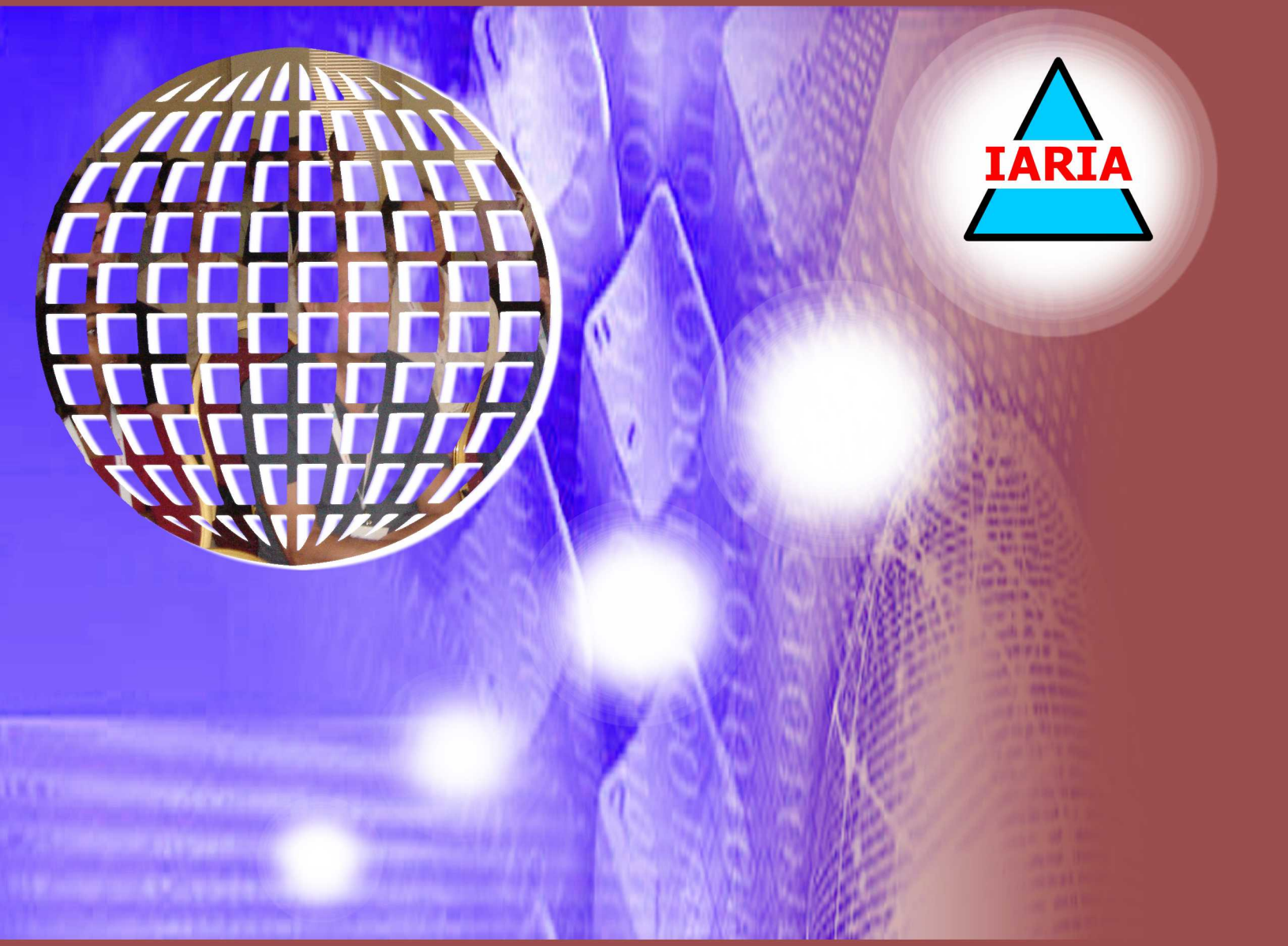
International Journal on Advances in Networks and Services



2026 vol. 19 nr. 1&2

The *International Journal on Advances in Networks and Services* is published by IARIA.
ISSN: 1942-2644
journals site: http://www.iariajournals.org
contact: petre@iaria.org

Responsibility for the contents rests upon the authors and not upon IARIA, nor on IARIA volunteers, staff, or contractors.

IARIA is the owner of the publication and of editorial aspects. IARIA reserves the right to update the content for quality improvements.

Abstracting is permitted with credit to the source. Libraries are permitted to photocopy or print, providing the reference is mentioned and that the resulting material is made available at no cost.

Reference should mention:

International Journal on Advances in Networks and Services, issn 1942-2644
vol. 19, no. 1 & 2, year 2026, http://www.iariajournals.org/networks_and_services/

The copyright for each included paper belongs to the authors. Republishing of same material, by authors or persons or organizations, is not allowed. Reprint rights can be granted by IARIA or by the authors, and must include proper reference.

Reference to an article in the journal is as follows:

<Author list>, "<Article title>"
International Journal on Advances in Networks and Services, issn 1942-2644
vol. 19, no. 1 & 2, year 2026, <start page>:<end page> , http://www.iariajournals.org/networks_and_services/

IARIA journals are made available for free, proving the appropriate references are made when their content is used.

Sponsored by IARIA
www.iaria.org

Copyright © 2026 IARIA

International Journal on Advances in Networks and Services
Volume 19, Number 1 & 2, 2026

Editor-in-Chief

Tibor Gyires, Illinois State University, USA

Editorial Board

Majid Bayani Abbasy, Universidad Nacional, Costa Rica
Muayad Al-Janabi, University of Technology, Iraq
Onur Alparslan, Osaka University / Doshisha University, Japan
Ilija Basicevic, University of Novi Sad, Serbia
Lasse Berntzen, University of South-Eastern Norway, Norway
Robert Bestak, Czech Technical University in Prague, Czech Republic
Razvan Bocu, Transilvania University of Brasov, Romania
Fernando Boronat Segui, Universitat Politecnica de Valencia, Spain
Marco Bruti, Telecom Italia Sparkle S.p.A., Italy
Albert M. K. Cheng, University of Houston, USA
Andrzej Chydzinski, Silesian University of Technology, Poland
Philip Davies, Bournemouth University, UK
Poonam Dharam, Saginaw Valley State University, USA
Kamil Dimililer, Near East University, Turkey
Jawad Drissi, Cameron University, USA
Mario Ezequiel Augusto, State University of Santa Catarina, Brazil
Rainer Falk, Siemens Technology, Germany
Mário Ferreira, University of Aveiro, Portugal
Steffen Fries, Siemens AG, Germany
Christos K. Georgiadis, University of Macedonia, Greece
Mohammad Reza Ghavidel Aghdam, Ozyegin University, Turkey
Juraj Giertl, Deutsche Telekom IT Solutions, Slovakia
Yi Gu, Middle Tennessee State University, USA
Xiang Gui, Massey University, New Zealand
Tibor Gyires, Illinois State University, USA
Fu-Hau Hsu, National Central University, Taiwan
Vasanth Iyer, Florida International University, Miami, USA
Jacek Izydorczyk, Silesian University of Technology, Poland
Yiming Ji, Georgia Southern University, USA
Maxim Kalinin, Peter the Great St.Petersburg Polytechnic University, Russia
György Kálmán, Obuda University, Budapest, Hungary
Ayad Ali Keshlaf, Sabratha University, Libya
İlker Korkmaz, Izmir University of Economics, Turkey
Dmitry Korzun, Petrozavodsk State University, Russia
Dragana Krstic, University of Nis, Serbia
Wen-Hsing Lai, National Kaohsiung University of Science and Technology, Taiwan
Wei-Ming Lin, University of Texas at San Antonio, USA
Jinwei Liu, Florida A&M University, USA
Maryam Tayefeh Mahmoudi, ICT Research Institute, Iran
Chengying Mao, Jiangxi University of Finance and Economics, China
Bruno Marques, Polytechnic Institute of Viseu, Portugal

Christopher Nguyen, Intel Corp., USA
Khoa Nguyen, Carleton University, Canada
Tudor Palade, Technical University of Cluj-Napoca, Romania
Constantin Paleologu, National University of Science and Technology Politehnica Bucharest, Romania
Paulo Pinto, Universidade Nova de Lisboa, Portugal
Agnieszka Piotrowska, Silesian University of Technology, Poland
Yenumula B. Reddy, Grambling State University, USA
Antonio Ruiz Martínez, University of Murcia, Spain
Addisson Salazar, Universitat Politècnica de València, Spain
Ioakeim K. Samaras, Intracom-Telecom, Software Development Center, Thessaloniki, Greece
Michael Sauer, Corning Incorporated, USA
Pushpendra Bahadur Singh, LTIMindtree, USA
Florian Skopik, AIT Austrian Institute of Technology, Austria
Vasco N. G. J. Soares, Polytechnic Institute of Castelo Branco | Instituto de Telecomunicações, Portugal
Dora Souliou, National Technical University of Athens, Greece
Pedro Sousa, University of Minho, Braga, Portugal
Álvaro Suárez Sarmiento, University of Las Palmas de Gran Canaria, Spain
Yongning Tang, Illinois State University, United States
Orazio Tomarchio, University of Catania, Italy
Božo Tomas, University of Mostar, Bosnia and Herzegovina
Kasturi Vasudevan, Indian Institute of Technology Kanpur, India
Yean-Fu Wen, National Taipei University, Taiwan
Mudasser F. Wyne, National University, USA
Cong-Cong Xing, Nicholls State University, USA
Martin Zimmermann, Offenburg University, Germany

International Journal on Advances in Networks and Services
Volume 19, Numbers 1 & 2, 2026

CONTENTS

pages: 1 - 13
Advanced Techno-Economic Modeling of Industrial 5G Non-Public Networks
Jyrki T. J. Penttinen, Alphacore Inc., United States
Jari Collin, Telia Company, Finland
Jarkko Pellikka, Nokia, Finland

pages: 14 - 19
Design Concepts to Satisfy User Data Accessibility of IoT Devices Postulated by the EU Data Act with Bosch Smart Home as a Case Study
Felix Fischer, Hochschule Mittweida, DE
Paul Seidel, Hochschule Mittweida, DE
Dirk Labudde, Hochschule Mittweida, DE

pages: 20 - 29
Information-Centric Wireless Sensor Network Framework for Smart Agriculture: Concept Proposal, System Design, and Preliminary Evaluation
Shintaro Mori, Fukuoka University, Japan

pages: 30 - 50
HTN Planning Deserves Better Domains: Engineering Realistic Models for Autonomous Vehicles in Space and on Earth
Ebaa Alnazer, University of Stuttgart, Germany
Ilche Georgievski, University of Stuttgart, Germany
Marco Aiello, University of Stuttgart, Germany

pages: 51 - 69
Design of a Hierarchical Communications Network for Public Services
Maria-Georgiana Butaru, National University of Science and Technology POLITEHNICA, Romania
Eugen Borcoci, National University of Science and Technology POLITEHNICA, Romania

pages: 70 - 82
Aerial 5G Non-Public Networks: Link Budget, Coverage, and Capacity Analysis
Jyrki T. J. Penttinen, Alphacore Inc., United States
Garth V. Rees, VERN Consulting, Spain
Urs C. Schlegel, Independent Researcher, Germany

pages: 83 - 97
Spatio-Temporal and Context-Aware Dataset for Mobile Network Forecasting
Mathias Gerstner, OTH Regensburg, Germany
Sebastian Fischer, OTH Regensburg, Germany
Rudolf Hackenberg, OTH Regensburg, Germany

pages: 98 - 107
Some Deficiencies of TCP BBR Implementation in Ns-3 Network Simulator Detected under High Speed and Heavy Load Conditions

Toshihiko Kato, University of Electro-Communications, Japan
Takahiko Kato, University of Fukui, Japan
Ryo Yamamoto, University of Electro-Communications, Japan
Satoshi Ohzahata, University of Electro-Communications, Japan

Advanced Techno-Economic Modeling of Industrial 5G Non-Public Networks

Jyrki Penttinen
Alphacore Inc.
Tempe, AZ, USA
email: jyrki.penttinen@outlook.com

Jari Collin
Telia Company
Helsinki, Finland
email: jari.collin@telia.com

Jarkko Pellikka
Nokia
Helsinki, Finland
email: jarkko.pellikka@nokia.com

***Abstract*— Non-Public Networks (NPNs), commonly referred to as Private Networks, are emerging as a pivotal component in commercial deployments across a wide range of industrial verticals, offering tailored connectivity solutions that meet specific operational needs. The development of NPNs is being spearheaded by the 3rd Generation Partnership Project (3GPP), with current models spanning multiple generations of mobile communication technology. This paper provides a review of the state-of-the-art standardized NPN solutions, with a particular focus on Fifth Generation (5G) mobile communications and critically evaluates their suitability for industrial applications identifying gaps in the existing NPN models that need to be addressed to fully realize their potential. Using selected industrial use cases, the paper explores the practical implications of deploying NPNs, analyzing key factors such as network performance, reliability, and security that are crucial for Industrial IoT (IIoT) scenarios. Furthermore, the paper assesses the key techno-economic considerations involved in NPN model selection, providing a framework for evaluating the cost-effectiveness and scalability of NPN solutions in industrial contexts. This paper offers insights and proposes base modeling and further evolved methodologies for optimizing the deployment and management of NPNs to support the growing needs of the industrial ecosystem.**

Keywords—NPN; private network; 5G deployment modeling; NPN optimization; network planning.

I. Introduction

A Non-Public Network (NPN), referred also to as Private Network, provides network services in isolated environment. It can be based on cellular networks or other wireless technologies, and it does not depend on numbering of regulated Public Land Mobile Networks (PLMN). Building upon previously published research [1] and the posterior advances of private networks, this paper explores the feasibility of NPN for serving industrial 5G applications. In addition to the assessment of available information from the standardization and field in general, this paper expands and deepens our originally presented model that helps select feasible private network model implementation considering specific requirements and needs of the users.

5G NPN refers to a 3GPP cellular system to deliver its capabilities for NPN use cases, such as businesses and municipalities. The 5G NPN can reside partially or completely in physical premises of an organization using it, e.g., within a factory or campus area, so that an external entity separate from a Mobile Network Operator (MNO) assumes the responsibilities of the isolated part offering its services to a limited group. NPN does not typically allow inbound roamers, although the NPN users may have roaming capabilities to use other PLMNs.

The benefits of NPN deployment include the possibility to control the Quality of Service (QoS) and protection by isolation. NPN service can include voice connectivity in defined geographical areas, or it can focus on Internet of Things (IoT) and respective industrial applications; today, there are many test projects and commercial setups involving industrial devices [2].

This paper explores the applicability of NPNs for Industrial 5G applications. “Industrial 5G” refers to Industrial Internet of Things (IIoT) with real-time data of networked sensors, assets, objects, and people, 5G network, and edge computing enabling ultra-reliable, low-latency, and high-bandwidth communication.

The presented model aims to provide means to evaluate the feasibility of available private network deployment options considering key requirements of use cases of interest, and it supports techno-economic assessment considering realities of various network deployment scenarios.

This paper introduces standardization of private network landscape in Section II and summarizes the key architectures of the available 3GPP models in Section III. Section IV discusses private network deployment models including 3GPP and other relevant options highlighting their key aspects. Section V summarizes the most important benefits and drawbacks of the presented private network variants, comparing their suitability in different scenarios. Section VI presents how techno-economic modeling can be applied in the selection of the most feasible variants based on key cost items related to Radio Access Network (RAN), Core Network (CN), Trasport Network (TN), and respective Capital Expenditure (CAPEX) and Operational Expenditure (OPEX) items. Section VII presents our further developed model, and Section VIII summarizes the findings.

II. Standardization

Private networks can be formed by a variety of technologies. The 3rd Generation Partnership Project (3GPP) has designed technical architecture specifications that build upon mobile network technologies, and industry bodies complement the effort through deployment guidelines.

A. 3GPP

3GPP has designed 4G and 5G NPN specifications in Release 16 providing enablers for 5G IoT, including IIoT [3] [4]. The areas include Time Sensitive Networking (TSN), NPN, and Local Area Network (LAN) type services. Release 17 evolve these capabilities further defining NPN in the Technical Specifications TS 23.251 (architecture and functional description of network sharing) [5], TS 22.104 (service requirements for cyber-physical control applications in vertical domains) [6], and TS 23.501 (5G system architecture) [7].

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

As stated in [8], Release 18 studied a set of features to evolve NPN. These study items include solutions for the support of non-3GPP access such as Wireless LAN (WLAN), support of mobility between Standalone NPNs, (SNPNs) without a need for new network selection, and means to provide access to local services such as enablement of localized service provider that can interact with NPN for local services.

Also, clarified specifications for charging aspects were noted important because, although no complex charging functions are needed for use cases involving internal SNPN communications, interaction between external entities may require support for more complete charging models.

As the NPN ecosystem evolves, a variety of new use cases and scenarios arise including, e.g., interaction involving one or more SNPNs, Public Network Integrated NPNs (PNI-NPNs), and MNO networks, and this expanding environment requires adequate solutions for a fluent charging. Release 18 has identified also the need for further enhancement of security and management of NPNs including end-to-end communications through radio and core networks, as well as capability exposure management for NPN variants.

As a result, the 3GPP Release 18 specified functions for the support: 1) of mobility between equivalent SNPNs, 2) for trusted and untrusted non-3GPP access for SNPN, 3) for connecting devices that lack support for 5G Core Non-Access Stratum (NAS) signaling over non-3GPP access to SNPN, and 4) for localized services during limited time within limited area. The Release 18 use cases focus largely on scenarios involving generic public in, e.g., campus, exposition, stadium, and concert venues. [9]

B. Industry Bodies

The 5G-ACIA (5G Alliance for Connected Industries and Automation) summarizes industrial IoT deployment scenarios for 5G NPNs based on 3GPP-defined 5G NPN [10]. It presents deployment models to complement their architectural design.

Also, GSMA (GSM Association) considers NPN, and their guidelines provide overview to deploy 5G industry campus NPN by 3GPP definition, which is one of the key 5G concepts to support “to business” models [11].

III. 5G NPN STANDARD ARCHITECTURES

As per the 3GPP Release 16 definitions, an NPN enables deployment of 5G System (5GS) for private use. The NPN can be deployed as a SNPN or PNI-NPN variant. An NPN operator manages SNPN without relying on the functions of a PLMN, whereas PNI-NPN deployment depends on those [12].

Figure 1 depicts NPN variants as interpreted from [11], [10].

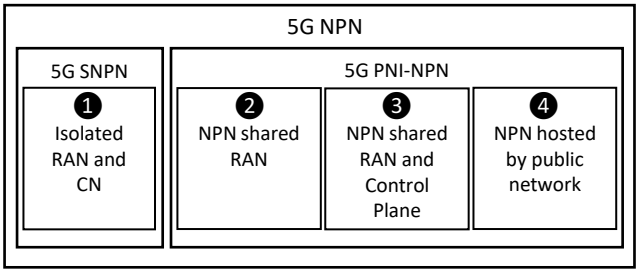


Figure 1 3GPP and 5G-ACIA NPN variants

A. SNPN

The SNPN uses combined PLMN Identifier (PLMN ID) and Network Identifier (NID). 5G User Equipment (UE) supporting SNPN can attach to it based on 5G Subscription Permanent Identifier (SUPI) and credentials. The RAN of SNPN broadcasts the combined PLMN ID and NID in the System Broadcast Information (SBI) signaling and supports network selection and re-selection, load and access control, and barring. The NIDs can be self-assigned individually to the SNPN NIDs upon its deployment.

The active NIDs may not be unique, but they use different numbering space than the other scenario, coordinated NID-assignment, that can have either 1) globally unique NID-assignment independent of the respective PLMN ID, or 2) globally unique NID/PLMN ID combination.

B. PNI-NPN

The PNI-NPN uses PLMN ID whereas Closed Access Group Identity (CAG ID) indicates the CAG -enabled 5G radio cells. Within a PLMN, a CAG cell can broadcast one or more CAG IDs, in which case PLMN ID is the base for the network selection and reselection whereas the network uses CAG ID for the cell selection and re-selection, as well as for the control for letting only CAG-enabled UEs access the network.

C. Implementation Aspects

As depicted in Figure 1, these two options result in practical scenarios of isolated deployment of standalone non-public network (1) or non-public network in conjunction with public networks. The latter breaks down into three scenarios including the Industrial and IoT environment: shared RAN (2); shared RAN and Control Plane (CP) (3); and NPN hosted by the public network (4) [10].

- SNPN: the NPN is separated from the public network, and all network functions reside inside the organization’s premises. The possible communication between the NPN and the public network takes place via a firewall, e.g., through Non-3GPP Interworking Function (N3IWF).
- Shared radio access network: the NPN and public network share part of the radio access network as per the 3GPP TS 23.251. Communication stays within NPN in this model.
- Shared RAN and CP: the NPN and the public network share the RAN for the defined premises while the public network does the network control tasks, the NPN traffic remaining within the premises. Network Slice (NS) or 3GPP Access Point Name (APN) can realize this case.
- PLMN-hosted NPN: the enterprise is served by a 5G NS.

IV. NPN DEPLOYMENT MODELS

A. Standalone NPN

Figure 2 depicts the principle of SNPN as interpreted from [10] [13]. In this scenario, the User Plane Function (UPF) works as a data router to connect Multi-access Edge Cloud (MEC) and possible LAN. The RAN manages the connectivity of 5G gNB (Next Generation NodeB) and UE of the SNPN users on licensed or unlicensed band. 5G Core (5GC) houses the NFs including Unified Data Management (UDM) of user credentials.

In this model, Network Functions (NFs) reside within the operational area of related entity, e.g., factory, the SNPN being

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

isolated network from the PLMN, allowing the communication between the PLMN and NPN through an optional firewall which isolates the NPN. In this way, Operational Technology (OT) company can operate the NPN and its services, including the NPN IDs to provide additional PLMN services in the NPN coverage area, NPN-subscribers to roam the public networks, and public networks' subscribers to roam the NPN depending on roaming agreement. NPN users may also have dual subscription for PLMN use.

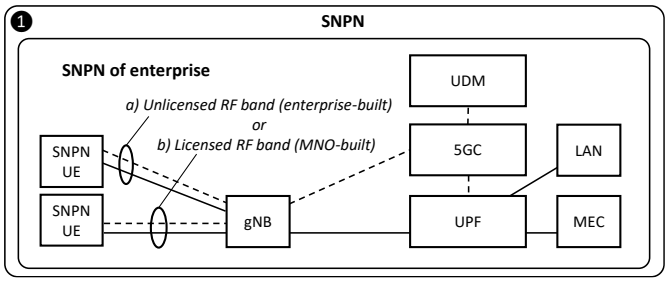


Figure 2 Principle of a SNPN

An example of the NPN 5G deployment on IIoT scenarios is the interconnection with TSN as per 3GPP TS 24.519 [14]. TSN is a set of new open standards that provide deterministic, reliable, high-bandwidth, low-latency communication [15].

The standalone private network can be built in various ways: by dedicated Service Level Assurance (SLA), local PLMN, PLMN by dedicated proportion of operator's licensed spectrum, or by SNPN using unlicensed or private spectrum. Each deployment model has its pros and cons. As an example, licensed spectrum is one of the most expensive single items in the commercial network. Feasible way to set up this type of network is to construct mm-Wave radio access points in a limited enterprise area and virtualized cloud core functions in nearby edge, one option being a broker managing the NPN [16].

B. Shared RAN

Shared RAN involves NPN and PLMN with certain part of its RAN for joint use whereas other network functions remain separated. Figure 3 depicts the principle of this model showing the connectivity of the NPN RAN to PLMN core while the own core network of the NPN is isolated from the external world as interpreted from [10], [11], and [17].

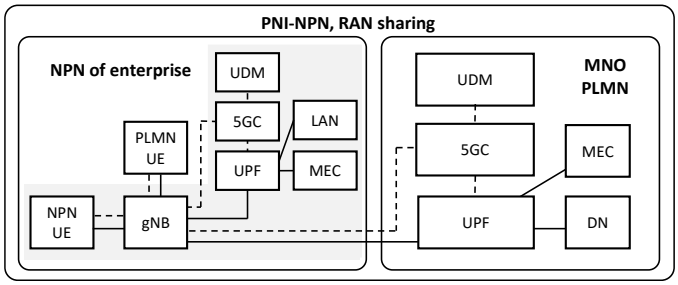


Figure 3 Shared RAN NPN deployment; the grey area indicates the private network slice that forms the NPN

In this deployment model, NPN traffic stays internal and within the logical, defined area, such as factory premises.

The 3GPP TS 23.251 details the network sharing model in its architectural and functional description and scenarios for network sharing usable also in NPN environment, which are Gateway Core Network (GWCN) and Multi-Operator Core Network (MOCN) [5].

C. Shared RAN and Control Plane

As depicted in Figure 4, interpreted from [10], [11], and [17], in this deployment, both NPN and PLMN share the RAN within defined business area premises and PLMN takes care of the control plane so that the internal NPN traffic stays always within the logical network related to the business. NS is one way to set up this scenario creating logically independent networks within a shared physical infrastructure. The isolation of the private network portion is possible by using unique NS identifiers. Using 3GPP-defined APN is another method to implement this scenario. In this case, the APN indicates the target network with opportunity to differentiate traffic.

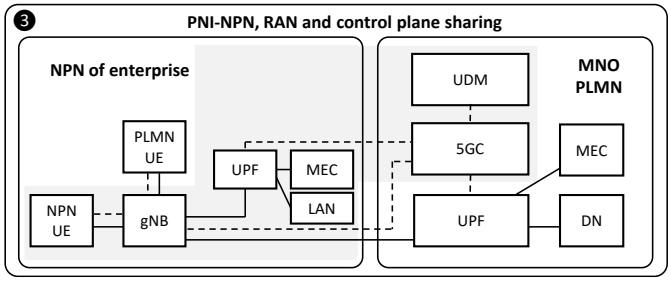


Figure 4 Deployment for shared RAN with control plane; the grey area indicates the private network slice that forms the NPN.

In the shared RAN scenario with shared Control Plane, the PLMN hosts the NPN so that the devices are a subset of PLMN subscribers. This arrangement eases the contractual aspects of PLMN and NPN operators, and the NPN devices can connect, apart from the NPN itself, also to the PLMN services and roaming. The NPN services may connect to PLMN services, which require optional interface between the NPN and PLMN services, so NPN devices can connect to NPN services via the PLMN if the device is located outside of the NPN coverage and still within the PLMN. Logically, if the NPN devices can access the PLMN services, this interface is not needed.

In Shared RAN and Control Plane deployment models, a particularly important evolution is the use of Local Breakout (LBO) architectures, where the UPF and selected application services are deployed locally within or near the enterprise premises while the Control Plane remains integrated with the MNO's infrastructure. This arrangement creates a practical balance between fully isolated SNPNs and fully outsourced public-network-hosted solutions. The enterprise benefits from reduced infrastructure ownership complexity because mobility management, authentication, orchestration, and subscriber control can remain under the operator's domain, while latency-sensitive industrial traffic is processed locally without traversing distant transport and core network paths. This is especially important for Ultra-Reliable Low-Latency Communications (URLLC) use cases such as robotic automation, autonomous vehicles, motion control, machine vision, and TSN integration.

The LBO also improves deterministic performance, data sovereignty, operational resilience, and cybersecurity isolation while still allowing efficient integration with public network services, roaming, and cloud-native management platforms.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

D. PLMN-Hosted NPN

In this case, as depicted in Figure 5 and interpreted from [10], [11], [17], both the PLMN and NPN traffic are external to the business area.

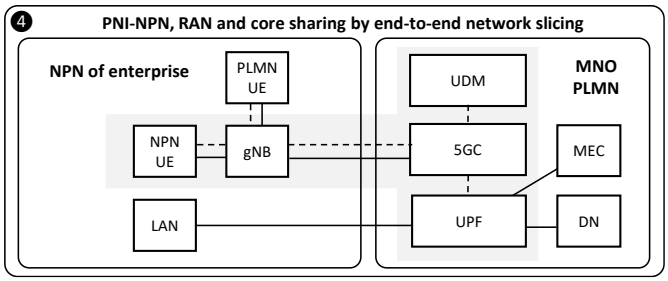


Figure 5 PLMN-hosted NPN; the grey area indicates the private network slice that forms the NPN

This scenario is facilitated by network virtualization and cloudification. It means that these traffic flows are served by different networks, and the NPN subscribers are in fact public network subscribers. The NPN-PLMN roaming implementation is straightforward as the traffic routes via the PLMN.

E. Private Network on Network Slice

Although the differentiation between certain user types is possible in 4G networks, it is limited to techniques, such as the isolation of services in a common infrastructure; the means for this include APN Routing, MOCN, and Dedicated Core Network (DECOR) [18]. Built upon Service Based Architecture (SBA), which enables the use of common hardware that executes the NFs as instances, 5G has been designed to support also network slicing with respective QoS assurance.

In this manner, the Network Slice Provider (NSP) can offer suitable characteristics within their different NSs fulfilling variety of different requirements for their subscribed verticals, also in NPN environment. NS provides means to differentiate the network resources and performance figures that can create also new business models. As an example, an operator can offer their customers gold, silver, and bronze categories, each having their personalized NS price and QoS levels [19]. The NSP can be either MNO or 3rd party. Technically, the NSP could be also enterprise taken their skillset suffices to manage slices.

In NS-based NPN, it is important to adequately interpret the end-user requirements. GSMA provides guidelines for NS setup based on requirements [20], and clarifies how the requirements can be captured from the vertical field [19].

NSs are not yet used widely in commercial Standalone (SA) 5G networks, despite the statistics indicating steady growth and about 25% use base in 5G by 2025 [21]. Furthermore, the optimal NS functionality in practice, especially in the end-to-end scenarios, might require still further development to cope with the impacts of real-world non-idealities in synchronization, near real-time orchestration, and overall management of the slices.

F. Open RAN as a Private Network

Open RAN refers to the overall movement of the telecom industry to disaggregate hardware and software to create respective open interfaces in between [22]. O-RAN Alliance publishes RAN specifications, releases open software for the RAN, and supports O-RAN Alliance members in integration and implementation testing. O-RAN Alliance works on open, interoperable interfaces, RAN virtualization, and big data enabled RAN intelligence [22] [23].

Open RAN may offer feasible small-scale, isolated shared RAN network deployment options also as NPNs. The concept is still evolving, though, and may not provide optimal techno-economic solutions quite soon; nevertheless, as the technology matures, Open RAN may offer competitive 4G and 5G NPN variants.

G. Legacy Network as a Service

Increasing number of incumbent MNOs have switched off their 2G and/or 3G legacy mobile communications networks or are aiming to sunset them soon. Nevertheless, there will remain scenarios involving IoT and voice services via legacy systems, especially in developing markets. According to GSMA statistics, the 2G and 3G systems combined still more than 20% of the global footprint in 2025 [21]. This can be a niche opportunity to manage part of this lessening infrastructure on the remaining spectrum, maintaining a minimum feasible type of infrastructure for IoT and other 2G/3G services via private networks for verticals needing only low capacity.

H. Local and Fixed Wireless Access (FWA)

The LTE-WLAN Aggregation (LWA) as per 3GPP Release 13 allows mobile device configuring on simultaneous LTE and Wi-Fi links [24]. 5G can work also in parallel with non-3GPP access to comply with a light-weight private networks' need. 3GPP Release 15 defines N3IWF allowing Wi-Fi access points delivered through 5G infrastructure as Wi-Fi hotspots or Fixed Wireless Access (FWA).

Release 16 defines further a Residential Gateway (RG) to interconnect end-users' devices via trusted access points. These gateways complement the N3IWF.

FWA can provide solutions to variety of use cases, such as tethering and Mobile Broadband (MBB), best effort FWA, and speed-based QoS [25]. Reflecting these use cases, the FWA could serve as a small-scale home office solution, too, with quick and economic deployment without need for optic fiber.

I. Non-3GPP Wireless Private Network

Technically, it is possible to set up a simple private network using any wireless access beyond the 3GPP specifications on unlicensed, shared spectrum. An example of this is a Wi-Fi hotspot network using common applications within the network providing Over the Top (OTT) voice and messaging. Also, Low Power Wide Area Networks (LPWANs) can provide a feasible communications channel to many IIoT use cases.

V. Considerations of Scenarios

The feasibility of each NPN deployment depends on the foreseen use cases. As an example, the roaming requirements impact the needed coverage and mobility of the end-users of the NPN. This is one of the examples that can dictate the selection between Standalone and PLMN-assisted scenarios.

A. SNSP Deployment

The main aspects of the previously presented deployment models for SNPN scenarios are summarized in the following list, with statements on their suitability for selected use cases.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

5G SNPN. An OT operates the NPN and its services behind a firewall independently. This solution provides good level for security isolation, e.g., to IIoT applications as data is not exposed externally. Optional PLMN interconnectivity can be realized via firewall. The operation and management of SNPN requires sufficient level of skillset from OT company. The solution provides an opportunity to build a very secure environment, but it can turn out to be more expensive than only partially owned, or completely outsourced network.

5G SNPN with SLA. The agreed SLA level impacts the business case in this model, so, the higher the SLA, more costly the CAPEX and OPEX are due to, e.g., active-active network mode and reliability of, e.g., 99.999% as per 3GPP Release 15 URLLC performance, or up to 99.9999% as per Release 16 performance for, e.g., TSN interconnectivity to serve critical IIoT applications.

PLMN with local infrastructure. This is a special case of PLMN that isolates part of the infrastructure and spectrum for sole use by private network devices in limited geographical areas. It can be set up technically, e.g., by barring access from others than specifically defined set of subscribers.

PLMN on part of MNO's licensed spectrum. Spectrum is typically very expensive investment for the license holder, so dedicating part of it needs to be designed carefully applying techno-economic optimization to adequately balance the cost and expected quality.

SA NPN on unlicensed spectrum. The RAN deployment and RAN business case are light in this scenario as there is no license fee involved. Nevertheless, the QoS cannot be ensured due to load of shared spectrum with possible other users.

B. PNI-NPN Deployment

The following list summarizes key aspects of this model.

NPN shared RAN. NPN and PLMN share part of the RAN, but the NPN communications stay within the defined premises. 3GPP defines well the technical RAN sharing options that can be applied in this model.

NPN shared RAN and CP. The NPN and PLMN share the RAN for the defined premises while the PLMN has network control and the NPN traffic remains within the defined premises. Network slicing serves this model as per 3GPP specifications, complemented by industry forums' guidelines for slice template setup. Alternatively, the setup can be based on APN.

NPN hosted by public network. PLMN and NPN traffic are external to the business area in this scenario so that these traffic flows are served by different networks, and the NPN subscribers are effectively public network subscribers.

C. Pros and Cons of Deployment Models

a) SNPN

Pros: Access for customization, independent controlling; high security by full isolation; RAN functions are within reduced geographical area favoring low-latency applications.

Cons: Deployment cost; expertise required for deployment. Dedicated network for sole enterprise includes the cost of the whole system in the geographic area.

b) Shared RAN

Pros: Optimizes RAN infrastructure costs while the internal data remains within the NPN infrastructure providing good protection; PLMN RAN connectivity serves for delivering the data meant for outside of the NPN as per need. Within the NPN, part of the base stations can be connected to PLMN according to the RAN sharing agreement between the PLMN and NPN operators while the rest can be kept internal. Licensed spectrum copes interferences; deployment less expensive compared to SNPN; uses typically local functions favoring low-latency applications.

Cons: External interferences can be higher than in SNPN, and the overall control of the network is less independent; need for local expertise, although less than in SNSP.

c) Shared RAN and CP

Pros: Licensed spectrum for controlled interferences; lower deployment expenses compared to SNPN and PNI-NPN; SLA can be applied between the NPN and public network.

Cons: Less independent from public networks; latency typically higher than in SNSP and PNI-NPN deployments; some local expertise required.

d) Hosted solution (Network Slicing by NSP/MNO)

Pros: Facilitated by NSP, no need for local expertise; fast to set up and adjust based on expressed requirements.

Cons: Less control for adjustments as the NS is managed by the NSP; technology not yet final, practical deployments require SA 5G network that are not yet many in markets.

e) Open RAN as an NPN service

Pros: The cost can be low; easy to set up by provider; can be hosted as "light-weight" 5G SNPN or NPI-NSN.

Cons: technology is still evolving and the realistic products for Open RAN -based NPN can take time.

f) 5G FWA (home office use case)

Pros: Customer Premises Equipment (CPE) easy to install; replaces fixed cabling.

Cons: use case adequate in a home office environment but limited for larger enterprise NPN use.

g) Non-5G-based solutions (Wi-Fi hotspots)

Pros: Wi-Fi hotspot deployment is rather straightforward within an enterprise area. The radio coverage does not require license, and it can also be extended to reach any Wi-Fi device external to the enterprise premises.

Cons: Low security, limited mobility, and lack of QoS.

h) LPWAN

Pros: Many options available basing on both cellular and non-cellular radio technologies. Cellular-based LPWANs are services integrated to system, easy to deploy.

Cons: Non-3GPP-based LPWANs have varying security and protection levels, and they require a separate infrastructure.

VI. TECHNO-ECONOMIC OPTIMIZATION MODELING

A. Current Deployment Models

Each NPN deployment model has its advantages and disadvantages. Understanding the requirements of the use cases and applying tecno-economic optimization assessment that considers the key variables, the selection of the most adequate deployment model will benefit favorable business.

In the selection of the deployment model, the task is to understand the realistic needs of the final users of the NPN access, performance, security aspects, mobility, capacity, QoS, and other key factors, and how they can be served by applying

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

cost-efficient technical solutions. It is important to understand also the changing requirements in the foreseen future because along with the evolution of the environment, originally selected, initially optimal model, can turn out to be less optimal in longer term.

As described in [19], related to NS scenarios, the collection of the verticals' requirements can be done via practical means to interpret the needs of the end-users. From the operators' perspective, this can be somewhat challenging task as the verticals may often express their requirements using non-standard terminology or are not able to formulate the actual requirements.

The methodology in [19], despite its original focus on the assessment of vertical requirements for forming NS templates, can be useful to be extended to cover additional aspects the other NPN deployment models. This methodology suits to interpret practical vertical needs based on the foreseen use cases, and to form technical requirements that can be finally be mapped to represent input parameters for the NPN modeling, whether it is about tailored 5G NS template for which the GSMA PRD NG.116 serves as a base, more "traditional" standalone NPN setup, or shared network model.

The aim of the requirement list is to ensure a common understanding of the environment and to set the expectations also for service level.

B. Base Techno-Economic Optimization

The assumption of the modeling is that the NPN is a business between an entity capable of deploying adequate wireless network (such as MNO, NSP, network equipment vendor, or system integrator) and an enterprise desiring to facilitate the mobile communications for their end-users via Industry IoT applications (such as port or energy company monitoring and controlling their workflow via intelligent sensor network).

The model for the selection of the most adequate NPN deployment option can build upon modular elements, which are:

1: *Interpretation* of the enterprise and end-user needs (e.g., via survey) to form technical requirements statement as a base for the input parameters; e.g., capacity (number of expected users/devices), coverage, QoS, need for roaming/local-only utilization; and services (IoT, voice, other).

2: *Business aspects* assessment (understanding possibility for investment in terms of CAPEX and OPEX; flexibility for initial and longer-term investments).

3: *Forecast* of near-future and longer-term outlook for the possible need for expansion of the network, capacity, and evolving QoS (which is important to avoid investing to multiple types of NPN as the requirements evolve).

4: *Any other relevant information* on the deployment aspects.

The assessment of the feasibility of the deployment options relevant to the scenario under evaluation can be carried out based on these results in a comparative manner. The base for the economic assessment is the cost for enterprise in terms of the CAPEX (initial deployment and forecasted posterior need for new infrastructure investments) and OPEX (yearly cost that is required to operate NPN).

The cost estimate of each scenario presented in the NPN Deployment Models Section considers the key attributes, such as area of deployment, device number (total expense x_d), and radio performance indicators, which together result in the required bandwidth and number of radio cells, and finally in the total cost of cells x_{gnb}. As an example, in very high data rate scenario requiring large indoor and outdoor NPN, the number of mm-Wave small cells, each resulting in, e.g., 80-100 m cell range, can be in order of dozens per km^2. Other attributes consider the cost of transmission network x_{tn}, and spectrum x_s.

For the core network, the cost includes the licenses and other cost items to activate the needed network functions NF x_{nf}. Cost of the needed applications / services x_a refers to the support of, e.g., voice service (that requires either own or outsourced IMS core for integrated Voice over New Radio) and IoT service license, and Location Based Service (LBS) deployment. Roaming and interconnectivity cost x_r is related to the agreements with national and international networks.

The cost x_v of other variable items can include, e.g., the actual installation of RAN, TN, CN equipment or cloud / edge environment, including antenna systems, base station shields, cabling, and any other expense that is required to set up an NPN for enterprise.

For the estimate of yearly operating costs y, the same main components as presented in CAPEX analysis generate expenses, such as licensing fees and electricity consumption, whereas an additional item to be considered in operations is the maintenance cost y_m. The resulting Equations for the initial costs (1) and operating costs (2) are thus:

$$CAPEX = x_{gnb} + x_{tn} + x_s + x_a + x_d + x_v + x_r + \sum_{nf1}^{nfn} x_{nf} \quad (1)$$

$$OPEX = y_m + y_{gnb} + y_{tn} + y_s + y_a + y_d + y_v + y_r + \sum_{nf1}^{nfn} y_{nf} \quad (2)$$

In this base techno-economic assessment, the resulting graphs turn out to be linear over time, providing a simplified manner to form an estimated CAPEX and OPEX behavior as a function of time. It should be noted, though, that the variables in Equations (1) and (2) can have non-linear inter-dependencies in practice; as an example, the volume discount of the number of radio cells can also lower the relative cost of core software and cloud feature licensing. This scenario will be discussed in Section VII, Further Developed Model.

The assessment presented hereby results in a statement of the suitability of deployment scenarios to indicate their level of compliance with the requirements. This method can be visualized in terms of the total cost per area as a function of time, considering attributes of interest, such as maximum supported device number or maximum data rate.

The method serves thus to estimate the rough level of initial and longer-term cost of deployment models under evaluation. Most importantly, the model indicates if the initially most cost-efficient option might turn out to be less optimal in longer run.

C. Return on Investment

The described modeling can be extended to estimate Return on Investment (RoI) of private network, including the business of MNO, enterprise, or 3rd party.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The RoI depends on the deployment and operational costs, share of ownership of private network components (hardware, software) versus outsourced items (e.g., 5G core that runs in virtualized environment served by cloud provider) in different deployment scenarios of interest, the generated savings compared to reference deployment scenario (as an example, enterprise can compare MNO-operated scenario against completely or partially enterprise-owned network), as well as potential earnings for different stakeholders.

As an example, enterprise managing completely or partially owned private network, either on shared or own spectrum, could allow also additional users to roam into that network for a fee that depends on the data consumption or time.

Although the pricing of network components is business between the vendors and customers and thus largely non-public information, it can be assumed that large entities investing to either own network infrastructure or outsourced solutions to provide the private network services to the end-users, may benefit from lower costs due to scale of economies compared to smaller entities.

Nevertheless, the private network ecosystem is expected to grow significantly at present, so it can thus also be assumed that small and medium sized entities could reach fortified position for own network component price negotiation in their private network market, which also can impact positively on the pricing models.

D. Expectations of the Modeling

As can be seen from Equations 1 and 2, the selected items result in a linear presentation of initial and operating costs. That said, the Equations represent snapshots of scenarios, and each parameter value may have either linear or non-linear behavior as a function of time, number of components, etc. As an example, the expense related to gNB can be either fixed per the number of gNBs, or there could be a volume-based discount granted by the vendor as a function of the number of gNBs.

As can be expected, the values of the cost items in Equations 1 and 2 depend on the markets, vendor pricing strategies, competitive landscape, and many more variables, so the further analysis of scenarios would be merely speculative without the availability of concrete parameter values. Nevertheless, an example of potential possible behavior can be presented by testing different scenarios and cost estimates of the parameters to understand the business impact in short, mid, and long-term operation of a private network.

The scenarios can be divided into following categories for the assessment of the total cost of a network, that a) is completely owned, partially owned, or completely outsourced ownership; b) uses licensed, shared, or unlicensed spectrum; c) has no roaming (completely isolated), or has inbound roaming, outbound roaming, or bilateral roaming.

To complement the evaluation, additional criteria can be assessed, too, for validating the level of compliance for end-user requirements such as QoS, latency, maximum and average data rate, reliability, etc. The level of compliance of different scenarios can be compared by using numeric values and their weights of importance.

Let us assume an enterprise desires to compare the techno-economic feasibility of a) completely own and isolated small-scale (10 mm-Wave gNBs), 5G network (SNPN) that is based on unlicensed 5 GHz spectrum and 5GC NFs on cloud, with b) MNO-operated private network that is based on an NS dedicated to the enterprise with gNBs that are already partially deployed for PLMN users in the area complemented by five additional, enterprise-owned indoor mm-Wave small cells that are located in the enterprise's operational premises.

Based on these assumptions, Figure 6 depicts theoretical example of the relevant key expense behavior over time using the parameters of Equation 1 and 2 and certain estimated values so that they are normalized having the SNPN CAPEX as the reference at year 0.

In this example, the assessment considers a relatively modest base of 50 mid category user smart devices, with 20% yearly renewal rate.

A considerable cost item in the SNPN deployment can be the price of the radio base stations (gNB). It should be noted that this item may vary largely and it depends on the business models applied between selected network vendor and enterprise customer, as well as the final set of quality, functional, capacity, and performance requirements.

Although the 3GPP technical specifications set the reference for the minimum viable performance, for example the level of radio and core network quality needs to be dimensioned based on the desired key performance indicators that may vary based on, e.g., on the criticality of the served area.

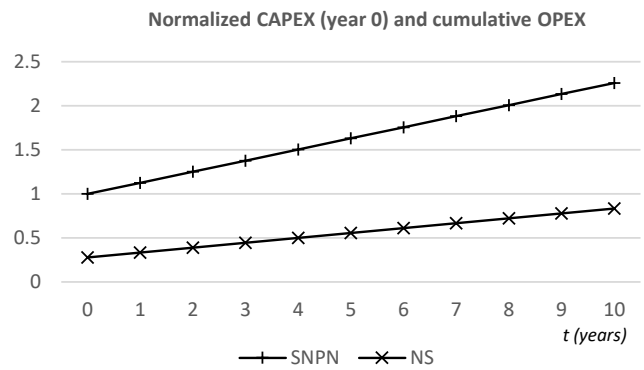


Figure 6 Example of the model's outcome comparing SNPN (reference) and NS deployment scenarios

In the highest quality service provisioning scenarios, a licensed spectrum is important, but at the same time, depending on the national regulation and type of license agreements, the license fees may represent a major cost item.

As the SNPN in this initial base assessment uses unlicensed spectrum, the analysis assumes there are no related fees in the use cases or the results presented in Figure 6.

Another cost item for an enterprise-owned and operated SNPN is the transmission from gNB to the rest of the network. Its significance depends on the respective provider and cost models. It can be assumed that this cost item is not of concern in completely outsourced NS deployment model.

As can be seen from Figure 6, the initial cost of completely own network of enterprise can be considerably higher than a subscription to MNO's NS-based service to form a private network due to required investments on the infrastructure. In this scenario, the OPEX of the SNPN adds up faster, too, compared to the dedicated MNO NS due to need for maintenance and licensing expenses of the own network.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

VII. Further Developed Model

The original techno-economic assessment presented in the conference version of this work provided a high-level framework for comparing different NPN deployment options using simplified CAPEX and OPEX representations [1].

Although this approach served as a baseline for illustrating the economic differences between deployment models, in practical industrial deployments, the behavior of private network costs, reflecting operational requirements, is considerably more complex than what purely linear models can adequately describe.

A. New Aspects of the Model

Industrial 5G deployments evolve over time. The operational environment may change, e.g., due to increasing numbers of connected devices, evolving QoS and other performance and functional requirements, software modernization, expansion of edge computing resources, changes in energy pricing, and increasing cybersecurity requirements. The deployment costs may exhibit nonlinear behavior, too.

As an example, the unit cost of radio access equipment may decrease due to economies of scale or customer-specific discounts over time and bulk orders, while maintenance aspects may vary disproportionately as the network expands. Equally, virtualization and cloud-native architecture models may initially reduce infrastructure investments but introduce recurring operational costs that can grow over time as per traffic and computational demand.

For these reasons, the presented enhanced model aims to extend the original framework into a dynamic lifecycle-oriented techno-economic model, which can reflect realistic operational behavior of industrial 5G NPNs. The updated model includes weighted multi-criteria assessment and takes better into account nonlinear parameter dependencies, deployment specialization coefficients, technology renewal cycles, and time-dependent operational growth functions. The resulting framework enables comparative evaluation of different deployment scenarios, including SNPNs, PNI-NPNs, NS deployments, Open RAN implementations, and hybrid ownership models.

The objective of the new model is still to identify the lowest-cost deployment alternative considering technical requirements, but considering more detailed operational constraints, scalability expectations, and long-term economic sustainability.

B. Lifecycle Modeling

The enhanced model presented hereby evaluates the Total Cost of Ownership (TCO) of a private network deployment over a selected operational lifetime. The model considers both initial investments and operational costs while discounting future expenditures to present value using a configurable discount rate. The lifecycle cost model can be expressed as:

$$TCO(t) = CAPEX(t) + \sum_{i=1}^{t} \frac{OPEX(i)}{(1+r)^i} \qquad (3)$$

where $TCO(t)$ represents cumulative lifecycle cost at year t, $CAPEX(t)$ denotes cumulative infrastructure investments up to time t, $OPEX(i)$ represents yearly operational expenditures, and r is the discount rate reflecting the time value of money.

The lifecycle-oriented TCO formulation used in the model applies discounted cash flow principles to represent the economic value of future expenditures in present-value terms. The denominator term $(1+r)^2$ represents the discounting factor applied to future operational expenditures occurring at year i. The purpose of this term is to account for the time value of money, based on the principle in techno-economic assessment, financial analysis, and infrastructure investment planning [26].

The rationale behind the denominator is that a monetary amount spent in the future does not have the same economic value as the same amount spent today. Capital available today can potentially be invested elsewhere to generate returns, while future expenditures are economically worth less when converted into present-value terms. Thus, future operational costs must be discounted back to the reference year to compare investment alternatives consistently over long operational periods.

The estimation of discounted lifecycle costing is important because industrial private networks can be expected to be long-term strategic investments whose economic feasibility depends on initial infrastructure expenses and recurring operational and modernization costs over several years. In practical industrial deployments, it is not uncommon that a deployment model with lower initial CAPEX may ultimately become more expensive due to, e.g., increasing operational complexity, subscription-based service costs, or software licensing fees.

The enhanced model separates the overall lifecycle behavior into interacting categories including infrastructure deployment costs, operational costs, network growth dynamics, technology refresh cycles, QoS-driven scaling effects, and deployment-specific specialization factors.

C. Enhanced CAPEX Modeling

The original model represented the CAPEX as a sum of infrastructure-related cost components. The new, enhanced framework still preserves this modular structure with additional functional dependencies and deployment-specific parameters. The total CAPEX is represented now as:

$$\begin{aligned} CAPEX(t) &= C_{RAN}(t) + C_{CN}(t) + C_{TN}(t) + C_S(t) + C_A(t) \\ &+ C_I(t) + C_{SEC}(t) + C_{EDGE}(t) \end{aligned} \qquad (4)$$

The model separates the investment items into radio access network costs (RAN), core network infrastructure (CN), transport network deployment (TN), spectrum-related costs (S), application platforms (A), installation and integration expenses (I), security infrastructure (SEC), and edge computing resources (EDGE). This categorization reflects the fact that industrial private networks can be assumed to become increasingly dependent on distributed cloud-native architectures and edge computing systems in addition to traditional radio infrastructure.

In modern industrial deployments, edge computing and security systems may represent, in fact, a substantial share of overall investment due to the need for low-latency processing, deterministic industrial communications, cybersecurity hardening, and local analytics capabilities.

The model also enables differentiation between fully enterprise-owned deployments, partially shared infrastructures, and outsourced service-based implementations. In practice, these deployment alternatives differ in how investments are

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

distributed between local infrastructure, subscription-based services, and cloud-native operational environments.

D. Radio Access Network Cost Behavior

The RAN remains one of the most significant contributors to the total infrastructure investment of industrial 5G private networks. However, the behavior of the cost of the RAN is not linear in practical deployments. The enhanced model represents the RAN investment as:

$$C_{RAN}(t) = N_{gNB}(t) \cdot P_{gNB} \cdot D_v(N_{gNB}) \cdot S_{QoS} \cdot S_f \quad (5)$$

where the cost depends on the number of deployed radio cells (N_{gNB}), the average unit price of each gNB (P_{gNB}), a nonlinear volume discount function D_v, a QoS multiplier S_{QoS}, and a frequency-dependent deployment coefficient S_f.

The inclusion of the nonlinear volume discount function reflects realistic procurement behavior. In large deployments, vendors may provide reduced relative unit pricing due to economies of scale, framework agreements, software bundling, and shared deployment costs, and this behavior can be modeled using the following equation:

$$D_v(N_{gNB}) = N^{-\alpha} \quad (6)$$

where the coefficient α reflects the magnitude of the volume-based cost reduction. The coefficient α represents the strength of the economies-of-scale effect associated with increasing deployment volume. The purpose of this coefficient is to model the realistic observation that the relative unit cost of infrastructure components decreases as the number of deployed radio nodes increases due to factors such as bulk procurement, shared installation resources, standardized deployment processes, centralized management efficiencies, and vendor discount structures.

In practice, when α=0, no volume discount exists and the deployment behaves linearly. When α>0, the effective unit cost decreases progressively with deployment size, whereas larger values of α correspond to stronger economies of scale. As an example of this behavior, a small industrial deployment of, say, five indoor gNBs may require largely manual installation, custom integration, and individually licensed software, whereas a deployment involving hundreds of gNBs may benefit from framework agreements, automated provisioning, shared transport infrastructure, optimized logistics, and lower per-unit software licensing costs.

This means that the total RAN cost does not scale linearly with the number of deployed cells, even though the absolute infrastructure investment continues to increase. The nonlinear coefficient is especially relevant in industrial 5G deployments because dense private networks may require large numbers of small cells, cloud-native orchestration enables centralized scaling efficiencies, and industrial campuses may reuse shared infrastructure components such as transport, edge computing, and security systems. This coefficient is thus selected because the use of power-law scaling to model economies of scale is well established in engineering economics and infrastructure cost modeling and can be found in well-established references [27].

The QoS multiplier models the impact of reliability and latency requirements on infrastructure costs. In critical industrial environments, such as automated manufacturing, mining, and port automation, the network may require, e.g., redundant radio coverage, active-active operation, and hardened resilience mechanisms, which means that the respectively implemented infrastructure can become more expensive compared to best-effort connectivity solutions.

In a similar fashion, the frequency bands impact deployment economics. Millimeter-wave deployments may provide very high capacity and low latency but require significantly much more dense radio deployments due to limited propagation characteristics, whereas lower-frequency deployments may reduce the required number of cells but provide lower overall throughput capacity.

E. Core Network and Virtualization Scaling

The economic behavior of the 5GC infrastructure differs from traditional hardware-centric deployments because cloud-native architecture models allow dynamic scaling of NFs. The enhanced model expresses the core network cost as:

$$C_{CORE}(t) = (C_{NF} + C_{LIC}) \cdot U(t)^{\beta} \quad (7)$$

where the cost depends on the infrastructure supporting the network functions (*NF*), software licensing (*LIC*) expenses, and the scaling behavior associated with the number of users *U* (that is, connected devices).

This formulation allows modeling of elastic and non-elastic deployment behaviors. In lightweight, cloud-native deployment scenarios, the scaling exponent β may remain below unity because virtualization enables efficient resource pooling and dynamic orchestration, whereas in highly isolated industrial private networks, the operational complexity and infrastructure cost may increase faster than the number of users or devices because strict reliability and security requirements often require duplicated network functions, geographically or logically separated resources, dedicated security domains, failover mechanisms, and increasingly complex orchestration and lifecycle management systems.

The model also reflects the growing importance of software licensing and orchestration platforms in modern private network deployments. While virtualization may reduce initial hardware investments, software-driven infrastructure can involve recurring operational cost items that evolve dynamically according to traffic growth and computational demand.

F. Transport Network and Edge Computing

Transport infrastructure plays a central role in industrial private networks because ultra-reliable and low-latency communication require deterministic transport characteristics. The transport network cost can be modeled as:

$$C_{TN}(t) = L_{fiber} \cdot P_{fiber} + N_{link} \cdot P_{mw} \quad (8)$$

where the total cost depends on deployed fiber infrastructure and microwave transport links. The first part of the Equation represents the total cost of deployed fiber-optic infrastructure, where L_{fiber} is the total length of the deployed fiber transport network (in m or km), and P_{fiber} is the unit deployment cost of fiber per distance unit, including expenses such as trenching, cabling, connectors, installation labor, permits, and optical transmission equipment.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The second part of the Equation represents the cost of microwave-based transport links, where N_{link} is the number of microwave transport links deployed in the network, and P_{mw} is the average cost per microwave link, including antennas, radios, mounting infrastructure, alignment, and installation. Together, these two terms model the total transport network investment required to interconnect radio access nodes, edge computing platforms, and core network functions using either fiber-based or wireless backhaul infrastructure, or a combination of both.

The model supports different transport architectures including fully fiberized industrial campuses, microwave-based industrial deployments, hybrid transport systems, and shared operator transport infrastructures.

Edge computing infrastructure is represented separately because industrial applications increasingly rely on localized processing for deterministic operation, artificial intelligence, analytics, digital twins, and operational automation. The edge computing investment is represented as:

$$C_{EDGE}(t) = N_{edge} \cdot P_{edge} \cdot S_{latency} \quad (9)$$

where the latency multiplier reflects the increasing complexity of ultra-low-latency processing environments.

G. Dynamic Operational Expenditure

Operational expenditures are time-dependent and exhibit nonlinear growth behavior during the operational lifetime of practical private industrial networks. The yearly operational expenditure can be modeled as:

$$OPEX(t) = O_{energy}(t) + O_{maint}(t) + O_{license}(t) + O_{cloud}(t) + O_{satff}(t) + O_{security}(t) + O_{roaming}(t) \quad (10)$$

This formulation captures the operational impact of energy consumption, maintenance, software licensing, cloud resource utilization, staffing requirements, security operations, and roaming-related expenses. Unlike static models, the enhanced framework assumes that these operational components evolve dynamically over time. As an example, cloud-related costs may increase due to growing AI workloads and expanding edge analytics, while staffing costs may increase due to operational complexity and cybersecurity requirements.

H. Energy Consumption and Traffic Growth

Energy consumption has become an increasingly important economic factor in private industrial networks, particularly as radio densification, edge computing, and AI-driven analytics continue to expand. The model represents energy-related operational cost items in the following way:

$$O_{energy}(t) = E_0 \cdot (1 + g_T)^t \cdot P_e(t) \quad (11)$$

where the operational energy cost depends on the baseline energy consumption (E_0), yearly (t) traffic growth (g_T), and the evolving electricity price P_e. The term $(1+g_T)^t$ represents the compounded growth of network traffic and computational load over time. The reasoning for this term is because industrial private networks can experience progressive expansion, and the formulation reflects the reality that industrial network traffic rarely remains constant during the lifecycle of deployment.

As enterprises continue their digital transformation, the number of connected devices, autonomous systems, sensors, and edge applications typically increases significantly over time. Furthermore, energy consumption does not scale linearly because increased traffic often requires additional radio processing, computational resources, and cooling capacity. For example, if yearly traffic growth is 20% (i.e., $g_T = 0.20$), then after the first year, the term results in $(1+0.20)^1=1.20$, and after the third year $(1+0.20)^3=1.728$, meaning the traffic-related energy demand has increased by approximately 72.8% compared to the initial deployment year.

I. Maintenance Growth and Technology Refresh Cycles

Industrial network infrastructures are subject to technology evolution and aging effects throughout their operational lifetime. The respective maintenance cost can be represented as:

$$O_{maint}(t) = M_0 \cdot (1 + \gamma t) \quad (12)$$

where the M_0 is the baseline yearly maintenance cost at the beginning of operation, and coefficient γ models the increasing operational complexity and maintenance burden associated with aging infrastructure, and t is the operational time in years.

In addition to gradual maintenance escalation, industrial private networks periodically require modernization investment related to, e.g., hardware replacement, migration to newer 3GPP releases, cybersecurity upgrades, AI-assisted optimization platforms, and edge computing expansion. Equation (12) thus assumes that maintenance expenses increase progressively due to factors such as, hardware aging, increasing failure probability, software patching requirements, and other issues that increase over time.

It should be noted that the expression is linear with time because many maintenance-related operational costs tend to increase approximately gradually during the normal operational lifetime before major modernization or replacement occurs.

As an example, if the baseline maintenance cost is USD 100,000/year, and the aging coefficient is γ=0.08, then after the first year, the term results in (1+0.08·1)=1.08, and after the fifth year, it is (1+0.08·5)=1.40 meaning the yearly maintenance cost has increased by 40% compared to the initial operational year.

Thus, the term models the lifecycle behavior in which older industrial network infrastructure generally need progressively greater operational effort and maintenance expenditure over time. The model captures these reinvestment events using discrete refresh cycles:

$$C_{refresh}(t) = \begin{cases} R_f, & t \in T_r \\ 0, & otherwise \end{cases} \quad (13)$$

This reflects the realistic lifecycle behavior of industrial deployments where significant reinvestments may occur every several years.

J. Weighted Multi-Criteria Suitability Assessment

It can be reasoned that pure cost optimization alone is insufficient for industrial NPN deployment decisions because many industrial environments prioritize factors such as reliability, security, isolation, and operational control. For this reason, the enhanced framework introduces weighted multi-criteria assessment. The overall suitability score is:

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$$S = \sum_{i=1}^{n} w_i \cdot p_i \quad (14)$$

where each criterion has relative importance weight and a normalized performance score. The evaluated criteria may include, e.g., latency, reliability, security, isolation, scalability, mobility, roaming support, energy efficiency, operational flexibility, and regulatory compliance.

In this equation, S represents the total suitability score of the deployment scenario, w_i represents the importance weight assigned to a specific evaluation criterion, p_i represents the normalized performance score of that criterion, and n is the total number of evaluated criteria.

The principle of the equation is that each of the selected criterion contributes to the final suitability score according to how important the criterion is for the enterprise, and how well the evaluated deployment fulfills that requirement. Typical criteria may include latency, reliability, security, and scalability, among other selected criteria.

This approach allows comparison of deployment scenarios from a broader industrial perspective. As an example, SNPN deployment may provide superior isolation and security, whereas a network slicing solution may provide lower initial investment and better scalability. The weighting mechanism enables the model to adapt to different industrial verticals whose priorities may differ significantly depending on the operational environment.

TABLE I presents an example of the weighting, assuming an industrial enterprise, which evaluates a private network deployment using three criteria.

TABLE I. EXAMPLE OF WEIGHTING

Criterion	Weight w_i	Performance p_i
Security	0.5	0.9
Latency	0.5	0.8
Cost Efficiency	0.2	0.6

The resulting suitability score becomes $S = (0.5 \times 0.9) + (0.3 \times 0.8) + 0.2 \times 0.6) = 0.45 + 0.24 + 0.12 = 0.81$. Thus, the deployment of this example has a suitability score of 0.81 on a normalized scale between 0 and 1. This behavior allows different industrial verticals to prioritize their most important requirements. For example, a factory automation environment may assign high weight to latency and reliability, whereas an energy utility may prioritize security and resilience, and a logistics deployment may emphasize scalability and mobility.

K. Dynamic Evolution of Enterprise Requirements

The enhanced model assumes that industrial digitalization evolves continuously during the operational lifetime of the network. The number of connected devices is modeled as:

$$N_d(t) = N_0 \cdot e^{\lambda t} \quad (15)$$

where the exponential growth coefficient reflects the pace of industrial automation and digital transformation. This approach reflects realistic industrial evolution trends involving increasing sensor density, autonomous robotics, AI-driven analytics, digital twins, and machine-to-machine communication. As deployment evolves, the respectively increased traffic volume, demand for computing, orchestration complexity, and exposure to cybersecurity create secondary impacts throughout the techno-economic model.

In this Equation, $N_d(t)$ is the number of connected devices at time t, N_0 is the initial number of devices, λ is the digitalization growth coefficient, t is time in years, and e is the base of the natural logarithm. The exponential term reflects that industrial device growth can be assumed to accelerate over time rather than increase linearly. As enterprises adopt more automation, AI-assisted monitoring, robotics, digital twins, and Industrial IoT services, the number of connected endpoints may expand rapidly.

As an example, let us assume that an industrial campus initially deploys 1,000 connected devices in such a way that the yearly digitalization growth coefficient is λ=0.18 and the enterprise expands automation continuously. The device growth becomes:

$$N_d(0) = 1000 \cdot e^{0.180} = 1{,}000$$

$$N_d(3) = 1000 \cdot e^{0.180 \cdot 3} = 1{,}716$$

$$N_d(5) = 1000 \cdot e^{0.180 \cdot 5} = 2{,}460$$

As can be seen, in this scenario, the deployment has grown to approximately 1,716 devices after 3 years, and after 5 years, the deployment reaches approximately 2,460 devices.

It can be assumed that this type of behavior is realistic in industrial environments where additional sensors are gradually deployed, autonomous systems increase, AI-driven analytics expand, production monitoring becomes more granular, and operational systems become increasingly interconnected.

L. Practical Implementation in Excel

The developed Excel-based implementation enables practical evaluation of industrial private network deployment scenarios by allowing users to modify parameter values dynamically and visualize the resulting lifecycle cost behavior. The implementation includes configurable input parameters for deployment architecture, number of users and devices, traffic growth, SLA level, gNB pricing, spectrum costs, cloud elasticity, maintenance escalation, technology refresh cycles, and discount rate.

The resulting calculations generate yearly CAPEX, OPEX, cumulative lifecycle cost, and discounted TCO curves based on the input parameters. This implementation enables comparative evaluation of deployment alternatives and supports sensitivity analysis of key techno-economic variables under realistic industrial deployment assumptions.

It should be noted that the most realistic analysis requires valid commercial values of devices and solutions. Nevertheless, based on a high-level estimate of the input values can provide useful examples of the output. Figure 7 presents such a theoretic assessment for cumulative discounted TCO, and Figure 8 for annual OPEX. The presented curves are normalized setting the SNPN Discounted TCO as the common reference (value 1). In these cases, the basic assumptions are the following: discount rate 8.0% annual; inflation 2.5% annual, device base 50 in year 0, device annual growth 0.2, traffic base per device 0.5, traffic annual growth 0.12. It should be noted that the assessment uses assumptions for the base unit CAPEX and annual OPEX values

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

through best effort estimate covering user devices, gNB units, transmission, spectrum, core network (NFs), applications and services, roaming and interconnect, installation, as well as maintenance and variable cost items.

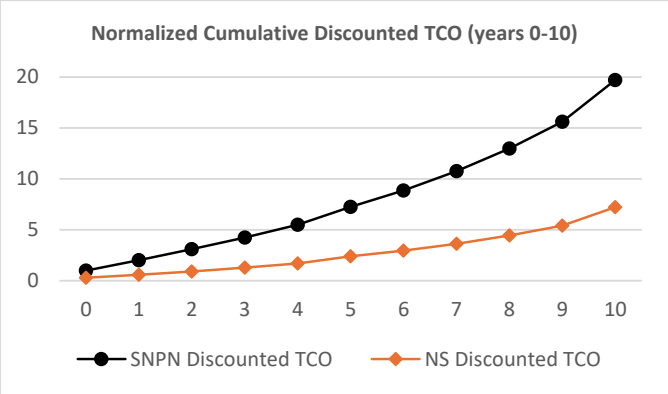


Figure 7. Example of the evolved model's outcome comparing SNPN (reference=1) and NS deployment TCO over 10 years

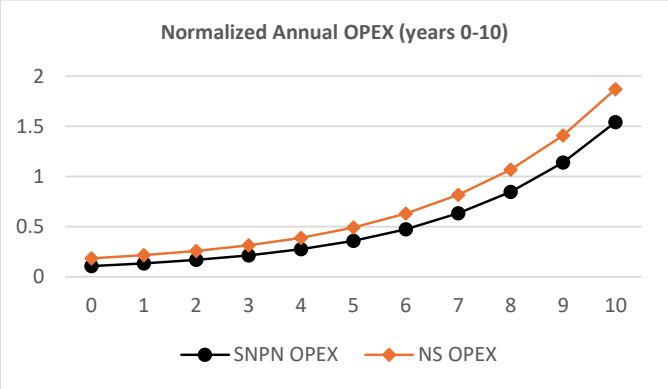


Figure 8. Example of the evolved model's outcome comparing SNPN and NS deployment Annual OPEX over 10 years; the presented values are normalized (the reference is the initial SNPN TCO)

As can be seen from Figure 7 and Figure 8, in this specific scenario, the TCO of SNPN deployment increases faster over time than deploying NS. Nevertheless, the OPEX of the NS deployment model grows faster than the SNPN deployment with the selected parameter values. In this scenario, the NS deployment remains less expensive throughout the lifecycle, but it is possible that the lowest initial investment in some other deployment scenario may turn out to be more expensive over time.

VIII. Summary and Future Work Opportunities

An NPN can serve many verticals and their use cases in a more optimal way than PLMN may be able to, to comply with special requirements for, e.g., hardened security by isolation, or high flexibility for network settings adjustment, for which the pros and cons of enterprise-owned vs. operators' components need to be evaluated. There are variety of deployment and ownership models, so the assessment of the scenarios prior to business decisions for the most feasible deployment and ownership model is beneficial.

The presented method provides the enterprise with a tool to estimate the cost difference of deployment models of interest over time and to assess whether certain variants are acceptable for deployment regardless of projected, potentially higher cost to balance the key requirements of the enterprise considering, e.g., the level of independent network control and security.

This paper presents means for the assessment of the techno-economic feasibility of NPN models and examples of the applicability through base and evolved variants of the models. For the model to perform adequately, insights into realistic OPEX and CAPEX values of the parameters are important. Thus, feedback from NPN proof of concepts and trials serve to calibrate this modeling and helps identify and focus on the evaluation of the most essential cost items.

Private networks are becoming reality, and they provide a functional base for many verticals and use cases to cope with special requirements. Stakeholders considering deployment and use of private networks benefit from adequate platform. As this study shows, even a relatively simple model can support the ecosystem to better understand the differences of the business cases related to a variety of private network models.

Abbreviations

3GPP, 3rd Generation Partnership Project
5GC, 5G Core
5GS, 5G System
ACIA, Alliance for Connected Industries and Automation
APN, Access Point Name
CAG, Closed Access Group
CAPEX, Capital Expenditure
CN, Core Network
CP, Control Plane
DECOR, Dedicated Core Network
gNB, Next Generation NodeB
GSMA, GSM Association
GWCN, Gateway Core Network
ID, Identifier
IIoT, Industrial Internet of Things
IoT, Internet of Things
LAN, Local Area Network
LBS, Location Based Service
LBO, Local Breakout
LPWAN, Low Power Wide Area Network
LWA, LTE-WLAN Aggregation
MBB, Mobile Broadband
MEC, Multi-access Edge Cloud
MNO, Mobile Network Operator
MOCN, Multi-Operator Core Network
N3IWF, Non-3GPP Interworking Function
NAS, Non-Access Stratum
NF, Network Function
NID, Network Identifier
NPN, Non-Public Network
NS, Network Slice
NSP, Network Slice Provider
OPEX, Operational Expenditure
OT, Operational Technology
OTT, Over the Top
PLMN, Public Land Mobile Network
PNI, Public Network Integrated

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

QoS, Quality of Service
RAN, Radio Access Network
RG, Residential Gateway
SA, Standalone
SBA, Service Based Architecture
SBI, System Broadcast Information
SLA, Service Level Assurance
SNPN, Standalone NPN
SUPI, Subscription Permanent Identifier
TCO, Total Cost of Ownership
TN, Trasport Network
TS, Technical Specification
TSN, Time Sensitive Networking
UDM, Unified Data Management
UE, User Equipment
UPF, User Plane Function
URRLC, Ultra-Reliable Low-Latency Communications
WLAN, Wireless LAN

REFERENCES

[1] J. T. J. Penttinen, "On Techno-Economic Optimization of Non-Public Networks for Industrial 5G Applications," in *The Nineteenth Advanced International Conference on Telecommunications (AICT), June 26-30, 2023*, Nice, France, 2023.

[2] GSMA, "Internet of Things," GSMA, 15 December 2020. [Online]. Available: https://www.gsma.com/iot/manufacturing/private-networks/private-networks/. [Accessed 18 May 2026].

[3] 3GPP, "TR 23.700-20: Study on enhanced support of Industrial Internet of Things (IIoT) in the 5G System (5GS)," 3GPP, 2021.

[4] 3GPP, "TS 23.502: Procedures for the 5G System (5GS)," 3GPP, 2026.

[5] 3GPP, "3GPP TS 23.251 Network Sharing; Architecture and functional description (Release 17)," 3GPP, March 2022.

[6] 3GPP, "TS 22.104: Service requirements for cyber-physical control applications in vertical domains," 3GPP, 2024.

[7] 3GPP, "TS 23.501: System architecture for the 5G System (5GS), Stage 2," 3GPP, 2026.

[8] D. Kim, "Non-Public Networks (NPN)," 3GPP, 15 December 2022. [Online]. Available: https://www.3gpp.org/technologies/npn. [Accessed 18 May 2026].

[9] 5G-ACIA, "NPNs for Industrial Scenarios," 5G-ACIA, 20 March 2024. [Online]. Available: https://5g-acia.org/whitepapers/npns-for-industrial-scenarios/. [Accessed 18 May 2026].

[10] 5GACIA, "5G Non-Public Networks for Industrial Scenarios," 5GACIA, July 2019.

[11] GSMA, "5G industry campus network deployment guideline," 10 November 2020. [Online]. Available: https://www.gsma.com/newsroom/wp-content/uploads//NG.123-v1.0.pdf. [Accessed 18 May 2026].

[12] D. Chandramouli, "5G for Industry 4.0," 13 May 2020. [Online]. Available: https://www.3gpp.org/news-events/3gpp-news/tsn-v-lan. [Accessed 18 May 2026].

[13] H. J. Son, "7 Deployment Scenarios of Private 5G Networks," NetManias, 21 October 2019. [Online]. Available: https://www.netmanias.com/en/?m=view&id=blog&no=14500. [Accessed 18 May 2026].

[14] 3GPP, "TS 24.519: Time-Sensitive Networking (TSN) Application Function (AF) to Device-Side TSN Translator (DS-TT) and Network-Side TSN Translator (NW-TT) protocol aspects," 3GPP, 2021.

[15] 5G-ACIA, "Integration of 5G with Time-Sensitive Networking for Industrial Communications," 5G-ACIA, 2021. [Online]. Available: https://5g-acia.org/whitepapers/integration-of-5g-with-time-sensitive-networking-for-industrial-communications/. [Accessed 18 May 2026].

[16] Syniverse, "Managing the Complexities of Connectivity for Enterprises and Operators," Syniverse, 24 March 2022. [Online]. Available: https://www.syniverse.com/blog/connectivity/managing-the-complexities-of-connectivity-for-enterprises-and-operators/. [Accessed 18 May 2026].

[17] M. Dryjański, "Private 5G Networks – Technology, Architecture, and Deployment," Rimedo Labs, 4 January 2021. [Online]. Available: https://rimedolabs.com/blog/private-5g-networks-technology-architecture-and-deployment/. [Accessed 18 May 2026].

[18] Samsung, "Technical whitepaper: network slicing," Samsung, 2020.

[19] GSMA, "Network Slicing: North America's Perspective V1.0," GSMA, 3 August 2021. [Online]. Available: https://www.gsma.com/newsroom/wp-content/uploads//NG.130-White-Paper-Network-Slicing-NA-Perspective-1.pdf. [Accessed 18 May 2026].

[20] GSMA, "Generic Network Slice Template V 7.0," GSMA, 17 June 2022. [Online]. Available: https://www.gsma.com/newsroom/wp-content/uploads//NG.116-v7.0.pdf. [Accessed 18 May 2026].

[21] GSMA, "Mobile economy report 2026," GSMA, February 2026. [Online]. Available: https://www.gsma.com/solutions-and-impact/connectivity-for-good/mobile-economy/wp-content/uploads/2026/02/The-Mobile-Economy-2026.pdf. [Accessed 18 May 2026].

[22] M. Shelton, "Open RAN Terminology," Parallel Wireless, 20 April 2020. [Online]. Available: https://www.parallelwireless.com/open-ran-terminology-understanding-the-difference-between-open-ran-openran-oran-and-more/. [Accessed 18 May 2026].

[23] O-RAN Alliance, "O-RAN: Towards an Open and Smart RAN," O-RAN Alliance, October 2018.

[24] Intel, "LTE-WLAN Aggregation (LWA): Benefits and Deployment Considerations," Intel, 2016. [Online]. Available: https://www.intel.com/content/dam/www/public/us/en/documents/white-papers/lte-wlan-aggregation-deployment-paper.pdf. [Accessed 18 May 2026].

[25] Ericsson, "Fixed Wireless Access handbook 2023 edition," 2022. [Online]. Available: https://www.ericsson.com/en/fixed-wireless-access. [Accessed 18 May 2026].

[26] W. Sullivan, E. Wicks and C. Koelling, Engineering Economy, 16th Edition, Pearson, 2014.

[27] H. Lee and C. Tang, "Modeling the Costs and Benefits of Delayed Product Differentiation," *Management Science,* vol. 43, no. 1, pp. 40-53, 1997.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Design Concepts to Satisfy User Data Accessibility of IoT Devices Postulated by the EU Data Act with Bosch Smart Home as a Case Study

Short Paper

1st Felix Fischer
Faculty Applied Computer Sciences and Biosciences
Hochschule Mittweida University of Applied Sciences
Mittweida, Germany
fische11@hs-mittweida.de

2nd Paul Seidel
Faculty Applied Computer Sciences and Biosciences
Hochschule Mittweida University of Applied Sciences
Mittweida, Germany
seidel6@hs-mittweida.de

3rd Dirk Labudde
Faculty Applied Computer Sciences and Biosciences
Hochschule Mittweida University of Applied Sciences
Mittweida, Germany
labudde@hs-mittweida.de

***Abstract*—The Data Act of the European Union (EU-2023/2854) came into effect on the 11th of January, 2024. By September 2025, the 20-month grace period for adoption will end, after which full compliance with the regulation will be necessary. Implementation depends on multiple factors, including the infrastructure in place currently. Accordingly, a typical operation of a smart home device is chosen as the basis for this paper. Sharing the data with the user is of particular interest to the user in this scenario. We describe different implementations designed to satisfy the requirements established by the EU Data Act. In our view, there exist four distinct design solutions to address those requirements. Access can be granted via the data-generating device, a user-hosted server, a smart home hub or the existing cloud. Design proposals are discussed to explain the benefits and disadvantages of each solution. We arrive at the conclusion that expanding existing cloud Application Program Interfaces (APIs) would be the preferred method for most Internet of Things (IoT) devices. However, sharing data via a local controller could also be an effective solution.**

Keywords-eu data act; data availability; iot; smart home; design.

I. Introduction

This paper is an extension of our previous work on the design concepts to satisfy user data accessability of IoT devices postulated by the EU Data Act, presented at IARIA INTERNET 2025 [1].

Under the General Data Protection Regulation (GDPR) framework, users can currently request access to their data, usually provided within a three-month time frame following the request [2]. However, the upcoming European Union (EU) Data Act will impose stricter requirements on data accessibility [3]. Specifically, the EU Data Act mandates that manufacturers ensure that users have direct access to product and related service data. Manufacturers have to fulfill the requirements before the 12th September, 2025. These data must be provided "by default, easily, securely, free of charge, in a comprehensive, structured, commonly used, machine-readable format" [3].

Wolfgang Kerber already looked at the text of the law from a legislative perspective. He came to the conclusion that the EU Data Act does not achieve its goals. "(a) empowering the users of IoT devices (especially the consumers), (b) unlocking large amounts of IoT data for innovation (for IoT-related services and across sectors), and (c) contributing to a fair sharing of the value from the generated IoT data" are mentioned as objectives. All are missed in his view [4].

In this paper, the technical implementation design is mainly considered. Consequently, the scenario of consideration of this paper will be defined in Section Section II. To fulfill these new requirements, various technical approaches are being considered. These approaches will be judged by selected criteria, which are discussed in Section III. Three primary methods for granting users continuous access to their data have been identified. The first approach is pulling data from the data-generating device, allowing users to extract data directly from the origin. The technical advantages and limitations of this approach are discussed in Section IV. The second method involves enabling users to set up a dedicated server for data collection, allowing the device to transmit data both to a cloud-storage service and to the user-managed server. The feasibility, advantages, and drawbacks of this approach are analyzed in Section V. Extending the previous method, a local server could be provided by other existing smart home devices, which then become a hub for all connected devices. This approach is presented in Section VI. The last option that comes to mind provides user access to data via the existing cloud infrastructure, potentially through an Application Programming Interface (API). This method is detailed in Section VII. This paper concludes with a discussion on the most suitable design choice in Section VIII, followed by a case study of real implementation from Robert Bosch Smart Home GmbH in Section IX. Finally, we close with recommendations for future research in Section X.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

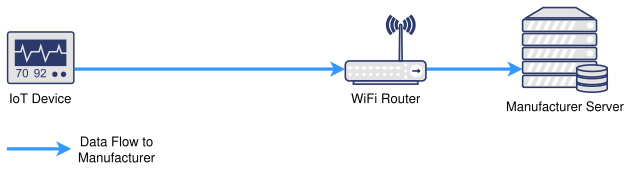


Figure 1. Typical data flow of IoT-devices at home.

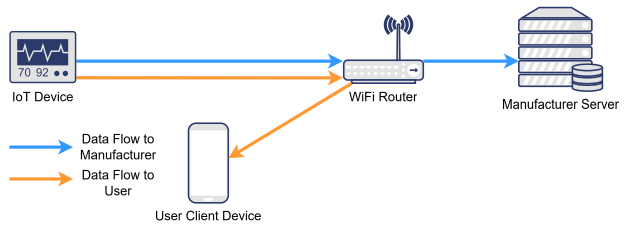


Figure 2. User pulls data from device directly.

II. Scenario under consideration

The term Internet of Things (IoT) device covers a wide range of devices. The considerations in this paper are limited to the classification of IoT devices in the home sector. The term smart home is often used for that. Despite this focus on a small sub-area, it should be possible to cover many other areas affected by the EU Data Act. Therefore, the term IoT device continues to be used here.

For this paper, a typical network setup of an IoT device is assumed. This is shown in Figure 1. The end device connects to a multifunctional device (WiFi Access Point + Switch + Router + Modem), which is commonly referred to as a "WiFi Router". This multifunctional device forwards the recorded data to a server of the respective manufacturer. There, data are collected and prepared for the user, but also for the manufacturer itself. The data flow is unidirectional towards the manufacturer's infrastructure, shown with blue arrows in Figure 1. The user can then typically access a visual representation of the data of their own devices via a web interface or a smartphone app. The access of the raw values collected by the IoT device is not mediationary at the moment.

Depending on the area of application, however, the user does not receive a complete data set. Such visual representations of data are rarely machine-readable without some form of preprocessing, a fact the EU Data Act aims to change. Furthermore, the manufacturer can generate broad knowledge by combining data from different users. In Figure 1 and the following figures, only data streams that are necessary for data provision under the EU Data Act are illustrated.

III. Implementation Considerations

The solutions presented here offer various advantages and disadvantages. In order to be able to evaluate the different approaches, the most important factors to be considered are included.

First, a change in the way of working can require a change on the IoT device itself. Consequently, the IoT device must be able to perform a firmware update. Especially for small devices such as wireless contact sensors for doors and windows or motion sensors, the capability to perform a firmware update is not provided and, therefore, impossible. Thus, a replacement of the hardware would be necessary. This could potentially result in a massive one-time payment.

However, there may also be additional costs due to other factors. This includes, among other things, the implementation costs for changing the software. These costs for programming new functionality come into play both when changing firmware and when adapting software running in the cloud. Especially because the data must be provided free of charge, the cost factor can not be neglected. Additional costs reduce profits. Special attention should therefore be paid to minimizing operating costs. This also includes costs for the provision of network traffic. In addition to computing power, the data stream itself has to be paid for. In the case of one-time paid devices, the running costs for data provision could otherwise wipe out any gained profit.

Finally, the provision of new interfaces increases the attack surface. By choosing a smart design, the attack surface can be reduced. Therefore, possible hacking attacks can be mitigated in advance.

In the analysis carried out here, the respective factors are presented from the point of view of a manufacturer of that IoT device.

IV. Access via the data generating device

The intuitive solution to provide the data according to the requirements listed above is to provide the user with an interface on the data-collecting device itself. As can be seen in Figure 2, the user can collect the data at their own will using this new interface (orange arrows). The current data flow to the manufacturer's server remains untouched (blue arrows). Consequently, the server software does not have to be modified. This means that there are no additional costs for the manufacturer due to the operation.

However, the device needs to be updated with a new firmware, which not all IoT devices support. This means that this option cannot be technically implemented for these devices.

Another positive aspect is that the user can be provided with an arbitrarily dense temporal resolution of the data. In addition, it can be positively emphasized that the data are immediately available to the user. However, most IoT devices lack sufficient storage capacity for a continued retention of the collected data. This means that the user has to query their data at an increased frequency.

IoT devices usually work according to the push principle. This means that they send data and are not actively waiting for a connection. This allows them to switch to a so-called deep sleep mode after sending the data, saving power. In this mode, the Central Processing Unit (CPU) switches to a very economical co-processor, which only waits for an expiring timer. As soon as the time elapses, the CPU is woken up again. This technique can massively reduce electricity consumption

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

and, therefore, the operating cost of a device. However, if the IoT device will await for a connection from the user, it cannot switch to deep sleep mode. As a result, electricity consumption increases. Thus, this will render small battery-powered appliances unusable within a few days or even hours.

In the event that electricity consumption does not have a restrictive effect, this method can be used to provide the data cost-effectively. This is especially the case if a so-called smart home controller is used. Such a controller is mainly present when a transmission technology that differs from WiFi is implemented and the device cannot communicate directly with the WiFi router. Thus, the controller acts as an access point for the alternative protocol. Common alternative protocols include ZigBee, Digital Enhanced Cordless Telecommunications (DECT), Bluetooth Low Energy (BLE), and Matter. In such a case, the data could be cached on the controller. Figure 3 illustrates, how data can be accessed from there, according to the solution described above in this section. The orange arrows show the data collection using the controller cache. The blue arrows illustrate the unchanged data delivery to the manufacturer server via the WiFi Router. For this implementation, an update of the smart home controller is required.

V. Access via User Hosted Cloud

One way to change the method from Section IV to a push method is to have the IoT device send the data twice: First, as before, to the manufacturer's server (blue arrows) and second, to a local server in the user's network (orange arrows). Strictly speaking, the server operated by the user can also be connected to the Internet. From this second server, the user can access the data at any time and without any other restrictions (green arrows). Figure 4 illustrates this setup.

By changing the access procedure from pull to push, the IoT device can switch back to deep sleep mode between transmission phases. Therefore, the energy consumption will have increased somewhat, but not drastically changed. Because transmission technologies, that are already in use are implemented, no hardware replacement is necessary. However, the firmware must be adapted in the sense that the IoT device sends the data twice and the user can enter to which server the data should be sent the second time. Thus, an update of the firmware will be necessary.

After the data have been transferred, the user has full control, but also full responsibility for their data. Especially

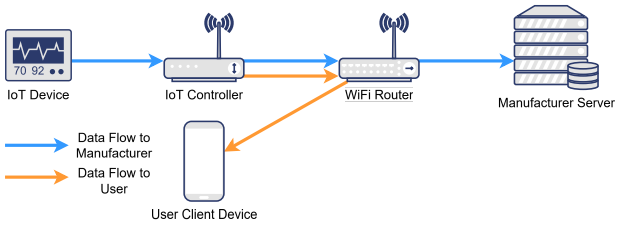


Figure 3. User pulls data from controller.

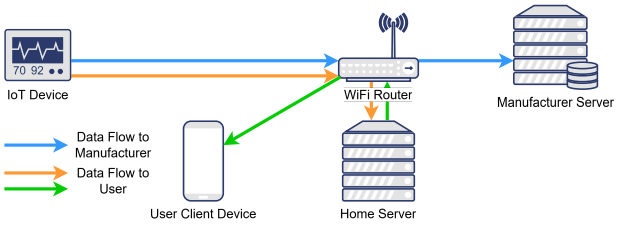


Figure 4. IoT device sends data twice and user accesses data via his local server.

from a data protection perspective, this fact is relieving for the manufacturer. Likewise, there are no further costs for the manufacturer's cloud operation. The management of the data, for its part, remains untouched.

It can be assumed that users expect that server software must be provided by the manufacturer for such a scenario. This must be additionally programmed and maintained, thus causing additional costs. However, these are likely to be quite limited, as the software is based on the already existing server software of the manufacturer's server.

The user has to operate their own server for this approach. It is unclear how this will be assessed as an obstacle to compliance with the EU Data Act by a judge. Making the data accessible in this way could be seen as a contradiction to making it available free of charge. Thus, there is a possibility that this approach will not be classified as meeting the requirements of the EU Data Act.

VI. Access via Smart Home Hub

Building on the approach in the previous Section V, existing hardware can be used as a local server. Some examples include Network Attached Storage (NAS) systems or smart home control solutions such as Alexa that are already in households [5]. Therefore using a pre-existing hub like a NAS or Alexa as the "user-hosted server" as described in Section V. The previous approach could be integrated into these systems. However, this results in the same advantages and disadvantages, except that no new hardware has to be employed.

VII. Access via Existing Cloud

The last approach builds on currently existing infrastructure. In this way, the data are already transferred to the manufacturer's server. Accordingly, it makes sense to set up direct access to the data for the user. As a result, an additional API will be provided or the existing one will be expanded. Figure 5 visualizes this implementation variant. The IoT device transmits data in the way currently used, illustrated by blue arrows. Afterwards, the user accesses the data from the manufacturers server, shown with orange arrows.

If the manufacturer currently does not store its data on its own server, this approach offers a clear disadvantage. The data is stored at the manufacturer's expense. However, the data are of no use to him. The manufacturer's server would, therefore, act as the user's free cloud storage. However, the authors are

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

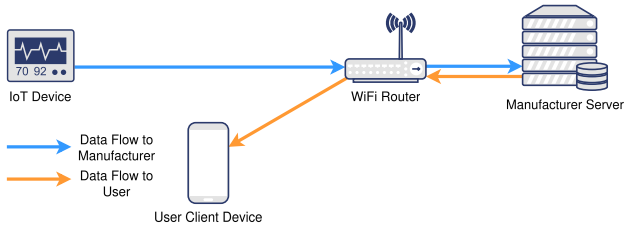


Figure 5. User pulls data from server of manufacturer.

not aware of any smart home device where this described case is observed. Manufacturers always have access to the data.

Expanding existing cloud access offers the clear advantage that no changes or updates of the Smart Home Network or the IoT devices therein are necessary. No firmware update or hardware replacement is required. Since there is no change to the existing hardware, there is no additional energy consumption.

In addition, an extension of server access to user data can be implemented comparatively quickly. Supplementary changes can be continuously integrated in the future. However, this method of implementation generates additional costs in computing power and network traffic on the server.

Access to the data for third parties could be realized via the same interface. For example, the user could grant third parties access to his data by releasing an API token. On the other hand, the manufacturer could also share the collected data with third parties via the same API.

Finally, it can be positively emphasized that this form of transition to the fulfillment of the EU Data Act is not perceptible to the user.

VIII. Discussion

Table I summarizes the analysis of the previous Sections. We provide an impact classification in the discussed aspects.

TABLE I
Summary of Design Concepts

Aspect	Device (S. IV)	User Cloud (S. V)	SH HUB (S. VI)	Cloud (S. VII)
Update needed on IoT device	yes	yes	yes	no
Implementation costs	medium	medium	medium	low
Operating costs for company	none	none	none	cloud capacity
Operating costs for user	electricity	hardware, electricity	hardware, electricity	none
power consumption of IoT device	high	medium	medium	low
Increased attack surface company	no	no	no	yes
Increased attack surface user	yes	yes	yes	no

Since every approach except the one in Section VII requires a firmware update, it should first be clarified whether a firmware update is technically feasible. Without the possibility of adapting the software on the IoT devices or the associated controller, the data provision can only be realized via an API on the manufacturer's server. In particular, if the additional cost that can be expected from making the data available via an API from the cloud is negligible, this implementation should be chosen. It represents the smallest change compared to the status quo.

If the positive factors from Section VII outweigh the negative factors and it is technically feasible, we recommend the implementation from Section IV. This is especially true if a controller is already in use, as shown in Figure 3. Here, it should be checked whether all user-related data are already provided. If this is the case, no changes are necessary to comply with the EU Data Act. Otherwise, the EU Data Act can be implemented cost-effectively by updating and caching the data on the controller with this procedure.

There are valid reasons for the other implementation options. However, it should be considered whether the two options mentioned above are not more suitable for one's own starting position. In particular, due to the uncertain legal situation described in Section V.

IX. Case Study Robert Bosch Smart Home GmbH

The deadline of the grace period ends on the 12th of September 2025. Robert Bosch Smart Home GmbH provided APIs for the user in advance to the EU Data Act. In the following, Robert Bosch Smart Home GmbH extended these APIs to fulfill the requirements given by the EU Data Act. We use this example to analyze the methods used to realize the EU Data Act demands. As a reminder, the requirements for data provision should be mentioned again. These data must be provided "by default, easily, securely, free of charge, in a comprehensive, structured, commonly used, machine-readable format" [3].

A. Requirement to be provided by default

Robert Bosch Smart Home GmbH uses github to document the API available in the Smart Home Controller (SHC). The public git repository https://github.com/BoschSmartHome/bosch-shc-api-docs can be opened by any user, even without an github account [6]. In addition, Robert Bosch Smart Home GmbH provides a website with an in-depth documentation of the Representational State Transfer Application Programming Interface (REST) API. The REST-API documentation can be reached by using URL https://local.apidocs.bosch-smarthome.com/ [7]. Both are lizenced under the Attribution-NonCommercial-NoDerivatives 4.0 International Lizence. Therefore, every customer could access the data cached in the smart home controller. Further more, the API is available without the need for changes on the controller by the user and therefore, active by default.

B. Requirement to be provided easily

In the github documentation at https://github.com/BoschSmartHome/bosch-shc-api-docs/tree/master/postman, Robert Bosch Smart Home GmbH includes an example of applying the REST API by utilizing the API platform

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Postman [8]. By doing so, even unexperienced users can make use out of the provided interface. Furthermore, Robert Bosch Smart Home GmbH wrote down best practices to elevate the benefit of unexperienced users.

C. Requirement to be provided securely

The connection between the collecting device and the SHC is secured by a Transport Layer Security (TLS) Layer. The version 1.2 of TLS is implemented. Client authentication is provided using a self-signed X.509 certificate. This must be uploaded by the user to the SHC together with the password used in the SHC. Thus, prevent unauthorized addition of new clients. The user is responsible for the integrity of the private key used in the certificate. Each client will be registered separately with different certificates, improving the integrity of the system. A compromised client could be denied access in the future without affecting the availability of other clients' connections. The used certificate has to utilize an 2048 bit RSA key. This is still widely used in the Internet, but at least 3000 bits are required by the German Federal Office for Information Security, Bundesamt für Sicherheit in der Informationstechnik (BSI) [9]. TLSv1.2 still ensures confidentiality and integrity of the transferred data, but TLSv1.3 introduces new security mechanisms. Therefore, the provided REST-API could be improved using TLSv1.3.

D. Requirement to be provided free of charge

Access to the git repository and the documentation provided is available free of charge. In addition, data access to the local SHC is possible at any time without payment. The SHC is necessary for regular usage of the Bosch Smart Home Devices. Thus, there are no additional hardware costs for the user. However, the customer pays for the use through the power consumption of the network transmission in the local network and the operation of the SHC.

E. Requirement to be provided in a comprehensive, structured, commonly used, machine-readable format

Data collection is done via a REST-like API. Representational State Transfer, or REST for short, is defined as a uniform network interface based on a web server architecture. When it comes to the requirements for an interface, REST focuses primarily on machine-machine communication. Data structuring under REST can be ensured with various data formats such as Extensible Markup Language (XML) or JavaScript Object Notation (JSON). Robert Bosch Smart Home GmbH chose the latter for the implementation of the API on the SHC. The data is transmitted in the formatting of a JSON. JSON is a widely adapted format for data exchanges this kind of. The name JavaScript Object Notation suggests that this data format can only be used in the JavaScript programming language. However, this is not the case. JSON can be used with all common programming languages. It is one of the most commonly used data formats for exchanging data between machines and therefore a great choice of Robert Bosch Smart Home GmbH to make the API usable for the most people feasible. Thus, the JSON data format meets the requirement of comprehensive, structured, commonly used, machine-readable format.

F. Assessment of the realization by Robert Bosch Smart Home GmbH

In addition to the requirements of the EU Data Act, Robert Bosch Smart Home GmbH provides some recommendations for optimal implementation with best practice. These include, for example, Long Polling Subscribe for an up-to-date notification of changed values without delay. "Long Polling is a mechanism in which the client makes a request, and the server keeps the connection open until there is new information available. Once available, the server responds by sending the new information to the client and closes the connection afterwards. As soon as the client receives the new information, it immediately sends another request and the process is repeated." [10]

In summary, it can be said that Robert Bosch Smart Home GmbH decided to implement the EU-Data-Act like described in Chapter IV. The Figure 3 shows the implementation graphically. In this specific case, the orange arrows in Figure 3 symbolize the access to the REST API on the SHC. Since the Robert Bosch Smart Home GmbH provides all the recorded data via the interface and also enables active control, operation is technically feasible completely independently of the Robert Bosch Smart Home GmbH cloud. As a result of the selected implementation, Robert Bosch Smart Home GmbH will not incur any other running costs in addition to the development costs. On the contrary, costs could even be saved if users only used their own infrastructure with the new Rest API. To do this, however, Robert Bosch Smart Home GmbH would have to make it possible to switch off data transmission to the cloud. This is not currently the case.

Users of the API can suggest changes through PULL requests in the git repository and are thus in direct exchange with developers. Users are already creating issues on identified errors and missing documentation. This gives Robert Bosch Smart Home GmbH insight into the use of its customers' smart home hardware and can incorporate this into the design of future products.

The open documentation enables networking with cross-platform systems such as openHAB. As early as November 2020, Robert Bosch Smart Home GmbH devices began to be integrated into openHAB [11].

OpenHAB is a platform for linking a wide variety of smart home devices from various manufacturers on a common control unit. OpenHAB uses Robert Bosch Smart Home GmbH's REST API to integrate it and created matching bindings in the smart home appliance [12]. We have successfully integrated contact sensors, motion detectors, smoke detectors, smart sockets and water detectors from Robert Bosch Smart Home GmbH into OpenHAB in a smart home. The data can be collected without having to make any changes to the device. Thus, it was checked that the data is retrievable by default. Not only can data be read, but requests for actions can also

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

be transmitted. For example, the smoke detector can also serve as a siren for opening contact sensors. This was also successfully tested. In addition to the IoT devices mentioned above, Robert Bosch Smart Home GmbH also offers other technology for smart hones. These are not available in our laboratory. As a result, it was not possible to fully check whether the requirements of the EU Data Act were met for all Robert Bosch Smart Home GmbH devices.

X. Conclusion and Future Work

In summary, it can be said that there are different approaches to making data available to the user according to the EU Data Act. These different approaches allow a solution to be cleverly adapted to the current situation. In the case study of the Smart Home devices, Robert Bosch Smart Home GmbH has chosen the implementation as described in chapter IV with the special case of an local controller. Robert Bosch Smart Home GmbH provides a REST API for data access and direct device control on the local Smart Home Controller.

This work can be followed by demonstrations of implementation proposals of the designs shown here. For example, the approach from Section VII alone could be implemented in many ways. Depending on which framework is used on the server, the corresponding implementation changes.

For the design of various solutions in this paper, the scenario described in Section II was limited. Thus, the designs presented here do not basically cover all scenarios. In particular, we consider connected cars and cloud services to be worthwhile, advanced fields of research for a submission regarding the EU Data Act.

In this study, only implementations for private devices specifically in smart homes were discussed. In the business-to-business sector, there are different starting points and different requirements are placed on the finished product. This allows for other perspectives on possible solutions.

Acknowledgment

This paper was funded by the European Union and the Free State of Saxony (Germany).

Kofinanziert von der Europäischen Union

Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des vom Sächsischen Landtag beschlossenen Haushaltes.

References

[1] Felix Fischer, Paul Seidel, and Dirk Labudde. “Design Concepts to Satisfy User Data Accessibility of IoT Devices Postulated by the EU Data Act”. In: *INTERNET 2025: The Seventeenth International Conference on Evolving Internet*. Lisbon, Portugal, Mar. 2025, pp. 1–4. ISBN: 978-1-68558-234-0.

[2] Council of European Union. *Council regulation (EU) no 2016/679*. 2016. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679 (visited on 06/01/2026).

[3] Council of European Union. *Council regulation (EU) no 2023/2854*. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=OJ:L_202302854, Article 3, last access: 2026.06.01. 2023.

[4] Wolfgang Kerber. “Governance of IoT Data: Why the EU Data Act Will not Fulfill Its Objectives”. In: *GRUR International* 72.2 (Oct. 2022), pp. 120–135. ISSN: 2632-8550. DOI: 10.1093/grurint/ikac107. eprint: https://academic.oup.com/grurint/article-pdf/72/2/120/49174428/ikac107.pdf. URL: https://doi.org/10.1093/grurint/ikac107.

[5] Shubham Jain. “Amazon Alexa Enabled Smart Wi-Fi Switch”. In: *Journal of Multi Disciplinary Engineering Technologies* 13.1 (July 2019). eprint: http://jmdet.com/wp-content/uploads/2019/09/JMDET_13_1_15-NEW.pdf.

[6] *Bosch Smart Home Controller Local API*. Robert Bosch Smart Home GmbH, May 2023. URL: https://github.com/BoschSmartHome/bosch-shc-api-docs (visited on 06/01/2026).

[7] *Bosch Smart Home Local API for Main Resources*. Robert Bosch Smart Home GmbH, Sept. 2025. URL: https://local.apidocs.bosch-smarthome.com/ (visited on 06/01/2026).

[8] *Postman*. Postman Inc., Sept. 2025. URL: https://www.postman.com/ (visited on 06/01/2026).

[9] *Technische Richtlinie - Kryptographische Algorithmen und Schlüssellängen*. BSI TR-02102-1. Jan. 2025. URL: https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102.pdf (visited on 06/01/2026).

[10] *Best Practice*. Robert Bosch Smart Home GmbH, Dec. 2022. URL: https://github.com/BoschSmartHome/bosch-shc-api-docs/tree/master/best_practice (visited on 06/01/2026).

[11] Christian Oeing. *More information about the provided device data*. Nov. 2020. URL: https://github.com/BoschSmartHome/bosch-shc-api-docs/issues/40 (visited on 06/01/2026).

[12] *Bosch Smart Home Binding*. OpenHAB, Sept. 2025. URL: https://www.openhab.org/addons/bindings/boschshc/ (visited on 06/01/2026).

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Information-Centric Wireless Sensor Network Framework for Smart Agriculture: Concept Proposal, System Design, and Preliminary Evaluation

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

***Abstract*—Widespread adoption of smart agriculture is crucial not only for food security but also because agriculture is a significant industry outside of urban areas. Smart-agriculture applications must collect, share, distribute, and manage video and image data in real time. Our previous studies developed a network that integrated information-centric networking, wireless sensor networks, and millimeter-wave-band (wireless) communications. We recently conducted proof-of-concept experiments in a test field for typical smart-agriculture applications on a new platform that combines the previous platform with visual sensor nodes and artificial intelligence, called a millimeter-wave-band communication information-centric wireless visual sensor network. This paper describes the platform, the system design required to implement it, and simulation results on the wireless network characteristics of the system in the target environment. In addition, this paper identifies technical issues regarding implementation, evaluation, deployment, and demonstration on real farms and strategies for their solution.**

Keywords—Information-centric networking; millimeter-wave-band wireless communications (mmWaves); platform deployment for smart agriculture; wireless sensor networks

I. INTRODUCTION

Internet connectivity is accelerating as new online social networking, live streaming, video sharing, multi-user online gaming, and Internet of Things (IoT) applications emerge. These applications are the elements of smart cities, and the Wireless Sensor Network (WSN) is a critical enabler technology. Internet semantics have shifted from host-centric to content-centric, as can be seen in the emergence of content delivery networks, web caching, and peer-to-peer networking. A paradigm shift is sorely needed toward Information-Centric Networking (ICN), which is a potential network architecture for the next-generation Internet [2]. Furthermore, ICN and WSNs concepts can be integrated to yield Information-Centric Wireless Sensor Networks (ICWSNs) framework.

ICN names individual data rather than addresses. Publisher (sensor) nodes generate sensing data as named data and commit it to the ICN architecture. In the data-retrieval procedure, the subscriber (user) node sends an 'interest message' (a data-acquisition request), and the nearest node replies with the requested (cached) data. If the node does not have the data, the interest message is forwarded to other nodes. To implement this procedure, ICN nodes have three databases: Pending Interest Table (PIT), Forwarding Information Base (FIB), and Content Store (CS). The PIT registers the interest messages, including in- and out-of-interface names enabling requested data to be forwarded to the data requester by back-tracing. If the ICN node has the data, the node replies with its data; otherwise, it relays the interest message. The CS manages cached data; i.e., nodes can use it to check whether the requested data is available. During data forwarding, intermediate nodes along the routing path copy and store the data in their cache memories.

Thanks to its in-network caching mechanism, ICN can provide efficient data sharing without relying on its location; as a result, network traffic can be significantly reduced in an autonomous, decentralized network environment. In general, in-network caching schemes are classified into on-path and off-path methods [3]. Typical ICN platforms support on-path methods. In on-path caching, the intermediate nodes along the routing path perform caching. On the other hand, in the off-path method, nodes outside the route cache the data. Here, the off-path method can be effectively investigated in a multicast network environment; ICWSN is a suitable scenario due to its distributed, decentralized network, and broadcasting and overhearing communications. The use of coding to improve the effectiveness of the off-path caching is under consideration. For example, the most well-known technique, Network Coding (NC), can improve network throughput and scalability in multisource and multicast scenarios. The basis of Superposition Modulation (SM) and Hierarchical Modulation (HM) has been investigated as a way for code-based caching in the physical layer to superimpose signals during modulation [4].

At the same time, these technologies are critical components underpinning the smart-city-as-a-service frameworks. Among these frameworks, agriculture is extremely important for national food security, but the agricultural workforce is aging and shrinking. Here, the deployment of information and communication technologies can help to solve these problems [5]. Smart agriculture has many applications, including anti-theft systems for agricultural products, pest-control systems (trapping traps), farm-management robots and tractors (e.g., automatic weeding, cultivating, irrigating, and fertilizing), plastic greenhouse management systems, and disaster-prevention systems. Introducing a smart-agriculture platform can improve and stabilize productivity in response to various changes in circumstances. In Japan, there are generally two styles of agricultural business: large-scale commercial and

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

small-scale family farms. Large-scale farms aim to supply food to domestic or international markets, whereas, small-scale farms protect land in the countryside and serve as hubs supporting local industry, the economy, and the community.

Previous studies on small-scale family farms identified two barriers to the deployment of smart agriculture [1]. One is that some farmers often have no idea how to use remote-sensing data effectively; thus, they think that smart agriculture systems are costly and yet do not directly increase productivity. In other words, as the cost of agricultural materials rises (from global inflation), these farmers would rather focus on managing financial and human resources and regard the purchase of seeds, seedlings, fertilizer, and plastic greenhouses as factors that directly affect agricultural productivity. On the other hand, some young farmers are interested in using smart agriculture to improve their work environment. Given these findings, we consider that a common platform should be provided rather than a separate system for each application. Table I summarizes the requirements of common technologies for smart-agriculture applications on the basis of our research and experience. The platform requires sufficient flexibility to support diverse application workflows, including Quality-of-Service (QoS)-required and streaming service workflows for real-time video and field images, that are distinct from the analyzed sensing data.

In this paper, we present the details of study items in our ongoing research and development project [1][6]. In particular, we identify and discuss potential challenges and examine the results of simulations to make preparations for a practical implementation.

This paper presents the following research contributions.

- Considerations based on the findings of previous theoretical studies, the protocol design, and a preliminary evaluation to determine the parameters of the wireless network.
- Expected situations and their solutions, including key technologies for ICWSN and for supporting smart-agriculture and associated applications.
- Identifying specific scenarios regarding the social implementation of ICWSN, including IoT frameworks based on the ICN platform. To the best of the author's knowledge, there are few studies on this aspect.

The remainder of this paper is organized as follows. Section II provides related work. Section III describes the development items needed to complete a new ecosystem as an outcome of the (future) research project and ongoing work. Section IV presents the research contributions to future wireless technological development. Section V describes the protocol design. Section VI presents the numerical results of a preliminary evaluation. Finally, Section VII summarizes the outcomes and future perspectives of our project.

II. Related Work

Bilal et al. [7] proposed a solvable linear network-coding scheme for the ICN architecture, and Llorca et al. [8] formulated the ICN-based multicasting model and analyzed its use. using it. Ramakrishnan et al. [9] adopted an NC-based ICN scheme to video streaming and Yang et al. [10] used it as an energy-saving technology in WSNs. Yin et al. [11] investigated the issue of cooperative communications where two adjacent nodes mutually recover lost packets in a lossy wireless broadcast channel. You et al. [12] examined joint use of the NC and HM techniques to provide asymmetric data rates. In particular, Tang et al. [13] presented combined HM and physical-layer NC techniques for downlink asymmetric two-way relay channels. Tharranetharan et al. [14] proposed to combine the HM and NC techniques by using a bit-interleaved coded modulation method to improve the bit-error probability.

TABLE I. COMMON TECHNOLOGIES FOR SMART-AGRICULTURE APPLICATIONS

Applications	*Requirements for common technologies*
Theft prevention	Monitoring using security cameras
Pest control	Monitoring of local circumstances
Field management	Remote control based on real-time video
Greenhouse management	Monitoring for damage and collapse
Weather and disasters	Remote monitoring of fields and rivers

There have been many studies on smart agriculture, and smart-agricultural equipment is available from many vendors [15]. Their proposals and solutions are primarily aimed at large-scale farmers; i.e., they are not cost-effective for small-scale farms to deploy. The wireless networks underpinning real-time video and image applications require large capacity and low latency due to the forwarding of streaming data. The proposed scheme uses IEEE 802.11-compliant Wireless Local Area Networks (WLANs) because they can easily integrate the millimeter-wave (mmWave), microwave, and sub-gigahertz-wave spectrums. There have been many studies on the construction of outdoor WLANs for smart agriculture [16]; however, there are only a few studies involving the mmWave spectrum; e.g., a field trial of mmWaves was conducted in Georgetown, Malaysia [17]. The ecosystem is typically implemented on the basis of cloud-native or edge (fog) designs in which the sensing data are centralized in the cloud area (or the data are distributed in the edge-node storage). In contrast, the proposed scheme adopts the ICN design in which all data are distributed among the edge-side nodes. The related studies introduced ICN to edge networks, particularly wired networks; the proposed scheme expands the scope to wireless network.

III. Research and Development Items for Smart-City Deployment for Smart Agriculture

This section presents the research and development challenges of deploying a smart-agriculture platform based on the proposed elemental technologies.

A. *Key technology: Cooperative communications scheme based on coded off-path caching method*

Figure 1 provides an overview of the proposed scheme, including the research and development items. Our previous studies preliminarily investigated a novel NC-based

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

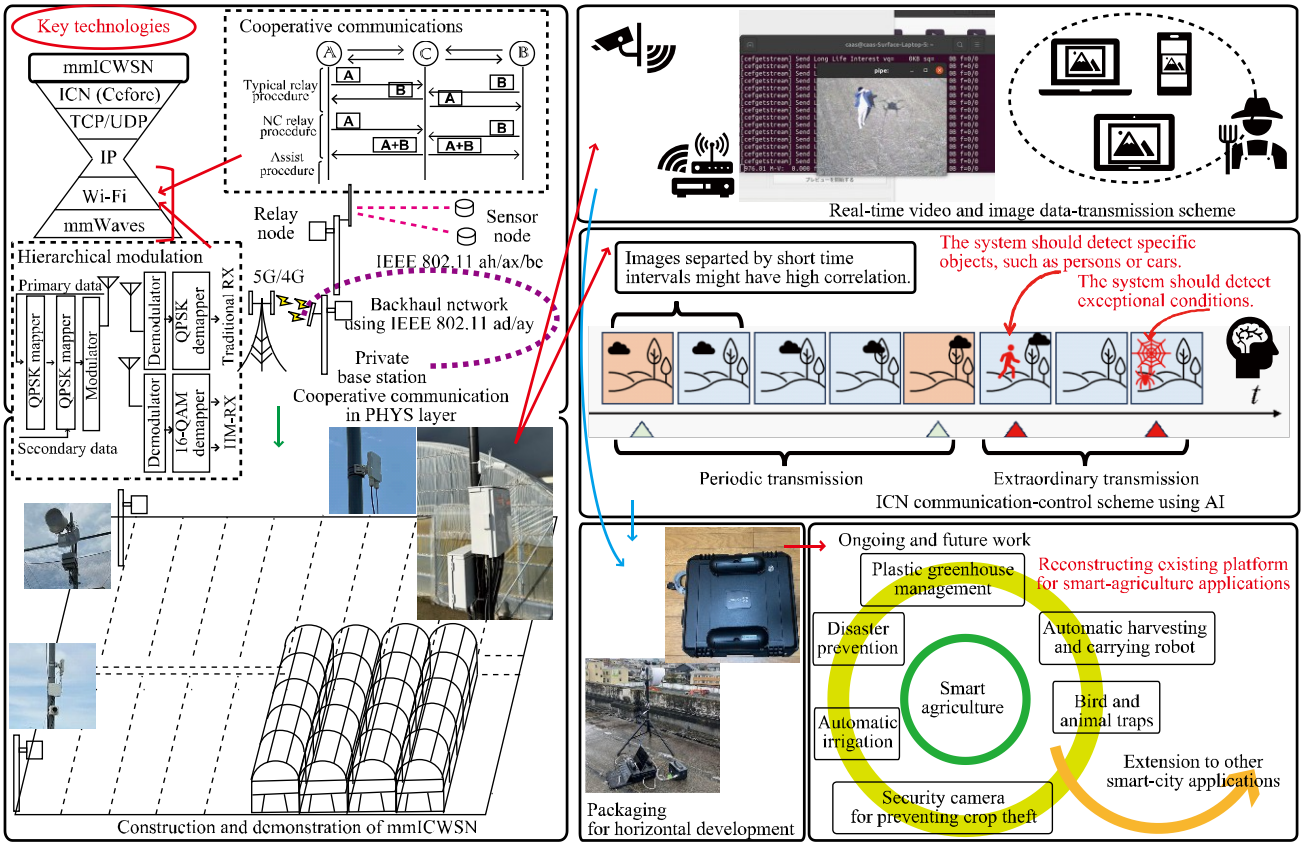


Figure 1. Overview of research and development items for smart-city (smart-agriculture) applications

cooperative communication scheme with constructive interference [4]; this scheme used the HM technique to reduce overhead, i.e., eliminate one of the data transmissions, during the cooperative phase. The transmitter-side node encodes two data bits using the Exclusive OR (XOR) operation bit-by-bit. In contrast, the receiver-side node decodes all the received data once. In this way, NC offers both distributed data storage and efficient multipath solutions to ICN. HM divides the data into two or more streams with different degrees of protection. All receivers can retrieve the most important data bits (primary data stream). The less important data bits (secondary data stream) can be received by some receivers that can decode the HM-based superposition data or by receivers in better reception conditions, such as those near the transmitter. For example, the combination of 4 and 16-Quadrature Amplitude Modulation (QAM) provides a two-level hierarchy. In this modulation scheme, the primary and secondary data streams are assigned to upper and lower bits, respectively, at the bit-to-symbol mapper. All receivers can demodulate (detect) the primary data in the received signal by using the symbol-to-bit demapper in 4-QAM (i.e., QPSK; Quadrature Phase-Shift Keying), while HM-available receivers can demodulate both primary and secondary data by using 16-QAM.

B. *Development and deployment of the ICWSN platform*

In our previous studies, we developed a reliable and self-organized testbed for co-creative smart cities [18]. The testbed is equipped with a zero-touch device designed for extreme outdoor conditions. Namely, the inside and outside of the device are connected via water-resistant connectors for a waterproof design, and mechanical components, such as motor drives and cooling fans, are omitted for greater reliability. According to feedback we often receive from farmers testing the equipment, the device remains unsuitable for outdoor environments because it requires a commercial power supply. Accordingly, a commercial power supply will be needed at the locations where the system is deployed; i.e., this issue is not considered critical.

The testbed will be deployed on an actual farm as a demonstration of the ICWSN platform. So far, the efficiency and fault-tolerance characteristics have been evaluated through computer simulations [19][20]. This paper focuses on the feasibility of achieving such advantages by presenting a protocol design prior to a real-world evaluation. The network of the ICWSN platform is composed of WLANs based on the IEEE 802.11 standard, in which multiple license-free radio-frequency bands, such as 920 MHz, 2.4, 5, 6, and 60 GHz, are integrated to provide sufficient coverage. The mmWave of our platform use Terragraph for the backhaul network (core

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

network) between agricultural fields and access points. Terragraph was developed by Meta (Facebook) as a wireless alternative to optical fiber and can be used to construct a 60-GHz-band wireless mesh network platform based on IEEE 802.11 ad/ay [17]. In our platform, a BeMap MLTG-360 will be used as the Distributed Node (DN), and MLTG-CN (standard type) and MLTG-CNLR (long-distance type) will be used as the Client Node (CN) [21]. Our previous network-construction experiment, which involved a field test on the testbed, demonstrated medium- to long-term operations. In addition, an experiment on long-distance mmWaves was conducted over a 1-km distance in a line-of-sight environment (in Nogata City Fukuoka, Japan) [22].

C. *Real-time visual data-transmission scheme*

This section describes the key technology for transmitting real-time visual data, such as video and images, via the ICWSN platform. The proposed data-transmission scheme is designed as a comprehensive information system that includes farmers, installers, and other relevant people. Our previous study [23] conducted a field test that included an evaluation of network performance and real-time video streaming testing between the ground node (MLTG-360) and the aerial node (MLTG-CN). The aerial node was a mooring node at a 5-m altitude, and it was implemented using an industrial drone. Cefore [24] was used on the test devices as an open-source ICN platform compatible with CCN/CCNx in a Linux (Ubuntu) environment. Using Cefore's standard commands, video data can be transmitted in real time. However, it was found that the video sometimes froze even when network conditions were stable. This issue will be tackled in the future, but the nodes will initially be placed on the ground.

Furthermore, the ICWSN platform will be developed to account for human-centered aspects pointed out in interviews and questionnaires with farmers, equipment installers, and government and organization staff. In particular, the previous implementation used Cefore's standard commands via the Character-User-Interface (CUI)-based terminal, but study's participants indicated that it was not user-friendly. To improve accessibility and usability, the ICWSN node's dashboard should instead be visualized based on Graphical User Interface (GUI), such as personal-computer, tablet-terminal or mobile-phone applications. Eventually, the above mentioned ICWSN solutions should be implemented and its effectiveness evaluated using the ICWSN platform in the on-site farms.

D. *Wireless communication control based on AI for ICN*

The wireless network and its infrastructure require high capacity, low latency, and high reliability. Its architecture is thus based on IEEE 802.11-compliant WLANs. WLANs can support different radio-frequency bands for short-, medium-, and long-distance coverage; however, bottlenecks remain. It is difficult to transmit all the visual data collected by the platform, so we will try to overcome this issue by using Artificial Intelligence (AI)-based ICN controls. A general characteristic of visual data is that nearby pixels are often similar, and their color and intensity variations depend on the time and location. Since the visual data generated by smart-agriculture applications does not change significantly over short intervals, the data can be downsampled (compressed). Moreover, the transmission interval can also be adjusted using an event as a trigger. When executing the trigger, if the correlation value between adjacent images along the time axis changes significantly or the AI detects an object, the AI-based node devices can be sent as an exception.

Our preliminary study [6] analyzed photographic data from an actual farm and found a strong correlation between image data nearby on the time axis. We also found that the correlation value decreased over time. In addition, by using YOLO [25], a well-known object-detection AI, we verified that persons and vehicles could be detected with reasonable accuracy. Here, we used the standard trained YOLOv11x model. However, it was not sufficiently accurate and detected many false positives. To reduce implementation costs, we plan to combine the Python-based Cefpyco provided by Cefore and a Python-based AI. AI is used at edge-side nodes to reduce network traffic within the network and to the cloud. Regarding the relationship between the accuracy of AI-detected events and network traffic, higher accuracy does not necessarily lead to a reduction in traffic. If only carefully selected higher-quality data is sent to the network, it will reduce network congestion. A detailed evaluation in the AI environment described in this section will be carried out once the mmWave-band ICWSN platform is deployed to the farm.

E. *Packaging and its validation for horizontal scalability and applicability*

At the end of this research project, the ICWSN platform will be packaged for horizontal deployment. The packaging involves integrating a suite platform as a smart-agriculture ecosystem for practical use. The packaged platform will be a modified version of a zero-touch-design node device with mmWave support. Thanks to its portability, the device can be placed anywhere outdoors, enabling quick deployment to meet the diverse demands of smart-agriculture applications. In addition, for scalability perspective, the platform can be used in other smart-city applications, such as smart factories (industries) and smart things.

IV. Discussion on Future Deployment to the Field

Here, we will discuss the requirements that must be met by key technologies in the physical (PHY) and Media Access Control (MAC) layers and identify the issues regarding their deployment within smart cities. In particular, we will discuss their potential in future wireless communications from the aspects of efficient spectrum use and migration to upper (higher) frequency bands.

A. *Validity of the network structure in proposed scheme*

The mmWave-complient ICWSN platform will be implemented, constructed, and deployed on actual farms and agricultural worksites. Here, let us discuss the wireless communication systems supporting those scenarios. Table II is a qualitative comparison of wireless communication systems for smart agriculture (outdoor environment) across different criteria. The individual evaluations indicate strengths and weaknesses relative to the other network types in the table. The compared networks include commercial cellular (4G/5G)

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE II. COMPARISON OF WIRELESS NETWORKS FOR SMART AGRICULTURE

Network system	*Coverage*	*Capacity*	*Economic cost*	*Technical cost*
Cellular (4G/5G) satellite networks	**o**	**o**	**x**	**o**
LPWANs (e.g., LoRa)	**o**	**x**	**o**	**x**
PAN (e.g., ZigBee)	**x**	**x**	**o**	**x**
Optical wired network	**x**	**o**	**x**	**o**
Proposed mmICWSN	*	*	*	*

o denotes suitable, **x** denotes not suitable, and * denotes suggested.

and satellite networks, Low-Power Wide-Area Networks (LPWANs), Personal Area Networks (PANs) based on IEEE 802.15.4, optical (wired) networks, and the network layout of the proposed ICWSN platform. The criteria are communication coverage, network communication (wireless) capacity, and the economic and technical costs of implementation, construction, and deployment.

Cellular and satellite networks are used as de facto standards for wireless communications in outdoor environments. These networks offer superior coverage and communication quality, but their operational costs are high, and their coverage flexibility is relatively limited since it is provided from cellular operators, i.e., the users cannot be decided to the coverage area. As such, small-scale farmers tend not to approve of them, which is one factor preventing proliferation of their smart-agriculture applications [26]. LPWANs [27], which can construct a private network with wide-area coverage and low energy consumption, such as LoRa and SigFox, have been widely investigated. However, LPWANs can only transfer small amounts of data, such as text or low-resolution (time-lapse) images; the 100-Hz bandwidth in the 1-GHz band is not sufficient for transferring streaming data. PANs using the 920-MHz band [28], such as ZigBee, 6LoWPAN, Wi-SUN, and Bluetooth, have been used in WSN research. PANs can be deployed only in small environments, such as university campuses and plastic greenhouses, even with multi-hop communications enabled. Wired networks are the primary option in areas where optical fiber lines have already been deployed; however, deploying new fiber lines is unrealistic in rural areas due to economic constraints.

The network architecture of the proposed scheme is based on a WLAN compliant with the IEEE 802.11 standard, integrating multiple license-free radio-frequency bands to provide sufficient coverage. Gigabit-class data transmission rates are required for the backhaul (core) network between agricultural fields and access points, and mmWaves (Terragraph) have this capacity. Regarding cost-effectiveness, wireless communication devices, modules, and terminals that adhere to IEEE 802.11 are inexpensive and readily available. Regarding technical implementation costs, since the proposed scheme can be built on an IP network, the system can provide simple connectivity to various nodes, such as personal computers, tablet computers, and smartphones.

B. Contribution in terms of efficient use of spectrum

The key technologies that enable effective use of the radio spectrum are the ICN scheme and the communication-control technique using AI. ICN can help improve frequency efficiency thanks to its pull-type network design and caching mechanism. Specifically, the ICN-based IoT framework sends data only when the user requests it, thereby reducing data transmissions. In addition, with caching, the network node returns the data from its cache when requested, reducing duplicate data transmissions. On the other hand, AI-controlled communications enable real-time streaming data to be transferred across the inevitable bottlenecks in practical wireless networks. To summarize the relevant parts of the preliminary study, it can be concluded that data-compression effects based on data correlation and AI-based object detection have the potential to enhance the system's effectiveness. Furthermore, the ICN-based system can also reduce energy consumption as a side effect.

C. Contribution in terms of migration to upper spectrum

Shifting to higher frequency bands anytime, anywhere has been investigated as a way of dealing with the spectrum shortage. Here, the use of mmWaves could help to achieve meaningful outcomes for future research and development activities. However, although mmWave technology has been deployed in trials in an actual city, Georgetown (Penang, Malaysia) [17], to the best of our knowledge, there are no other examples. In addition, in our previous studies, network performance evaluations (measuring, e.g., TCP and UDP throughput) indicated that the TCP congestion control mechanism was not suitable for mmWave transmissions. This is because typical congestion controls that are suitable for wired networks and current microwave-band WLAN are not designed for the specific characteristics of mmWaves, such as dynamic throughput variation, high packet-error probability, and channel conditions affected by obstacles like humans, trees, and even leaves. In other words, the characteristics of mmWaves affect the upper-layer protocols. Note that the radio-propagation characteristics of mmWaves have been investigated in other studies, but the performances physical-, datalink-, network-, and transport-layer protocols well as those of the application-layer protocols have yet to be clarified.

D. Realistic workload model

The proposed ICWSN platform targets a smart agriculture scenario using cameras and visual data transmissions. The visual data is from cameras monitoring farms, greenhouses, and the surrounding environment, e.g., irrigation, canals, and weather conditions. It is expected that most of a monitoring session will consist of periodic images with minimal variation that is sent as sequential data. In fact, an analysis of images from an actual greenhouse confirmed this expectation [6]. On the other hand, when the AI in the proposed scheme detects people, vehicles, or wildlife in the framed image, it triggers communications and sends the relevant data. In traditional IP-based WSNs, because of their push-based architecture, data must be continuously sent to the cloud. In contrast, since ICWSN is a pull-type system that retrieves data only when necessary, it is expected to reduce network traffic compared

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

with conventional systems. Note that the ICWSN cannot receive any abnormal events unless the user requests them at a desired time. To tackle this issue, the proposed scheme incorporates an AI-based control.

V. Protocol Design for Future Deployment

Here, we discuss the feasibility of deploying a PoC-based system in the field. Prior to the experiment, we specify the protocol design and its procedure.

A. Network model

Figure 2(a) compares the proposed scheme's network layout with that of a typical WSN. The WSN is divided into several groups of Sensor Nodes (SNs), and each group selects a Cluster Head (CH) from its members. The CH collects the data from the SNs and sends it to the sink. In the proposed scheme, the network consists of SNs, Relay Nodes (RNs), and a Private Base Station (PBS). Moreover, while WSNs use the microwave wireless communications, the proposed scheme uses both mmWaves and microwaves, and the PBS andRNs work across both air interfaces. As shown in Figure 2(b), the proposed scheme uses RNs, which are higher in performance than SNs, instead of CHs, and a PBS rather than a sink node. The computing performance, reliability, battery capacity, and communication performance of the RNs and the PBS are higher than those of the SNs. The proposed scheme envisions the use of one PBS, several RNs, and dozens of SNs in the observation field.

The SNs acquire the information, e.g., a sensor measurement (text-based data) or an image captured by a camera (visual data). They packetize the data and send it as sensing data to the network. The SNs and RNs are distributed and have a leader-follower relationship via the shared wireless links. The wireless link between each SN and RN has a forward channel (in uplink) for sensing data and a backward channel (in downlink) for control messages. Note that this paper assumes that the SN can access the nearest RN in a single hop. The sensing data are finally forwarded to PBS, which coordinates the RNs in the local area in an edge-side network with a mesh topology. PBS includes the RN functionality and a proxy for external networks. When necessary, it shares a portion of the sensing data with the cloud. Users who request data acquisition are assumed to be located within the internal network; i.e., they access the nearest RN and exchange sensing data and control messages in accordance with the ICWSN procedure described later.

B. Procedure of passing control messages

Figure 3 shows the procedure of passing control messages. To shift computational functions, such as analysis and command determination, from the cloud to edge-side nodes, the RN is equipped with a functor that uses queues and a database. As shown in Figure 3(a), when an SN joins the ICWSN, the cover-node list of SNs, RNs, and the PBS located within the coverage area held by the SNs has to be updated. To perform this operation, the new SN advertises (broadcasts) a scan message to neighbor nodes, and then the SNs that received it respond after a backoff time has elapsed to prevent packet collisions. At this time, the responding SNs store this

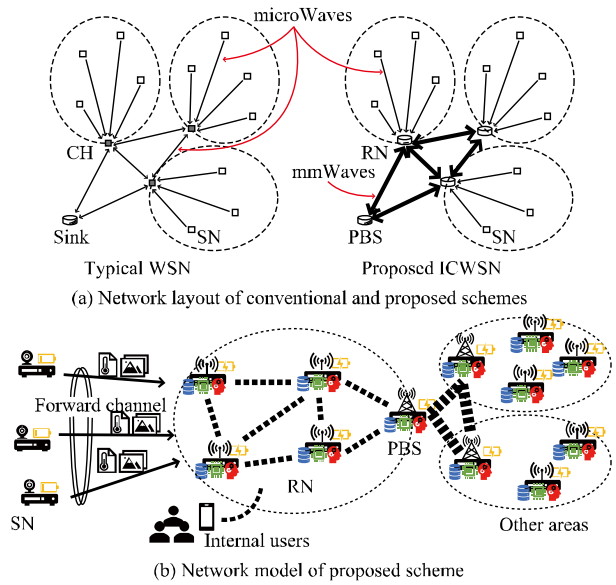


Figure 2. Network layout and model of the framework

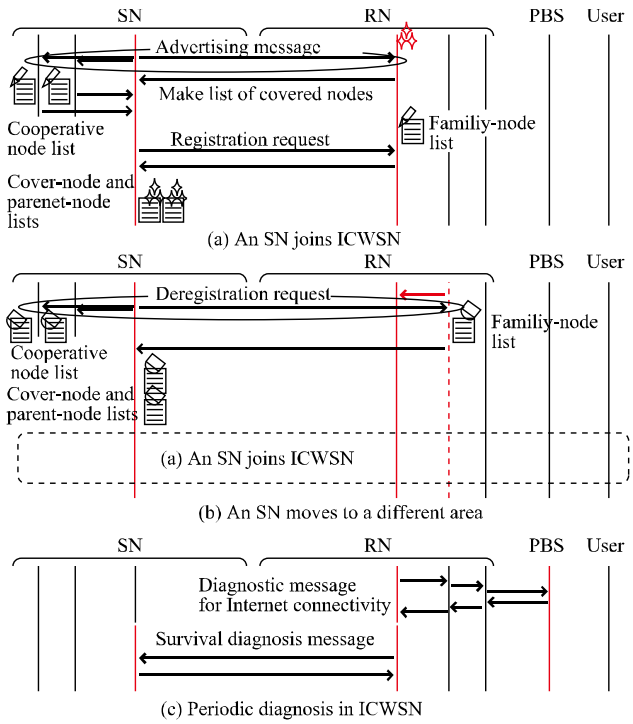


Figure 3. Procedure of sending control messages in proposed scheme

information in their cooperative node lists, which are used in the cooperative communication mechanism described in Section III.A. Next, the new SN selects the nearest RN as its parent node from the cover node list and sends a unicast registration message for the SN to join the RN's family (member). Upon receiving and accepting the proposal, the RN appends the SN to its family node list and sends a response message. Finally, the SN appends the RN to its parent node

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

list as its parent node. Note that the parent node list and family node list are the databases associated with the FIB in the ICN layer, and the SNs (and RNs) can only have RNs or the PBS as parent nodes.

As shown in Figure 3(b), when an SN moves, it sends a deregistration request to the (parent) RN to remove the registration. The RN then removes its registration from the SNs' cooperative node list. Then, a new connection in the new location is established according to the procedure in Figure 3(a). To ensure ICWSN's availability, it is necessary to periodically confirm the node's survival and verify connectivity to the external network. As shown in Figure 3(c), the RNs periodically verify the connectivity to the PBS, and the survival and connectivity of the SN are guaranteed through periodic one-hop connectivity verification between the SN and the (parent) RN.

C. Procedure of data collection (publishing) and retrieval

Figure 4 shows the procedure for handling the named data. As shown in Figure 4(a), cooperating SNs share part of this data with each other and with the RN. Note that, the AI-based method described in Section III.D filters the data to reduce network traffic. In addition, to use this mechanism, the data can be replied to even if the SN cannot answer; in such case, the RN (or a cooperative SN) responds as a proxy by using the SN as a caching mechanism. Regarding data retrieval, as shown in Figure 4(b), send an interest message (i.e., data retrieval request) to the network, similar to in the typical ICN framework. The interest message includes information about the desired data, such as its time and location. On the basis of the information in the interest message, it is forwarded to the RN, which stores the data to meet the requirements of the request. In accordance with the in-network caching mechanism of ICN, if the PBS (an RN along the previous routing path) has the data in its cache, it can respond to the request. Note that, when placing the RN in the configuration shown in Figure 4(b), the RNs closer to the PBS are more likely to be found before reaching the final RN. In addition, data retrieved at this time can improve the probability of content hits if it is frequently retrieved; therefore, it should be stored in the RN.

When the interest message reaches the final RN, there are different response scenarios depending on the message's time and location. If the time and location conditions are relatively relaxed, the RN will reply with the cached data. In other words, it will reply with the cached data when the time restriction is sufficiently large for data to be acquired and shared in short intervals. In this case, the response data is the caching data periodically shared in Figure 4(a). When time and location constraints are strict, and the user requires the latest data, the RN forwards the interest message using the family node list. The response data is the original data, and the RN caches it using a first-in-first-out algorithm for subsequent interest messages to improve the cache-hit probability. A key feature of ICN is that it does not require data from a specific node, unlike traditional host-centric networks. In other words, the response data need not be obtained by a specific SN; it only needs to be at a particular time and place. Therefore, if the data was retrieved from a specific area, it might have been replaced

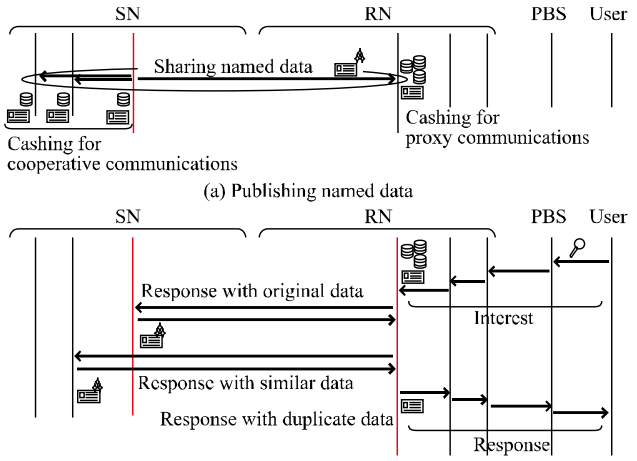


Figure 4. Procedure of data collection and retrieval in proposed scheme

with the neighbor node. If the response can be provided using similar data rather than the original data, it can improve the response probability.

D. PHY/MAC and ICN layers protocol design

The ICN protocol and packet format have been defined in IETF RFCs, such as RFCs 8569 and 8609. Therefore, if the proposed framework is to be used universally and extensively, its format must be fully compliant with existing ICN standards and be formally approved through standardization. This section focuses on the functionalities required to operate ICWSN rather than strict definitions. Note that standardization and related considerations are out of scope, but they will be continuously investigated as part of our future work. This section highlights how the ICN l ayer is stacked on top of the PHY/MAC layer in ICWSNs.

As a network design principle, the ICN layer's data transmission is based on multicast. Specifically, the data retrieval requests (interest) are multicast to the destination specified in the FIB, while the responses are transferred via a multipath-based unicast method according to the PIT. Also, the PHY/MAC layer's wireless communication is an over-the-air (overhearing), flooding-based broadcast method. However, the intermediate layer between them, i.e., a network layer like TCP (UDP)/IP, is sandwiched between them and generally uses a unicast communication method. Therefore, it is necessary to bridge the gap between the upper layer (ICN) and the lower layer (PHY/MAC). Although IP multicast primarily supports UDP-based data streaming applications, such as real-time video broadcasting, the reliability of data transmission over WLANs remains a concern.

For named data, the naming rule, data freshness, and security will be based on the same mechanisms as in the general ICN platform. Since the proposed scheme is designed for regional ICWSNs, using flat addresses is adequate. At the same time, it is necessary to determine a unique conversion rule for each data attribute (such as time and location) within each ICWSN. For data freshness, ICN cannot use the same latency-related parameters as used in conventional networks. This is because the traditional network provides user

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

responses from a fixed server, whereas ICN allows any node that stores cached data to act as a data provider. This aspect is currently under study as a new design based on the Age of Information (AoI) [29]. A detailed discussion regarding this matter is beyond the scope of this paper.

The proposed scheme seeks to resolve this issue by using cross-layer design [30], as shown in Figure 5. Cross-layer design is a methodology that intentionally breaks the traditional layered hierarchy to optimize overall system performance by enabling information sharing and coordinated operations across non-adjacent layers. For instance, the proposed scheme bypasses the middle layer. The cross-layer design presented here can be well-designed and well-explained in purely theoretical and academic settings, but it is not practical. Instead, a practical solution with a practical implementation is outlined as follows. The FIB settings of the unicast-based ICN platform are dynamically configured while considering compatibility; this enables multiple unicast communications to operate in parallel and thereby emulates the behavior of multicast communications. On the other hand, with such a simple design, the data transmission request and the target node generate exponential traffic. This problem can be alleviated by incorporating a proxy caching scheme into the RNs (mesh network). At the same time, the proxy caching scheme can reduce traffic in the wireless communication section by aggregating cached data to a manageable level. This mechanism is the core idea of the cooperative communication scheme described in Section III.A, and it can be enabled through the above alternative approach. A detailed validation, including implementation, network construction, and proof-of-concept demonstrations, remains as future work.

E. Channel allocation and assignment

As a network model based on WLANs, the proposed scheme assumes the use of microwaves (in the 2.4-GHz and 5-GHz bands) and mmWaves (in the 60-GHz band). Microwave WLANs are common, inexpensive commodity systems, making it suitable for the SNs, RNs, and PBS. However, since the mmWave WLAN system is not yet available, it will be used in the RNs and PBS as a backhaul. Moreover, WLANs are commonly standardized, but the available spectrum is regulated by individual countries and regions. As shown in Figure 6, four and 20 microwave channels and four mmWave channels are available in Japan. Note that a practical WLAN system can provide a multiplexing scheme, but here we count the number of non-overlapping wireless channels. In addition, the mmWaves have a 2.16 GHz bandwidth per channel, and high-speed data transmission is performed across the entire frequency band. Here, we suppose a 20 MHz bandwidth per channel for mmWaves, similarly to microwaves; 108 channels are available.

VI. Computer Simulations

The proposed scheme's feasibility was estimated in terms of its characteristics in the PHY and MAC layers by using computer simulations. In particular, the evaluations focused on whether the system is fundamentally usable, especially its

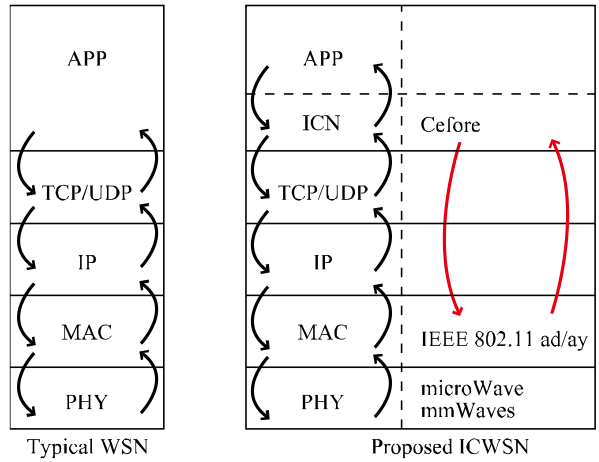


Figure 5. Protocol stack design: Typical WSN versus proposed ICWSN

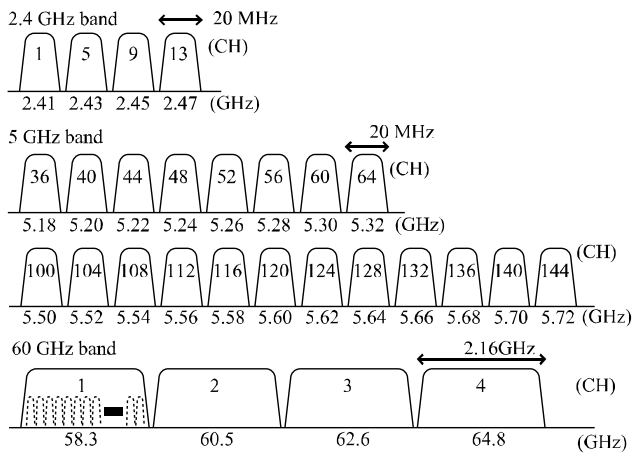


Figure 6. Channel-allocation plan in proposed scheme in compliance with Japan's regulations

resilience to the network traffic potentially generated by the SNs.

We assumed that the RNs (and PBS) support mmWave connectivity; and are installed in fixed locations, as shown in Figure 7(a). Furthermore, a mesh network based on mmWaves was constructed using four-sided antennas. In accordance with the Terragraph specifications [21][22][23], the communication range was set to 200–300 meters and the MCS index was set to 12 and 9, i.e., the catalog specifications; this would give a communication range of 250 meters. We assumed that the mmWave device could use four channels and that devices in adjacent areas could not use the same channels because of interference. Furthermore, the antenna was directional (beamforming). Given the above constraints, as shown in Figure 7(b), 25 fixed nodes (24 RNs and a PBS) were distributed in a latticed layout, and their channel assignments were determined. Note that if nodes are distributed completely at random, some of them might not be able to reach the PBS. The signal strength of mmWaves varies depending on objects in the surrounding environment, such as farms, trees, humans, and vehicles; i.e., they can be disrupted

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

by intermittent signal interference and obstructions. Here, we assumed a flat, obstacle-free countryside, free of the aforementioned signal disruptions, to simplify the computer simulation.

Figure 7(b) shows the scenario with 100 SNs distributed randomly in the field. Note that the simulation covered scenarios with 100 to 400 SNs. When the communication range was set to 200 m, all SNs were within the coverage area of the RNs. Here, we will call an RN that an SN can access a parent node and call the SNs accommodated by the same RN team nodes. Since RNs can guarantee connectivity to PBS via a mutual mesh network, if any SN can connect directly to the RN (parent node), all SNs remain network members without isolation, making it possible to collect and access the sensing data. The network layout and channel allocation described here enable a broadband wireless video solution, as described in Section III.C.

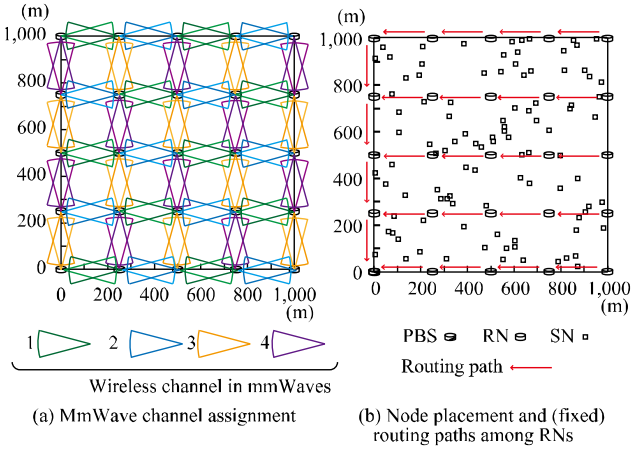
Figures 8–10 show the simulation results regarding the PHY/MAC layer capacity to support traffic on the ICWSN. The simulations account for the worst-case scenario in which all SNs (100–400) generate call requests simultaneously where they continuously transmit data. In other words, this evaluation is of how tolerant the wireless channel is to multiple SNs sending messages simultaneously. In addition, mutual interference among SNs was considered, but the channel allocation was performed ideally. Figure 8 shows the offered load divided by the capacity for all relay nodes as a statistical Cumulative Distribution Function (CDF). Since the offered load over the capacity was for a simulated environment, it was calculated by assuming up to twice the actual channel capacity that could be utilized. Here, when 100 SNs were simulated, the offered load over the capacity for all RNs was 72%, indicating sufficient wireless resources. However, wireless capacity was insufficient in the other cases (200, 300, and 400 SNs), resulting in a significant reduction in data transmission performance (offered load over capacities of 112%, 120%, and 140%, respectively).

Figure 9 shows a heatmap of overall wireless channel occupancy rates in the ICWSN field for the microwave spectrum. The results indicate high channel occupancy in the central field area. In contrast, Figure 10 indicates that millimeter-wave relay stations can sufficiently accommodate the traffic generated by 100–400 SNs because mmWaves can use a larger bandwidth (4.5 times as many channels as microwaves). Note that the fan-shaped radiation patterns in Figure 10 are due to directionality, and the fixed orientations are caused by the fixed routing path shown in Figure 7. Overall, these results indicate that it is necessary to determine the placement of nodes in the field.

VII. Conclusion

We presented a blueprint of the ICWSN platform for smart agriculture. The proposed platform integrates ICN, WSN, mmWaves, AI, and related technologies. The system design for implementing the platform was presented and preliminary evaluations were conducted.

In future work, we will carry out our plans and expect to achieve our goals. In particular, regarding the proof of concept for the network construction, it will be necessary to ensure



(a) MmWave channel assignment (b) Node placement and (fixed) routing paths among RNs

Figure 7. MmWave channel allocation and node placement

sufficient coverage to support the development of the network ecosystem. In addition, we will demonstrate the scalability and extensibility of the proposed scheme which is expected to operate stably for medium- to long-term periods. Furthermore, the platform developed in this research project will be used for other smart-city applications. In addition, it will be necessary to consider a broadcast-based wireless communication system combining ICN and WSN for edge-side networks. On the basis of the strategies identified in this study, the requirements for the key elemental technologies must be provided as feedback, including limitations and potential challenges.

Acknowledgement

This work was partly supported by SCAT and MIC, Japan.

References

[1] S. Mori, "Concept of ecosystem for smart agriculture: Millimeter-wave information-centric wireless visual sensor network," *Proc. IARIA IARIA Congress*, pp. 33–37, Venice, Italy, July 2025.

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

[3] M. Zhang, H. Luo, and H. Zhang, "A survey of caching mechanisms in information-centric networking," *IEEE Commun. Surveys & Tutorials*, vol. 17, no. 3, pp. 1473–1499, Apr. 2015.

[4] S. Mori, "Fundamental analysis of cooperative communication scheme using hierarchical modulation for information-centric wireless sensor networks," *Proc. IEEE CCNC 2025*, Las Vegas, USA, Jan. 2025 pp. 1–2, doi: 10.1109/CCNC54725.2025.10976088.

[5] N. Victor et al., "Remote sensing for agriculture in the era of industry 5.0—A survey," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 17, pp. 5920–5945, Feb. 2024.

[6] S. Mori, "A study on information-centric wireless visual sensor network for smart agriculture applications," *Proc. RISP NCSP 2025*, Pulau Pinang, Malaysia, Feb.–Mar. 2025, pp. 485–488.

[7] M. Bilal and S. G. Kang, "Network-coding approach for information-centric networking," *IEEE Systems Journal*, vol. 13, no. 2, pp. 1376–1385, June 2019.

[8] J. Llorca, A. M. Tulino, K. Guan, and D. C. Kilper, "Network-coded caching-aided multicast for efficient content delivery," *Proc. IEEE ICC 2013*, Budapest, Hungary, June 2013, pp. 3557–3562, doi: 10. 1109/ICC.2013.6655103.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

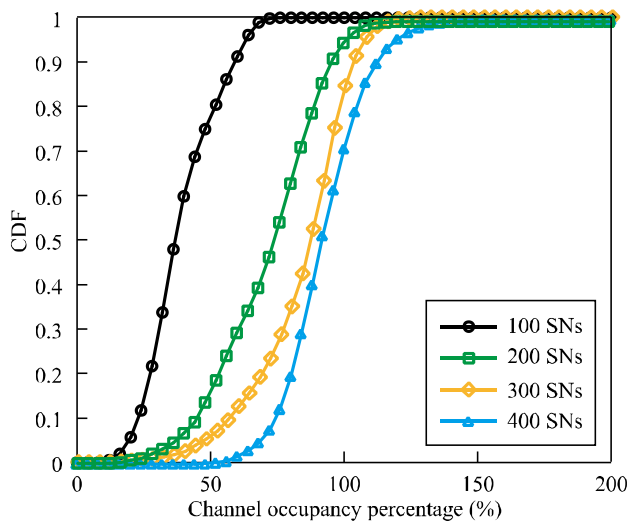


Figure 8. Statistical charactristics of channel occupancy in microWaves

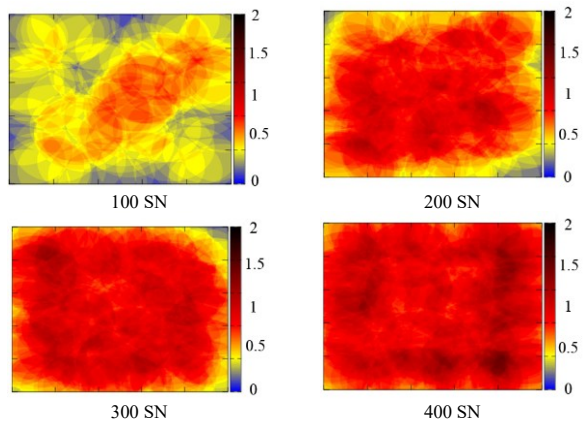


Figure 9. Heatmap of ICWSN deployment field in microWaves

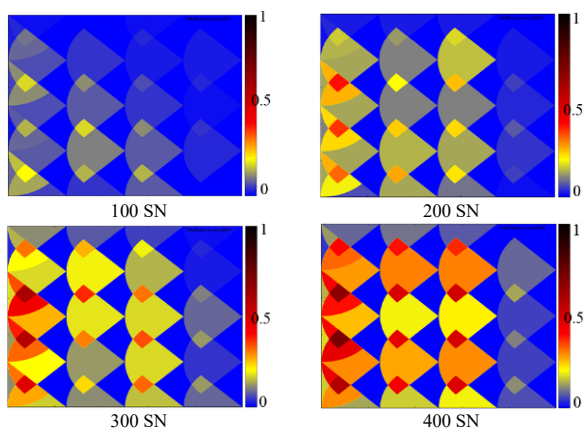


Figure 10. Heatmap of ICWSN deployment field in mmWaves

[9] A. Ramakrishnan, C. Westphal, and J. Saltarin, "Adaptive video streaming over CCN with network coding for seamless mobility," *Proc. IEEE ISM 2016*, San Jose, USA, Dec. 2016, pp. 238–242, doi: 10. 1109/ISM.2016.0054.

[10] Y. Yang and T. Song, "Energy-efficient cooperative caching for information-centric wireless sensor networking," *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 846–857, Jan. 2022.

[11] J. Yin, et al., "On the network coding-based D2D cooperative recovery for lossy feedback-free broadcasting," *IEEE Commun. Letters*, vol. 27, no. 9, pp. 2283–2287, Sept. 2023.

[12] D. You, Y. H. Lee, S. H. Choi, D. H. Kim, and S. Cho, "Joint hierarchical modulation and network coding for asymmetric data rate transmission over multiple-access relay channel," *Proc. ICUFN 2016*, Vienna, Austria, July 2016, pp. 96–98, doi: 10.1109/ICUFN.2016. 7536991.

[13] M. Tang, J. Chen, Y. Zhang, and Y. Zhang, "Performance analysis for physical-layer network coding with hierarchical modulation," *Proc. IEEE ICCSN 2016*, Beijing, China, June 2016, pp. 33–37, doi: 10.1109/ ICCSN.2016.7586682.

[14] S. Tharranetharan and M. J. Hossain, "A hierarchical modulation and network coding-assisted BICM system for asymmetric two way relay channels," *Proc. IEEE VTC2014-Fall*, Vancouver, Canada, Sept. 2014, pp. 1–5, doi: 10.1109/VTCFall.2014.6965815.

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

[16] M. N. Mowla, N. Mowla, A. F. M. S. Shah, K. M. Rabie, and T. Shongwe, "Internet of Things and wireless sensor networks for smart agriculture applications: A survey," *IEEE Access*, vol. 11, pp. 145813–145852, Dec. 2023.

[17] Terragrah, https://terragraph.com/ (retrieved: May 2026).

[18] K. Kanai, et al., "(Invited) D2EcoSys: Decentralized digital twin ecosystem empower co-creation city-level digital twins," *IEICE Trans. Commun.*, vol. E107-B, no. 1, pp. 50–62, Jan. 2024.

[19] S. Mori, "A cooperative and coded communication scheme using network coding and constructive interference for information-centric wireless sensor networks," *International Journal on Advances in Networks and Services*, vol. 15, no. 3&4, pp. 54–61, Dec. 2022.

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

[21] BeMap, https://www.bemap.co.jp/ (retrieved: May 2026).

[22] S. Mori, "Millimeter-wave information-centric wireless-sensor-network ecosystem: Evaluation under non-terrestrial environment and demonstration in actual city," *International Journal on Advances in Networks and Services*, vol. 18, no. 3&4, pp. 81–89, Dec. 2025.

[23] S. Mori, "MmWave UAV-assisted information-centric wireless sensor network for disaster-resilient smart cities: Preliminary evaluation and demonstration," *Proc. IARIA ICN 2024*, pp. 1–4, Barcelona, Spain, May 2024.

[24] Cefore, https://github.com/cefore/cefore/ (retrieved: May 2026).

[25] YOLO, https://ultralytics.com/ (retrieved: May 2026).

[26] M. R. Mahmood, M. A. Matin, P. Sarigiannidis, and S. K. Goudos, "A comprehensive review on artificial intelligence/machine learning algorithms for empowering the future IoT toward 6G era," *IEEE Access*, vol. 10, pp. 87535–87562, Aug. 2022.

[27] A. Pagano, D. Croce, I. Tinnirello, and G. Vitale, "A survey on LoRa for smart agriculture: Current trends and future perspectives," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 3664–3679, Feb. 2023.

[28] R. Pahuja, H. K. Verma, and M. Uddin, "A wireless sensor network for greenhouse climate control," *IEEE Pervasive Computing*, vol. 12, no. 2, pp. 49–58, Apr.–June 2013.

[29] S. Mori, "A study on delay-sensitive information-centric wireless sensor and actuator/actor networks," *Proc. IARIA ICN 2026*, pp. 1–5, Venice, Italy, May 2026.

[30] V. Srivastava and M. Motani, "Cross-layer design: A survey and the road ahead," *IEEE Commun. Mag.*, vol. 43, no. 12, pp. 112–119, Dec. 2005.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

HTN Planning Deserves Better Domains: Engineering Realistic Models for Autonomous Vehicles in Space and on Earth

Ebaa Alnazer, Ilche Georgievski, Marco Aiello
Service Computing Department, IAAS
University of Stuttgart
Stuttgart, Germany
e-mail: {ebaa.alnazer|ilche.georgievski|marco.aiello}@iaas.uni-stuttgart.de

***Abstract*—Developing operational AI planning systems requires domain models that faithfully capture the characteristics of their corresponding real-world applications. However, existing planning domain models exhibit two key limitations: they often underrepresent realistic aspects, such as alternative ways to accomplish tasks, uncertainty, and risk, and they provide limited coverage of societally relevant applications. We address these limitations in the context of Hierarchical Task Network (HTN) planning, an expressive AI planning technique well-suited for representing rich domain knowledge. We extend well-established domains of Rover and Satellite by incorporating overlooked realistic aspects, and engineer a new Self-Driving Car domain that has not previously been considered in the HTN planning literature. We design and apply a systematic domain-engineering approach grounded in existing methodologies and perform a quantitative structural analysis of the resulting domains. The results show that these domains exhibit increased structural diversity and richer representations of application knowledge. Beyond these results, the analysis highlights structural patterns in how HTN domains evolve when enriched with realistic aspects. We argue that these types of contributions not only bring planning domains closer to real-world applications but also provide a stronger foundation for advancing HTN planning frameworks and for developing and evaluating HTN planners.**

Keywords-HTN planning; knowledge engineering; domain models; risk; uncertainty; alternative methods.

I. Introduction

Intelligent autonomous systems are increasingly deployed in safety-critical and high-stakes environments, such as self-driving cars, satellites, and planetary rover missions to Mars. To operate reliably in such settings, these systems must reason about their actions and generate plans that achieve complex objectives under operational and safety constraints. These capabilities are studied in AI planning, a subfield of AI concerned with computing courses of action that, when executed from a given initial state, achieve specified objectives [2].

Among planning paradigms, HTN planning has attracted sustained interest due to its expressive power and its ability to represent structured application knowledge [3]. By decomposing tasks into subtasks using methods, HTN planning supports the explicit modelling of procedural knowledge and multiple alternative ways to achieve tasks. This makes it particularly suitable for complex, real-world applications.

While planning algorithms have advanced significantly, comparatively less attention has been given to the systematic engineering of domain models, including HTN ones. AI planning is a model-based technique, meaning to construct valid and executable plans, accurate and adequate knowledge about the application domain must be elicited, conceptualised, and formalised as a domain model. In HTN planning, this dependency is especially strong, as rich application knowledge is explicitly encoded in task hierarchies and decomposition methods. Constructing such domain models requires eliciting domain expertise, identifying relevant operational aspects, and translating them into formal structures, a knowledge-engineering process that remains labour-intensive and often ad hoc [4][5].

Many benchmark HTN domains, particularly those derived from classical benchmarks and introduced through International Planning Competitions (IPCs), are intentionally simplified to align with planners' capabilities and support their evaluation [6]. They often (1) restrict alternative task decompositions to a minimal set, (2) assume deterministic action outcomes or fixed costs, and (3) omit explicit modelling of risk and uncertainty. As a result, domains that shape much of HTN planning research abstract away characteristics intrinsic to real-world environments. This creates a discrepancy between the expressive potential of HTN planning and the simplified benchmarks typically used to evaluate the planners. This discrepancy motivates our central focus: how to engineer HTN planning domain models that more faithfully capture realistic, application-inherent aspects while remaining structurally analysable and comparable.

Building on our previous work [1][7][8], here we integrate three complementary elements into a single perspective on HTN domain engineering: the structured identification of realistic domain aspects, the incorporation of richer alternative task structures and explicit modelling of uncertainty and risk in HTN domain models, and the quantitative structural analysis of the resulting domain models. Together, these elements allow us not only to enrich domain models but also to assess how such enrichment changes their properties.

We propose a systematic domain-engineering approach grounded in existing knowledge-engineering processes [5][9][10]. We apply a conceptual framework for identifying and capturing realistic aspects [11], which guides the elicitation and structured incorporation of alternative courses of action, uncertainty in action costs, and risk. These aspects are modelled within the state-based framework of HTN planning [3] and its risk-aware extension [8], where risk and uncertainty are represented through probability distributions over action costs. Finally, we propose a set

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

of metrics to enable a quantitative assessment of domain model properties, such as structural characteristics and domain richness, thereby enabling a more systematic comparison between simplified benchmarks and their enriched or newly engineered counterparts.

To demonstrate and analyse our approach, we pursue two complementary strategies. First, we revisit two benchmark domains, Satellite and Rover, which are widely used in both classical and HTN planning research (e.g., [6][7][12]). Their prominence makes them representative of modelling assumptions prevalent in the field. At the same time, they represent autonomous vehicles in space operations, an inherently uncertain and risk-sensitive application area. They also capture important real-world planning challenges, such as coordinating multiple satellite or rover missions under strict resource constraints, including limited power, restricted target access, and high operational costs [13][14]. This makes them suitable case studies for extending simplified benchmarks to incorporate richer alternative task structures and probabilistic cost representations within risk-aware HTN planning.

Second, we engineer a new HTN domain for self-driving cars. Unlike Satellite and Rover, this domain is not derived from an existing benchmark but developed from scratch, informed by expert knowledge collected from the literature, allowing realistic aspects to be embedded during initial domain construction. The terrestrial autonomous driving setting provides a contemporary example of a realistic domain whose characteristics have received limited attention in AI planning. Many of these characteristics pose significant challenges for both modelling and solving planning problems. They arise from the domain's inherent dynamics and uncertainty, including road incidents, varying road conditions, risk factors, diverse driving tasks, and resource constraints, all of which require continuous tracking of the vehicle's state and its surrounding environment.

Taken together, we position HTN domain engineering and analysis as the central focus for realistic domain models for the first time. By integrating structured aspect identification, risk-aware modelling, systematic domain enrichment, and quantitative structural analysis, we provide a consolidated and extended account of how HTN planning domains can be developed to better reflect the complexity of real operational environments.

The remainder of the paper is organised as follows. Section II provides the necessary background. Section III contains the approach we follow to engineer HTN planning domains. Sections IV and V present our extensions and engineering of existing Satellite and Rover HTN planning domains with more alternatives and risk-awareness, respectively. Section VI contains our approach to engineering the Self-Driving Car domain. Section VII provides our quantitative analysis of the developed HTN planning domains, and Section IX presents discussions beyond the presented domains and concluding remarks.

II. Background

HTN planning is a widely used AI planning technique across many real-world domains, such as robotics, e.g., [15], healthcare, e.g., [16], and building automation, e.g., [17]. It enriches planning models with procedural knowledge about how tasks can be accomplished [3]. This domain knowledge, together with the intrinsic hierarchical structure of tasks, enables HTN planning to naturally capture structured decision-making processes.

An HTN planning problem is defined by an initial state, an initial *task network* representing the objective to be achieved, and domain knowledge consisting of networks of *primitive* and *compound* tasks. A task network represents a hierarchy of tasks, each either primitive or compound, and may include constraints that restrict their ordering. Primitive tasks correspond to actions that can be executed directly, whereas compound tasks must be decomposed into subtasks via *methods*, which describe how to accomplish them. In *state-based HTN planning*, the search starts at an initial state with an empty plan. The objective is to construct a plan that accomplishes the initial task network. During the search, when a compound task is encountered, it is decomposed using an *applicable* method, i.e., a method that can decompose the task and whose *preconditions* are satisfied in the current state. Decomposition refines the task network at the next hierarchical level while remaining in the same state. When a primitive task is encountered, it is executed, removed from the task network, and appended to the plan, after which the search proceeds to the resulting successor state.

Risk-aware HTN planning is a framework that extends classical HTN planning with constructs that consider the *uncertainty* of real-world environments [8]. It enables modelling of risk- and uncertainty-inducing actions through probability distributions over their costs and effects, where costs are defined as unbounded negative functions. The framework can be tailored for planning problems where actions have deterministic effects but variable costs. In our modelling approach, we adopt this variant as an initial step toward incorporating risk and uncertainty into HTN planning domain models.

In risk-aware HTN planning, the cost functions of actions can be of different types depending on the factors/sources of the costs. The cost function can be (1) *external* ($c^{ie}(a)$), i.e., it depends on external factors not explicitly modelled in the domain, such as market electricity prices or taxes, (2) *state-dependent* ($c^{is}(a)$), i.e., it is based on the system's current state, like the vehicle's charging level or position, (3) *constant* ($c^{ic}(a)$), i.e., it remains the same for every execution of an action, such as a fixed charging price, or (4) external and state-dependent, i.e., a *hybrid function* ($c^{ies}(a)$), which depends on both current state and external factors, where a denotes an action.

III. Engineering HTN Domain Models Approach

We design a systematic approach to ensure that HTN planning domains are systematically enhanced or engineered with realism in mind, supporting the development of planning

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

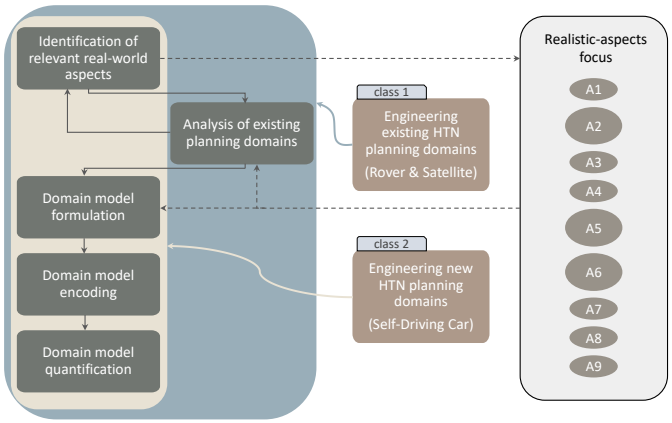


Figure 1. Approach for engineering HTN planning domains and classes of HTN planning domains.

models that are both expressive and applicable to real-world settings.

The approach is illustrated in Figure 1. It consists of several sequential phases, as suggested in the existing literature on knowledge engineering for planning [4][5][9][10], and distinguishes two classes of domains based on whether domain models are already available. *Class 1* concerns domain models that already exist and can be extended to increase realism, while *Class 2* focuses on domain models for which no domain model exists and must be engineered from scratch.

The approach begins by identifying the relevant aspects of the application to be reflected in the resulting domain model, regardless of class. When engineering knowledge for a Class 1 domain model, this phase must be combined with an analysis of the existing domain model. Be it a Class 1 or Class 2 domain, the next phase is to map the identified aspects into appropriate HTN constructs to formulate the extended or newly engineered domain model. The formulation of domain knowledge is followed by a phase in which the final outcome is produced by encoding the domain model using a planning language. Finally, the last phase is a quantitative assessment to evaluate how well the domain models reflect targeted aspects.

A. Identification of Relevant Real-World Aspects

HTN planning is a model-based technique in which planning systems rely on relevant and adequate knowledge of the domain in which they are supposed to operate [4][10]. Engineering such knowledge requires domain requirements analysis, which entails identifying relevant aspects that characterise the application domain [5]. We therefore apply an existing conceptual framework for identifying and categorising aspects of real-world planning domains [11]. We select seven aspects of the framework (**A1**–**A7**) and complement them with two additional necessary aspects (**A8** and **A9**) related to the real-world grounding of planning domains [7]. We refer to those two as *foundational aspects*. We place particular emphasis on **A2**, **A5**, and **A6**, as they capture alternative ways of accomplishing tasks, uncertainty, and risk associated with action costs.

A1: An important aspect in engineering domain models is defining their *objectives*. We consider hard goals or requests, which define a mandatory behaviour of plans generated by the planning system, according to which planning problems in the domain should be solved.

A2: Another key aspect concerns the granularity of *tasks and their hierarchical relationships*. Higher-level tasks represent abstractions that can be decomposed into subtasks, forming a hierarchy that naturally introduces structured causality and supports reasoning across different abstraction levels. A high-level task may admit multiple refinement options, i.e., alternative ways for accomplishing the same task. These refinements may only be valid under specific *constraints*.

A3: In addition to constraints restricting which refinements are possible in a given world state, we consider constraints governing the order in which tasks should be performed. Depending on the scenario, tasks may follow a total order, a partial order, or no order at all.

A4: It may be necessary to define who or what executes actions in the domain. These actors may include human agents, non-human agents, or mixed teams. We refer to them collectively as *agents*.

A5: Understanding the sources of uncertainty present in real-world environments is particularly important. When considering the executing agents, uncertainty can originate internally, from the agent itself (e.g., system reliability, human limitations, irrational behaviour), or externally, from environmental factors beyond the agent's control. Both internal and external sources can further be classified as either random (stochastic, rare, and unpredictable) or regular (pattern-driven and consistent).

A6: Executing actions alters the environment and incurs costs, i.e., consumes resources, such as money, time, fuel, or effort, which are predictable in a certain world. These costs represent one type of *quantity* that should be considered. In a fully predictable world, such costs are fixed. Under uncertainty, however, action costs become variable and may differ each time an action is executed. This variability stems from the underlying source of uncertainty. When the source is random, cost variability is unpredictable and cannot be addressed in offline planning. When the source is regular, the variability can be more precisely defined. When cost distributions of actions are known or statistically inferable, actions are considered *risk-inducing*. When the distributions instead reflect the decision maker's beliefs, the actions are *uncertainty-inducing*.

A7: Another salient consideration when identifying relevant aspects is the *structural diversity* of a domain. Planning domains should exhibit structural diversity by capturing the specific characteristics of the real-world applications they represent, rather than merely reproducing tasks that appear in other, similar domains. Many benchmark domains, for example, *Logistics*, *Rover*, and *Transport*, share common task structures, such as moving between locations. While such tasks are essential, modelling domains mainly around these generic task structures risks producing domain models that are structurally similar and insufficiently representative of their intended applications. To achieve meaningful structural

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

diversity, domain models should incorporate knowledge specific to the application context.

A8: One foundational aspect concerns the extent to which a domain model represents an *actual or plausible application* of planning technology, rather than an oversimplified or toy problem.

A9: The second foundational aspect addresses *evolving technological awareness*. It highlights the expectation that domain models should, when appropriate, reflect recent technologies and emerging trends in real-world applications. By doing so, planning models remain relevant and aligned with technological progress rather than being anchored solely in static or outdated assumptions.

B. Analysis of Existing Planning Domains

The Analysis of Existing Planning Domains phase involves examining the current domain representation, including its tasks, methods, actions, and structural relations, to determine how the identified aspects are currently reflected in the domain model. In doing so, the analysis highlights gaps or simplifications in the existing representation that prevent the domain from capturing important characteristics of the underlying application. The insights gained during this analysis may also prompt refinement of the aspects identified in the previous phase, leading to an iterative process between aspect identification and domain analysis until a consistent understanding of the domain model is achieved.

C. Domain Model Formulation

The next phase concerns formulating domain knowledge to construct the actual domain model [4]. A *domain model* is an abstract conceptual description of an application domain used to represent knowledge within a planning application. This conceptual description is derived from knowledge obtained in the preceding phase(s) and covers domain dynamics in relation to the types of problems the planning engine must solve, and the types of plans (solutions) that need to be produced as output [9]. Our focus is on state-based HTN planning and its risk-aware variant.

The domain model formally describes persistent knowledge and represents entities that are invariant across all planning problems [9][18]. These include objects with their relations and properties, actions that can change the environment's state, and other constructs, such as tasks and methods in HTN planning. A corresponding *problem instance* is needed to formally describe specific planning scenarios, which include the initial world state and the goal to be achieved.

HTN methods are particularly important as they model alternative ways to achieve tasks. We propose a categorisation of HTN methods based on their potential to serve as alternatives to other methods, distinguishing between strict alternatives, weak alternatives, and no alternatives [7]. Methods are considered strict alternatives if they share the same components and their preconditions do not contradict each other, allowing them to be simultaneously applicable. In contrast, two or more methods are classified as weak alternatives when their preconditions differ but are not mutually exclusive. In such cases, the domain model alone cannot determine whether the methods will be applicable in the same world state; this can only be resolved during planning, when variables are instantiated with concrete objects from the problem instance. A method is classified as having no alternative if it is the sole method capable of decomposing a given task, or if its preconditions (or the preconditions of the actions resulting from the decompositions of its corresponding task) contradict those of all other methods associated with that task. Formulating all types of alternatives entails that the added methods for task decomposition may introduce new task networks that include compound and primitive tasks not present in the original model of Class 1 domains.

Similarly, incorporating risk awareness is important as real-world applications often involve uncertainty that affects the outcomes and resource consumption of actions. In many situations, the exact cost of performing an action (e.g., resource consumption) cannot be defined in advance but can instead be characterised by a probability distribution. When such distributions are known or can be inferred, i.e., when risk-inducing actions are taken, the situation corresponds to decision-making under risk. Risk can be incorporated into HTN planning domains by modelling action costs as probability distributions. This representation allows planning decisions, such as choices between alternative methods, to be evaluated with risk attitude. Thus, incorporating risk into HTN planning domains may also entail enriching them with multiple alternatives for achieving tasks.

D. Domain Model Encoding

The next phase is to encode domain models (and problem instances) in well-defined syntax, such as the Hierarchical Planning Definition Language (HPDL) [19], Hierarchical Domain Definition Language (HDDL) [20], or another suitable planning language. In our approach, we use HDDL and our own extension of it to enable risk-aware encoding [1][8]. We refer to this HDDL variant as *risk-aware HDDL* (rHDDL), which allows encoding actions with probability distributions over their costs, representing each possible cost alongside its associated probability.

E. Domain Model Quantitative Analysis

As a final phase of our approach, we analyse the resulting domain models quantitatively to better understand their properties. The purpose of this analysis is not to evaluate planner performance but to examine how the undertaken engineering efforts affect the structural characteristics of the resulting HTN domain models.

Several of our introduced aspects, such as risk-aware action costs, influence planning behaviour during execution rather than the structural composition of a domain model itself. Since no metrics currently exist to quantify these properties at the modelling level, our analysis focuses on aspects that can be assessed directly from the domain structure. In particular, we focus on structural characteristics of engineered domains and analyse them using the following *structural metrics* [7]:

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

- total number of methods **#m**,
- number of methods that are strict alternatives **#ms**,
- number of methods that are weak alternatives **#mw**,
- number of methods that do not form an alternative **#mn**,
- number of tasks **#t**,
- number of direct recursive tasks **#rt**, where the direct recursive task is the task that is contained in the task network of one of the methods that can decompose it, and
- number of methods that have x subtasks **#m-tn**.

Beyond structural metrics, further insight into domain realism can be obtained by examining *knowledge richness*, a characteristic enabled by the expressive power of HTN planning. Since no widely accepted definition of knowledge richness exists, one might argue that knowledge richness can be quantified by the number of tasks, methods, alternative methods, or expressive constructs. However, objectively assessing whether the addition of such constructs truly enriches domain knowledge remains challenging. Instead, we adopt a more pragmatic approach and measure knowledge richness using metrics that capture the diversity of domain objects and their relationships. The *domain-richness metrics* are [7]:

- the number of predicates **#p**,
- the number of predicates that define relations between two or more objects **#pm**,
- the number of child object types, corresponding to the leaves of the type hierarchy **#ot**,
- the number of object types that serve as parents in the type hierarchy **#otp**, and
- the number of actions **#a**.

F. Application Domains

We apply our domain-engineering approach to three application domains. To illustrate how different realistic aspects can be incorporated into HTN planning domains, we consider domains that emphasise different characteristics of their underlying applications. For Class 1, we focus on space missions involving rovers and satellites. In particular, we select the Rover domain model [21],[1] and the Satellite domain [22],[2] The Rover domain has been used as a benchmark in IPCs, while the Satellite domain we consider is a hierarchical adaptation of a classical IPC benchmark, used to evaluate specific planning approaches, e.g., [22]–[24]. We extend the Rover domain by focusing on alternative decompositions, reflecting the multiple operational strategies available in planetary exploration missions. In contrast, we extend the Satellite domain by focusing on risk awareness, capturing the uncertainty and variability inherent in space operations. For Class 2, we focus on terrestrial driving involving autonomous cars. We develop an HTN domain model from scratch while focusing on both alternative decompositions and risk awareness, reflecting the dynamic and uncertain nature of autonomous driving environments.[3]

IV. Class 1: Rover Domain with Alternative Decompositions

The Rover application domain involves multiple autonomous rovers operating in planetary exploration scenarios. These rovers must navigate across a set of predefined waypoints to perform scientific tasks such as collecting rock or soil samples, capturing photographs of designated targets, and transmitting the gathered data back to a lander for further analysis.

We enrich the Rover domain by introducing additional alternative task decompositions. Accordingly, our analysis focuses on aspects related to task structure and decomposition, namely **A1**, **A2**, **A3**, **A4**, and **A7**. As this extension does not introduce risk awareness, aspects related to uncertainty and quantities (i.e., **A5** and **A6**) are outside the scope of the engineering process. We additionally discuss the domain with respect to the foundational aspects **A8** and **A9**. In the following, we describe all phases of the approach, but the last one, which has its own dedicated section. Moreover, we merge the first and second phases for a cleaner presentation and discussion.

A. Identification of Relevant Real-World Aspects and Analysis of the Existing Domain

Rovers have the objective of exploring planetary environments by investigating targets and performing multiple tasks, such as collecting data and samples and taking images of various objects [25] (**A1**). The existing planning benchmark [26][27] reflects several of these activities, where tasks are organised across multiple abstraction levels and exhibit structured causality, with complex tasks achieved through the decomposition into multiple subtasks (**A2**). In this domain model, multiple rovers explore a planetary environment by navigating between locations, collecting rock and soil samples, and taking images of target objects. Getting a rock or soil sample involves navigating to the data location, emptying the rover's storage, collecting the sample, navigating to a location from which the lander is visible, and, lastly, transmitting the collected data to the lander. The execution of these tasks should follow this total order (**A3**). Getting images follows a similar structure. It involves calibrating the camera, navigating to a location from which the target is visible, capturing the image, and relaying it to the lander by navigating to a location from which the lander is visible and transmitting it to the lander. The rovers and the lander represent the agents operating in this domain (**A4**).

Rovers are designed to operate on specific types of terrain and may therefore be unable to traverse certain areas. With technological advancements, however, the capabilities of rovers for planetary exploration have expanded. In particular, *space drones* have been introduced to support rovers, and perhaps astronauts in the future, in exploring terrains that are hard to reach by rovers. Space drones are "small uncrewed vehicles

[1] https://github.com/panda-planner-dev/ipc2020-domains/tree/master/partial-order/Rover

[2] https://github.com/panda-planner-dev/domains/tree/master/total-order/Satellite-PANDA

[3] The full domain models are available on Github: https://github.com/PlanX-Universe/htn-planning-domain-models.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

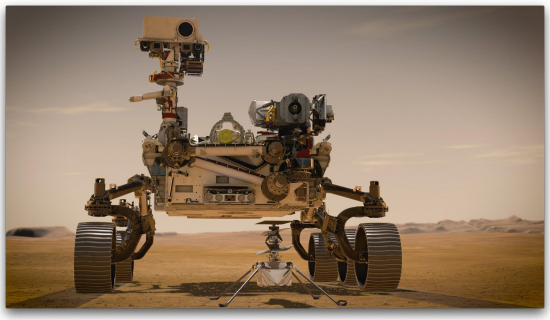


Figure 2. NASA's Mars 2020 Perseverance rover and NASA's Ingenuity Mars Helicopter. Courtesy NASA/JPL-Caltech.

guided by remote or autonomous means" [28]. Figure 2 shows a notable example, which is *Ingenuity* (also known as the Mars Helicopter), the first space drone sent to Mars by NASA in July 2020. This drone is attached to the *Perseverance rover*, which landed safely on Mars in February 2021 [29].

Extending the Rover domain by allowing rovers to request assistance from space drones during exploration missions, therefore captures additional characteristics of contemporary planetary exploration missions. In particular, adding space drones as helpers for rovers broadens the range of achievable tasks and objectives and introduces alternative ways of accomplishing space mission tasks (**A2**), an aspect lacking in the original domain.

The extension can include several alternatives: a space drone may send collected data to a lander, a space drone may collect data instead of a rover, and data gathering may be a collective effort between rovers and drones. Extending with the first alternative, for example, requires the extended domain to contain drones located at different positions such that each drone belongs to a rover. In this setting, each task of communicating image, soil, or rock data to the lander can be accomplished in two ways. The first option is for the rover to communicate the collected data directly to the lander by navigating towards it. The second option allows the rover to delegate this task to a drone. To send data to the lander using a space drone, additional tasks with their ordering constraints should be introduced (**A2** and **A3**). In particular, the drone navigates to the rover's location, the rover transfers the data to the drone, the drone navigates to a location from which the lander is visible, and lastly, it transmits the data to the lander. In the extended domain, space drones also represent an additional category of agents (**A4**).

Regarding **A7**, the original Rover domain already includes tasks that capture the specificities of the application, such as communicating rock data to the lander. It therefore goes beyond merely restating generic tasks that appear across many domains, such as navigation. Extending the domain with space drones to help rovers further exploit these application-specific characteristics by introducing operational capabilities.

The original Rover domain also covers **A8**, as it was modelled on the basis of an actual application of planning technology, namely the Mars exploratory rover mission of 2003, rather than an oversimplified or toy scenario. However, planetary exploration missions have evolved significantly since then, with substantial advances in capabilities and technology. One such development is the integration of space drones as assistants to rovers, which can be viewed as a transformative technology in rover missions. By incorporating drones, the extended Rover domain captures these developments and therefore addresses **A9**.

B. Domain Model Formulation

Having identified the relevant realistic aspects of the Rover domain, we now formulate them within the HTN domain model. The knowledge about rover exploration activities is represented using compound and primitive tasks. Compound tasks capture abstraction levels, causal relationships, recursion, conditions, and the representation of alternative ways of achieving tasks. Each compound task can be decomposed into subtasks through multiple methods.

An abstract graphical overview of the resulting HTN domain model is presented in Figure 3, where blue nodes denote compound tasks, orange nodes denote primitive tasks, and grey nodes $v_{m_0}, v_{m_1}, \ldots, v_{m_{13}}$ denote methods. Nodes highlighted in green correspond to our extensions. For clarity, the figure shows only the portion of the domain related to sampling and transmitting soil data. The domain components for rock sampling follow analogous tasks and methods, while the image-taking part differs slightly. Nevertheless, the extensions that allow rovers to obtain assistance from space drones when transmitting data to the lander involve similar tasks and methods to those used in soil sampling. In the domain model formulation and encoding, we explicitly implement the alternative in which space drone assistance is used during data transmission to the lander. The remaining alternatives, where drones collect data independently or jointly with rovers, can be formulated and modelled using the same approach.

The domain includes three top-level tasks, `get_soil_data`, `get_rock_data`, and `get_image_data` to obtain soil data, rock data, and take images, respectively. The `get_soil_data` compound task is decomposed using method v_{m_0} into four totally ordered tasks; `navigate_abs`, `empty_store`, `sample_soil`, and `send_soil_data`. The `navigate_abs` compound tasks is further decomposed using v_{m_1} into the primitive task `visit`, which marks the waypoints that the rover visits as visited, the compound task `navigate_abs1` to navigate through waypoints to the data location, and `unvisit` to mark the waypoints that the rover is not at anymore as unvisited when the rover arrives at the data location. The `navigate_abs1` compound task is further decomposed by three methods v_{m_2}, v_{m_3}, and v_{m_4}. v_{m_2} decomposes the task into an empty task network when the rover is already at the destination. v_{m_3} decomposes the task into a single primitive task `navigate` if the rover can directly traverse to the destination waypoint and that waypoint is visible from the rover's location. v_{m_4} decomposes the task into four further tasks: `navigate` to move to the next waypoint, `visit` to

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

mark the current location as visited, navigate_abs1 which is recursively decomposed to traverse all waypoints until reaching the destination, and lastly, unvisit to mark the traversed waypoint as unvisited. The empty_store task is decomposed by two methods; v_{m_5}, which results in an empty task network if the store is already empty, or v_{m_6} if the store is full and needs to be emptied by executing the primitive task drop. The task sample_soil is primitive and can be executed if the rover is equipped for soil sampling and the rover's store is empty. Lastly, the send_soil_data task is a compound task for sending the data to the lander and can be decomposed by two alternative methods that the rover can choose from v_{m_7} and v_{m_8}. The first method is for the rover to send the data to the lander without the space drone and it decomposes the task into two tasks navigate_abs for the rover to navigate to a location from which the lander is visible, and communicate_soil_data, which is a primitive task for sending the data to the lander. Method v_{m_8} is for getting help from the space drone to send the data to the lander. This method decomposes the task into 4 tasks. The first task is navigate_abs_drone, which is for the drone to go to the location of the rover, and it involves similar tasks and methods as the navigate_abs, with the exception that the space drone, unlike the rover, can traverse all locations without restrictions. This task is followed by the transfer_soil_data_to_drone primitive task to transfer the data from the rover to the space drone and navigate_abs_drone, where the drone navigates to a location from which the lander is visible. Lastly, the primitive task communicate_soil_data_drone is performed to transmit the data to the lander.

The get_rock_data task is decomposed similarly to the get_soil_data task. However, get_image_data has slight differences. This task is decomposed by a single method to the tasks calibrate_abs, navigate_abs, take_image, and send_image_data. To calibrate the camera, the calibrate_abs task is decomposed into two tasks navigate_abs to navigate to the calibration target, and calibrate to calibrate the camera. After calibrating, the navigate_abs, take_image, and send_image_data tasks are performed to navigate to the location from which the targeted object is visible, take an image, and send the image data to the lander similar to the tasks for sending the soil and rock data, respectively.

In Appendix A-A, we provide an example of possible planning problems in this domain.

C. Domain Model Encoding

To implement the proposed extensions, we encode the extended Rover domain in HDDL, the same language used in the original domain specification. Listing 1 shows the method that can be used to communicate soil data to the lander by the drone. The method decomposes the send_soil_data task into the following subtasks: the drone navigates to the rover's location, the rover transfers the data to the drone, the drone navigates to a location from which the lander is visible, and lastly, it communicates the data to the lander. This method requires that the drone belongs to the rover and that the lander is visible from the location to which the drone should navigate.

Listing 2 shows the method used to decompose the task of navigating the drone to a desired location. This method is modelled similarly to the navigation method of the rover. In particular, it is defined as a recursive method that eventually decomposes navigate_drone_abs into primitive tasks for navigating one step at a time between connected locations. However, unlike rovers, which may be restricted by terrain types, a drone can move to a location provided that the location is visible from its current position. Finally, Listing 3 shows the action used to communicate soil data from a drone to a lander. The action requires that the location of the lander is visible from the drone's location and that the drone has the collected soil data. As an effect, the soil data has been transmitted, the drone no longer holds the data, and the communication channel of the lander becomes available again.

Listing 1: Method for sending soil data by drone.

```
(:method m-send_soil_data_drone
 :parameters (?rover - rover
              ?waypoint ?x ?y ?z - waypoint
              ?l - lander
              ?drone - drone)
 :task (send_soil_data ?rover ?waypoint)
 :precondition (and (at_lander ?l ?y)
                (visible ?x ?y)
                (belongsTo ?drone ?rover)
                (at ?rover ?z))
 :ordered-subtasks (and
          (navigate_abs_drone ?drone ?z)
          (transfer_soil_data_to_drone
                ?rover ?drone ?waypoint)
          (navigate_abs_drone ?drone ?x)
          (communicate_soil_data_drone
            ?drone ?l ?waypoint ?x ?y)))
```

Listing 2: Recursive method for navigating the drone to the desired location.

```
(:method m-navigate_abs_drone-1
 :parameters (?drone - drone
              ?from ?to - waypoint)
 :task (navigate_abs_drone ?drone ?to)
 :precondition (at_drone ?drone ?from)
 :ordered-subtasks (and (visit ?from)
            (navigate_abs_drone-1
                    ?drone ?from ?to)
            (unvisit ?from)))
```

Listing 3: Action for communicating soil data from the drone to the lander.

```
(:action communicate_soil_data_drone
 :parameters (?r - drone
              ?l - lander
              ?p ?x ?y - waypoint)
 :precondition (and (at_drone ?r ?x)
            (at_lander ?l ?y)
            (have_soil_analysis_drone ?r ?p)
            (visible ?x ?y)
            (channel_free ?l))
  :effect (and (channel_free ?l)
```

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

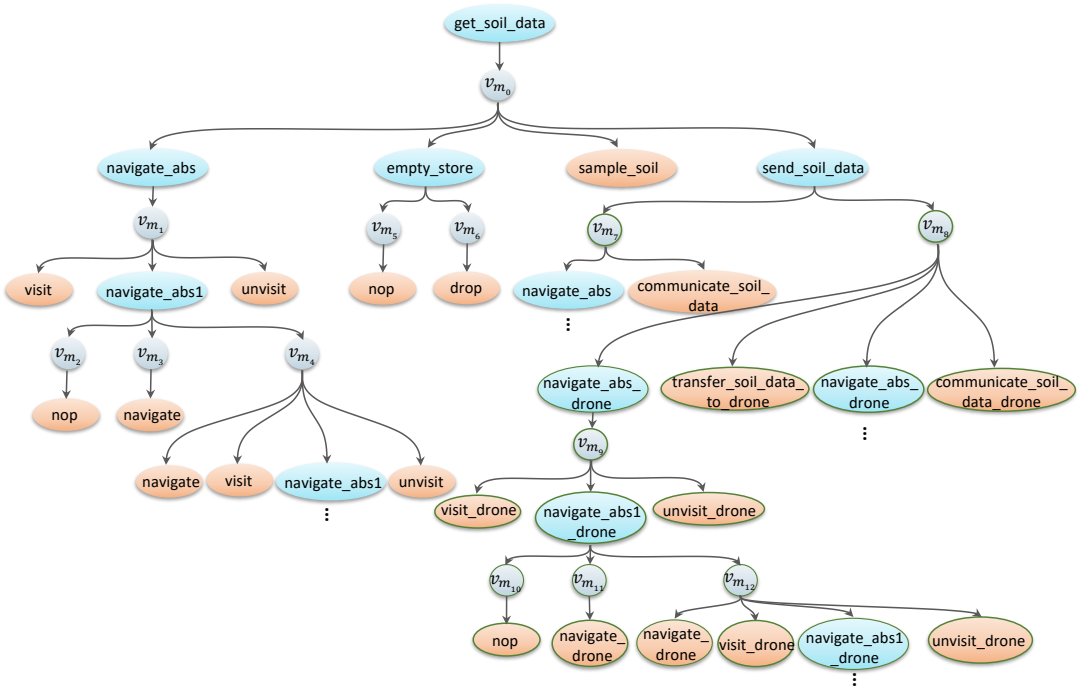


Figure 3. HTN model for the Rover domain.

```
        (communicated_soil_data ?p)
    (not (have_soil_analysis_drone ?r ?p))))
```

V. Class 1: Satellite Domain with Risk Awareness

The Satellite application domain involves space applications with autonomous orbiting spacecraft that perform tasks such as imaging, data collection, navigation, and scientific research. Figure 4 shows an example of such a spacecraft, which is the Ocean Surface Topography Mission/Jason 2 Earth satellite designed to make observations of ocean topography for investigations into sea-level rise and the relationship between ocean circulation and climate change.

We enrich the Satellite domain by introducing risk awareness. Accordingly, our analysis focuses on aspects related to task structure and decomposition, namely **A1**, **A2**, **A3**, **A4**, and **A7**, and aspects related to non-determinism and quantities, namely **A5** and **A6**). We additionally discuss the domain with respect to the foundational aspects **A8** and **A9**. The rest of the section follows the same structure as for the Rover domain.

A. Identification of Relevant Real-World Aspects and Analysis of the Existing Domain

The objective of satellites is to perform stellar observations while optimising factors such as mission time and power consumption (**A1**). This objective must be achieved while considering aspects, such as risk and uncertainty, and the states of satellites and their instruments. Achieving this objective requires satellites to perform multiple activities that have different abstraction levels and causal relationships (**A2**).

Satellites are equipped with several observation instruments, each with specific operation modes such as infrared, spectrograph, X-ray, and thermography, and have defined calibration

Figure 4. The Ocean Surface Topography Mission/Jason 2 Earth satellite. Courtesy NASA/JPL-Caltech.

targets (directions). Conducting a space observation mission is a complex task that involves preparing a satellite and then capturing an image of the target phenomenon. Preparing the satellite requires routing energy to the instrument, properly calibrating it, and orienting the satellite toward the phenomenon to be captured. Once these steps are completed, the satellite can capture the desired image. These activities must follow a specific execution order (**A3**). Note that the satellite itself represents the main agent in this domain (**A4**).

In practice, multiple instruments often need to be activated simultaneously due to time and resource constraints [30]. Consequently, the satellite must be capable of powering on several instruments at once and choosing how many instruments to activate, while accounting for faults, such as

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

power failures. Thus, extending the Satellite domain to support the selection of the number of instruments to power on reflects additional characteristics of the application domain and enables the modelling of realistic alternatives for performing stellar observation tasks (**A2**).

The complexity of satellite missions also arises from the inherent uncertainty of space environments and the technologies involved in satellite operations and instrumentation (**A5**). Planning stellar observations, therefore, requires reasoning about uncertainty and its inherent randomness, specifically the distinction between information available during planning and information revealed only during execution. These uncertainties directly affect resource consumption, such as time and power required to perform actions (**A6**).

Uncertainty in satellite missions comes from both internal and external sources. *Internal sources* are related to the satellite platform and instruments. For example, limitations of physical sensors may affect data quality (e.g., resolution, accuracy, noise), introducing uncertainty into action costs when these are tied to the required data quality [31]. Other internal uncertainties arise from the potential of power system failures, such as battery or solar array malfunctions [32]. This can be caused by power overload if the satellite's energy source (e.g., solar panels, radioisotope thermoelectric generators, or helium cells) cannot meet the demand [33]. Such power overload and potential power loss can occur when powering several instruments in parallel, ultimately increasing execution time and delaying recovery.

External sources stem from the satellite's operating environment. Space weather events may cause anomalies such as temporary outages, power failures, and solar cell degradation [34]. Similarly, solar energetic particles or protons can penetrate satellite electronics and cause electrical failures [35].

The Satellite domain is structurally diverse and includes tasks that capture the specific characteristics of satellite operations, such as routing energy to instruments and calibrating them (**A7**). Thus, similarly to the Rover domain, it goes beyond merely restating common or generic tasks that appear across many planning domains. Adding the ability to power multiple instruments at once, however, is a step forward in exploiting these application-specific characteristics.

The domain satisfies **A8**, as it was originally modelled on the basis of an actual application of planning technology at NASA. It also reflects emerging characteristics of contemporary stellar observation missions (**A9**). In particular, modern satellites support multiple observation instruments operating in different modes may activate several instruments simultaneously to maximise observation opportunities.

B. Domain Model Formulation

Next, we formulate the gathered knowledge as a prerequisite to model the Satellite domain. An abstract graphical overview of this HTN domain model is shown in Figure 5. The graphical notation follows the same conventions as in Figure 3, where colours distinguish compound tasks, primitive tasks, methods, and extensions.

Knowledge about stellar observation is represented as compound and primitive tasks. The domain contains a single top-level compound task, `do_observation`, which represents the execution of a stellar observation mission. This task consists of preparing an instrument and taking an image of the target phenomenon. It can be decomposed using four methods, v_{m_0} to v_{m_3}, which represent different situational contexts rather than alternative strategies of performing the observation task. Their main differences lie in the preparation phase. Specifically, v_{m_0} performs the full preparation sequence consisting of `activate_instrument`, `turn_to`, and `take_image`. Method v_{m_1} omits instrument activation, assuming that the instrument is pre-calibrated, for example, from previous observations. Method v_{m_2} skips the turning step, while method v_{m_3} performs only `take_image`.

While a satellite can have multiple instruments onboard, the original domain allows only one instrument to be powered at a time per satellite. This constraint is enforced by switching off any active instrument before powering on another and by making the power unavailable once an instrument has been activated. This behaviour is modelled through methods v_{m_4} and v_{m_7}. v_{m_4} decomposes `activate_instrument` into three subtasks: `switch_off`, `switch_on`, and `auto_calibrate`. The first two tasks switch off an already powered instrument and activate the instrument corresponding to the required observation mode. The `auto_calibrate` is further decomposed by $v_{m_{10}}$ and $v_{m_{11}}$. Method $v_{m_{10}}$ decomposes the task into `turn_to` and `calibrate` to slew the satellite toward the calibration direction and calibrate the instrument, respectively. Method $v_{m_{11}}$ skips turning, assuming the satellite is already aligned with the calibration target. Method v_{m_7} applies when no instrument is currently powered on, thus it skips switching off.

However, real-world satellite missions often require activating multiple instruments simultaneously due to time and resource constraints [30]. To capture this behaviour, we extend the domain introducing new methods for the `activate_instrument` task that allow satellites to power multiple instruments concurrently and incorporate permissive decomposition.

To model the situation of powering several instruments under the risk of power overload and recovery delays, we introduce methods v_{m_8}, v_{m_9}, v_{m_5}, and v_{m_6}. Methods v_{m_8} and v_{m_9} allow additional instruments to be powered by decomposing into `overload` or `superload`, which switch on a second or third instrument, respectively, followed by `auto_calibrate`. Methods v_{m_5} and v_{m_6} instead switch of multiple instruments at once by decomposing into `switch_off_overload` or `switch_off_superload`, followed by `switch_on` and `auto_calibrate`.

These extensions introduce alternative ways of activating instruments. For example, to power a second instrument, the planner may either switch off the first instrument using v_{m_4} or overload the satellite using v_{m_8}. Likewise, v_{m_5} and v_{m_9} offer a choice between switching off two instruments or activating a third, resulting in a "superload" state with three active

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

instruments.

To model non-determinism and action costs, we analyse how internal and external uncertainty sources affect actions that rely on a stable satellite power system. We focus on the cost variability of actions `overload` and `superload`, which place higher demands on power. Specifically, considering that costs are indicative of the time required for actions to be completed, activating a second instrument via the `overload` action has a 99% probability of taking 15 units of time and a 1% probability of taking 100 units of time due to power failure. Action `superload` is riskier as the unfavourable outcome incurs much higher cost, although it has a lower likelihood of occurrence, with a 95% probability of taking 15 units of time and a 5% probability of taking 400 units of time. All other actions are assigned a fixed cost of 15 units with 100% probability, enabling a clear comparison of planner behaviour under different risk attitudes concerning the simultaneous or sequential activation and deactivation of instruments. Here, we use the constant cost function type $c^{ic}(a)$. This means that the action costs are defined within the domain model and remain consistent across different problem instances. However, other types of cost functions and alternative cost distributions can also be considered, as the statistical inferences used to derive them may vary across different satellite domains.

In Appendix A-B, we give examples of possible planning problems in this domain.

C. Domain Model Encoding

We encode the planning knowledge formulated earlier directly into a risk-aware HTN planning domain model using rHDDL. Listing 4 shows a fragment of the domain model, including the actions `switch_on`, `overload`, and `superload`. The action `switch_on` can be executed only when the satellite's power is available and makes it unavailable once an instrument is activated. In contrast, actions `overload` and `superload` can be executed when the power is already unavailable, but they introduce a risk of power failure. In this model, we use a predicate `(power_avail)` that represents a safety constraint on the maximum power usage. This constraint can be deliberately ignored when choosing to overload or superload the satellite. The potential recovery time from such failures is represented using probabilistic cost distributions via the : $costdist$ construct of rHDDL. A cost distribution specifies alternative recovery costs together with their associated probabilities in the form $(or(p_1(c_1)(p_2(c_2))\ldots(p_n(c_n))$, where each expression $(p_i(c_i))$ denotes that cost c_i occurs with probability p_i.

Listing 4: Switch on, overload, and superload actions in the Satellite domain.

```
(:action switch_on
 :parameters (?so_i - instrument ?so_s -
    satellite)
 :precondition (and (on_board ?so_i ?so_s)(
    power_avail ?so_s))
 :effect (and (power_on ?so_i)(not(calibrated
    ?so_i))(not(power_avail ?so_s)))
 :costdist (or (1(15))))

(:action overload
 :parameters (?so_i - instrument ?so_s -
    satellite)
 :precondition (and (on_board ?so_i ?so_s)(not
    (overloaded ?so_s))(not(power_avail ?so_s)
    ))
 :effect (and (power_on ?so_i)(overloaded ?
    so_s)(overloads ?so_i ?so_s)(not(
    calibrated ?so_i)))
 :costdist (or (0.99(15)) (0.01(100))))

(:action superload
 :parameters (?so_i - instrument ?so_s -
    satellite)
 :precondition (and (on_board ?so_i ?so_s)(
    overloaded ?so_s)(not(power_avail ?so_s))(
    not(superloaded ?so_s)))
 :effect (and (power_on ?so_i)(superloaded ?
    so_s)(superloads ?so_i ?so_s)(not(
    calibrated ?so_i)))
 :costdist (or (0.95(15)) (0.05(400))))
```

VI. Class 2: Self-Driving Car with Alternative Decompositions and Risk Awareness

We now turn from extending existing domains to engineering a new planning domain from scratch. To demonstrate the generality of our approach across different operational environments, we consider the application domain of self-driving cars, for which, to our knowledge, no corresponding (HTN) planning domain model currently exists.

Self-driving cars are autonomous transportation systems that navigate with little or no direct human control, typically transporting passengers or performing tasks under human supervision. They operate in environments that are inherently complex and uncertain, characterised by factors such as varying road conditions, road incidents, risk, diverse driving tasks, and resource constraints, including travel time and fuel or battery levels. Safe operation, therefore, requires continuously tracking both the vehicle's internal state and its surrounding environment.

We engineer an HTN planning domain model for this application, which we call *Self-Driving Car*. We analyse all aspects (**A1**-**A9**), with a particular focus on task alternatives and risk awareness.

A. Identification of Relevant Real-world Aspects

The objective of the Self-Driving Car domain is to transport passengers or goods between locations while satisfying operational constraints such as travel time, energy consumption, and safety requirements (**A1**). Achieving this objective requires considering the vehicle's overall state (e.g., current location), the states of its components (e.g., headlights), and various environmental factors (e.g., weather and road conditions). It also involves a variety of driving activities organised across different levels of abstraction, where complex tasks are realised through the execution of multiple subtasks such as route planning, lane following, obstacle avoidance, and parking (**A2**).

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

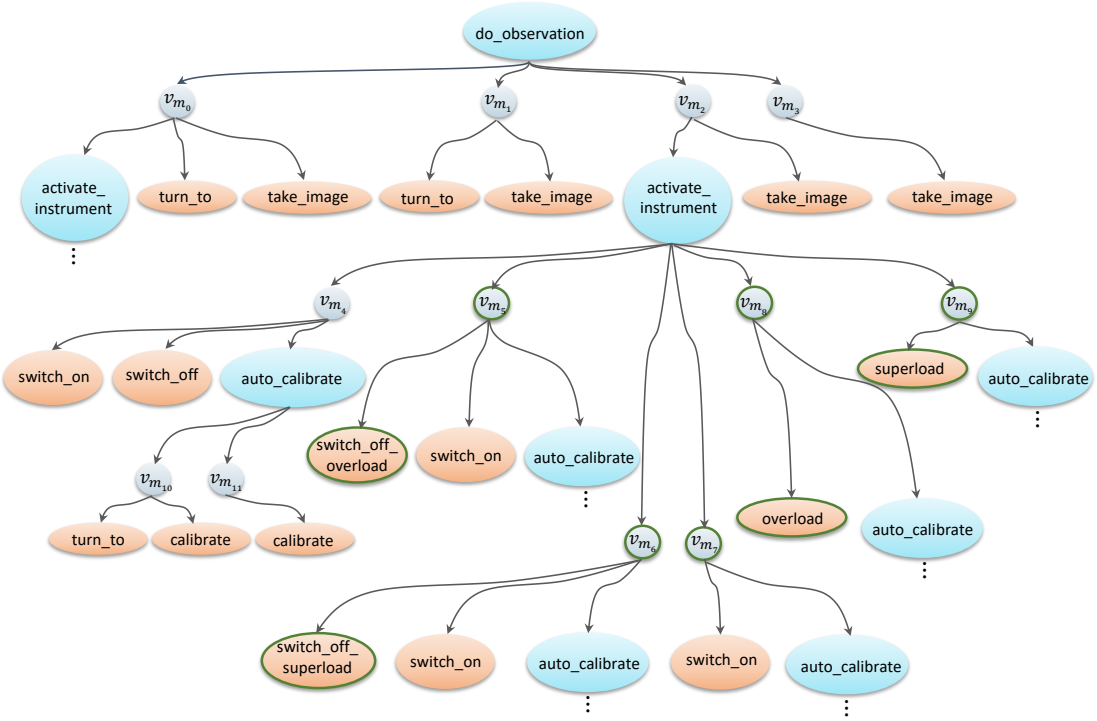


Figure 5. HTN model for the Satellite domain.

For example, reaching a destination may require navigating through intermediate locations, stopping or starting the engine, and managing turn signals. During operation, the vehicle must also handle road contingencies, such as pedestrians or construction zones, which may require actions like stopping, dodging the incident, or restarting the engine. Environmental factors, such as slippery roads, introduce additional complexity and may require actions like activating the Electronic Stability Program (ESP) and adjusting the car's speed.

These activities must also respect ordering constraints arising from traffic regulations and the dynamics of road environments (**A3**). Additionally, many driving activities can be accomplished in multiple ways (i.e., alternatives). For example, a car may address poor road conditions by adjusting speed, with or without activating ESP, and may choose from several alternative routes to reach the same destination. The self-driving car itself serves as the primary agent in this domain, performing all actions (**A4**).

The complexity of self-driving tasks largely arises from the dynamic and uncertain environments in which cars operate (**A5**). Factors such as driver behaviour, vehicle control variability, traffic conditions, and environmental changes introduce uncertainty into the execution of driving tasks. Following our previous work [8][11], we categorise uncertainty sources according to the level of vehicle autonomy: (1) non-autonomous (human-driven), (2) fully autonomous (no human intervention), and (3) semi-autonomous (shared control). These uncertainties may be internal, originating from the agent performing the driving tasks, or external, coming from the environment.

For human-driven cars, internal regular uncertainties often arise from differences in driving skills, habits, intentions, tactics, and speed preferences. As a result, travel time and energy consumption may vary depending on how a driver behaves. Additional uncertainties may arise from driver fatigue, which can lead to accidents, such as falling asleep at the wheel. The consequences of driver drowsiness have been studied statistically [36]. Unlike regular sources, which can be statistically anticipated, some internal uncertainty sources are rare and unpredictable, such as a driver experiencing a medical emergency (e.g., a stroke), making them difficult to anticipate.

For fully autonomous cars, internal regular uncertainties may stem from control variability. For instance, maintaining stability on slippery roads may require adjustments in control behaviour, which can lead to variations in travel time or energy consumption. Random internal sources of uncertainty may also exist, leading to unpredictable outcomes. For example, mechanical failures (e.g., a flat tyre) can interrupt the car's operation (e.g., the car stops).

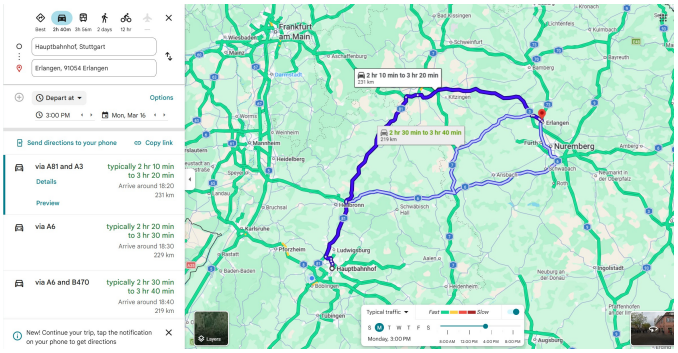
Semi-autonomous vehicles inherit elements of both cases, as the car's behaviour results from interactions between the human driver and the autonomous control system. Consequently, both human-related and system-related uncertainties may affect task execution.

External sources of uncertainty can also influence cost variability (**A6**). These sources may likewise be regular or random. Rare events, such as a vehicle being unable to charge due to an unexpected charging-station malfunction or encountering unexpected roadblocks caused by accidents, are difficult to predict during planning, making the outcomes of actions uncertain.

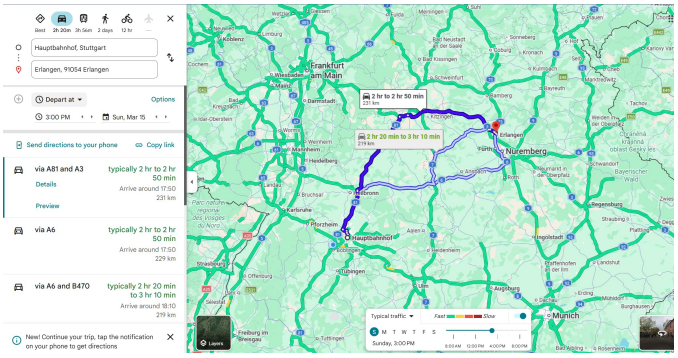
2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Now consider regular external sources of uncertainty that arise from environmental conditions that vary over time. The weather is a prominent example: due to the chaotic nature of the atmosphere and limitations in observation and modelling, forecasts inherently involve uncertainty [37]. Consequently, weather conditions are often reported using probabilistic forecasts in which variables such as temperature, wind, and precipitation are quantified probabilistically [38].

Traffic conditions represent another major source of regular external uncertainty. Route planning systems, such as Google Maps, illustrate this effect by providing ranges of estimated travel times for the same route depending on expected traffic conditions. As shown in Figure 6, these estimates of travel times vary between weekdays and weekends, with shorter times typically observed on weekends due to lighter traffic. Similarly, queues at charging stations introduce additional regular external uncertainty. Studies such as [39] estimate the probability of waiting times at charging stations, which directly affect charging duration and overall trip time.



(a) Travelling times on a workday [40].



(b) Travelling times on a weekend [41].

Figure 6. Variability of travelling times in Google Maps.

In addition to the factors discussed above, it is important to account for resource consumption, i.e., the costs associated with performing actions. These costs may include time, money, energy, effort, or even potential risks to human lives (**A6**). In the presence of uncertainty, such costs become variable and may differ across executions of the same action. Therefore, understanding and modelling cost variability is essential for effective planning.

For instance, uncertainty in weather conditions may significantly affect driving costs. Travelling between two locations in winter may take longer and require more energy if snow falls. The extent to which such variability can be characterised depends on the available knowledge about the underlying uncertainty. When uncertainty can be described probabilistically, such as through weather forecasts, the associated action costs, such as travel delays, can also be represented probabilistically. This situation corresponds to decision-making under risk. Conversely, random sources such as unpredictable accidents introduce uncertainty that leads to costs, e.g., delays, injuries, financial loss, and environmental impact, that are difficult to quantify probabilistically. Such events can reduce road capacity, increase congestion and travel time, and potentially cause further incidents [42].

Figure 7. Autonomous Self-Driving Car on Urban Street. Free for use from Pexels.

The domain also exhibits structural diversity, as it includes tasks specific to autonomous driving applications, such as activating the ESP, decelerating, and accelerating (**A7**). Moreover, the domain represents a realistic planning application, as shown in Figure 7 (**A8**), and reflects contemporary technological advancements in modern vehicles, such as stability control systems and sensory data (**A9**).

B. Domain Model Formulation

Having identified the relevant aspects of the self-driving car domain, we now formulate them as HTN planning constructs to build the corresponding domain model.

Driving activities are constructed as compound and primitive tasks in a hierarchy that captures how high-level driving objectives are achieved through decompositions to low-level actions. An abstract graphical overview of the resulting HTN domain model is shown in Figure 8. At the top of the hierarchy is the compound task `drive`, which represents travelling between two locations. This task can be decomposed using three methods, denoted as v_{m_1}, v_{m_2}, and v_{m_3}, which capture different driving situations. Method v_{m_1} is applicable when the vehicle does not have enough energy to reach the next location. In this case, the vehicle must first drive to a charging station (`drive`), recharge its battery (`recharge`), and then resume its trip by driving from the charging station to the intended destination (`drive`).

The second method, v_{m_2}, applies when the vehicle has sufficient energy but has not yet reached its destination. In this case, before moving, the vehicle prepares for driving

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

by cranking the engine if it is not already running (`start`) and turning on the lights if they are off and it is nighttime (`turnon`). The vehicle then moves one step to the next location (`move_step`). After each movement, the `drive` task is recursively decomposed, allowing the vehicle to continue travelling through intermediate locations until it reaches the destination.

The third method, v_{m_3}, becomes applicable when the vehicle has reached its destination. In this case, the `drive` task is decomposed into a single compound task `stop_vehilce`, which handles the termination of the trip. The task `stop_vehilce` can itself be decomposed by two methods, v_{m_4} and v_{m_5}. The former stops the engine using a primitive task, while the latter applies when the engine is already off ahd therefore performs no further action, indicated by the `nop` primitive task. This latter case corresponds to the phantomisation pattern commonly encountered in HTN domain models, where tasks may be reduced to an empty task network [43].

The `move_step` compound task mentioned above is decomposed by two methods v_m_{10} and v_m_{11}, depending on whether the road segment ahead is free of any complexity. When the road is clear, $v_{m_{10}}$ decomposes into a single primitive task `accelerate`, representing a straightforward movement to the next location. When the road involves additional complications, $v_{m_{11}}$ decomposes into two consecutive compound tasks: `handle_incidents` and `handle_road_conditions`. The `handle_incidents` task is decomposed by two methods, $v_{m_{13}}$ and $v_{m_{12}}$, depending on the type of incident encountered. For still incidents, such as rocks or construction work, the vehicle must decelerate, dodge the incident, and then accelerate again. For moving incidents, such as pedestrians crossing a street, the vehicle decelerates, brakes to allow the moving incident to pass, and accelerates again.

The `handle_road_conditions` task can be decomposed by four methods, $v_{m_{15}}$, $v_{m_{16}}$, $v_{m_{17}}$ and $v_{m_{18}}$. All these methods are applicable when the road conditions are abnormal, for example, when the road is icy, slippery, under construction, or covered with loose gravel. The methods correspond to different driving strategies: accelerating without adjusting for the road condition, decelerating only, activating the ESP while decelerating, and activating the ESP but accelerating. Since these methods share the same precondition, they are all applicable at the same time during planning assuming the planning state allows for it. This allows an HTN planner to choose among several alternatives, a modelling patter that we refer to as *permissive decomposition* or *non-exclusive decomposition*.

Both `handle_incidents` and `handle_road_conditions` also include an additional method ($v_{m_{14}}$ and $v_{m_{19}}$, respectively) that decomposes the corresponding task into an empty task network when there are no incidents or abnormal road conditions, respectively. These cases represent another instance of phantomisation. This modelling choice allows an HTN planner to handle situations in which incidents and adverse road conditions occur simultaneously between two connected locations.

We now formulate the aspects related to non-determinism and resource quantities. In particular, the domain model considers the effects of three sources of uncertainty. The first source is the speed at which pedestrians walk across the pedestrian crossing, which represents a regular external source of uncertainty. The second source is traffic on roads, which also constitutes an external regular source of uncertainty. The third source is the ability of the autonomous vehicle to stabilise on slippery roads, which represents an internal regular source of uncertainty related to vehicle control.

When cost variability comes from regular uncertainty sources, such as traffic jams, it can be described by a probability distribution that is either known or statistically inferred. In such cases, the corresponding actions are considered risk-inducing. Within this domain model, we define driving costs as the time required to traverse roads while handling the various road complexities. Traffic conditions, such as traffic jams, during planning make the estimation of travel time variable (as shown in Figure 6). Although we focus on risk-inducing actions and travel time as the primary cost metrics, other forms of uncertainty and cost measures could be considered. Examples include uncertainty-inducing actions or costs related to fuel or battery consumption, ride comfort, or road characteristics such as curvature.

Travel costs depend both on external factors, such as traffic conditions, and on properties of the road itself, such as length and surface conditions. To capture this dependency, we use a hybrid external and state-dependent cost function $c^{ies}(a)$ (see Section II). Estimation of uncertain traffic conditions is obtained through an external function, while road properties are represented by a state-dependent cost function defined with respect to the grounded planning problem. These two components are combined into a hybrid function that produces a probability distribution over travel times. Figure 9 shows example probability distributions over travel times for a planning problem in this domain. For instance, accelerating or decelerating on road (l_4, E) leads to variable travel times depending on whether the ESP is activated (state-dependent) and on external factors such as traffic and vehicle stability. When the ESP is active, deceleration guarantees arrival at destination E in ten hours. Without ESP, arrival may take eleven hours with probability 80% or six hours with probability 20%.

A detailed discussion of the planning problems in this domain, including a detailed reasoning behind the assigned cost probability distributions in a specific problem instance, is provided in Appendix A-C.

C. Domain Model Encoding

We encode the domain and problem instances using rHDDL. Compound tasks are modelled by specifying the task name together with its parameter list. For example, Listing 5 shows the definition of the compound task `handle_incidents`. Methods are modelled by specifying their parameters, the compound task they decompose, the corresponding preconditions, and the resulting task network. For example, `handle_incidents` can

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

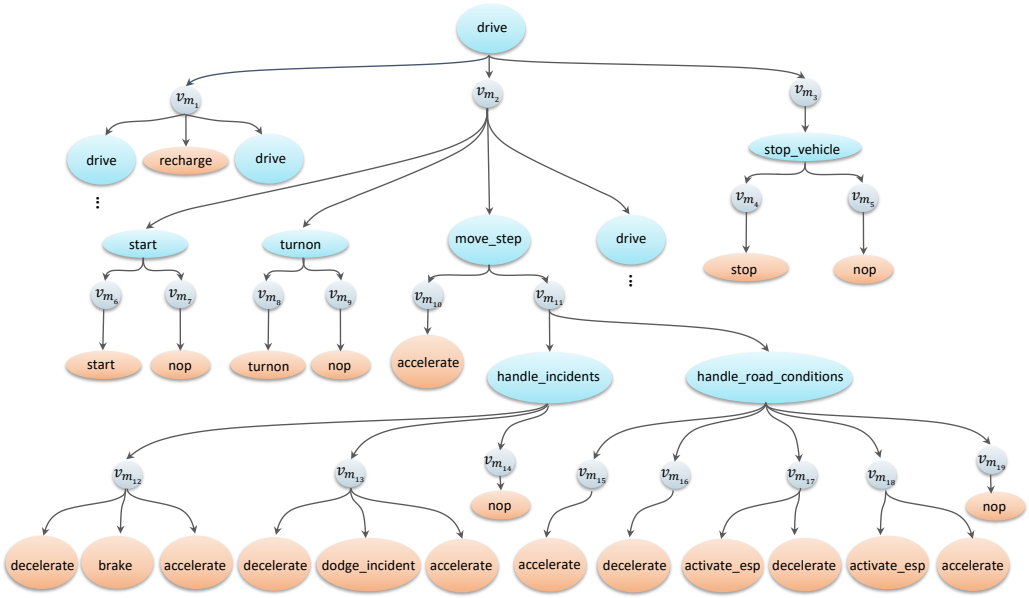


Figure 8. HTN model for the Self-Driving Car domain.

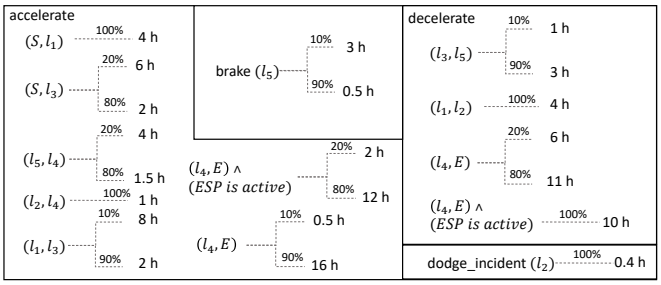


Figure 9. Actions with the corresponding possible bindings and probability distribution of costs (time).

be decomposed by three methods. Two of these methods apply depending on whether the encountered incident is static or moving, as specified in their preconditions. The third method represents a phantomisation case, allowing the task to be reduced to an empty task network when no incident is present.

Listing 5: Methods for handling incidents in the Self-Driving Car domain model

```
(:method handle_incidents_0
 :parameters(?v - vehicle ?l1 ?l2 ?l3 - loc ?
    movinc - movingobs)
 :task(handle_incidents ?v ?l1 ?l2)
 :precondition(and (connected ?l1 ?l3)(
    connected ?l3 ?l2)(in-inc ?movinc ?l3))
 :ordered-subtasks(and (decelerate_incident ?v
     ?l1 ?l3)(brake ?v ?l3)(
    accelerate_incident ?v ?l3 ?l2)))

(:method handle_incidents_1
 :parameters(?v - vehicle ?l1 ?l2 ?l3 - loc ?
    stillinc - stillobs)
 :task(handle_incidents ?v ?l1 ?l2)
 :precondition(and(connected ?l1 ?l3)(
    connected ?l3 ?l2)(in-inc ?stillinc ?l3))
 :ordered-subtasks(and(
    decelerate_still_incident ?v ?l1 ?l3)(
    dodge_incident ?v ?l3 ?l2)(
    accelerate_still_incident ?v ?l3 ?l2)))

(:method handle_incidents_2
 :parameters(?v - vehicle ?l1 ?l2 - loc)
 :task(handle_incidents ?v ?l1 ?l2)
 :precondition(clearroad ?l1 ?l2)
 :ordered-subtasks())
```

Actions are modelled with parameters, preconditions, and effects. Risk is represented using the `:costdist` construct. Listing 6 shows four actions related to decelerating and accelerating when handling bad road conditions. Each action is associated with cost distributions shown in Figure 9.

Since the cost functions are hybrid, the actions can be preprocessed and expanded into multiple variants corresponding to different road conditions. These variants directly reference the appropriate hybrid cost functions within the domain model.

Listing 6: Accelerating and decelerating actions of the Self-Driving Car domain model that deal with bad road conditions.

```
(:action accelerate_bad_road
  :parameters (?v - vehicle ?l1 ?l2 - loc)
  :precondition (and (not (in ?v ?l2))(not(
    activated_esp ?v)))
  :effect (and (in ?v ?l2)(highspeed ?v))
  :costdist (or (0.9(16)) (0.1(0.5))))

(:action accelerate_after_esp
  :parameters (?v - vehicle ?l1 ?l2 - loc)
  :precondition (and (not(in ?v ?l2))(
    activated_esp ?v))
  :effect (and (in ?v ?l2)(highspeed ?v))
  :costdist (or (0.8(12)) (0.2(2))))

(:action decelerate_after_esp
```

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

```
  :parameters (?v - vehicle ?l1 ?l2 - loc)
  :precondition (and (not(in ?v ?l2))(
    activated_esp ?v))
  :effect (and (in ?v ?l2) (not(highspeed ?v))
    )
  :costdist (or (1(10))))

(:action decelerate_bad_road
  :parameters (?v - vehicle ?l1 ?l2 - loc)
  :precondition (and (not(in ?v ?l2))(not(
    activated_esp ?v)))
  :effect (and (in ?v ?l2)(not(highspeed ?v)))
  :costdist (or (0.2(6)) (0.8(11))))
```

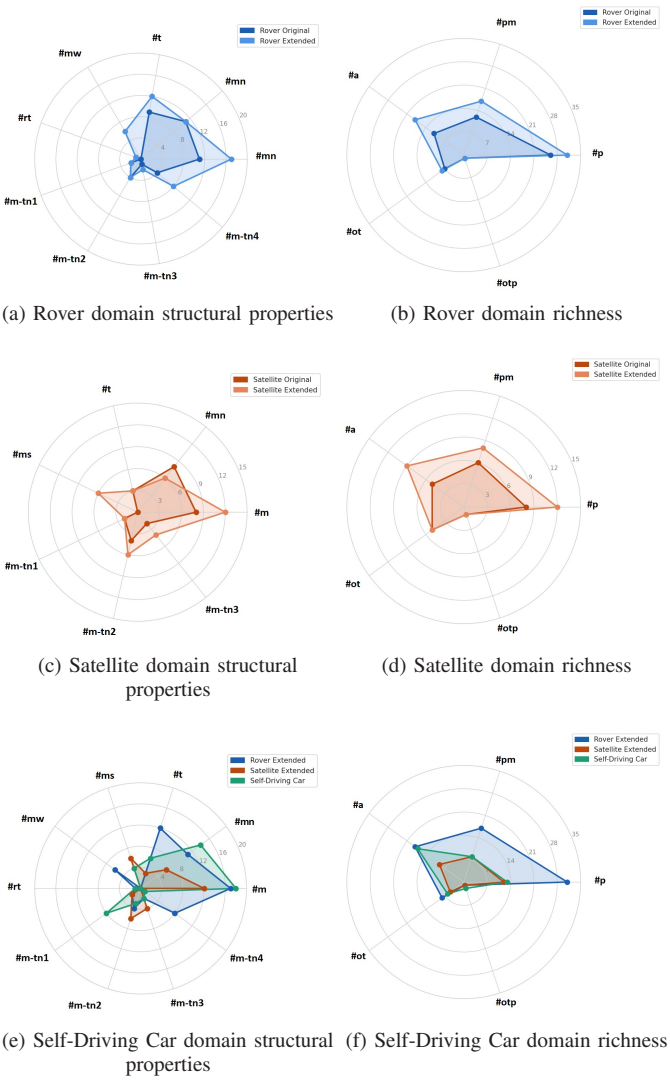
VII. Quantitative Analysis of the Developed HTN Domains

The developed HTN planning domains are developed following the systematic domain-engineering approach presented in Section III, which focuses on incorporating realistic, application-inherent aspects. As a result, the developed HTN domains include features that the original domains miss, making them qualitatively more advanced. However, quantitative analysis of these domains can provide additional insights into their structural properties. The results of the quantitative analysis of the analysed HTN planning domains are shown in Table I. The table compares the structural properties and domain richness of the original and extended versions of the Satellite and Rover domains. It also reports the same metrics for the newly engineered Self-Driving Car domain.

Regarding structural properties, we highlight that in the original versions of both the Satellite and Rover domains, no alternatives are available for accomplishing tasks. After the proposed extensions, the Satellite domain includes six strict alternative methods that can accomplish the instrument activation task, whereas the Rover domain includes six weak alternative methods that enrich data transmission to the lander offered using space drone's assistance. Since the extended Rover domain is modelled under the assumption that each rover is associated with a drone, these weak alternative methods are guaranteed to provide an option to perform the task. Thus, in this specific domain, these six methods can be interpreted as strict alternatives.

The Self-Driving Car domain has the largest number of methods overall. Among these, four are strict alternatives that offer the choice of how to perform driving tasks on roads with abnormal conditions. Both the Rover and Self-Driving Car domains include one recursive task related to navigation or driving. Such recursion is common in domains involving continuous behaviour, such as movement between locations until reaching a destination.

As expected, the introduction of additional alternatives increases the total number of methods in the extended Rover and Satellite domains. In the Satellite domain, the extension introduces additional methods with two and three subtasks, while in the Rover domain, the number of methods with three or four subtasks increases. However, the maximum number of subtasks across all methods remains unchanged in both domains.



(a) Rover domain structural properties (b) Rover domain richness

(c) Satellite domain structural properties (d) Satellite domain richness

(e) Self-Driving Car domain structural properties (f) Self-Driving Car domain richness

Figure 10. Comparison of the structural and domain-richness of the original, extended, and newly engineered domains.

Regarding the domains' richness, Table I shows that the number of actions, predicates, and multi-object predicates increases in both the Rover and Satellite domains after the proposed extensions. The values observed in the newly developed Self-Driving Car domain lie between those of the two extended domains. A larger number of objects and richer relationships between them provide more detailed information about the application domain. Similarly, a greater number of actions expands the range of achievable behaviours. Taken together, these metrics indicate that the developed HTN domains exhibit increased knowledge richness and more closely reflect the complexity of real-world scenarios.

The radar charts in Figure 10 provide a visual comparison of the structural and domain-richness profiles of the analysed HTN planning domains across their original, extended, and newly engineered configurations. Overall, the radar charts show that the proposed domain engineering process yields distinct struc-

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE I. STRUCTURAL AND DOMAIN-RICHNESS METRICS OF THE ORIGINAL AND EXTENDED VERSIONS OF THE SATELLITE AND ROVER DOMAIN AND THE DEVELOPED SELF-DRIVING CAR DOMAIN. THE METRICS ARE EXPLAINED IN SECTION III-E

Domain		Structural metrics										Domain-richness metrics				
		#m	#ms	#mw	#mn	#t	#rt	#m-tn				#p	#pm	#ot	#otp	#a
								1	2	3	4					
Rover	Original	11	0	0	11	9	0	2	4	1	4	26	12	7	1	11
	Extended	17	0	**6**	11	**12**	**1**	2	4	2	**8**	**31**	**17**	**8**	1	**18**
Satellite	Original	8	0	0	8	3	0	2	4	2	0	8	6	5	1	5
	Extended	12	**6**	0	6	3	0	2	**6**	**4**	0	12	8	5	1	9
Self-Driving Car	Original	**18**	4	0	**14**	6	**1**	**8**	3	2	1	13	8	6	**2**	17

tural outcomes across domains. The Rover domain primarily grows through additional methods, tasks, predicates, and actions (Figures 10a and 10b). The growth in the number of predicates and the growth in predicates that define relations between two or more objects appear to be comparable (Figure 10b). This suggests that the extended domain introduced additional predicates primarily to capture more complex relationships among the objects, rather than merely increasing the number of stand-alone properties. The Satellite domain expands through the addition of new methods, predicates, and actions, while the set of tasks remains unchanged (Figures 10c and 10d). This indicates that the extended domain introduces alternative ways to accomplish the existing tasks, providing greater flexibility in how the available tasks can be executed. The Satellite and Rover domains introduce new alternatives through restructuring of task decompositions, as shown in Figures 10c and 10a. The Self-Driving Car domain exhibits comparatively high procedural complexity due to the diversity of driving behaviours represented in the model, as shown in Figure 10e.

For the Rover domain, the transition from original to extended is characterised by a radial expansion across all axes, as shown in Figure 10a and 10b, with the most pronounced gains observed in the number of methods, particularly alternative methods and those with four subtasks in their task network, actions, predicates, and task arity. This indicates that the extension introduces enrichment of both procedural and operational expressivity while preserving the overall structure of the domain.

The Satellite domain, by contrast, exhibits a different extension pattern. While the total number of methods and predicates increases moderately compared to the Rover domain, the number of non-alternative methods decreases, as shown in Figure 10c. This reflects a deliberate structural reorganisation of the domain model in which some previously flat task decompositions are replaced by alternative hierarchical decompositions, thereby increasing the availability of alternatives. In general, Figures 10c and 10d show that the transition from the original to the extended version of the domain exhibits a radial expansion across all axes, except for the non-alternative methods.

The radar comparison across all three engineered domains (Figures 10e and 10f) shows that the Self-Driving Car domain has the largest number of methods and a relatively high number of actions. This indicates a comparatively high level of procedural complexity in the domain model. At the same time, the number of predicates remains lower than in the extended Rover domain and comparable to the extended Satellite domain, suggesting that the Self-Driving Car domain encodes much of its complexity through task decompositions and actions rather than through a large set of predicates.

Across all domains, the number of object types remains relatively stable. This observation suggests that the object-type hierarchy, once defined during the initial modelling stage, tends to remain largely unchanged during subsequent domain extensions.

VIII. RELATED WORK

Most existing AI planning domains have been developed either as benchmarks for IPCs or as illustrative examples for evaluating planning techniques. In the context of IPCs, domains are often intentionally simplified in order to facilitate systematic evaluation and comparison of planners, e.g., [6]. While this simplification is necessary for benchmarking purposes, it can limit the suitability for modelling actual planning problems. This limitation has been characterised as the *sanitised benchmark domain barrier*, referring to the limited utility of simplified benchmark domains for real-world environments [12].

Other domains are introduced to demonstrate the capabilities of domain modelling languages (e.g., [44]) or to illustrate or evaluate specific planning approaches. As a result, many planning domains are designed with particular experimental objectives in mind rather than through a systematic domain engineering process [4]. Some works also propose domains that address aspects different from those considered in our domains, e.g., [45][46].

In the following, we review existing work that models the Rover, Satellite, and Self-Driving Car domains, or closely related planning domains.

A. Rover Domains

Research on AI planning for planetary rover missions is closely related to our work. However, most existing rover domain models generally do not reflect recent technological developments, such as the use of drones as assistants in planetary exploration. The Rover domain we extend was used in the HTN planning track of IPC-2020 [21]. While it provides a useful benchmark, it does not incorporate recent advancements in exploration technologies or capture several realistic aspects considered in our extension.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Several studies have modelled rover domains for AI planning using techniques other than HTN planning and with a focus on incorporating aspects different from those we focus on. In [47][48], for example, a rover domain is formulated as a hybrid domain exhibiting both continuous and discrete behaviour. The work aligns with the goal of providing more options for achieving tasks by allowing alternative methods for energy generation and management. To transmit data, the rover can either draw power from its batteries or generate energy via solar panels. The domain is modelled in PDDL+ for hybrid planning rather than HTN planning and incorporates different alternatives from those introduced in our extension. Other works focus on incorporating temporal and uncertainty aspects into rover-related planning domains. In [44], a rover domain is presented using PDDL2.1, with particular emphasis on temporal constraints. Similarly, in [49], a model of the rover domain is proposed, which integrates temporal and uncertainty aspects within a framework combining HTN and timeline-based planning. In [45], an Europa lander domain is developed for HTN planning. Although it shares similarities with rover missions in terms of planetary exploration tasks, it focuses primarily on uncertainty in energy consumption and resource quantities, representing these through action costs related to battery life and predefined task utilities. Overall, while these studies provide valuable insights into modelling rover-related planning domains, they address different planning techniques or incorporate a distinct set of aspects from those considered in our work. In particular, most existing HTN or HTN-like Rover domains were developed without following a clear methodological domain-engineering process. As a result, they often overlook several realistic aspects that we explicitly target in our extensions, including risk awareness and the systematic modelling of alternative task decompositions.

B. Satellite Domains

Existing Satellite planning domains lack several aspects that we incorporate in our work, such as explicit modelling of risk and uncertainty. A version of the Satellite domain was used in [22][24][50]. This version is developed on the basis of its classical planning counterparts, which were originally introduced for earlier IPC benchmarks. As a result, it inherits classical planning assumptions, thereby simplifying several characteristics of real-world satellite operations. In our work, we build on this benchmark version by analysing sources of uncertainty and their effects on action costs, enabling the incorporation of risk-aware decision-making. In addition, we expand the domain by introducing additional tasks and alternative methods for completing space missions.

In [46], a Satellite domain for onboard and online planning is presented using a language based on an extended HTN representation. The model captures several operational aspects, including unexpected events and resource consumption, but does not incorporate risk. Another line of work on satellite planning focuses on broadcasting and communication satellites, e.g., [51]. These domains address scheduling and resource allocation problems that differ semantically from the space observation and exploration scenarios considered in our work.

C. Self-Driving Car Domains

Existing AI planning works that model self-driving car domains are rather scarce. Several studies have explored traffic control and multi-vehicle navigation problems using classical planning and its extensions, focusing on coordinating vehicles to improve traffic flow and management [52]–[56]. While these works consider vehicle movement and routing tasks, they primarily focus on traffic optimisation rather than modelling the behaviour of individual autonomous vehicles.

Several works use probabilistic planning to model the uncertainty of the self-driving car environments, e.g., [57]–[59]. These works account for uncertainty through Partially Observable Markov Decision Processes (POMDPs), thereby relying on a different planning technique than our approach. Moreover, they do not model risk.

Research on route planning in AI planning is also relevant to self-driving car applications. For example, several works investigate route planning under uncertainty in marine environments, such as those involving autonomous underwater vehicles [60]. Although such vehicles share some navigation tasks with autonomous terrestrial vehicles, their operational environments and mission objectives differ significantly. Other works address route planning within a temporal HTN planning framework in the aerial domain, e.g., [61], focusing on challenges specific to aerial vehicles, such as flight dynamics and airspace constraints, which differ significantly from those encountered in autonomous ground vehicle scenarios.

IX. Conclusions

Developing operational AI planning systems requires domain models that capture the characteristics of real-world environments. However, many existing planning domains remain simplified and omit important aspects such as uncertainty, risk, and diverse ways of achieving tasks. We addressed this limitation by focusing on the systematic engineering and analysis of realistic HTN planning domains.

Our contributions are twofold. First, we proposed a systematic domain-engineering approach for developing realistic HTN planning domains. The approach integrates the identification of relevant domain aspects, their formulation using HTN constructs, their encoding in a domain model using a planning language, and a quantitative analysis of the resulting domain models. Second, we applied this approach to develop and analyse three HTN planning domains: two extended benchmark domains, Rover and Satellite, and a newly engineered domain named Self-Driving Car. In these domains, we incorporated several aspects that are often absent from existing domain models, including richer alternative task decompositions, explicit representations of uncertainty and risk, and considerations of recent technological developments in the respective application areas.

Our analysis shows that incorporating these aspects leads to domains with increased structural diversity and richer representations of application knowledge. The quantitative evaluation

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

demonstrates that the developed domain models contain more alternative methods, more expressive task structures, and richer object and predicate relationships compared to their original counterparts. These properties allow the domain models to better reflect the complexity of real-world environments and provide a more informative basis for evaluating planning systems.

The quantitative analysis also reveals recurring structural patterns in how HTN domains evolve when enriched with realistic aspects. Extensions of existing domains manifest either as *uniform growth in structural constructs* or as *structural reorganisation* that introduces additional alternatives and hierarchical relationships. In contrast, engineering domains from scratch may exhibit *high procedural complexity* from the outset through richer task decompositions and action sets. These patterns suggest that systematic domain engineering not only increases domain richness but also influences how planning knowledge is conceptualised and structured.

Finally, richer and more realistic domains may also stimulate advances in planning technology itself. As demonstrated by the historical influence of benchmark domains introduced through IPCs, the availability of more representative domains can drive the development of planners capable of handling increasingly complex planning problems.

As future work, we plan to engineer more HTN planning domains by incorporating realistic aspects, considering also aspects not addressed in this treatment. We also aim to explore systematic methods for evaluating realism in planning domains and to investigate how richer domains influence the design and performance of HTN planning systems. Together, these directions will contribute to bridging the gap between domain models and operational planning applications.

Acknowledgements

During the preparation of this work, the authors used ChatGPT to polish self-authored text and Claude to visualise the radar charts. After using the tool, the authors reviewed and edited the content as needed, and take full responsibility for the content of the work.

References

[1] E. Alnazer, I. Georgievski, and M. Aiello, “Risk-Aware HTN Planning Domain Models for Autonomous Vehicles and Satellites”, in *International Conference on AI-based Systems and Services (AISyS)*, 2025, pp. 36–45.

[2] M. Ghallab, D. Nau, and P. Traverso, *Automated Planning: Theory and Practice*. Elsevier, 2004.

[3] I. Georgievski and M. Aiello, “HTN Planning: Overview, Comparison, and Beyond”, *Artificial Intelligence (AIJ)*, vol. 222, pp. 124–156, 2015.

[4] T. L. McCluskey, T. S. Vaquero, and M. Vallati, “Engineering Knowledge for Automated Planning: Towards a Notion of Quality”, in *International Conference on Knowledge Capture (K-CAP)*, Association for Computing Machinery, 2017, pp. 1–8.

[5] I. Georgievski, “Software Development Life Cycle for Engineering AI Planning Systems”, in *International Conference on Software Technologies (ICSOFT)*, 2023, pp. 751–760.

[6] J. Hoffmann et al., “Engineering Benchmarks for Planning: The Domains Used in the Deterministic Part of IPC-4”, *Journal of Artificial Intelligence Research (JAIR)*, vol. 26, pp. 453–541, 2006.

[7] E. Alnazer, I. Georgievski, and M. Aiello, “On Bringing HTN Domains Closer to Reality-The Case of Satellite and Rover Domains”, in *International Conference on Automated Planning Systems (ICAPS) Workshop on Scheduling and Planning Applications (SPARK)*, 2022.

[8] E. Alnazer, I. Georgievski, and M. Aiello, “Risk Awareness in HTN Planning”, *arXiv preprint arXiv:2204.10669v2*, 2025.

[9] M. Vallati and L. McCluskey, “A Quality Framework for Automated Planning Knowledge Models”, in *International Conference on Agents and Artificial Intelligence (ICAART)*, SciTePress, 2021, pp. 635–644.

[10] I. Georgievski, “Conceptualising Software Development Lifecycle for Engineering AI Planning Systems”, in *International Conference on AI Engineering – Software Engineering for AI (CAIN)*, IEEE, 2023, pp. 88–89.

[11] E. Alnazer and I. Georgievski, “Understanding Real-World AI Planning Domains: A Conceptual Framework”, in *Symposium and Summer School on Service-Oriented Computing (SummerSoC)*, Springer, 2023, pp. 3–23.

[12] I. Georgievski, *Engineering AI Planning Systems*, Habilitation thesis, University of Stuttgart, 2025.

[13] E. Turan, S. Speretta, and E. Gill, “Autonomous Navigation for Deep Space Small Satellites: Scientific and Technological Advances”, *Acta Astronautica*, vol. 193, pp. 56–74, 2022.

[14] D. Long and M. Fox, “The 3rd International Planning Competition: Results and Analysis”, *Journal of Artificial Intelligence Research (JAIR)*, vol. 20, pp. 1–59, 2003.

[15] M. Weser, D. Off, and J. Zhang, “HTN Robot Planning in Partially Observable Dynamic Environments”, in *International Conference on Robotics and Automation (ICRA)*, IEEE, 2010, pp. 1505–1510.

[16] J. Fdez-Olivares, L. Castillo, J. A. Cózar, and O. Garcia Perez, “Supporting Clinical Processes and Decisions by Hierarchical Planning and Scheduling”, *Computational Intelligence (CI)*, vol. 27, no. 1, pp. 103–122, 2011.

[17] I. Georgievski et al., “Planning Meets Activity Recognition: Service Coordination for Intelligent Buildings”, *Pervasive and Mobile Computing (PMC)*, vol. 38, no. 1, pp. 110–139, 2017.

[18] J. R. Silva, J. M. Silva, and T. S. Vaquero, “Formal Knowledge Engineering for Planning: Pre and Post-Design Analysis”, *Knowledge Engineering Tools and Techniques for AI Planning*, pp. 47–65, 2020.

[19] I. Georgievski, “Hierarchical Planning Definition Language”, University of Groningen, Tech. Rep. JBI 2013-12-3, 2013.

[20] D. Höller et al., “HDDL: An Extension to PDDL for Expressing Hierarchical Planning Problems”, in *AAAI Conference on Artificial Intelligence*, AAAI Press, 2020, pp. 9883–9891.

[21] G. Behnke, D. Höller, and P. Bercher, Eds., *Proceedings of the 10th International Planning Competition: Planner and Domain Abstracts – Hierarchical Task Network (HTN) Planning Track (IPC 2020)*, 2021.

[22] P. Bercher, G. Behnke, D. Höller, and S. Biundo, “An Admissible HTN Planning Heuristic.”, in *International Joint Conferences on Artificial Intelligence (IJCAI)*, 2017, pp. 480–488.

[23] G. Behnke, D. Höller, and S. Biundo, “totSAT-Totally-ordered Hierarchical Planning Through SAT”, in *AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[24] B. Schattenberg, “Hybrid Planning & Scheduling”, Ph.D. dissertation, Ulm, Univ., Diss., 2009, 2009.

[25] R. C. Anderson et al., “Collecting Samples in Gale Crater, Mars; An Overview of the Mars Science Laboratory Sample Acquisition, Sample Processing and Handling System”, *Space*

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Science Reviews (Space Sci. Rev.), vol. 170, no. 1, pp. 57–75, 2012.

[26] D. Höller, P. Bercher, G. Behnke, and S. Biundo, "A Generic Method to Guide HTN Progression Search with Classical Heuristics", in *International Conference on Automated Planning and Scheduling (ICAPS)*, vol. 28, 2018, pp. 114–122.

[27] D. Pellier and H. Fiorino, "From Classical to Hierarchical: Benchmarks for the HTN Track of the International Planning Competition", in *International Planning Competition (IPC)*, 2020.

[28] C. Carr et al., "Space Drones: An Opportunity to Include, Engage, Accelerate, and Advance", in *Bulletin of the American Astronomical Society*, vol. 53, 2021.

[29] N. Potter, "A Mars Helicopter Preps for Launch: The First Drone to Fly on Another Planet will Hitch a Ride on NASA's Perseverance Rover", *IEEE Spectrum*, vol. 57, no. 7, pp. 06–07, 2020.

[30] C. Powell, C. S. Ruf, S. Gleason, and S. C. Rafkin, "Sampled Together: Assessing the Value of Simultaneous Collocated Measurements for Optimal Satellite Configurations", *Bulletin of the American Meteorological Society (BAMS)*, vol. 105, no. 1, E285–E296, 2024.

[31] V. Maggioni, C. Massari, and C. Kidd, "Errors and Uncertainties Associated with Quasiglobal Satellite Precipitation Products", in *Precipitation Science*, Elsevier, 2022, pp. 377–390.

[32] G. A. Landis, S. G. Bailey, and R. Tischler, "Causes of Power-Related Satellite Failures", in *World Conference on Photovoltaic Energy Conference (EU PVSEC)*, IEEE, 2006, pp. 1943–1945.

[33] P. S. Morgan, "Fault Protection Techniques in JPL Spacecraft", in *Integrated System Health Engineering and Management in Aerospace (ISHEM)*, 2005.

[34] H.-S. Choi et al., "Analysis of GEO Spacecraft Anomalies: Space Weather Relationships", *Space Weather*, vol. 9, no. 6, 2011.

[35] R. Horne et al., "Space Weather Impacts on Satellites and Forecasting the Earth's Electron Radiation Belts with SPACECAST", *Space Weather*, vol. 11, no. 4, pp. 169–186, 2013.

[36] C. C. Liu, S. G. Hosking, and M. G. Lenné, "Predicting Driver Drowsiness Using Vehicle Measures: Recent Insights and Future Challenges", *Journal of Safety Research*, vol. 40, no. 4, pp. 239–245, 2009.

[37] N. R. Council et al., *Completing the Forecast: Characterizing and Communicating Uncertainty for Better Decisions Using Weather and Climate Forecasts*. National Academies Press, 2006.

[38] *Enhancing Weather Information with Probability Forecasts*, https://shorturl.at/RFuna, Accessed: 2026-05-15, 2026.

[39] J. Antoun, M. E. Kabir, R. F. Atallah, and C. Assi, "A Data Driven Performance Analysis Approach for Enhancing the QoS of Public Charging Stations", *Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 23, no. 8, pp. 11 116–11 125, 2021.

[40] *Google Maps Directions for Driving from Stuttgart to Erlangen*, https://shorturl.at/O2jdL, Accessed: 2026-03-15, 2026.

[41] *Google Maps Directions for Driving from Stuttgart to Erlangen*, https://shorturl.at/H7riv, Accessed: 2026-03-15, 2026.

[42] P. Farradyne, "Traffic Incident Management Handbook", *Prepared for Federal Highway Administration, Office of Travel Management*, 2000.

[43] I. Georgievski and M. Aiello, "Phantomisation in State-Based HTN Planning", in *International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI)*, 2019, pp. 39–44.

[44] M. Fox and D. Long, "PDDL2. 1: An Extension to PDDL for Expressing Temporal Planning Domains", *Journal of Artificial Intelligence Research (JAIR)*, vol. 20, pp. 61–124, 2003.

[45] D. Wang, J. A. Russino, C. Basich, and S. Chien, "Analyzing the Efficacy of Flexible Execution, Replanning, and Plan Optimization for a Planetary Lander", in *International Conference on Automated Planning and Scheduling (ICAPS)*, vol. 32, 2022, pp. 518–526.

[46] F. D. S. Cividanes, M. G. V. Ferreira, and F. de Novaes Kucinskis, "An Extended HTN Language for Onboard Planning and Acting Applied to a Goal-Based Autonomous Satellite", *Aerospace and Electronic Systems Magazine*, vol. 36, no. 8, pp. 32–50, 2021.

[47] W. M. Piotrowski, M. Fox, D. Long, D. Magazzeni, F. Mercorio, et al., "Heuristic Planning for PDDL+ Domains", in *AAAI Workshop: Planning for Hybrid Systems*, vol. 16, 2016, p. 12.

[48] W. M. Piotrowski, "Heuristics for AI Planning in Hybrid Systems", Ph.D. dissertation, King's College London, 2018.

[49] J. D. Victoria, N. Policella, Y. Gao, and O. von Stryk, "Design Concepts for a New Temporal Planning Paradigm", in *International Conference on Automated Planning and Scheduling (ICAPS) Workshop on Planning and Scheduling with Timelines (PSTL)*, 2012, p. 16.

[50] P. Bercher, S. Keen, and S. Biundo, "Hybrid Planning Heuristics Based on Task Decomposition Graphs", in *International Symposium on Combinatorial Search (SOCS)*, vol. 5, 2014, pp. 35–43.

[51] M. D. Rodríguez-Moreno, D. Borrajo, and D. Meziat, "An AI Planning-based Tool for Scheduling Satellite Nominal Operations", *AI Magazine*, vol. 25, no. 4, pp. 9–9, 2004.

[52] F. Jimoh, L. Chrpa, T. L. McCluskey, and S. Shah, "Towards Application of Automated Planning in Urban Traffic Control", in *International Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2013, pp. 985–990.

[53] M. Vallati, D. Magazzeni, B. De Schutter, L. Chrpa, and T. McCluskey, "Efficient Macroscopic Urban Traffic Models for Reducing Congestion: A PDDL+ Planning Approach", in *AAAI Conference on Artificial Intelligence*, vol. 30, 2016.

[54] M. Gulić, R. Olivares, and D. Borrajo, "Using Automated Planning for Traffic Signals Control", *PROMET-Traffic&Transportation*, vol. 28, no. 4, pp. 383–391, 2016.

[55] T. McCluskey and M. Vallati, "Embedding Automated Planning Within Urban Traffic Management Operations", in *International Conference on Automated Planning and Scheduling (ICAPS)*, vol. 27, 2017, pp. 391–399.

[56] F. Ivankovic and M. Roveri, "Planning with Global State Constraints for Urban Traffic Control", in *CEUR WORKSHOP PROCEEDINGS*, CEUR-WS, vol. 2987, 2021, pp. 1–5.

[57] X. Dong et al., "Why Autonomous Vehicles Are Not Ready Yet: A Multi-Disciplinary Review of Problems, Attempted Solutions, and Future Directions", *Journal of Field Robotics*, 2025.

[58] S. Brechtel, T. Gindele, and R. Dillmann, "Probabilistic Decision-making Under Uncertainty for Autonomous Driving Using Continuous POMDPs", in *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2014, pp. 392–399.

[59] S. Ulbrich and M. Maurer, "Probabilistic Online POMDP Decision Making for Lane Changes in Fully Automated Driving", in *International IEEE Conference on Intelligent Transportation Systems (ITSC)*, IEEE, 2013, pp. 2063–2067.

[60] T. X. Lin, M. Hou, C. R. Edwards, M. Cox, and F. Zhang, "Bounded Cost HTN Planning for Marine Autonomy", in *Global Oceans 2020: Singapore–US Gulf Coast*, IEEE, 2020, pp. 1–6.

[61] J. J. Kiam, P. Bercher, and A. Schulte, "Temporal Hierarchical Task Network Planning with Nested Multi-Vehicle Routing Problems — A Challenge to be Resolved", in *International Conference on Automated Planning and Scheduling (ICAPS) Workshop on Hierarchical Planning (HPlan)*, 2021, pp. 71–75.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Appendix A
Problem Instances

A. Rover Domain

Planning problems in the Rover domain represent the exploration environment through the specification of the initial state, the initial task network, and the domain objects. The initial state captures both static and dynamic knowledge of the environment. Static knowledge describes properties that remain unchanged during execution. These include the visibility relationships of the waypoints, the visibility of calibration objectives from the waypoints, the locations of rock, soil, and image samples, and the capabilities of each rover. The rover capabilities specify whether a rover can sample soil, rock, or image data, and which terrain types it can traverse between waypoints. Additional static knowledge includes the lander's location, the cameras mounted on each rover, and the calibration targets and operating modes supported by these cameras. Dynamic knowledge describes the evolving state of the system during execution. This includes the current locations of rovers and drones and the status of the rover's storage units, i.e., whether they are empty or contain collected samples. The initial task network includes tasks for sampling rock and soil data from specified locations and for taking images in a specific mode. The objects of the planning problem include various waypoints, drones, available cameras and their operating modes, rovers and their storage units, the lander, and the objectives for taking images.

B. Satellite Domain

Planning problems in the Satellite domain represent the observation environment, where the initial state captures both static and dynamic knowledge of the environment. Static knowledge includes properties about each satellite's instrument, supported modes, and the calibration target of each instrument. Dynamic knowledge includes properties about the available power and current pointing direction of individual satellites. The initial task network includes all required observation missions, each targeting a phenomenon with a specified mode. The objects of the planning problem include the satellite, instruments, calibration directions, modes, and image directions.

C. Self-Driving Car Domain

Knowledge about planning problems is formulated such that the initial state contains the knowledge about the static and the dynamic states of the environment. The static state includes the road network, i.e., locations and routes connecting them, the location of still and moving incidents (e.g., construction works and pedestrians), the conditions of roads (e.g., slippery roads, roads with gravel, and normal roads), and the length of the roads. The dynamic states include the location of the car. The initial task network in the planning problem is to move the car from one location to another. The objects are the locations that the car can navigate to and the various incidents.

Let us consider an example of a planning problem with seven different locations denoted as S, l_1, l_2, l_3, l_4, l_5, and E, depicted in Figure 11. The car is initially at S and should navigate to E while handling the various road complexities. We assume that the car has enough power to navigate all the roads, and it is nighttime, so the car has to turn on the lights. The roads between S and l_1 and between S and l_3 are complexity-free. However, the first road is longer than the second. The road between l_1 and l_4 has constructions at location l_2. The road between l_3 and l_4 has a school area that is very crowded with pedestrians and traffic jams, i.e., moving incidents, at location l_5. The road between l_1 and l_3 is a highway free from any complexities, but has a variable level of traffic congestion. Finally, the road between l_4 and E is icy and slippery.

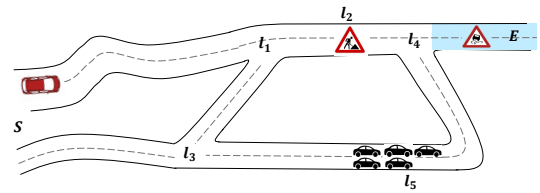


Figure 11. A problem instance in the Self-Driving Car domain with seven locations S, l_1, l_2, l_3, l_4, l_5, and E, and various road complexities.

Figure 9 shows some possible bindings of the actions with the corresponding probability distribution of costs with respect to the ground planning problem and external traffic as sources of uncertainty. Note that we only show feasible bindings and ground actions that can be part of the computed plans. For example, the action of accelerating has a probability distribution of costs when the car is accelerating on the road between S and l_1, different from the probability distribution of costs when accelerating on the road between S and l_3, since the length and traffic jams of these roads are different. The logic behind our assignments of the action variable costs is as follows. The road between S and l_1 is complexity-free and free of traffic jams. Thus, the time needed to drive through this road is certain, i.e., driving through this road takes four hours with 100% probability. On the other hand, although shorter, the road between S and l_3 has a variable traffic jam throughout the day. Driving through this road can take six hours with 20% probability, or two hours, in the best case, with 80% probability. Thus, taking the short road is riskier than taking the long road. The road between l_1 and l_4 has construction work at location l_2. Since the construction is considered a still incident, passing through this road, i.e., decelerating, dodging the constructions, and accelerating again, is done in a certain time under the assumption that this road does not include any traffic jam. In particular, cumulative deceleration on the road between l_1 and l_2 requires four hours, dodging the incident requires 0.4 hours, and accelerating on the road between l_2 and l_4 takes one hour. Unlike the road between l_1 and l_4, the road between l_3 and l_4 has multiple schools and heavy traffic at location l_5, which can lead to long waiting times. This is an external source of uncertainty since pedestrians have uncertain times and speeds at which they cross the road, which can make this area congested. Additionally, the roads between l_3 and l_5, and l_5 and l_4 have an uncertain level of traffic congestion. Thus, driving from l_3 to l_4, i.e., decelerating, braking, and

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

accelerating, requires a variable amount of time, as shown in Figure 9. In particular, decelerating on the road between l_3 and l_5 takes one hour with 10% probability and three hours with 90% probability. Consider the school area is very crowded, and the car might need to brake for a long time, waiting for the pedestrians to walk. Thus, braking before this area can take half an hour with a high probability of 90% and can, in the worst case, take three hours with a probability of 10%.

The road between l_1 and l_3 is a highway that is complexity-free. However, it has an uncertain level of traffic congestion. Thus, although the car can accelerate on this road, with a small probability of 10%, it can take eight hours to reach l_3 when there is a high traffic congestion. With a high probability of 90%, the car can travel from l_1 to l_3 within two hours since the highway is mostly free of traffic jams. Lastly, the road between l_4 and E is icy and slippery. If the agent chooses to decelerate without activating the ESP, the time needed to reach E will be variable and is based on the ability of the car to stabilise on this slippery road. This choice can lead to six hours of driving with 20% probability and eleven hours of driving with 80% probability. If the agent chooses to decelerate after activating the ESP, it will need 10 hours, i.e., a known and certain amount of time, to reach E since this is the safest and most guaranteed option to choose. However, suppose the agent accelerates after activating the ESP. In that case, it will need an uncertain amount of time, depending on the car's stability. This option is very risky since it might require 12 hours in the worst case with an 80% probability as the car will probably lose stability. At the same time, there is a 20% probability that the car will have good stability and reach its destination in two hours since it is accelerating. An even riskier option is to accelerate on this road without activating the ESP. In that case, the car might need sixteen hours to reach location E with a probability of 90%, and, in the best case, it needs half an hour with a probability of 10%. Comparing the option of decelerating after activating the ESP with the option of decelerating without activating the ESP, the first option has a more guaranteed outcome, i.e., execution time, although both options have the same expected value, which is 10 hours. We can also see that the option of accelerating after activating the ESP is riskier than decelerating without activating the ESP, since, with an 80% probability, it might lead to a higher time than the outcome of only decelerating. Lastly, when comparing the options of decelerating without activating the ESP and only decelerating, we see that both options involve risk. However, unless the agent is highly risk-seeking, it is less likely that the first option is preferable since it has the probability of 20% of costing six hours compared to the second option, which has the probability of costing four hours less with the same 20% probability. At the same time, the first option has an 80% probability of costing eleven hours compared to twelve hours for the second option, with the same probability, which means a one-hour difference only. Note that, despite the differences in the risk level, all three options have the same expected value, which is ten hours. The only option that has a higher expected travelling time compared to all other options is accelerating without activating the ESP. That is why this option is the riskiest one and is only chosen if the agent is extremely risk-seeking. Additionally, we extend the possible roads that the car can take by adding the possibility of taking a shortcut road that has a 90% probability of taking two hours and a 10% probability of taking eight hours. This extension increases the number of choices the agent should make according to its risk attitude, since taking the shortcut road can have a high risk of incurring long travelling times compared to taking several longer roads.

We model the objects existing in the planning problem, the initial task network is to drive to a destination, and the initial state is a list of predicates. Listing 7 shows the initial state of the problem instance illustrated in Figure 11.

Listing 7: Initial state of the problem instance illustrated in Figure 11.

```
(:init (connected s l1)(connected s l3)(
    connected l1 l4)(connected l3 l4)(
    connected l4 e)(connected l1 l2)(connected
     l2 l4)(connected l3 l5) (connected l5 l4)
    (clearroadlong s l1)(clearroad s l1)(
    clearroadshort s l3)(clearroad s l3)(
    in-inc construction l2)(in-inc bottleneck
    l5)(clearroad l4 e)(badroad l4 e)(in v0 s)
    )
```

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Design of a Hierarchical Communications Network for Public Services

Maria-Georgiana Butaru
National University of Science and Technology
POLITEHNICA
Bucharest, Romania
e-mail: georgianabutaru8@gmail.com

Eugen Borcoci
National University of Science and Technology
POLITEHNICA
Bucharest, Romania
e-mail: eugen.borcoci@elcom.pub.ro

***Abstract* – Wide Area Networks (WAN) dedicated to public services for large communities of users are usually hierarchically organized, consisting of regional and local networks to support media and data services. Media communications (e.g., voice) benefit from specialized protocols designed to meet real-time communications requirements. A regionalized hierarchical infrastructure solution determines the scalability of the overall system. A centralized monitoring system is included, supervising the equipment and services status, thus achieving a high-performance network. In this paper, the design of such a network and verification of all protocols and services are performed, using the Graphical Network Simulator-3 (GNS3). This work builds on previous research that developed a communications network providing public services for citizens. The current paper extends that work by including in-depth analysis, additional configured services, relevant examples, further experimental evaluations and additional developments considerations. Also, useful insights are provided, for network designers and operators involved in the design and implementation and management of large-scale communication infrastructures.**

Keywords - communication networks; technical requirements; public services; experiments; centralized administration and monitoring.

I. INTRODUCTION

This paper is an extended version of the paper [1], which was dedicated to the design and implementation of a communications infrastructure to ensure the provision of public services to citizens. The system represents a complex environment with multiple functional and non-functional requirements. User requirements (functional, quality of service (QoS, availability, security, etc.) must be fulfilled. The architecture should be open to new networking technologies. This work addresses in more detail the designed and configured infrastructure.

In the public sector, communities should be provided with easy access to government information, forms or online services. The availability of such services for a public environment stimulates transparency and interaction with public authorities [2]. At the level of public institutions, it also facilitates communication between different government and public entities at local, regional or national. Communication networks play an essential role in deploying services, automating operations, and stimulating innovation. Therefore, digitalization contributes not only to the improvement of individual life, but also to the modernization of society as a whole.

The main objective of this paper is the design and implementation of a communications infrastructure to provide public services to citizens, e.g., issuing a passport, issuing an identity card, issuing a criminal record, driving license or health card.

The aim is to offer all public services intended for citizens into a single physical location, practically an integrated environment that brings numerous benefits.

The network is intended to be deployed at the national level and will be based on a WAN architecture, composed of medium-sized networks regional networks. Each regional network will consist of several local networks arranged at the county level. At the county level, it is possible to obtain a public service from an integrated environment, having all the options in a single location. Such feature will improve convenience and efficiency for citizens. Including technological innovations, will results in a higher performance, shorter response time in the process of obtaining a requested public service.

The designed proposed network, as well as the necessary protocols and functionalities will be implemented and simulated in GNS3 [3].

The initial paper [1] on which this extension is based, developed a WAN network capable of integrating multiple services scalable, reliable and convergent.

This extension brings additional improvements by defining a disaster recovery plan to ensure the optimal functioning of the services offered even in the event of natural disasters. Another improvement is the possibility of migrating the data center network topology to an Application Centric Infrastructure (ACI) network that brings to the forefront network centralization and its automation through application policy infrastructure controller (APIC) [4].

This paper is structured in 8 sections. Section I is an introduction with information about communications networks, technological innovation and digitalization. Section II identifies the system technical requirements. Section III includes the definition of the system architecture based on the technical and user requirements. Section IV focuses on the simulation scenarios of the network, some tests to verify the functionalities of the network before going

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

to production. Section V will present the experiments and results and also contain other special subsections that represent all the data flows, protocols and functionalities implemented. Section VI outlines some possible extensions of the network and shows how a disaster recovery plan can be made. Section VII contains details about how the network can be deployed, how the entire infrastructure can be improved. Section VIII contains the conclusion of this paper and future work.

II. GENERAL REQUIREMENTS OF THE NETWORK FOR PUBLIC SERVICES AND POTENTIAL SOLUTIONS

Still, in practice, to obtain a public service (obtaining a passport, identity card, criminal record, driver's license or health card), a citizen must physically travel to several locations in a city. A solution to solve this could be a centralized network, where all the public services listed above are integrated, would create a more comfortable environment for a citizen to access and complete the entire process of obtaining a document, and also will allow a centralized and efficient administration.

Such a system should meet a series of general requirements, related to functionality, performance/ efficiency, security, and availability of services,

This section will summarise the requirements and also outlines some possible solutions. We classify the requirements in two main categories: technical (functional, performance) and business derived requirements.

Functional requirements

- *Service availability*- all services must be available while being conformant to their specific characteristics, for different, defined classes of users. Service Level Specifications will provide qualitative and quantitative characteristics of each service.
- *Infrastructure scalability* it has two complementary dimensions: horizontal, through the ability to extend the system coverage, i.e., add new nodes without downtime or architectural redesign, and vertical, through upgrading the capacity of existing equipment. The implemented Three-Tier architecture can allow vertical scalability at each layer level, but introduces limitations at the Distribution level, where uplinks become a potential bottleneck as the number of devices increases [5].
- *Infrastructure resilience* – can be quantified by two fundamental indicators specified in the literature: Recovery Time Objective (RTO) and Recovery Point Objective (RPO). In this paper, the Hot Standby Routing Protocol (HSRP) mechanism with optimized timers (Hold=300ms) ensures an RTO of less than 1 second at the distribution gateway level, a value that falls into the High Availability category according to the ISO 22301 standard [6]. Redundancy in terms of links and servers will assure a high degree of resiliency.
- *Infrastructure security* - is structured on the Defense in Depth principle, by overlapping multiple layers of independent technical control, so that compromising a single layer does not lead to exposure of the entire system. At the physical network access level, Port Security with static MAC addresses offers a basic first line of defense. However, its scalability limitations make it unsuitable for larger or more dynamic environments, necessitating a transition to more robust and scalable access control mechanisms. At the perimeter level, a Next Generation Firewall (NGFW) with Deep Packet Inspection (DPI) inspection capabilities can filter the traffic, based on application content, not just port and protocol, providing granular visibility and control. VLAN segmentation, correlated with strict inter-VLAN routing policies, limits lateral propagation in the event of a segment compromise, implementing the principle of least privilege at the network level [7].
- *Access management* - administrative access to network equipment is a critical component of infrastructure security, often neglected in favor of perimeter protection. As an example, the Terminal Access Controller Access-Control System Plus (TACACS+) protocol, developed by Cisco and widely adopted in enterprise environments, provides a clear separation of the three AAA functions: Authentication (verifying the administrator's identity), Authorization (defining the commands allowed per user or group), and Accounting (complete recording of all actions performed). Unlike the old RADIUS, which encrypts only the password field, TACACS+ encrypts the entire session payload, providing superior protection against eavesdropping. The granularity of authorization allows the definition of differentiated access profiles: network operators can be limited to monitoring (show) commands, while senior administrators have full access, with all actions being recorded in a centralized audit trail with a minimum retention of 12 months according to the NIS2 Directive [8].

Performance requirements

- *Quality of Service (QoS)* – In an infrastructure that centralizes public services, several types of flows are transported on shared paths: media ones (e.g. VoIP) but assigns differentiated weights to each class, while best-effort traffic is served from the remaining available resources [10].

  In terms of bandwidth allocation, QoS policies define minimum guaranteed reservations per traffic class. Example: VoIP (RTP) traffic, marked with Expedited Forwarding (EF – DSCP 46), benefits from a reservation of 10–15% of the total bandwidth of a path, Session Intiation Protocol IP signaling (CS3 – DSCP 24) 5%, critical applications (Assured Forwarding AF41 – DSCP 34) 30%, administrative data (AF21 – DSCP 18) 25%, network management (CS6 – DSCP 48) 5%, and best-effort traffic consumes the remaining available resources [11].

  From a latency and jitter perspective, the ITU-T G.114 standard requires VoIP traffic to have an end-to-end latency of no more than 150 ms and a jitter of

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

no more than 20 ms for acceptable call quality. Exceeding these thresholds results in perceptible degradation of voice quality, quantifiable by a decrease in the Mean Opinion Score (MOS) below 4.0, considered the minimum threshold for good quality [12].

The identification of the technical set of requirements, will provide inputs in order to proceed with the system designed for implementing the target communications.

Business requirements

These are related to the business aspects and operation of the system also web applications and other data applications flows Different services requires require specific level of QoS assurance. The classical and scalable model of mechanims to control QoS in IP networks is the Differentiated Services (DiffServ), which classifies traffic into service classes by marking the Differentiated Services Code Point (DSCP) field in the IP header. Traffic classification and marking is performed by the source, or at the network access level. The set of mechanism like classification, marking, policing, shaping, queuing and packet scheduling can implement QoS with different levels of guarantees and also prevent network congestion [9].

From a prioritization perspective, traffic can be organized into service queues with differentiated treatment. VoIP traffic benefits from Low Latency Queuing (LLQ), a strict priority queue that guarantees immediate service of voice packets before any other type of traffic. Critical application traffic can be managed through (Class-Based Weighted Fair Queuing (CBWFQ).

- *Avoid vendor lock-in* - exclusive dependence on a single manufacturer of network equipment (vendor lock-in) is a risk that should be avoided for a public infrastructure. Therefore, the adoption of open and standardized protocols at all architectural levels is a solid choice. For instance, at the routing level, Open Shortest Path First (OSPF, RFC 2328) and Border Gateway Protocol (BGP, RFC 4271) standard protocols ensure interoperability between equipment from different manufacturers. The vendors used in this infrastructure are Cisco and Fortinet [13], [14].
- *System management and monitoring* - a monitoring system will be developed supervising both functional and performance aspects including assurance of a rapid response in case of urgent events. Active and continuous supervision of the network status is necessary to ensure correct operation, identify problems and optimize performance.

Implementation approach

The GNS3 simulator has been used in this paper to design and simulate the entire infrastructure. It consists of servers and clients. The client is installed directly on the physical equipment. The client-server interconnection is based on the server's Internet Protocol (IP).

Also, for configuring the network equipment and making connections between them, IOS images containing the operating system are required depending on the type of functional block: switch, router, security equipment (firewalls), images with the Linux operating system for network monitoring and voice communications implementation.

Last but not least, images with the Windows 10 and Windows Server 2012 operating systems are used for configuration of the infrastructure domain, but also for adding workstations at the local network levels in the domain for their uniform administration.

III. COMMUNICATIONS NETWORK ARCHITECTURE

The proposed system is hierarchically organized. It contains a WAN communications network, implemented at the national level. At lower level are county networks. Several county networks form a regional network that will be connected through concentrators to the single data center network of this infrastructure. The data center will collect and process all the information from the local networks. A simplified overall view of this hierarchy is presented in Figure 1.

The county-level network links will not be connected directly in the data center, but will be interconnected by two concentrators, in redundancy, and then, through external firewalls; the connection to the data center with the network equipment in the WAN area, (Figure 2) of the data center network topology will be established. In addition to the implemented network links. the data center will store the system administration and monitoring servers within this network. All data will be processed only at the data center level at the central point of the network. At the county level only network equipment and workstations will be used to handle the data taken and to transmit them to the central network. In our experiment two regional networks will be connected to the central network through two concentrators (1 and 2), with traffic being filtered and redirected through the two external firewalls (1 and 2).

In addition to local and regional networks, several zones are defined within the network (to complete the infrastructure), such as the central zone, which centralizes all data via the WAN network to the data center network.

The WAN network provides connections for internal users to the infrastructure resources through the two configured hubs.

The design enhances security through firewalls and additional protection mechanisms designed to improve performance and reliability. For monitoring. the Zabbix solution will be used; it will collect in real time the data, visualizes and analyzes them; based on these it creates graphs and statistics necessary for achieving awareness on the system and optimizations

A special service configured in this infrastructure is Voice over IP (VoIP) telephony, integrated with the SIP (Session Initiation Protocol) control protocol. VoIP technology allows the transmission of packetized voice flows over IP networks, offering much cheaper and flexible solutions w.r.t traditional telephony. SIP protocol is used to initiate, manage and terminate real-time communication sessions.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

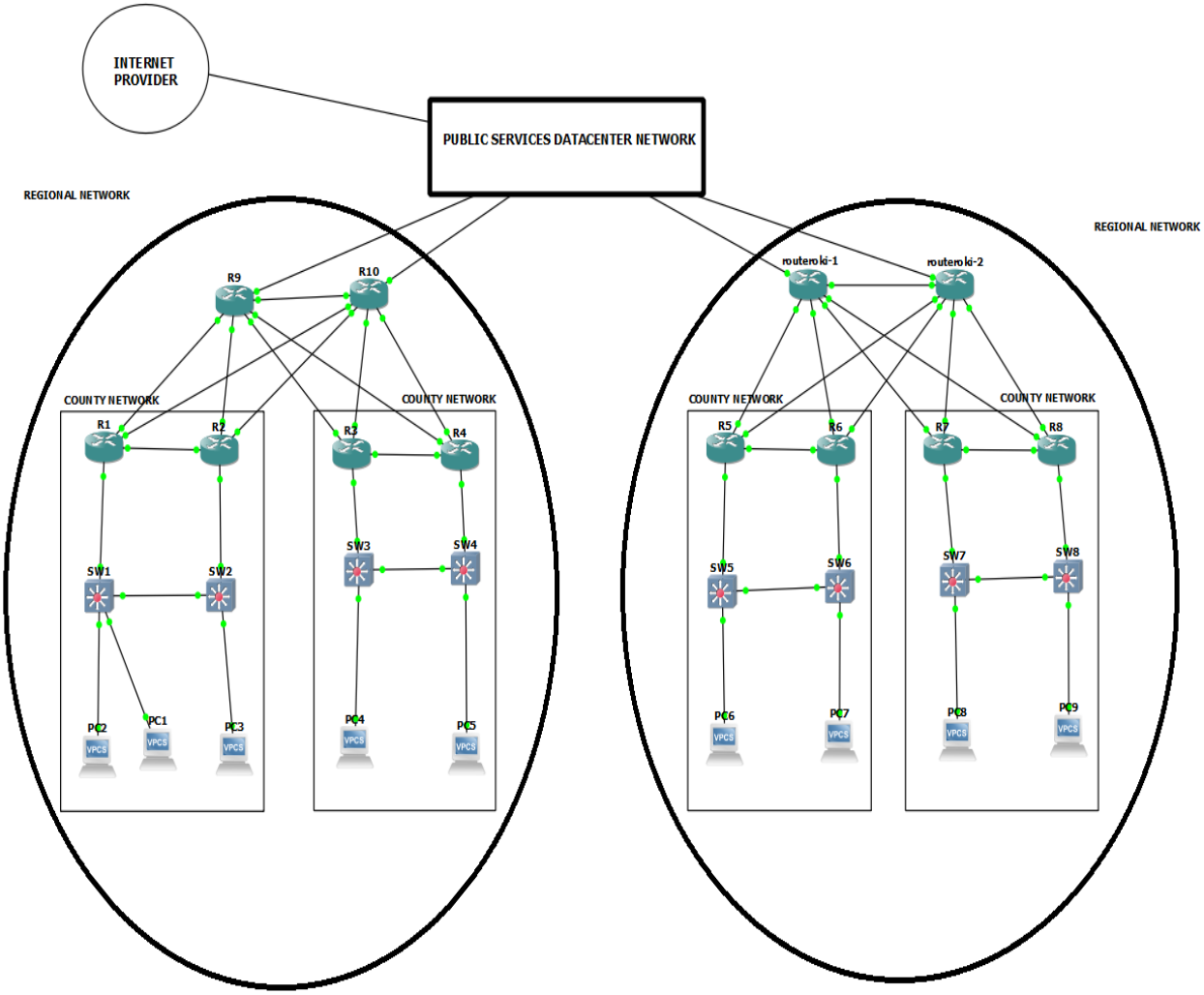


Figure 1. Communication network – overall view

In this work, a complete telephony solution based on SIP integration in a communications infrastructure is implemented, using Asterisk as a communications server and Twinkle [15] as a client application.

In addition to the integrated firewall filters, a complementary security-related measure refers to GRE (Generic Routing Encapsulation) tunnels, to support their confidentiality and integrity. Through the firewalls plus the configured GRE tunnels, the confidentiality, integrity and access of data is ensured. To complete the set of security measures, port-security will also be configured at the access level, on the switch interfaces that connect workstations or IP phones to the communications network. Moreover, port-security configured at the network equipment level, significantly contributes to the security of the entire communications infrastructure.

Given the large number of entities, dynamic IP routing will be used, which supports the network scalability. Since it is a large network, adding or removing equipment can be frequent. To establish connections between local networks and the central area of the network, the following routing protocols are used: OSPF, EIGRP (Enhanced Interior Gateway Routing Protocol), MPLS (Multiprotocol Label Switching), as well as BGP for connections between regional networks and the central network (data center). Depending on the selected metric (e.g., distance or speed), the dynamic routing protocols choose the optimal path. Also, multi-path is possible in this network. Dynamic routing protocols can identify and use alternative paths in case some current ones become unavailable. This ensures adaptability by solving topology changes events.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Redundancy is one of the essential pillars of this communications network; it must exist both at the hardware level and software level.

The entire proposed network architecture will be designed and configured in the GNS3 simulator. All protocols, technical solutions and security measures implemented in the network infrastructure will be validated and tested.

IV. SIMULATION SCENARIOS

Simulations of large communications networks are very useful prior to actual design, implementation and deployment. These can validate the architectural solution and initial design from several points of view: functional operation, reliability, security and performance. Different scenarios allow the network evaluation under different contexts, Simulations can assess not only basic functionality aspects but also robustness (traffic variation, increased number of users etc.) but also network resiliency (behavior in extreme or abnormal situations), in order to identify possible points of vulnerability or bottlenecks that could affect the performance and stability of the network.

We consider several basic scenarios to be tested.

Scenario 1: Expanding the network by increasing the number of devices connected to the network.

Objectives: Assessing the horizontal scalability of the network and its ability to manage the increase in the number of connected devices.

Scenario 2: Modifying the network configuration to test the flexibility and stability when changes appear.

Objectives: Verifying how the network adapts to topology and configuration changes.

Scenario 3: Simultaneous generation of multiple types of network traffic to verify network security.

Objectives: Verifying the functionalities of firewalls implemented in the network regarding the filtered, controlled and authorized data flows to test the security level of the entire infrastructure.

During execution of these scenarios, the status of different subsystems will be visualized by monitoring the network.

The phases of this work have been: identification of requirements from both users and technical side; architectural specification; system/network infrastructure and topology design; selection of appropriate solutions for network equipment and their installation and configuration; based on the network topologies established for each part of the network, they are interconnected and configured; routes are created for testing network functionality at the local level; after the local network connections at the county level are made, the regional connections will be made, as well as the implementation of routes to the equipment in the regional area; this region aims to interconnect several counties; as soon as the regional networks have been formed, the connections between the local networks and the data center network are made.

These experiments and tests aim to validate the solution, in terms of functionality, scalability, flexibility and stability, before moving on to the actual implementation of the network.

V. EXPERIMENTS AND RESULTS

According to the technical analysis, scenarios and intended experiments, the communication network was implemented in the simulator (see Figure 3). This communication infrastructure is WAN, composed of numerous local LANs and a data center network. The data center network is implemented in a classic Three-Tier topology.

A. Data center network. The Three-Tier topology is a well-established architecture [16] used in network design that provides a hierarchical structure, organized into three logical levels, (Figure 2) which aims to improve the performance of the implemented network.

The topology is composed of three main layers:

1. Access layer. This is the basic layer that provides the initial connections of end users. Authentication, access control and traffic segmentation are managed in this area.

2. Distribution Layer. It is the second layer and has the role of interconnecting the Access layer with the Core layer. To ensure the performance, as well as the availability of the services offered, a method of aggregating physical links into a single logical link, called EtherChannel, was used. As a result, between the Distribution and Core layers, several physical links were grouped to create a single logical Ethernet link to provide fault tolerance over time and high-speed links between the equipment at these levels.

3. Core Layer. This contains the brain of the network and represents the central part of the network, responsible for the efficient transport of data between the equipment configured in this communications infrastructure.

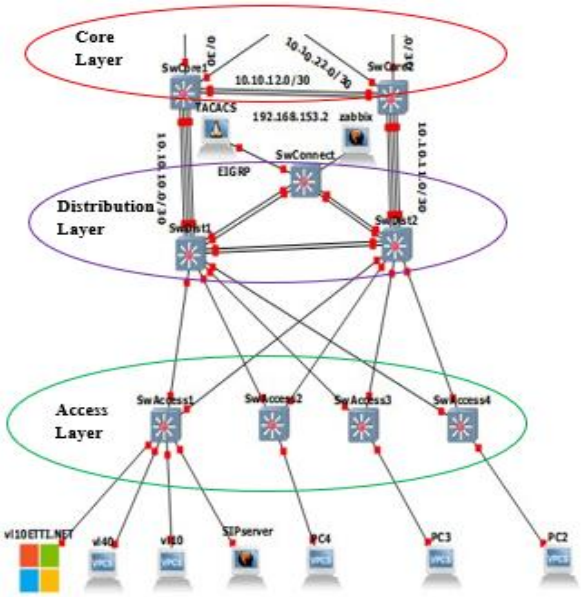


Figure 2. Three-Tier topology of the data center network

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

B. Implementing local networks and defining traffic types. The counties chosen as examples for these traffic types are (Romanian counties): Gorj, Sibiu, Brasov and Constanta.

In order to distinguish between different traffic flows within the network and to facilitate their efficient management, several virtual local area networks (VLANs) were created as follows:

• Vlan 10 – intended for intranet traffic
• Vlan 20 – intended for internet traffic
• Vlan 30 – intended for voice traffic
• Vlan 40 – intended for network management

VLAN is an important solution, ensuring logical separation between subnets, but also guaranteeing flexibility and security. They were configured on the switches in the local networks, but also on the switches at the Access level of the data center network.

C. Establishing physical connections between devices. Each VLAN has an addressing plan so that traffic between subnets can be differentiated. The role of VLAN addressing classes is to route traffic through dynamic routing protocols configured so that users in different locations can communicate.

In this work the following dynamic routing protocols were used:

Enhanced Interior Gateway Routing Protocol (EIGRP) with message authentication in all local networks throughout the infrastructure.

Open Shortest Path First (OSPF) to route traffic from regional networks and the WAN network, as well as to establish interconnections with Fortinet firewalls Another reason for selecting OSPF is that EIGRP is a Cisco proprietary protocol and cannot be implemented on equipment from other vendors, such as Fortinet devices. Multiprotocol Label Switching (MPLS) and Border Gateway Protocol (BGP) for interconnecting county networks with the data center of the entire infrastructure.

EIGRP is used with VLANs to provide dynamic routing between different VLAN subnets on Layer 3 switches or routers, facilitating inter-VLAN communication, load balancing, and redundancy. It enables efficient routing by advertising VLAN interface (SVI) subnets, usually requiring trunk links between switches to carry VLAN traffic.

EIGRP message authentication enhances network security and stability while protecting routing information exchanged between routers. Authentication verifies the identity of participating routers, so that only authorized devices can communicate. This mechanism prevents the introduction of false data and maintains the integrity of routing information, which contributes to the optimal functioning of the network [17].

Since one of the requirements was the integration of several vendors in this implementation, Cisco and Fortinet equipment were configured in the communications network. Since the EIGRP routing protocol is developed by Cisco, it was necessary to introduce another protocol to integrate all the devices. Consequently, the Open Shortest Path First (OSPF) protocol was chosen to perform routing on other equipment, than Cisco ones.

OSPF is a widely adopted link-state routing protocol based on Dijkstra algorithm, which calculates the most efficient routes within a network [18]. In our network it is mainly used at the data center network level. At the same time, to establish connections at the regional network and WAN network level, information is transported via Multiprotocol Label Switching (MPLS) protocols accompanied by Border Gateway Protocol (BGP).

BGP manages data exchange between autonomous systems (ASs), configured both at the level of each LAN network and at the level of the two regional networks, in order to ensure better routing control, isolate traffic between networks and prevent conflicts. BGP uses flexible routing policies, based on path length, preference metric, and other customized criteria [19]. Data and voice traffic reach the core network from the local networks transported by the regional networks, and the two hubs collect all the traffic, interconnecting with the data center network through the two external firewalls.

D. Redundancy of equipment and links. Both at the local, regional and data center network levels, equipment is configured in redundancy to ensure high resiliency. Also at the firewall level, they are configured in redundancy, so that in the event of errors or technical failures, all traffic can be managed by the other equipment without causing interruptions. Each switch in the infrastructure has the Hot Standby Router Protocol (HSRP) configured, providing redundancy for user gateways. The HSRP [RFC 2281 - version 1] is Cisco proprietary redundancy protocol for establishing a fault-tolerant default gateway. Enhanced Version 2 supports IPv6. HSRP establishes an association between gateways in order to achieve default gateway failover if the primary gateway becomes inaccessible. HSRP allows multiple network devices to operate as a single redundancy group. In our network HSRP provides failover services for infrastructure users. A set of network devices, either routers or switches, are configured to represent a single logical device. Depending on the configuration, the device with the highest priority becomes active, and the one with the lowest priority goes into standby mode [20].

In this topology, HSRP is configured to deliver a virtual gateway for devices in the configured logical networks (VLANs). If the active switch becomes inaccessible, the other switch in the group will take over the active role. Accordingly, the end devices will not experience significant interruptions, because the virtual gateway remains the same (see Figure 4).

Each VLAN interface on both switches has an IP address and mask set, and the same IP address is used for the HSRP group, which will also represent the gateway address for network users.

The HSRP mechanism was implemented in version 2 on the two distribution switches (SwDist1 and SwDist2), but also at the level of the local networks located in the four counties with a dedicated group for each VLAN: Group 1 for VLAN10 (VIP: 192.168.10.1), Group 2 for VLAN20 (VIP: 192.168.20.1), Group 3 for VLAN30 (VIP: 192.168.30.1) and Group 4 for VLAN40 (VIP: 192.168.40.1). The configuration described below applies to all locations

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

implemented in the infrastructure in terms of the configured parameters.

Sw_GJ1 was configured with a priority of 150, compared to the priority of 100 for SwDist1, assuming the role of Active router for both groups. The preemption mechanism was activated with a 60-second delay, ensuring automatic return to the main device after fixing a fault, without the need for manual intervention. From a performance perspective, the HSRP timers have been optimized over the default values, using a Hello time of 100 ms and a Hold time of 300 ms. Consequently, if the active switch fails, the standby device detects the failure and assumes the active role in approximately 0.3 seconds, compared to the 10 seconds specific to the default configuration, representing an improvement of approximately 33 times.

The two state changes recorded in the output of the show standby command confirm that the failover mechanism worked correctly during testing.

Since GRE tunnels are configured at the infrastructure level between the county-level locations and the data center, the scenario regarding the combined impact of HSRP and GRE tunnels is considered. GRE encapsulation introduces an additional overhead of 24 bytes per packet, as well as an estimated reconvergence time between 200 and 500 ms when changing the Active device. Therefore, the estimated total downtime in the event of a failover is approximately 650–950 ms, remaining below the critical threshold of 1 second. Although the packets in transit through the tunnel at the time of the failure are lost, the TCP sessions are automatically recovered through the retransmission mechanism, and the application-level sessions are not terminated.

```
SW_GJ1#sh standby
Vlan10 - Group 1 (version 2)
  State is Standby
    4 state changes, last state change 00:26:43
  Virtual IP address is 192.168.53.1
  Active virtual MAC address is 0000.0c9f.f001 (MAC Not In Use)
    Local virtual MAC address is 0000.0c9f.f001 (v2 default)
  Hello time 100 msec, hold time 300 msec
    Next hello sent in 0.080 secs
  Preemption enabled, delay min 60 secs
  Active router is 192.168.53.125, priority 150 (expires in 0.352 sec)
    MAC address is 0c0a.d10a.800a
  Standby router is local
  Priority 100 (default 100)
  Group name is "hsrp-Vl10-1" (default)
Vlan20 - Group 2 (version 2)
  State is Active
    2 state changes, last state change 00:27:42
  Virtual IP address is 192.168.53.129
  Active virtual MAC address is 0000.0c9f.f002 (MAC In Use)
    Local virtual MAC address is 0000.0c9f.f002 (v2 default)
  Hello time 100 msec, hold time 300 msec
    Next hello sent in 0.032 secs
  Preemption enabled, delay min 60 secs
  Active router is local
  Standby router is 192.168.53.157, priority 100 (expires in 0.240 sec)
  Priority 100 (default 100)
  Group name is "hsrp-Vl20-2" (default)
```

Figure 3. HSRP configured on a switch in a local network for vlan 10 and vlan 20

At the same time, by implementing VoIP communications to support communication between company users, a scenario regarding the latency of these communications through GRE tunnels is also defined within this configuration.

```
Vlan30 - Group 3 (version 2)
  State is Active
    2 state changes, last state change 00:27:42
  Virtual IP address is 192.168.53.161
  Active virtual MAC address is 0000.0c9f.f003 (MAC In Use)
    Local virtual MAC address is 0000.0c9f.f003 (v2 default)
  Hello time 100 msec, hold time 300 msec
    Next hello sent in 0.048 secs
  Preemption enabled, delay min 60 secs
  Active router is local
  Standby router is 192.168.53.189, priority 100 (expires in 0.320 sec)
  Priority 100 (default 100)
  Group name is "hsrp-Vl30-3" (default)
Vlan40 - Group 4 (version 2)
  State is Active
    2 state changes, last state change 00:27:42
  Virtual IP address is 192.168.53.193
  Active virtual MAC address is 0000.0c9f.f004 (MAC In Use)
    Local virtual MAC address is 0000.0c9f.f004 (v2 default)
  Hello time 100 msec, hold time 300 msec
    Next hello sent in 0.032 secs
  Preemption enabled, delay min 60 secs
  Active router is local
  Standby router is 192.168.53.221, priority 100 (expires in 0.336 sec)
  Priority 100 (default 100)
  Group name is "hsrp-Vl40-4" (default)
```

Figure 4. HSRP configured on a switch in a local network for vlan 30 and vlan 40

VoIP functionality was validated by successfully registering SIP extensions on the Asterisk server (see Figure 15), extension 1101 confirming the registration with a TTL of 3600 seconds. A key aspect of the validation process is that extensions 1100 and 1101 reside in different LANs within the infrastructure, communication between them being possible exclusively through the GRE tunnel configured between the two network segments. The call initiated between the two extensions, visible in the Twinkle client interface, confirms that the SIP signaling correctly traverses the GRE tunnel, proving end-to-end connectivity between distinct network segments. From a latency perspective, the GRE tunnel introduces an additional overhead estimated between 4 and 8 ms, generated by the packet encapsulation and decapsulation process. Adding the local network latency of 1–2 ms, the estimated total latency for VoIP traffic through the tunnel is in the range of 5–10 ms, well below the critical threshold of 150 ms imposed by the ITU-T G.114 standard for ensuring optimal call quality. In an HSRP failover scenario, latency may temporarily increase during the GRE tunnel reconvergence (estimated at 200–500 ms), but this interruption remains below 1 second and does not terminate the active SIP session. This demonstrates that the GRE tunnel is not just a theoretical component of the architecture, but an active and functional element, indispensable for routing voice traffic between separate LANs, with a negligible impact on the quality of communications under normal operating conditions.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

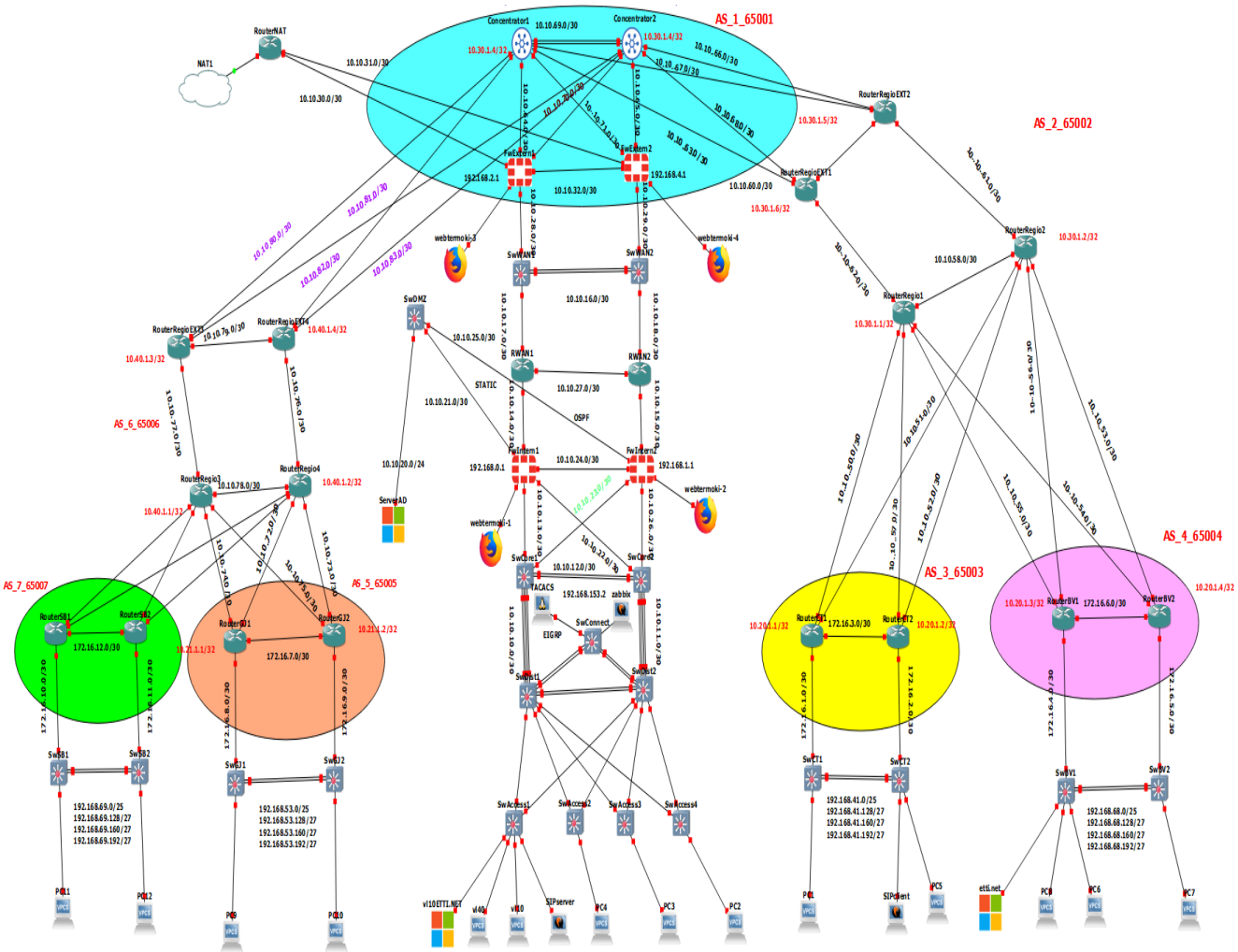


Figure 5. The communication network implemented in GNS3

To meet all technical requirements and user availability, redundant routes have been implemented for all configured areas (local networks, regional networks, data center network and WAN areas). Redundant connections are also present at the security equipment level, external firewalls 1 and 2 and internal firewalls 1 and 2, thus avoiding non-functioning situations. Below is the scenario of a redundant area with redundant routes and equipment, and the purpose of this scenario is to verify the functionality of the redundant route, as well as the main route. The same principle applies to all configured areas in the network.

Consequently, there are redundant paths between the Access and Distribution layers at the data center network level (see Figure 5). In this situation, the following case can be encountered. For this scenario, the interfaces in the main route are disabled (the route highlighted in green) between SwAccess4 and SwDist1, the packets will go on another path, the redundant path in this case (the route highlighted in red), that is, from SwAccess4 to SwDist2 and then SwDist1 (see Figure 5).

The Wireshark utility integrated into GNS3 allows packet flows along this route to be analyzed and visualized.

packets go on the main route, represented by SwAccess4 and SwAccess1.

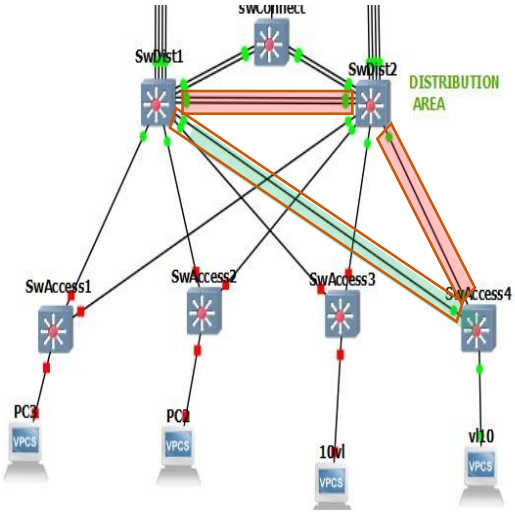


Figure 6. Check the main and secondary route of the network

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

This test will be performed with the ping utility from the VL10 PC to the VLAN interface with the IP address 192.168.10.1. The first step consisted of checking the main route from the PC located in Vlan10, whose IP addressing class is 192.168.10.0/24. From Figure 6, you can see how the packets go on the main route, represented by SwAccess4 and SwAccess1.

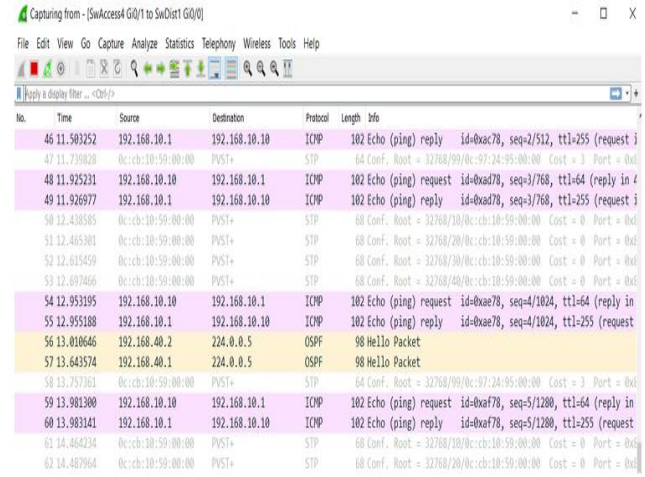


Figure 7. Testing main route connectivity between Access and Distribution layers – Wireshark capture – SwAccess4 – SwDist1

After disabling both interfaces on the main route, traffic is routed on the redundant path (see Figure 6 and Figure 7), meaning packets leave Pc-vl10, arrive at SwAccess4, then it can no longer redirect them on the main path and generates traffic to SwDist2, and SwDist2 routes them to SwDist1.

This scenario demonstrates the redundancy of connections between equipment and also highlights one of the characteristics regarding service availability, which is also strongly influenced by these physical connections between devices.

Therefore, packets originating from the same source reach their destination via the redundant path, even if the primary route is down.

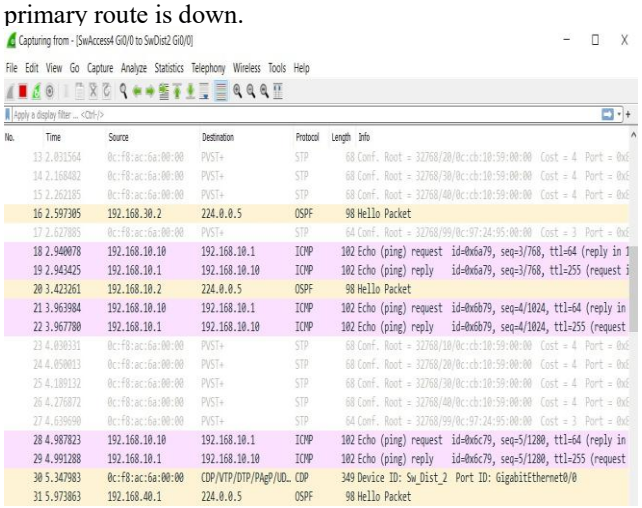


Figure 8. Testing redundant route connectivity between Access and Distribution layers – Wireshark capture – SwAccess4 – SwDist2

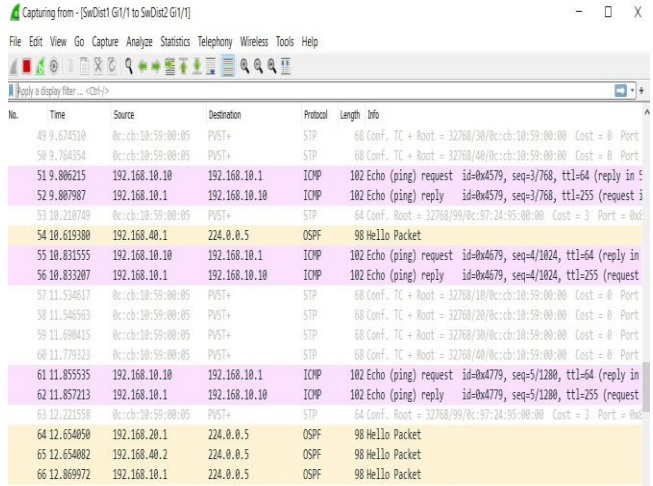


Figure 9. Testing redundant route connectivity between Access and Distribution layers – Wireshark capture – SwDist2 – SwDist1

E. Network Security. Firewalls play a fundamental role in the security of the communications network. Firewalls inspect network traffic and determine, according to predefined security policies, whether packets should be permitted or denied. The firewall then prevents unauthorized traffic from entering or leaving the network. In our design, the firewalls are implemented in a redundant configuration, furthermore ensuring continuous protection and constant availability (see Figure 9). A secondary firewall can automatically take over functionality in the event of a failure or maintenance operation of the primary one, thus guaranteeing the continuity of operations without interruption. Since both hardware and software network protection are equally important in a communications network, two firewall filters were implemented, in redundancy; one firewall filters for control, access and protection of the external part of the network, and the other firewall filters for access to the internal network, but also for routing traffic between devices.

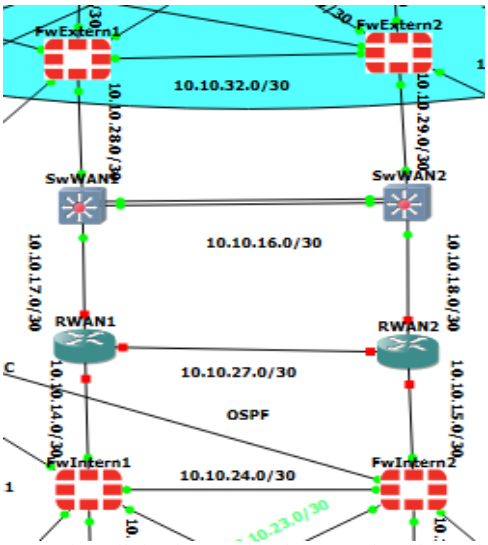


Figure 10. Firewalls implemented in the external and internal area

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Lists are created for filtering and monitoring access, both for incoming and outgoing traffic leaving the internal network through the firewall. Static routes and dynamic routes (via protocols) are configured to facilitate connectivity between devices in the network, the general purpose being the implementation of a secure, reliable, scalable and complex communications network.

Another important feature of firewalls in this topology is routing data. Through the dynamic OSPF, but also several static routes, it ensures the interconnections between the external and internal networks.

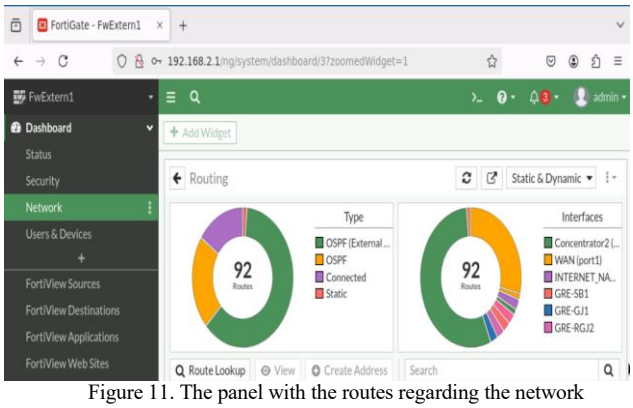


Figure 11. The panel with the routes regarding the network interconnection through the firewall

Figure 10 shows the web interface for managing one of the firewalls in the infrastructure. The information in the dashboard is presented in the form of statistics regarding the routing table and network connections. Since firewalls are used to interconnect local networks and the data center network through regional networks, these options allow you to see all routes, both static and dynamically learned through various routing protocols.

A special security solution implemented in this infrastructure is port-security which is an important functionality, offered by most switches. This feature allows network access control at the physical port level on the switch, and provides an added layer of network security [21]. Within this communication network, it was configured with the violation shutdown option. By enabling the port with this functionality, the switch is monitoring each configured port and allows only devices whose MAC address is known and accepted, the maximum MAC address allowed on a port being only one MAC address. Another enabled feature is sticky MAC learning, which automatically records and associates MAC addresses with specific switch ports, creating an increased level of network security without complex administration. If an unauthorized device tries to connect and send traffic, the respective port is automatically disabled (shut down). This immediate reaction eliminates the possibility of an attacker gaining access to the network, even temporarily, and also allows the administrator to investigate the incident before reactivating the port.

All switches in the access area of all subnets in the entire communications network have within their port-security configuration, with the two options implemented, deactivating the port in case of unauthorized access and dynamically storing MAC addresses on the port, which greatly contributes to the security of the communications network.

Port security based on fixed MAC addresses is an effective and easy-to-implement solution in small networks, providing granular control over which devices are authorized to access the infrastructure. In this paper, a method was used to restrict access at the switch port level, preventing unauthorized devices from connecting by blocking unknown MAC addresses. However, the scalability of this approach becomes a significant challenge in large networks, where thousands of devices may be replaced, moved, or added frequently, requiring manual reconfiguration of each port individually — a time-consuming process that is prone to human error.

A scalable and widely adopted alternative in enterprise environments is authentication based on the IEEE 802.1X standard, combined with a RADIUS server. Through this mechanism, network access is conditioned on the authentication of two distinct entities: both the user, through individual credentials or digital certificates, and the workstation, through centrally issued and managed machine certificates. This dual approach eliminates scenarios where an authorized user logs in from an unauthorized device or vice versa, ensuring that only the valid user-station combination receives access to network resources. The RADIUS server centralizes access policies, allowing unified management of thousands of endpoints within the network, regardless of their physical location or the port to which they are connected [22], [23].

In the context of the further evolution of the presented infrastructure, integration with Cisco Identity Services Engine (ISE) was identified as a future development direction and mentioned in the Future Works section of the paper.

F. Windows Server – Active Directory. Within this communications network, Windows Server 2012 was configured to implement the Active Directory service [24], in order to add existing workstations to the network, create users in the domain, and provide centralized management of the entire IT infrastructure. The domain was created to enable centralized (is that what you mean) administration, enhance security, and control user access to network resources.

As a result of the domain created for this infrastructure, remote connections between users are possible through the Remote Desktop protocol [25]. This is a Microsoft-developed secure network communication protocol that allows users to remotely control another computer's desktop interface. In our case, remote access to computers in the domain facilitates much easier administration, but also contributes significantly to solving technical problems that arise at the user level. An additional role that Active Directory has in the network is to define and apply security policies in the network, in a centralized way, so that they apply uniformly to users and devices in the domain.

Through Active Directory, users can be created and added to different groups according to needs and

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

requirements. In addition, based on the domain created in this infrastructure, called etti.net, computers can be added to the domain, thus leading to centralized management, where security policies can be applied uniformly across the entire domain. Figure 11 shows an example of a user created in Active Directory, a user who will be able to connect to any workstation that is part of this domain.

Also, both users and workstations can be organized into organizational units, so that their administration becomes much more useful and easier.

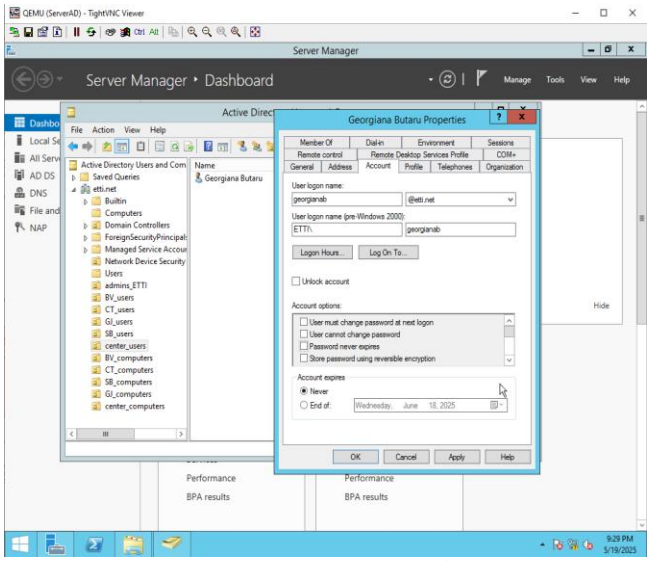


Figure 12. Active Directory configuration

G. Network Monitoring Continuous monitoring is essential for the early detection and prevention of operational issues and service disruptions. Zabbix [26] is the solution for monitoring this network. This solution collects all logs of the configured devices, in real time, helping to quickly identify any problem before it becomes critical. (see Figure 12). At the same time, it is possible to view whether the configured equipment is operating within normal parameters, preventing situations that could cause the interruption of the services.

Figure 13. Network monitoring example – with Zabbix

Simple Network Management Protocol (SNMP version 2) is used for monitoring and managing devices. It collects information about the status and performance of devices, and transmits it to the Zabbix platform. All this information is centralized in a database, and based on this, detailed graphs, alerts and reports are created to help to identify and resolve problems before they affect end users or cause significant interruptions.

In Zabbix, an important benefit is the automatic generation of graphs for each network interface, based on the traffic sent and received. These graphs show the evolution of traffic over time, so the administrator can easily see how much bandwidth is being used, if there are periods of overload or if anomalies occur in the network.

At the same time, the traffic that circulates through an interface is exposed to us in a graph in the Zabbix platform that is based on a legend that explains what is happening at the level of an interface, such as how much data was sent or received, the traffic that was removed from the interface or even the traffic that entered or exited the interface with errors.

Therefore, all the necessary and essential details are exposed centrally in this platform, contributing significantly to the management of the communications network.

H. Implementing TACACS+ in the Network. Terminal Access Controller Access Control System Plus (TACACS+) [27] is a protocol used to authenticate, authorize, and account for user access to network devices, providing enhanced security and more granular access control. TACACS+ is installed on a Linux machine within the data center network, connected to the infrastructure via the SwConnect device. User groups are configured and rights are assigned based on the activity performed and requirements, so that access to network equipment is centralized, via TACACS+ (see Figure 14).

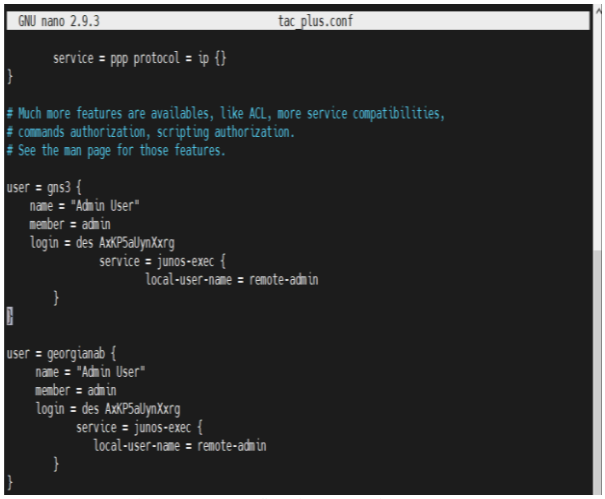


Figure 14. Configuring users in TACACS+

With TACACS+, user authentication is centralized, which means that each network device does not need to have its own authentication system. This allows for more efficient and secure user management in a complex network. Another

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

advantage of TACACS+ is that all communications between the client and the server are encrypted, thus protecting sensitive data. The choice of the TACACS+ protocol for authenticating administrators to network equipment is based on essential architectural differences compared to RADIUS, which make it superior in the specific context of network infrastructure management [28].

The first fundamental difference lies in the separation of AAA functions. TACACS+ treats authentication, authorization, and accounting as completely independent processes, allowing for granular policies for each function. RADIUS, by design, couples authentication and authorization within a single exchange, limiting the flexibility of access policies. This separation becomes critical in the context of network equipment management, where the same administrator may require different rights on different devices — for example, full access on distribution switches, but only monitoring commands on edge devices [29].

The second critical difference is the granularity of command-level authorization. TACACS+ allows explicit definition of permitted or prohibited commands per user or group, down to the command argument level. An operator might be authorized to execute show running-config but not configure terminal, or might be limited to viewing a single VLAN. RADIUS does not natively provide this level of control, its authorization operating at the session level, not the individual command level [30].

The third architectural difference concerns session encryption. RADIUS encrypts only the password field in the authentication packet, leaving the rest of the payload, including authorization attributes, transmitted in the clear. TACACS+ encrypts the entire payload of every packet exchanged during the session, providing substantially superior protection against man-in-the-middle attacks and interception of administrative traffic—critical in a public infrastructure where management sessions contain sensitive information about the network configuration [31].

Within this infrastructure, the protocol was configured on a Linux operating system, where users were created and the groups to which the users would belong were defined, either with full rights or only read rights.

I. Session Initiation Protocol. Session Initiation Protocol (SIP) is a communication protocol used to initiate, maintain, and terminate communication sessions in IP networks. It is used primarily for Voice over IP (VoIP) telephony, videoconferencing, and instant messaging applications, allowing users to establish and control real-time communication sessions [32].

In this communications infrastructure, SIP is implemented through Asterisk [33] configured on a Linux operating system – Ubuntu, which acts as a telephony server, and Twinkle, an application used as a softphone by users to initiate and receive calls. This architecture is commonly employed in VoIP environments, where packetized voice traffic is transmitted over IP networks, rather than through traditional circuit switching channels.

The Asterisk server acts as a software PBX (Private Branch Exchange), handling all calls between users. It receives SIP requests from Twinkle and establishes voice sessions between two SIP endpoints. In addition, Asterisk can apply call routing rules, record calls, enable voicemail, or integrate more advanced functionality such as conferencing or IVR (Interactive Voice Response).

In terms of user needs, Twinkle is a SIP application that connects to Asterisk over the network. The user enters their SIP credentials (username, password, Asterisk server IP or domain), and after connecting, he/she can initiate or receive voice calls. Twinkle uses SIP for signaling and RTP (Real-Time Transport Protocol) for the actual transmission of audio data.

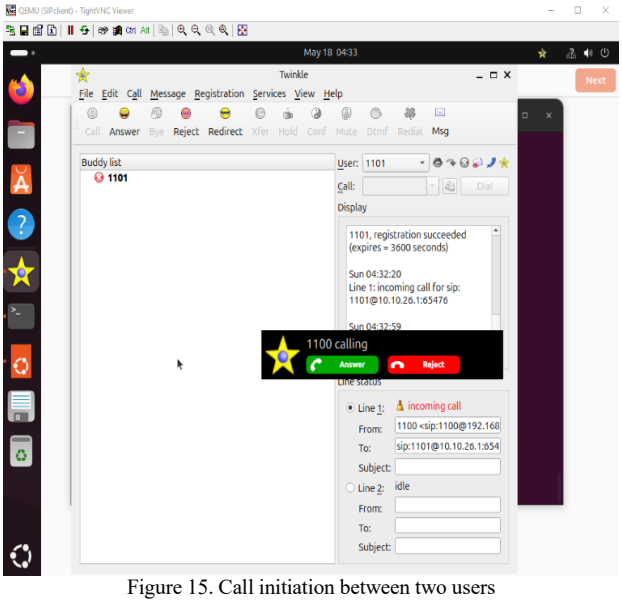


Figure 15. Call initiation between two users

During a call, Twinkle sends a SIP request to Asterisk. The server processes this request, checks if the recipient is available, and then initiates a session between the two Twinkle applications or between Twinkle and another compatible device (Figure 14). Once the call is accepted, audio data begins to flow between the two points, in real time, via RTP.

This architecture is efficient in the communications network, as an alternative to classic telephone systems. The advantages are multiple: low costs, flexibility in call management, the possibility of integration with other IT systems and easy scalability. In addition, the use of an open protocol such as SIP allows interoperability between different applications and devices.

J. Configuring Tunnels Between Local Area Networks and the Core Network. GRE (Generic Routing Encapsulation) tunneling is a technique used to encapsulate and transport data packets between two points on a network, using a tunneling protocol [34]. Basically, GRE allows the creation of a virtual "tunnel" between two network devices, in this case between routers in the LAN locations and external firewalls 1 and 2.

The main purpose of GRE is to allow the transport of packets using different protocols over an IP network, in a way that is transparent to the transport network. In other words, GRE allows non-native traffic (e.g., IPv6, IPX,

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

multicast) to be sent over a network that only supports IP (usually IPv4). GRE creates a logical point-to-point tunnel between two network devices (usually routers), called the tunnel source and the tunnel destination. The traffic from the source network is encapsulated in a GRE packet, which is then encapsulated in a standard IP packet. This IP packet is sent through the intermediate network (e.g., the Internet) to its destination, where it is decapsulated and forwarded to the target network.

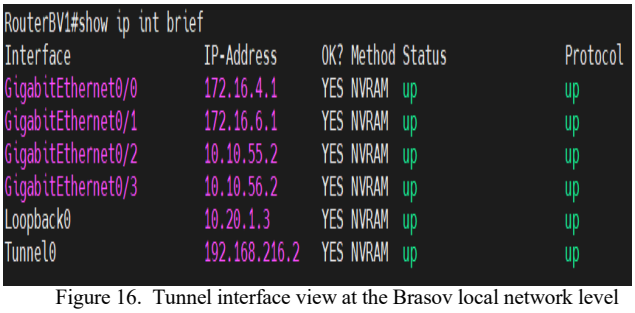
The structure of the packet looks like this:

• The original packet (which can be any protocol – IPv6, IPX, AppleTalk, etc.)

• The GRE header (which indicates that it is a tunnel and the type of packet being transported)

• The IP header (used to transport the GRE packet over the Internet or another IP network)

This double encapsulation mechanism allows data to be transported over networks that do not necessarily understand the original format of the packets [35] (see Figures 15 and 16).



```
RouterBV1#show ip int brief
Interface              IP-Address     OK? Method Status          Protocol
GigabitEthernet0/0     172.16.4.1     YES NVRAM  up              up
GigabitEthernet0/1     172.16.6.1     YES NVRAM  up              up
GigabitEthernet0/2     10.10.55.2     YES NVRAM  up              up
GigabitEthernet0/3     10.10.56.2     YES NVRAM  up              up
Loopback0              10.20.1.3      YES NVRAM  up              up
Tunnel0                192.168.216.2  YES NVRAM  up              up
```

Figure 16. Tunnel interface view at the Brasov local network level

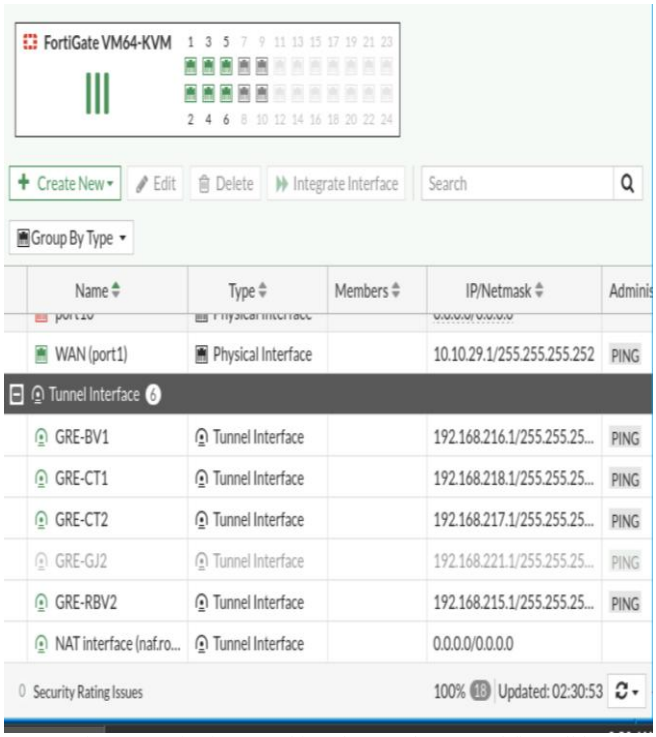


Figure 17. Viewing tunnel interfaces at the external firewall level

The communications network implemented in this paper plays an important role in modern society, where technology enables access to integrated digital services. Providing public services in a unified environment makes information easily accessible and improves efficiency. Implementing the network in the GNS3 simulator is valuable practice, as it supports real network device images and operating systems, offering a realistic environment for configuring, testing, and validating the network before deployment.

The implemented technical solutions demonstrate a solid balance between security, performance and efficient management. This communications network meets the conditions of a complex network, with high data traffic, and with high performance due to the configured dynamic routing protocols and the implemented security measures, easily ensuring the availability of services.

VI. INCREASING THE NETWORK RESILIENCE BASED ON DISASTER RECOVERY PLAN

A disaster recovery plan is vital for any organization. Numerous factors may lead to significant network disruption or infrastructure failure, floods, landslides, etc.).

Disaster recovery capability is essential for business continuity (BC). The concept consists of planning and preparing necessary resources to ensure the operationalization and availability of the services offered during and after a disaster recovery. The primary objective of business continuity (BC) is to maintain operational resilience and minimize downtime [36].

Business continuity and disaster recovery are dimensions often neglected in simulation-based technical projects, but essential in an infrastructure designed for public services for citizens. From a scientific perspective, any Business Continuity strategy must be anchored in two fundamental indicators: RTO, which defines the maximum acceptable interval for restoring a service after an interruption, and RPO, which establishes the maximum amount of data that can be lost without significant operational impact. These indicators, established in the specialized literature and standardized by ISO 22301 and NIST SP 800-34, constitute the basis of any documented recovery plan [37].

The infrastructure implemented within the present project ensures the first level of a continuity architecture through redundancy mechanisms at the network level: the HSRP protocol guarantees an RTO of approximately 300 ms at the distribution gateway level, and the GRE tunnel ensures the continuity of connectivity between network segments. These mechanisms fall into the Warm/Hot Standby category at the network infrastructure level, according to the taxonomy of recovery layers defined in the specialized literature [38].

However, a complete DR strategy for public data requires additional layers of protection, including real-time replication of databases, the existence of a secondary geographical site, and documented and periodically tested failover procedures at the application level. These aspects take on particular importance in the context of the European regulations in force: the NIS2 Directive (2022/2555) requires

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

operators of essential services to implement proportionate technical continuity measures and periodically test recovery plans, while the GDPR requires ensuring the integrity and continuous availability of citizens' personal data. Failure to comply with these requirements can attract significant administrative sanctions, which gives the Business Continuity dimension not only a technical relevance, but also a legal and institutional one, justifying its inclusion as a priority direction in the further development plan of the infrastructure [39], [40].

Recovery methods in the context of natural disasters are centered on critical infrastructures, with a predominant emphasis on the physical restoration of the services offered [41].

The strategy of this infrastructure is for the data center network to benefit from a disaster recovery plan. This plan must consist of a procedure for recovering data and returning to normal activities. The actions to be taken must be established in stages, so that the return to normal can take place in optimal conditions.

Since it is an important environment that manages and stores citizens' personal data, the degree of responsibility and liability in the event of unforeseen situations is equally important. Thus, in order to prevent such situations caused by normal and abnormal events of nature, it is essential to expand the data center network by creating another communications network in another physical location, at a considerable distance from the physical location of the existing data center, which serves as a data center in the event of such disasters. Therefore, to achieve this important objective, a project will be built that will contain all the necessary steps both for the implementation and construction of a new data center, but also for establishing the necessary connections, so that the transition can be achieved without significantly impacting the entire infrastructure.

The first step in implementing this plan will consist of choosing a new physical location, where the new data center infrastructure will be designed. Several geographical parameters will be taken into account, such as avoiding high mountain areas that would cause problems regarding physical accessibility and signal in the area, but also hilly areas whose clay soil could frequently produce landslides and would constitute an imminent danger to the infrastructure.

After the location has been chosen and all the criteria that must be met have been checked, the plan regarding the implementation of the data center network will be created. In this context, a replica of the existing data center network will be created. On a similar network topology, the equipment will be configured in redundancy to ensure availability in the event of a failure of one of them, and the links will also be configured in redundancy to avoid situations when one of the connections is down. In addition, for these connections, the EtherChannel protocol will also be used in certain segments of this network due to the benefits it brings. The benefits consist of increasing bandwidth by aggregating multiple physical ports into a single logical port, resulting in redundancy and fault tolerance.

Also, the network will be divided into zones, and traffic will be filtered through configured security equipment, on which multiple policies regarding access to equipment and resources in the infrastructure will be implemented.

Within the implemented infrastructure, in addition to network equipment, servers will be configured that will contain the information located in the main and active network. Based on back-up and replicate mechanisms, planned at a certain time interval, the data will be transferred to the secondary data center, so that in case of disaster recovery, the activity can be carried out in optimal conditions, without significant and long-term interruptions.

Connections between equipment will be made based on dynamic routing protocols, so that over time there can be multiple changes by adding or removing devices, which simplifies the entire infrastructure administration process.

Once the disaster recovery infrastructure has been designed and configured, a plan will be developed for the transition from the primary to the secondary network. The plan will contain the necessary steps, including training the people responsible for this infrastructure. All the benefits, but also the consequences that are brought by this network expansion, will be made known.

Another expansion through this disaster recovery, consists of implementing another area in the infrastructure in which there will be equipment that will contain only the backup of all configured equipment, the backup program to be done outside of working hours, so as not to overload the network. This implementation will ensure the network from a security perspective that if a vulnerable device or equipment fails and loses its configuration, even if they are configured in redundancy, to avoid such situations it is very important to have backup for any configured equipment, so that the network will be oriented towards performance, towards functioning in optimal parameters, towards availability in any situations and most importantly towards security at the highest level of a communications infrastructure.

VII. POSSIBLE ARCHITECTURAL DEVELOPMENTS OF THE IMPLEMENTED COMMUNICATIONS INFRASTRUCTURE

One of the most significant developments that can be brought to this implemented network is to update the data center network topology to novel approaches.

The Three-Tier (Core–Distribution–Access) architecture, although mature and well-documented, has structural limitations that become increasingly pronounced as the performance and agility requirements of the infrastructure increase. The first fundamental limitation is the reliance on Spanning Tree Protocol (STP) to prevent loops at the switching level. STP blocks active redundancy, allowing the use of only one active path between any two nodes, while the others remain in standby. This translates into a systematic underutilization of the physical infrastructure — half of the redundant links are inactive at any time — and a slow convergence in the event of a link failure, with reconvergence times that can exceed 30 seconds in classic

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

STP implementations, or 1–2 seconds in RSTP. The second structural limitation is the limited horizontal scalability of the distribution layer. In a Three-Tier architecture, all uplinks from the Access layer converge to a fixed number of distribution switches, which become potential bottlenecks as port density and traffic volume increase. Adding capacity at the distribution layer involves significant reconfiguration and potential maintenance windows, incompatible with the continuous availability requirements of public services. The third limitation concerns distributed and heterogeneous management. Each device in the Three-Tier stack requires individual configuration, and consistency of security, QoS, and VLAN policies across hundreds or thousands of devices becomes a major operational challenge, exposed to the risk of configuration errors and policy drifts that are difficult to detect and correct [42].

The Spine-Leaf architecture, introduced in response to the limitations of traditional hierarchical topologies in the context of modern data centers, is based on a non-blocking Clos topology, theorized by Charles Clos in 1953 and brought back into use by the scalability requirements of cloud infrastructures. The fundamental principle consists of connecting each Leaf node to all Spine nodes, guaranteeing that any communication between two Leaf nodes traverses exactly 2 hops, regardless of the size of the network or the position of the nodes in the topology. This property, called equal-cost multipathing (ECMP), eliminates the concept of active/standby path specific to STP, allowing the simultaneous use of all available physical links. Traffic is distributed evenly across all active paths through hash algorithms based on flow parameters (source IP, destination IP, source port, destination port), maximizing infrastructure utilization and eliminating bottlenecks at the distribution level. Aggregate network capacity increases linearly with the addition of Spine or Leaf nodes, without architectural redesign and without downtime.

At the routing protocol level, the Spine-Leaf architecture replaces STP with or OSPF at the underlay level, protocols designed for rapid convergence and large-scale scalability. The overlay is provided by VXLAN (Virtual Extensible LAN), standardized by RFC 7348, which extends the address space from 4,096 specific 802.1Q VLANs to over 16 million logical segments, eliminating one of the fundamental limitations of traditional VLAN segmentation [43].

The key distinction between a generic Spine-Leaf architecture and Cisco ACI (Application Centric Infrastructure) lies in the management paradigm and policy model. A standard Spine-Leaf implementation with VXLAN and BGP represents a significant improvement over Three-Tier from a performance and scalability perspective, but maintains the device-centric configuration paradigm — policies are defined per device, per port, per VLAN. ACI introduces a fundamental paradigm shift, moving to an intent-based model, in which the administrator defines what behavior he wants, not how each device should be configured [44].

Currently, the topology used is of the Three-Tier type, which has 3 layers: Access, Distribution and Core (see Figure 3). A more recent solution, proposed for the architectural development of this infrastructure is Cisco Application Centric Infrastructure ACI [45], which uses a two-tier spine-leaf topology. ACI consists of leaf switches (connecting to servers) and spine switches (forming the backbone), connected in a full-mesh to ensure high redundancy, scalability, and efficient east-west traffic. ACI also offers high east-west bandwidth; automation can be embedded, based on policies compared to the traditional architecture (see Figure 17).

The main elements and features are:

Leaf Layer: access switches that connect directly to servers, storage, and other end devices.

Spine Layer: backbone switches that interconnect all leaf switches.

Full-Mesh Topology: every leaf switch connects to every spine switch, ensuring no more than one hop between any two leaf switches.

Connection Type: typically uses Layer 3 routing between layers to optimize performance.

Traffic in this topology is optimized, east-west, from endpoint to endpoint, providing multi-path forwarding, with low latency and equal cost. The management is centralized through Application Policy Infrastructure Controller (APIC) by configuring policies, applied directly to the equipment in the topology.

Another important aspect generated by this infrastructure is defined by the elimination of Spanning Tree (STP). In a classic architecture if there are two links to two switches, STP blocks one of them to avoid loops.

One advantage of this architecture is the ability to implement advanced features such as Virtual Port Channel (VPC). This technology allows two different physical switches to appear as a single logical switch to the third device, which can be a server. This new concept ensures resilience at the hardware level, meaning that if the other switch fails, traffic continues to go through the other VPC switch without any interruption for the endpoint. It is an essential element for disaster recovery at the level of this topology [46].

Also, the bandwidth increases, since the server can send data simultaneously on both wires, thus using the full capacity of both switches.

The topology in Figure 17 is proposed for the development of the implemented infrastructure, by improving the data center network, by replacing the Three-Tier topology with this topology.

The ACI topology above is composed of 3 APIC servers, which represent the main component of this topology, managing, automating and monitoring the equipment through a unified platform, the platform where policies are created and pushed to the equipment in the topology.

The devices at the Spine and Leaf levels are Nexus switches that bring virtualization and redundancy technologies in addition to the devices in a classic topology. These devices are the basis of Software-Defined Network data center networks. Another feature that these devices bring is Virtual Extensible Local Area Network (VXLAN) that creates layer 2 networks (overlay) over layer 3 IP

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

networks (underlay), ensuring massive scalability in a data center [47].

Replacing the classic Three-Tier topology of the data center network with the Cisco ACI topology brings numerous benefits both from the point of view of equipment administration, offering centralized management, and from a technical perspective through the automated and unified mechanisms that this topology brings.

In this topology, the four leaves switches are not interconnected, and the two spine switches are also not directly connected to each other. Instead, each leaf switch connects to both spine switches, forming the core fabric of the network. The leaf switches serve as the access layer where end devices connect, which may be either physical servers or virtual machines.

Within the ACI architecture, endpoint groups (EPGs) are used to logically group servers that share similar network and security requirements. For example, servers in the same EPG may belong to the same security zone, have access only to specific networks, and be subject to granular traffic policies. This logical graph represents the relationships between applications and the rules that govern communication between them. The policy graph describes how groups of endpoints, called Endpoint Groups (EPGs), communicate with each other based on contracts.

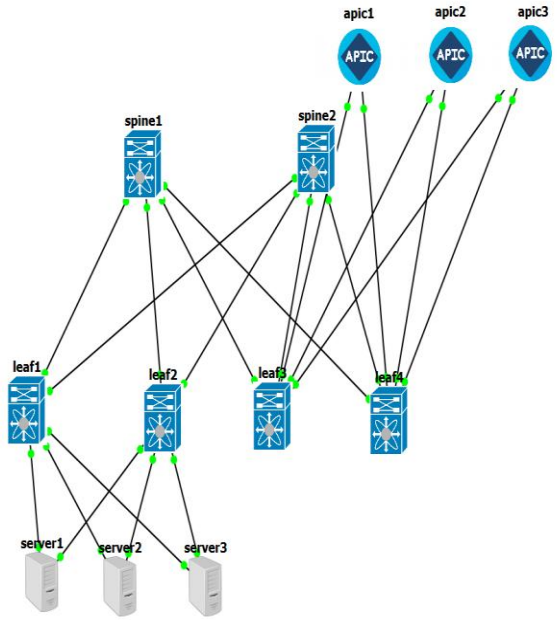


Figure 18. Application Centric Infrastructure Topology – Spine and Leaf for the data center network

In the absence of a contract, communication is blocked by default, which increases security and control. Some of the advantages of these graphs are the clarity and consistency they provide to network management. This feature of the ACI topology represents a significant evolution compared to traditional network configuration models [48].

The migration from Three-Tier to ACI Spine-Leaf architecture cannot be achieved through a sudden transition in an infrastructure that hosts active public services, but requires an incremental phased approach that guarantees service continuity throughout the entire process.

Phase 1 – Coexistence and Interconnection: The ACI fabric is implemented in parallel with the existing Three-Tier infrastructure, interconnected by L3Out nodes that provide routing between the two domains. Services are migrated incrementally, one segment or application at a time, with the possibility of rollback at any stage. This approach, called brownfield deployment in the specialized literature, minimizes operational risk and allows validation of each migration before continuing.

Phase 2 – Migration of critical services: Public services with high availability and performance requirements are migrated first to the ACI fabric, benefiting from ECMP and uniform latency. Existing security policies are translated into ACI Contracts and validated by comparison with previous behavior.

Phase 3 – Three-Tier Decommissioning: After all services are fully migrated and behavior is validated in the new fabric, the Three-Tier infrastructure is progressively decommissioned, freeing up resources for expanding the ACI fabric.

Throughout the migration process, the coexistence of the two architectures involves maintaining a unified addressing plan and consistent QoS policies across domains, aspects that must be planned and documented rigorously before initiating the migration [49].

Criteria	Three-Tier Topology	ACI Infrastructure
Topology	Hierarchical, 3 levels	Non-blocking cluster, 2 levels
Switching protocol	STP/RSTP	ECMP, BGP/OSPF
Link usage	50% (STP blocked)	100% (all active)
Latency variable	2-4 hops	Only 2 hops
VLAN Scalability	4,096 segments	16+ million (VXLAN
Manageme nt model	Device-centric, CLI	Intent-based, APIC
Security	Distributed ACLs	Native microsegmentation
NIS2 Compliance	Manual Audit	Automated Centralized Audit

Thus, this topology offers a new, scalable and automated architecture for data center networks, being a viable solution for dynamic environments with multiple data flows.

VIII. CONCLUSION AND FUTURE WORK

The communications infrastructure presented in this paper provides a practical and effective framework for supporting multiple integrated services. The system includes multiple protocols and services, which results in a network

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

oriented towards performance, security, scalability and resilience.

Several technical requirements and multiple challenges regarding their implementation have been addressed, so that the result was a secure network, a fast, flexible network, which adapts to the latest technologies.

The management and monitoring mechanisms allow to have an overall view of all configured and interconnected devices, a centralized management of the users, as well as the unified application of policies on the domain configured in this network. Voice communication between users located in different physical locations is supported by various protocols and applications that manage and prioritize voice traffic. These communications use modern technologies and can be carried out through existing computers, without requiring additional specialized equipment.

In the current context of cyberattacks, security measures have been provisioned, both through security equipment, but also through network equipment, events that are monitored through the configured platform.

All these technologies combined form a reliable, scalable communications infrastructure, capable of supporting critical operations and easily adapting to user needs. This communications network, configured with the aim of providing public services to citizens, represents a network model that easily adapts to the requirements and needs of both users and its administrators. The network is scalable; it can be expanded by adding more locations, small and medium-sized networks, or even by expanding the data center network, and adapting its network to state-of-the-art topologies, with multiple new mechanisms.

Another development consists of integrating a security platform and access policy management, such as the Cisco Identity Engine solution, which centrally manages an entire infrastructure through an Authentication, Authorization and Accounting (AAA) security mechanisms.

Furthermore, the infrastructure created in this work meets all the essential and necessary characteristics of a scalable, secure multi-services network.

REFERENCES

[1] M. G. Butaru and E. Borcoci, “Implementing a Communications Infrastructure to Optimize Public Services for Communities”, The Twentieth International Conference on Systems and Networks Communications, ICSNC, Lisbon, 2025, [Online]. Available from https://www.thinkmind.org/library/ICSNC/ICSNC_2025/icsnc_2025_1_10_20006.html, 03.01.2026.

[2] Digital technology, [Online]. Available from https://www.durham.gov.uk/article/29603/why-digital-technology-is-important, 05.01.2026.

[3] Graphical Network Simulator-3, [Online]. Available from https://docs.gns3.com/docs/, 04.01.2026.

[4] Cisco Application Policy Infrastructure Controller (Cisco APIC), [Online]. Available from https://www.cisco.com/c/en/us/products/collateral/cloud-systems-management/aci-fabric-controller/at-a-glance-c45-730001.html, 04.01.2026.

[5] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture”. A scalable, commodity data center network architecture. ACM SIGCOMM Computer Communication Review, 2008 38(4), 63–74, [Online]. Available from https://doi.org/10.1145/1402958.1402967, 03.05.2026.

[6] M. Swanson, P. Bowen, A. Phillips, D. Gallup, and D. Lynes, NIST Special Publication 800-34. NIST Special Publication 800-34 Rev. 1: Contingency Planning Guide for Federal Information Systems. National Institute of Standards and Technology, 2010 [Online]. Available from https://doi.org/10.6028/NIST.SP.800-34r1, 03.05.2026.

[7] S. Rose, O. Borchert, S. Mitchell, and S. Connelly, (2020). NIST Special Publication 800-207: Zero Trust Architecture. National Institute of Standards and Technology, [Online]. Available from https://doi.org/10.6028/NIST.SP.800-207, 04.05.2026.

[8] T. Dahm, A. Ota, D.C. Medway Gash, D. Carrel, and L. Grant, RFC 8907 – The Terminal Access Controller Access-Control System Plus (TACACS+) Protocol, 2020, [Online]. Available from https://datatracker.ietf.org/doc/html/rfc8907, 05.05.2026.

[9] K. Nichols, S. Blake, F. Baker, and D. Black, "RFC 2474 – Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," 1998, [Online]. Available from https://datatracker.ietf.org/doc/html/rfc2474, 05.05.2026.

[10] Cisco Systems. (2005). Enterprise QoS Solution Reference Network Design Guide, Version 3.3. Cisco Systems, [Online]. Available from https://community.cisco.com/legacyfs/online/legacy/3/2/6/64623-qossrnd.pdf, 06.05.2026.

[11] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski, "RFC 2597 – Assured Forwarding PHB Group," 1999, [Online]. Available from https://datatracker.ietf.org/doc/html/rfc2597, 07.05.2026.

[12] International Telecommunication Union. (2003). ITU-T Recommendation G.114 (05/2003): One-way transmission time. ITU-T, [Online]. Available from https://www.itu.int/rec/T-REC-G.114-200305-I/en, 07.05.2026.

[13] J. Moy, "RFC 2328 – OSPF Version 2," 1998, [Online]. Available from https://www.rfc-editor.org/rfc/rfc2328.html, 08.05.2026.

[14] Y. Rekhter, T. Li, and S. Hares, Eds., "RFC 4271 – A Border Gateway Protocol 4," RFC 4271 – A Border Gateway Protocol 4 (BGP-4), 2006, [Online]. Available from https://www.rfc-editor.org/rfc/rfc4271.html, 10.05.2026.

[15] Asterisk Project Documentation, [Online]. Available from https://docs.asterisk.org/, 05.01.2026.

[16] What is Three-Tier Architecture in Switch Networking, [Online]. Available from https://www.qsfptek.com/qt-

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

news/what-is-three-tier-architecture-in-switch-networking.html?srsltid=AfmBOoq-qV_3MysTzjZhnDTmFo01goIY5rD7yJMERRXeR21uxCCmkljt, 05.01.2026.

[17] EIGRP Message Authentication Configuration, [Online]. Available from https://www.cisco.com/c/en/us/support/docs/ip/enhanced-interior-gateway-routing-protocol-eigrp/82110-eigrp-authentication.html, 06.01.2026.

[18] A. S. Tanenbaum - Vrije Universiteit Amsterdam, The Netherlands, David J. Wetherall - University of Washington Seattle, WA, “Computer Networks”, p. 474 - 478, 5th edition, 2011.

[19] IP Routing: BGP Configuration Guide, [Online]. Available from https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_bgp/configuration/xe-16/irg-xe-16-book/configuring-a-basic-bgp-network.html, 07.01.2026.

[20] Understand the Hot Standby Router Protocol Features and Functionality, [Online]. Available from https://www.cisco.com/c/en/us/support/docs/ip/hot-standby-router-protocol-hsrp/9234-hsrpguidetoc.html, 08.01.2026.

[21] Port Security, [Online]. Available from https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/15-4SY/config_guide/sup6T/15_3_sy_swcg_6T/port_security.pdf, 08.01.2026.

[22] IEEE. (2020). IEEE Std 802.1X-2020 – IEEE Standard for Local and Metropolitan Area Networks: Port-Based Network Access Control. IEEE Standards Association, [Online]. Available from https://1.ieee802.org/security/802-1x/, 10.05.2026.

[23] C. Rigney, S. Willens, A. Rubens, and W. Simpson, "RFC 2865 – Remote Authentication Dial In User Service (RADIUS)", 2000. [Online]. Available from https://www.rfc-editor.org/info/rfc2865, 11.05.2026.

[24] What is Active Directory? A step-by-step tutorial, [Online]. Available from https://www.comparitech.com/net-admin/active-directory-step-by-step-tutorial/, 09.01.2026.

[25] Understanding the Remote Desktop Protocol (RDP), [Online]. Available from https://learn.microsoft.com/en-us/troubleshoot/windows-server/remote/understanding-remote-desktop-protocol, 09.01.2026.

[26] What is Zabbix, [Online]. Available from https://www.zabbix.com/documentation/current/en/manual/introduction/about, 10.01.2026.

[27] TACACS Protocol Explained, [Online]. Available from https://www.tacacs.com/security-hub/tacacs-protocol-explained/, 10.01.2026.

[28] Dahm, T., Ota, A., Medway Gash, D.C., Carrel, D., & Grant, L. (2020). RFC 8907 – The Terminal Access Controller Access-Control System Plus (TACACS+) Protocol, [Online]. Available from https://www.rfc-editor.org/rfc/rfc8907.html, 12.05.2026.

[29] Cisco Systems. (2015). TACACS+ Configuration Guide, Cisco IOS Release 15S – Configuring TACACS+. Cisco Technical Documentation, [Online]. https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_usr_tacacs/configuration/15-s/sec-usr-tacacs-15-s-book/sec-cfg-tacacs.html, 13.05.2026.

[30] Authentication Authorization and Accounting Configuration Guide, [Online]. Available from https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/sec_usr_aaa/configuration/xe-16/sec-usr-aaa-xe-16-book/sec-cfg-authorizatn.html, 13.05.2026.

[31] W. Stallings, Network Security Essentials: Applications and Standards, 2016, (6th ed.), pp. 161–169. Pearson Education. ISBN: 978-0134527338.

[32] RFC 3261 - SIP: Session Initiation Protocol, [Online]. Available from https://datatracker.ietf.org/doc/html/rfc3261, 13.01.2026.

[33] Asterisk VoIP System for Linux | How to Install and Use, [Online]. Available from https://ioflood.com/blog/install-asterisk-linux/, 13.01.2026.

[34] What is GRE tunneling? | How GRE protocol works, [Online]. Available from https://www.cloudflare.com/learning/network-layer/what-is-gre-tunneling/, 14.01.2026.

[35] Generic Routing Encapsulation (GRE), [Online]. Available from https://datatracker.ietf.org/doc/html/rfc2784, 14.01.2026.

[36] J. Moore and A. Owen, "Disaster Recovery and Business Continuity in Network Management”, pp 1-30, April 2023, [Online]. Available from https://www.researchgate.net/publication/389687488_Disaster_Recovery_and_Business_Continuity_in_Network, 08.05.2026.

[37] M. Swanson, P. Bowen, A. Phillips, D. Gallup, and D. Lynes, NIST SP 800-34 Rev. 1 – Contingency Planning Guide for Federal Information Systems. National Institute of Standards and Technology, 2010, [Online]. Available from nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-34r1.pdf, 15.05.2026.

[38] R. Hinden, "RFC 3768 – Virtual Router Redundancy Protocol (VRRP)," 2004, [Online]. Available from https://datatracker.ietf.org/doc/html/rfc3768, 20.05.2026.

[39] European Parliament & Council of the EU. (2022). Directive (EU) 2022/2555 – NIS2 – on measures for a high common level of cybersecurity across the Union, [Online]. Available from eur-lex.europa.eu/eli/dir/2022/2555/oj/eng, 17.05.2026.

[40] European Parliament & Council of the EU. (2016). Regulation (EU) 2016/679 – General Data Protection Regulation (GDPR), [Online]. Available from eur-lex.europa.eu/eli/reg/2016/679/oj/eng, 20.05.2026.

[41] Z. Zhao, Z. Li, T. Wang, Z. Lin, and D. Fang, "Post-disaster recovery planning for infrastructure systems based on residents' needs: A hypernetwork approach.",

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

International Journal of Disaster Risk Reduction Volume 118, 15 February 2025, 105258, [Online]. Available from https://www.sciencedirect.com/science/article/abs/pii/S2212420925000822?via%3Dihub, 18.05.2026.

[42] IEEE. (2018). IEEE Std 802.1Q-2018 – Bridges and Bridged Networks. IEEE Standards Association. DOI: 10.1109/IEEESTD.2018.8403927, [Online]. Available from https://ieeexplore.ieee.org/document/8403927, 19.05.2026.

[43] M. Mahalingam, D. Dutt, K. Duda, P. Agarwal, L. Kreeger, T. Sridhar, M. Bursell, and C. Wright, RFC 7348 – Virtual eXtensible Local Area Network (VXLAN), 2014, [Online]. Available from https://datatracker.ietf.org/doc/html/rfc7348, 19.05.2026.

[44] Cisco Systems. (2021). Cisco ACI Multi-tier Architecture, [Online]. Available from https://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/white-paper-c11-742214.html, 21.05.2026.

[45] Cisco Application Centric Infrastructure (ACI), [Online]. Available from https://www.cisco.com/c/en/us/td/docs/dcn/whitepapers/cisco-application-centric-infrastructure-design-guide.html, 18.01.2026.

[46] Cisco Application Centric Infrastructure Virtual Port Channel (VPC) in ACI, [Online]. Available from https://www.cisco.com/c/dam/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/aci-guide-vpc.pdf, 18.01.2026.

[47] VXLAN Forwarding in ACI, [Online]. Available from https://www.dclessons.com/vxlan-forwarding-in-aci, 19.01.2026.

[48] Secure Data Center – Cisco ACI, Secure Firewall, and Secure ADC Design Guide, [Online]. Available from https://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/sdc-aci-radware.html#AppendixDADCSecurityPagesBlockedMessage, 19.01.2026.

[49] Cisco Systems. (2022). Migrating Existing Networks to Cisco ACI – Network-Centric Migration Guide, [Online]. Available from https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/migration_guides/migrating_existing_networks_to_aci.html, 19.05.2026.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Aerial 5G Non-Public Networks: Link Budget, Coverage, and Capacity Analysis

Jyrki T. J. Penttinen
Alphacore Inc.
Tempe, AZ, USA
jyrki.penttinen@alphacoreinc.com

Garth V. Rees
VERN Consulting
Madrid, Spain
garth.rees@verncon.net

Urs C. Schlegel
Independent Researcher
Munich, Germany
urs.c.schlegel@mail.com

***Abstract*—Uncrewed Aerial Vehicles (UAVs) equipped with radios can establish rapid, ad-hoc connectivity in areas where terrestrial infrastructure is unavailable or compromised. Leveraging the virtualized architecture of Fifth Generation (5G) mobile networks, both base station and required minimal core functions can be hosted aloft, enabling agile Internet of Things (IoT) or evolved Mobile Broadband (eMBB) centric private networks for emergency response, expeditionary, military operations, and consumer events. This study evaluates the technical feasibility of UAV-mounted 5G Non-Public Network assembled from commercial off-the-shelf components, comparing the physical radio layer performance of IoT and evolved mobile broadband use cases. Candidate 3GPP architectural options are reviewed, and radio link budget calculations quantify physical layer performance in open and rural environments for a single-UAV. Simulations in dense area type complement the study presenting the impact of UAV height in a single UAV case and a formation of three UAVs. The obtained results highlight the trade-off between frequency band, UAV-altitude, and the resulting radio coverage and data rate, providing design guidance for lightweight, energy-efficient aerial 5G systems.**

Keywords-aerial 5G network; link-budget analysis; uncrewed aerial vehicle (UAV); non-public network (NPN); drone-mounted IoT and eMBB radio service; emergency communications; ad-hoc radio access network (RAN); radio coverage and capacity estimate.

I. Introduction

Apart from their commercial use, cellular systems can be deployed also as complementing ad-hoc networks, e.g., in emergency solutions after a natural disaster that has damaged telecommunications infrastructure, or in scenarios where non-permanent augmented capacity and radio coverage are desired. A Fifth Generation (5G) private mobile network model through temporally deployed base stations can provide a suitable platform for data transfer and signaling in such situations, enabling enhanced communications and situation awareness also for, e.g., defense groups.

However, in temporal ad-hoc use cases, 5G users may be highly mobile, so deploying terrestrial trailer-mounted radio base stations may not suffice, as the varying link conditions alter quality and can result in uncertainties, resulting in radio network outages when users are on the move. An aerial ad-hoc 5G network that follows the underlying users can provide an important opportunity to overcome these challenges.

Based on the originally published study "Design Feasibility and Link Budget Assessment of Aerial 5G IoT and eMBB Connectivity" [1], this paper presents an extended feasibility study of a UAV-based 5G radio network that can be used in various Line of Sight (LOS) scenarios through 5G Non-Public Networks (NPNs).

The 3rd Generation Partnership Project (3GPP) is developing the Internet of Things (IoT) concept further in 5G as a logical continuum from the 4G era, in terms of massive Machine Type Communications (mMTC). 5G IoT in 3GPP is realized through legacy Long-Term Evolution (LTE) -based Narrow-Band IoT (NB-IoT) and LTE for Machines (LTE-M) seamlessly attached to the 5G Core Network (CN) and, from Release 17 onward, through 5G New Radio (NR) Reduced Capabilities (RedCap), which is a native, reduced-bandwidth flavor of NR, and evolves further as of Release 18 [2]. Combining NPN and IoT through aerial platform enables novel means to develop and provide low-power, low-data rate services in very large areas.

The novelty of this research lies in the following: 1) it presents a concept based on 3GPP-defined NPN-architecture and available Commercial Off-the-Shelf (COTS) components to provide aerial IoT and data services to a variety of use cases and 2) it evaluates performance of such a solution comparing UAV-mounted 5G next generation Node B (gNB) performance of evolved Mobile Broadband (eMBB) and mMTC.

This study considers UAVs for local communication to a set of User Equipment (UE) underneath, paving the way for the forthcoming work that will consider the formation of a multi-UAV-based 5G RAN service and automatized location functions through advanced sensing and artificial intelligence.

Section II discusses the state-of-the art of UAV-assisted wireless communication and current gaps. Section III presents IoT and eMBB, and Section IV discusses 3GPP-defined 5G NPN. Section V describes a 5G-UAV concept and discusses UAV-mounted equipment, presenting an example of a feasible set. Section VI describes physical radio aspects, and Section VII presents link budget results for validation of radio network performance applying adequate radio propagation modeling for aerial network and comparing 5G eMBB and IoT use cases that represent two “extremes” in terms of achievable 3D-network coverage areas. Based on in-house -developed software, Section VIII presents radio network coverage and achievable data rate simulations in a combined dense urban and open environment of Sydney, Australia, considering the impact of UAV altitude for a single UAV-housed gNB and a set of gNBs in three UAV constellation. Section IX summarizes the observations of this study and Section X presents our plan for further research.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

II. UAV-Assisted Networking

The global 5G deployments are expanding. The GSM Association (GSMA) estimates that the adaption for 5G will surpass that of 4G in 2028, whereas the earlier networks, 2G and 3G, keep losing their customers; in fact, many of these networks have already been decommissioned [3].

Figure 1 presents an assessment of the data available from [3], showing that soon after 5G has become the dominating mobile communications technology, it also surpasses the sum of the users of all the previous generations 2G, 3G, and 4G.

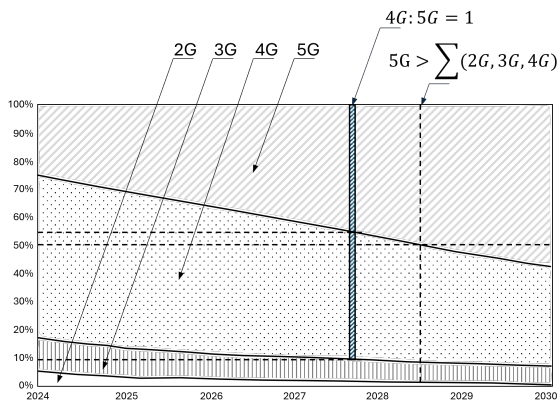


Figure 1. Mobile adoption by technology: percentage of total connections

The current 5G system architecture models enable various deployment options and variations for tailored solutions. Examples of these facilitators include new NPN architectures, Non-Terrestrial Networks (NTNs), Open RAN, and mMTC that are all evolving and being deployed in commercial networks.

While 5G matures, there are already concrete efforts to develop systems beyond 5G (B5G), paving the way for Sixth Generation (6G) [4]. During Release 20, 3GPP has carried out use case and feasibility studies, and the actual forming of technical 6G specifications begins along with the Release 21. The first commercial 6G networks can be expected to be available as of 2030 [5]. The 6G is anticipated to be particularly attractive for connected UAVs due to significant improvements, including ubiquitous 3D connectivity on the ground and in the air [6].

While 6G is still under development, the current 5G systems outperform the previous generations, and can be tailored to provide radio service also beyond traditional terrestrial base stations through Service-Based Architecture (SBA) and Service-Based Interfaces (SBIs) that handle specific needs of varying use cases and dynamically provide optimal sets of required and available resources to different usage types through Network Functions Virtualization (NFV).

The key benefit of 5G is its ability to run Network Functions (NFs) on COTS hardware. This evolution makes 5G a suitable candidate also for UAV-type networking, e.g., through NPN models as they can form an architectural base for isolation (with augmented security) or interconnection / roaming (providing wider connectivity) network segment. A Mobile Network Operator (MNO) or Network Slice (NS) Provider (NSP) can set up NSs, that can be used for deploying UAV-networks, too.

5G systems can be optimized further through Open RAN (Radio Access Network). Examples of the efforts driving Open RAN include Open RAN Alliance's O-RAN [7] and Telecom Infra Project's TIP [8]. Via Open RAN, vendor-specific internal RAN interfaces are opened so that an extended number of stakeholders can provide select RAN protocol layers independently, increasing efficiency and reducing costs [9], and it can be used also in UAV-based networking [10].

As for the State of the Art and challenges, the mobile networks' radio coverage extension mounting a base station or repeater on aerial vehicle has been studied from several points of view, such as how to maximize the radio coverage by optimal UAV positioning [11] and how to enable group handover for drone base stations [12] related to the mMTC, use of data services, and commercial needs for respective coverage and capacity extension to facilitate adequate data rates and Quality of Service (QoS) for the subscribers.

On radio link budget, there are various studies such as [13] (high-altitude platform for 5G access node) and [14] (system model for forward link transmissions in an Integrated Access and Backhaul (IAB) multi-tier drone cellular network). An example of real-world UAV-based networking is AT&T's 5G Cell on Wings (CoW), a drone-mounted cellular 4G or 5G base station that temporarily extends radio coverage, e.g., during disasters and large events [15].

Nevertheless, the available studies are not necessarily conclusive in terms of the tradeoffs of the UAV altitude and wider set of deployed frequency bands [16]. Furthermore, the adaptation of optimal architectural models of UAV-networking, considering the feasibility and gaps of COTS components, can benefit from additional research [17].

III. IoT vs. eMBB

The 5G eMBB and IoT (mMTC) represent opposites in terms of many aspects, like data rates, power consumption and number of simultaneously communicating devices. These elements dictate also the achievable radio coverage area size. For example, IoT has been optimized for bandwidth given that IoT payloads are small, and narrow band lowers the thermal noise floor. IoT relies on robust modulation schemes and Hybrid Automatic Repeat Request (HARQ) repeats balancing respective throughput and coverage. It is important to note that battery-driven IoT modules stay below 1 W to meet license and life constraints (e.g., NB-IoT Class 3 and Class 5 use 23 dBm and 20 dBm, respectively). Table I summarizes some of the key differences of IoT and eMBB.

IV. Aerial Private Network Considerations

3GPP has designed private network realizations through Standalone Non-Public Network (SNPN) and Public Network Integrated Non-Public Network (PNI-NPN) models [18] as presented in Table II. Although 3GPP designed these models

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE I. COMPARISON OF KEY ASPECTS OF 5G EMBB AND IOT

Item	eMBB	IoT
Channel bandwidth	20 – 400 MHz	180 kHz – 20 MHz
SNR (BLER<10%)	8 – 15 dB (64 / 256 QAM)	-13 – -3 dB (BPSK / QPSK, heavy coding)
Fade/penetr. marg.	3 – 5 dB	10 – 15 dB
Device TX power	23 dBm (smartphone)	14 – 23 dBm (sensor)
Data rate target	10 Mb/s – 1 Gb/s	50 b/s – 1 Mb/s

TABLE II. PRIVATE NETWORK TYPES IN UAV-BASED DEPLOYMENTS

Variant	Assessment
SNPN (fully standalone architecture)	Best for autonomous, localized, quick-to-deploy networks (no MNO dependency)
PNI-NPN with radio access network sharing	Enables UAVs to share ground RAN where available, while maintaining separate core
PNI-NPN with core network sharing	UAV-based NPN reuses public 5G core network, allowing leaner deployment
UE route selection via mobile network selection	UEs served by UAVs can select between private and public network profiles

for terrestrial networking, their principles can be extended to serve also in aerial networks.

For the architectural modeling of 5G-based UAV network, the following technical specifications form the base: 1) 3GPP TS 23.501 (System Architecture for the 5G System) defines high-level architecture for both SNPNs and PNI-NPNs [19]; 2) 3GPP TS 23.548 (5G System Enhancements for NPN) explores enhancements specific to NPNs (e.g., management, registration, selection) [20]; and 3) 3GPP TS 22.261 (Service Requirements for 5G System) covers typical service-level requirements relevant to NPNs including verticals, e.g., public safety [21] stating also that 5G is expected to support various enhanced UAV scenarios for applications and scenarios for low altitude UAVs in commercial and government sectors.

SNPN is self-contained and independently operated from public networks. It is adequate for rapid deployment for on-demand aerial networks with no dependency on commercial MNO infrastructure and is thus a match for UAV-based temporal networks serving field units. PNI-NPN, in turn, is an NPN deployed with integration into a public mobile network, and it may share infrastructure (e.g., RAN or CN). The variants of PNI-NPN include RAN-sharing with network slicing, core network sharing, and UEs with public and private subscriptions, referring to Public Land Mobile Network (PLMN) or Network Slice Instance (NSI) selection.

Table II summarizes the 3GPP-defined NPN types and presents their key benefits related to their applicability for forming UAV-based network services.

V. 5G-CAPABLE UAV REALIZATION

The system considered in this study is based on minimal viable 5G SNPN architecture and a single UAV equipped with a 5G gNB enabling local connectivity to the UEs underneath to provide temporarily deployable service if complementing MNO infrastructure is not available.

A. Architecture

Based 3GPP SNPN models [18], Figure 2 presents a respectively adapted UAV-monted SNPN realization in this study. UAV (or set of UAVs) can house radio functions, whereas the essential core network functions of gNBs can be implemented on the same UAVs, separate UAVs, or ground station.

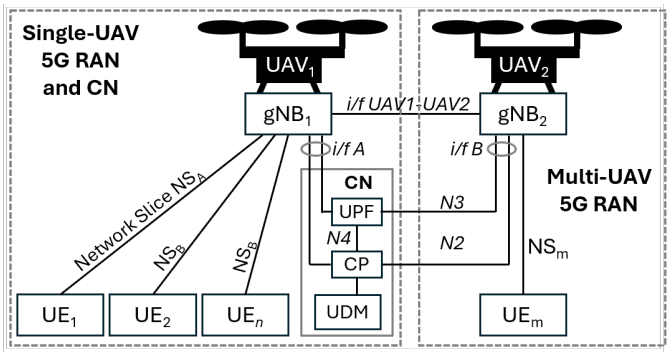


Figure 2. UAV-based, isolated SNPN realization that can be based on a single UAV or multiple interconnected UAVs that house 5G gNB

This model can be extended to cover additional UAVs and respective gNBs that are interconnected (e.g., through low-latency PC5 link between vehicles) forming a 5G RAN drone swarm, e.g., via 5G-based mesh between UAVs.

In multi-UAV 5G radio service provisioning, it can be reasoned that maintaining optimal 3D placement is hard as small altitude shifts swing path loss and backhaul Signal-to-Noise Ratio (SNR), so sufficient overlapping must be ensured. Some additional challenges include satellite positioning jitter or denial that can degrade the Time Division Duplex (TDD) timing and node loss fragments control. To overcome these challenges, control can be made hierarchical (central planner and local packet core), and use of multi-source timing can help in this. The rapid geometry changes are another challenge as they may trigger ping-pong handovers, especially in the case of narrow NR beams that raise beam-failure risk in UAV turns. Also, a dominating cell with a weak multi-hop backhaul can impact negatively the QoS. To overcome this challenge, it is possible to apply larger handover hysteresis (e.g., 3-5 dB) and time to trigger (e.g., 160-320 ms), implement dual connectivity feature (make-before-break), and use backhaul-weighted cell selection.

B. Equipment Considerations

To deploy a 5G SNPN UAV RAN, the UAV can host a basic integrated gNB (e.g., Amarisoft / Parallel Wireless). The basic location management is assumed to be manual and satellite positioning system -assisted, but automated methods can also be developed based on UE signals and using basic Radio Frequency (RF) heuristics (e.g., weighting received signal strength indicator) to position the UAVs according to user density.

Basing the solution on COTS devices, Table III presents a set of feasible candidate elements for simplified functional architecture.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE III. FUNCTIONAL ARCHITECTURE OF AERIAL 5G SYSTEM

Layer	Function	Realization
Radio Access (RA)	gNB (5G standalone) with full UE-to-UE routing support	Lightweight COTS integrated into small cell
Control Plane (CP)	Lightweight distributed UAV logic (also swarm consensus)	Simple microcontroller and onboard logic
Backhaul (BH)	None (fully isolated); direct local P2P 5G	PC5 direct-mode or local user plane function (UPF)
Intel-ligence	Initially manual; advanced version has UE-following	Position, RSSI-based positioning heuristics
UE Signal-ing	Simple beaconing uplink from UEs, e.g., by Synchronization Signal Blocks (SSBs)	Existing 5G UE support
Swarm Comms	When more than one UAV, 5G-based mesh between UAVs	5G PC5 (Sidelink)

Table IV presents examples of UAV-mounted equipment. For advanced alignment of the UAVs and UEs, downward-facing cameras can be considered for UE clustering estimation (COTS-based image processing) and barometers for altitude stabilization. Based on the selected options minimizing the weight of the components, the total UAV payload (essential RAN components) is at minimum approximately 2.5–3.0 kg, which is feasible for medium-class UAVs (e.g., DJI Matrice 300 RTK [22] or similar custom UAVs).

One of the key challenges of UAV-based networking is the limitations of power supply, which represents the major weight of the payload. As an example, the above-mentioned DJI Matrice 300 RTK supports up to 55 minutes operational flight time [22]. For the communication components, Table IV presents a rough estimate of the average and peak power drain. As can be seen, the gNB consumes major part of the total power. For the operational power of the UAVs and 5G RAN components, this feasibility study assumes ideal power management, but in practice, hybrid model can be used with tethered UAVs (permanent anchor nodes with gNBs) and rotating UAVs (fly, recharge, rotate). Complementary power sources can be, e.g., solar panels (small flexible panels on UAV structure extending flight by approximately 10-20%), hydrogen fuel cells (about 2-3× endurance of lithium polymer, LiPo), or tethered power supply forming wired UAVs with unlimited power (for anchor UAVs [23]).

TABLE IV. EXAMPLES OF 5G RAN KEY EQUIPMENT PER UAV (*POWER CONSUMPTION ESTIMATED IN TYPICAL AVERAGE / PEAK WATTS)

Component	Description	Example	Weight	Power*
Integrated 5G Small Cell	Embedded gNB; RU (SA mode)	Amarisoft Callbox Mini / Baicells Nova430	400 – 800g	25 / 40W
Light Compute Module	For basic UAV / swarm logic	Raspberry Pi CM4 or Jetson Nano	100 – 200g	8 / 12W
Simple Mesh Swarm Radio	IEEE 802.11s Wi-Fi 6 / V2V PC5	Compex WLE900VX / 5G module	50 – 100g	6 / 12W
Battery Pack	Standard UAV LiPo	6S 22000 mAh	1.5 – 2.0kg	0.6 / 1W
Positioning Sensors	GPS, IMU	COTS GPS + Pixhawk FC	<100g	3 / 5W

It should be noted that regulation limits the operational boundaries in terms of UAV altitude and maximum radiated power. As an example, there is no general airborne-gNB Effective Isotropic Radiated Power (EIRP) allowance in the USA and the power is what the experimental or carrier's license and service rules permit, often with tight coordination to prevent wide-area interference. US-rules dictate 400 ft for the maximum UAV altitude above ground level (AGL) under Part 107, although higher values can be permitted via waiver. In the EU, routine airborne gNBs are not covered by the standard local/private-5G licenses, so specific or certified flight approval is needed with European Union Aviation Safety Agency (EASA) framework as well as a trial or temporary spectrum license from the national regulator, and the maximum altitude is 120 m AGL.

VI. PHYSICAL RADIO INTERFACE

This feasibility study considers theoretical coverage limits of a single UAV applying relevant propagation loss models, with the aim to provide optimal 5G RAN quality and performance as a function of UAV altitude (and later, inter-UAV distance) in rural and open areas, and the following parameters: 1) Frequency band f = low (1 GHz), mid (3.5 and 6 GHz) and high (24 and 28 GHz); 2) UAV altitude h_{gNB} = 50m, 100m,..., 400m; 3) UE type = pedestrian; 4) Power P_{UE} = +23 dBm, P_{gNB} = +23 dBm; 5) UAV antenna type is omni-directional (0 dBi gain, uniform, ideal radiation pattern). The key results include achievable cell size and respective data rate estimate for both eMBB and IoT use cases.

The starting point of the study was to consider a single UAV that houses 5G SNPN equipment needed to form basic radio access for the UE-UAV$_{\text{gNB}}$-UE communications.

In rural and open-space areas, for the Line of Sight (LOS) scenarios, the Free Space Path Loss (FSPL) L_{FSPL} (in dB) can be estimated by applying the ITU-R P.525 model (within version 12 of ITU-R P.1411) that assumes minimal obstructions [24]. The LOS path loss equation is:

$$L_{FSPL} = 20log_{10}f + 20log_{10}d + 92.45 \quad (1)$$

In this equation, f is the frequency in GHz and d is the distance in kilometers between UAVgNB and UE.

In the feasibility study presented in this paper, Figure 2 depicts the principle of the UAV setup. For the baseline and comparative reference, this study is based on a theoretical omnidirectional UAV-mounted 0 dBi antenna. In practice, isotropic antenna gain is conservative whereas directive antenna enhances the link and optimizes the radio coverage also for minimizing the interference; an example of this is a cone-shaped vertical radiation pattern as depicted in Figure 3.

In field deployments, an adaptive Multiple-In Multiple Out (MIMO) antenna provides augmented performance, capacity, and interference mitigation, but the drawback is the increased complexity and power consumption. To keep the antenna

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

complexity and respective power consumption at minimum in this study, passive omni-directional approach provides means to materialize multi-UAV gNB connectivity without a need for a specific steering logic.

In practice, for the final selection of the antenna type and respective gain, detailed radio network planning is important for ensuring adequate inter-UAV connectivity and radio cell dimensioning. The impact of the final antenna gain can be considered adjusting the presented G_{TX} parameter values.

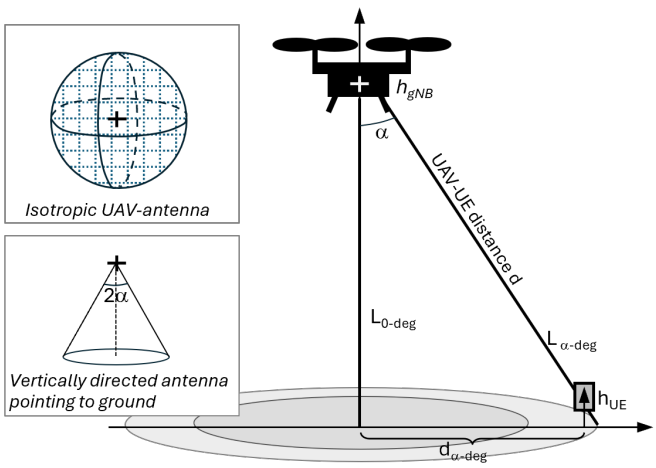


Figure 3. Principle of (theoretic) radiation patterns and respective radio coverage formation assumed in this study

VII. Results

Using ITU-R P.1411, Figure 4 presents the path loss UE-UAV at a distance *d* from UAV's location for isotropic UAV transmitter (TX) antenna at 100 m altitude (earth curvature limit 35 km) for frequency bands 1 GHz – 28 GHz.

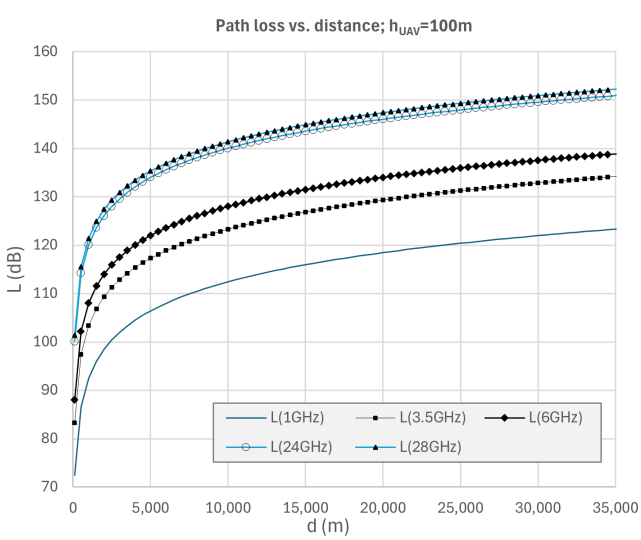


Figure 4. Path loss of UAV-UE as a function of the UE's distance from UAV's vertical reference location (UAV altitude is 100m)

Considering the same frequency range 1 GHz – 28 GHz, Figure 5 summarizes the estimated path loss values L (dB) directly beneath (α=0º) and off the vertical location of the UAV (α=30º) in open and rural areas, when the UAV altitude varies between 50 m and 400 m. The presented LOS path-loss estimates show that the altitude-distance trends for 1–28 GHz match the 3GPP TR 38.901 LOS baseline [25]. Comparable UAV-gNB studies at 28 GHz [26][13] show the same free-space path loss driven scaling, with practical shortfalls mainly from blockage, beam-pointing, and backhaul constraints rather than the LOS model itself. Consistent with geometry, off-nadir (30°) incurs about 1.25 dB extra loss from the longer slant range, and at 100 m altitude the footprint is curvature-limited. Deployment planning can thus start with FSPL-based bounds and subtract environment-specific losses for realistic coverage.

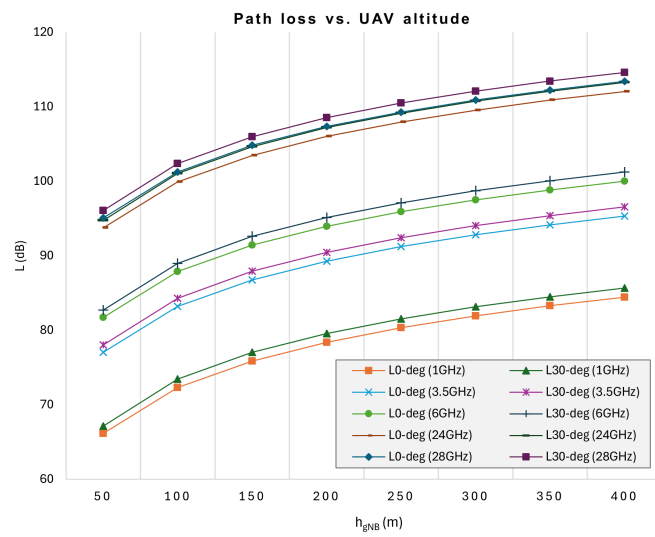


Figure 5. Path loss prediction, 1GHz–28GHz, for UAV altitudes of 50m–400m

A. Scenario: eMBB

Table V presents the key eMBB radio budget items, and Table VI presents an example of the radio link budget when the UAV altitude is 400 m and the UE is located underneath.

In these scenarios, the channel bandwidth is 20 MHz (1 GHz), 100 MHz (3.5/6 GHz), or 400 MHz (24/28 GHz). In these calculations, spectral efficiency model is η (bps/Hz) = $0.6\times\log_2(1+\text{SNR})$. The assumption for the 0.6 factor is due to scheduler, modulator and coding inefficiencies that lower

TABLE V. Applied radio link budget items, eMBB scenario

Parameter	Value	Notes
gNB transmitter power P_{TX}	+23 dBm	Typical small-cell power limit, applicable to a UAV-mounted gNB
TX antenna gain G_{TX}	0 dBi	Isotropic (no beamforming)
UE antenna gain G_{RX}	0 dBi	Smartphone baseline
Fade margin *M*	3 dB	Covers body loss, ageing, fading
UE noise figure *NF*	7 dB	5G NR handset typical
Thermal noise density	–174 dBm/Hz	$N_o=kT=1.38\times10^{-23}$ J/K × 290K

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE VI. EXAMPLE OF THE RADIO LINK BUDGET (D=400M)

Link budget, h_{UAV}=400m, α=0°	**1 GHz**	**3.5 GHz**	**6 GHz**	**24 GHz**	**28 GHz**
Path loss PL, dB at (2km)	84.5	95.4	100.1	112.1	113.4
Tx power (gNB), dBm	23.0	23.0	23.0	23.0	23.0
Tx antenna gain, dB	0.0	0.0	0.0	0.0	0.0
UE antenna gain, dBi	0.0	0.0	0.0	0.0	0.0
Implementation/fade margin, dB	3.0	3.0	3.0	3.0	3.0
UE noise figure NF, dB	7.0	7.0	7.0	7.0	7.0
Thermal noise density, dBm/Hz	-174	-174	-174	-174	-174
Channel bandwidth, MHz	20	100	100	400	400
Received power Prx, dBm	-61.5	-72.4	-77.1	-89.1	-90.4
Noise floor N, dB	-94.0	-87.0	-87.0	-81.0	-81.0
Operational SNR after margin, dB	29.5	11.6	6.9	-11.1	-12.5
Spectral efficiency SE	5.88	2.38	1.54	0.06	0.05
Data rate, Mb/s	**117.6**	**237.5**	**154.4**	**25.8**	**19.1**

the theoretic capacity of Shannon limit. The received power is $P_{RX} = P_{TX} + G_{TX} + G_{RX} - L$, and the noise floor is $N = -174 + 10 \log_{10}B + NF$. The operational signal to noise ratio (SNR) after margin is SNR = $P_{RX} - N - M$. The spectral efficiency $SE = 0.6\times\log_2(1+10^{SNR/10})$, and the data rate is $R = SE \times B$. Beyond the presented calculations, the effective SINR enhances through beamforming gain G_{BF} that is typically 3-10 dB whilst the level of interference lowers it respectively. MIMO, in turn, adds rank often 1-2 aloft scaling efficiency, and resource allocation gives a per-UE share 0.3-0.8. [27]

Figures 6 and 7 summarize the impact of h_{UAV} (50 m – 400 m) on the received single user data rate considering the space below the UAV and surrounding region. The results are based on analytical modeling of path loss prediction and a set of radio link budget attribute values (select examples presented in Table V and VI), varying the UAV altitude.

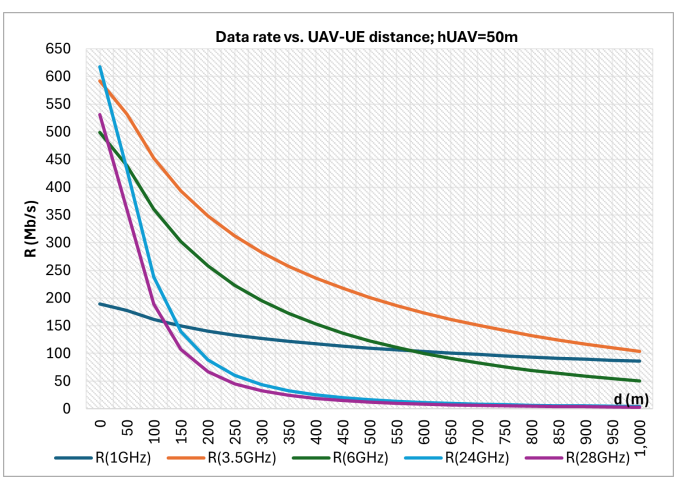


Figure 6. Data rate for UAV-mounted 5G gNB at 50 m altitude

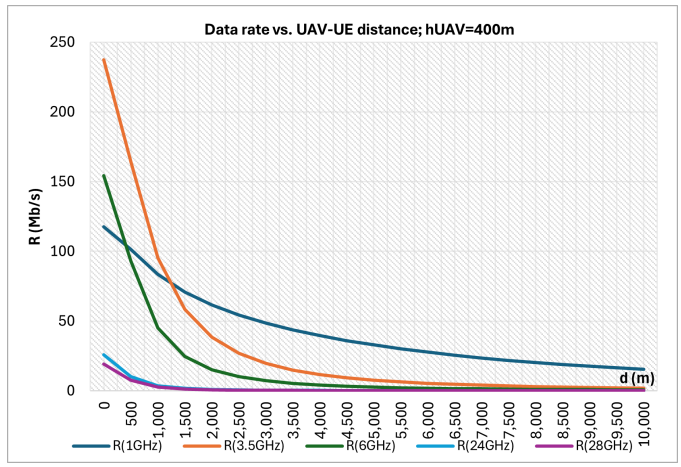


Figure 7. Data rate for UAV-mounted 5G gNB at 400 m altitude

As can be seen, high-band (24 GHz and 28 GHz) provides the highest rates when the distance between UAV_{gNB} and UE is relatively short, but the rate lowers drastically as the distance between UAV and UE increases over a few hundred meters due to the strong attenuation of this band. As can be expected, the mid-band (3.5 GHz and 6 GHz) performs more constantly at short distances and nearby regions.

Taking a closer look at the short distance (5–200 m) between the UAV and UE, Figure 8 shows more detailed behavior. As can be seen, high-band outperforms the other bands up to about 50 m distances providing 1–2 Gb/s data rates, but afterwards, mid-band provides the highest rate (250–500 Mb/s). The heavy attenuation of the high-band makes also the low-band data outperform it beyond 150–200 m distance.

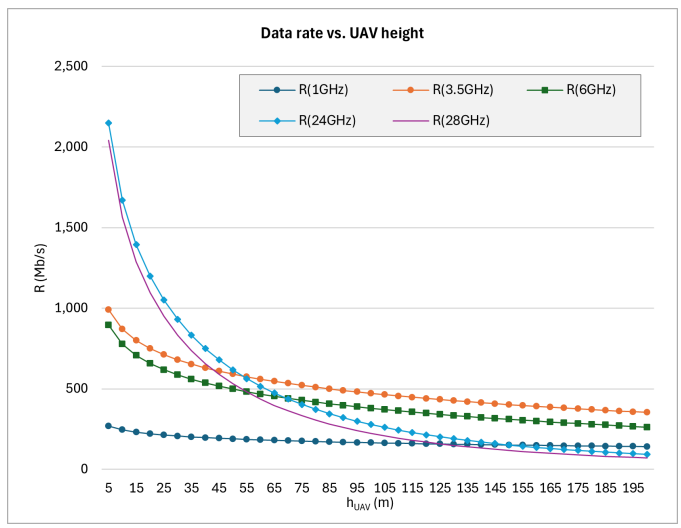


Figure 8. Data rate behavior when the distance between the UAV and UE is short (5-200m)

Figure 8 shows that for a critical mission in open and rural areas, assuming the key requirement is fast data connectivity and high capacity, e.g., for high-definition video contents, high-band provides optimal performance up to about 100 m.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

If, instead, the main requirement is a large coverage area (e.g., over 10 km), and the UAV operation is possible at high altitude (e.g., 400 m), low-band is adequate selection as Figure 8 indicates. Should there be limitations for the UAV altitude, such as nearby airports or other restricted areas, mid-band (particularly 3.5 GHz) provides the most adequate balance for h_{UAV} and radio performance, as can be seen in Figure 6.

B. Scenario: IoT

Table VII presents key radio budget items for the IoT scenario comparing NB-IoT class CE0 and CE2, LTE-M, and RedCap.

TABLE VII. APPLIED RADIO LINK BUDGET ITEMS, IOT

Link budget	**NB-IoT$_{CE0}$**	**NB-IoT$_{CE2}$**	**LTE-M**	**Red Cap**
Channel bandwidth, MHz	0.18	0.18	1.40	5.00
Thermal noise floor, dBm	-114.4	-114.4	-105.5	-100.0
Impl. / fade margin, dB	8.0	14.0	8.0	5.0
Required SNR / E_b/N_o	-5.0	-13.0	-7.0	-3.0
RX sensitivity, dBm	-119.4	-127.4	-112.5	-103.0
Peak data rate, b/s	25k	50-100	1M	150M
Path loss budget, dB	**134.4**	**136.4**	**127.5**	**121.0**
Distance (1GHz), km	»35	»35	»35	̃25
Distance (3.5GHz), km	̃35	>35	̃15	̃7

In all these cases, the gNB transmitter (TX) power is 23 dBm, TX and receiver (RX) antenna gains are 0 dBi, and RX noise figure is 7 dB (typical low-cost IoT modem). Again, the free-space path loss model is used in these calculations. Receiver sensitivity is driven by bandwidth, i.e., noise floor plus the required SNR for the lowest modulation / coding of each profile. Path-loss budget is the maximum loss the link can tolerate after allocating the fade / implementation margin. Converting that budget to free-space distance shows the theoretical cell radius; real-world coverage will be smaller due to UAV altitude (earth curvature), foliage, buildings and interference. The presented SNR / (E_b/N_0) values are for the lowest-order modulation / coding in each profile (reference sensitivity) as per the UE RF specifications [28][29], and the RX sensitivity is N + Required SNR (cross-checked with the specifications for minimum guaranteed UE sensitivity).

As can be seen, NB-IoT can reach the radio horizon even with only 23 dBm EIRP; coverage is limited by geometry, not RF. LTE-M at 1 GHz still covers tens of kilometers, whereas at 3.5 GHz it shrinks to about 15 km LOS. NR RedCap offers the highest data rates but requires roughly 10 dB more SNR than NB-IoT, confining its cell to single-digit-kilometer radii at 3.5 GHz. These Tables can be used directly to size UAV altitude, antenna gain, or additional power needed for a given IoT service profile. Compared with the broadband case (with MCL 95–105 dB), IoT enjoys tens of dB link budget head-room mainly from the narrow bandwidth and low SNR requirement. Thus, all the presented IoT cases outperform the radio coverage of the eMBB.

VIII. SIMULATIONS

As a step beyond conventional radio link budget estimates, our research team developed a simulation environment based on in-house software tools. The software was developed by the authors for use in normal terrain-based radio prediction modeling across 300-4,500 MHz Frequency Range 1 (FR1) bands and other modules for Frequency Range 2 (FR2) millimeter-wave systems, and has been modified to enable the generation of radio predictions from both static and mobile aerial base-station platforms.

The objective of these simulations was to evaluate practical deployment scenarios for a UAV-mounted 5G gNB and its associated antenna system. The simulation framework extends the default base-station configuration parameters used in typical terrestrial radio network planning tools, enabling the modeling of aerial radio nodes and their interactions with ground-based UEs.

The radio prediction modeling toolset is colloquially known as AARON. It is designed primarily to operate on servers with significant Graphics Processing Unit (GPU) capability, as it utilizes novel ray-tracing algorithms to develop the prediction layers, in conjunction with digital elevation models for the elevation data, with a resolution down to one square foot (0.093 m^2) per pixel, and supporting multiple different coordinate reference sets which may be used to ensure accurate alignment of terrain data and mapping representations.

The toolset allows for computation of Reference Signal Received Power (RSRP), Signal-to-Interference-plus-Noise Ratio (SINR), and cell capacity estimations, assuming fully utilized Resource Block (RB) allocations with consideration of the defined cells in the cell configuration file. This is available across the current approved range of Frequency Division Duplex (FDD) and Time Division Duplex (TDD) cellular frequencies, with differing bandwidths, from 1.4 MHz up to 100 MHz, and can be tailored for additional bandwidths, as required.

The AARON toolset contains a range of modules, including dedicated modules to extract and process elevation data from high-definition Light Detection and Ranging (LiDAR) data files (.LAZ format), which are typically available for many countries through central mapping agencies or data repositories.

The toolset can be configured for a range of computational parameters, particularly related to different antenna designs, where antenna beam roll-off, -3 dB vertical and horizontal beam width can be defined. There is support for massive MIMO antenna structures, with variable beam-counts, allowing for a User to define a specific antenna for a use case.

AARON supports different propagation models, with the use of free space loss as the default for the purposes described herein, with a modified COST-231 model is utilized for predictions using terrain-based base stations.

The toolset is configured through three Comma Separated Variable (.CSV) files, generated using an external text editor tool, providing the following files:

- Master File (for multiple project listings).
- Project Configuration file, where the defined project paths and tool configuration data is stored.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

- Cell configuration file, which provides individual cell parameters, such as site location, antenna height, -3dB beamwidths, mechanical and electrical tilt parameters, RF power (typically RS power for LTE, and PBCCH block power for 5G).

This allows for relatively straight-forward adaptations and modification of the tool's behavior, for different prediction scenarios.

The toolset has some newly developed techniques for allowing a set of sequential predictions to be generated to emulate one or more mobile aerial base stations, as well as fixed ground stations to be defined. This allows for generating predictions for given time sequences and paths, and displayed as animated graphical presentation for the users.

In conventional radio network coverage prediction tools, the underlying assumptions typically reflect a tower-mounted base station antenna configuration. These models incorporate standard link-budget components such as feeder cable losses, connector attenuation, sector antenna orientation, and mechanical or electrical downtilt optimized for terrestrial coverage scenarios. However, in the case of a UAV-mounted gNB, several of these assumptions change due to the three-dimensional mobility of the aerial platform and the absence of fixed infrastructure.

For aerial base stations, antenna orientation and coverage patterns differ significantly from traditional sectorized cellular deployments. Rather than directing the antenna main lobe horizontally or slightly downward toward a predefined service area (such as a roadway corridor or urban sector) a UAV platform typically operates in a dynamic 3D environment. In such conditions, the antenna must provide coverage not only to ground users but potentially also to other aerial nodes, including additional UAV-mounted gNBs or UAV-based user equipment. This is especially relevant in rapidly deployable temporal aerial networks for fast response and disaster recovery [30]. Consequently, a practical and lightweight solution is the use of a passive omnidirectional antenna mounted on the UAV platform, enabling approximately uniform radio coverage in the surrounding airspace and toward the ground users beneath the drone.

In the simplest implementation, a passive omnidirectional antenna configuration can be approximated using a half-wave or full-wavelength dipole antenna. Such antennas offer relatively uniform azimuth radiation without requiring power-consuming beamforming hardware. This is advantageous for UAV platforms where payload weight and onboard energy consumption are critical constraints. Passive antennas require no additional signal processing or electronic beam steering, which reduces system complexity, weight, and power consumption compared to active antenna arrays or MIMO beamforming systems.

Figure 9 illustrates the well-known conceptual principle of an omnidirectional UAV antenna radiation pattern. This type of antenna can serve both ground users and aerial nodes within the surrounding three-dimensional space.

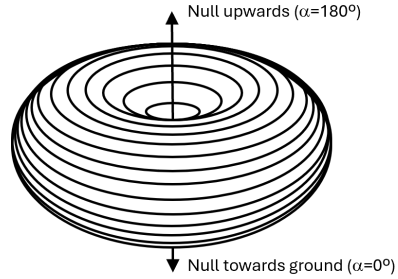


Figure 9. Principle of (close to) omnidirectional UAV antenna serving users on the ground and other UAV nodes in 3D space

To implement a simple, practical, and passive antenna element in the simulation environment, a 1-λ dipole antenna model was used. From a system engineering perspective, passive antennas provide several advantages for UAV-mounted gNB platforms. Active beamforming antennas, although capable of higher gain and adaptive beam steering, require additional RF chains, digital processing, and power amplifiers, which significantly increase both system mass and power consumption. For small and medium UAV platforms, the available payload capacity and onboard energy budget are limited. Therefore, passive antenna solutions often represent a realistic trade-off between performance, weight, complexity, and flight endurance. Several studies on UAV-based communication platforms emphasize that the communication payload, including the base station and antenna subsystem, must remain lightweight and energy-efficient in order to maintain practical flight durations and operational feasibility [30][31].

The radiation pattern of a dipole antenna is commonly described as "donut-shaped," where the maximum radiation occurs in directions perpendicular to the antenna axis, while nulls appear along the antenna axis itself. In the context of a UAV-mounted antenna, this means that the strongest radiation occurs horizontally around the UAV, while weaker radiation exists directly above and below the antenna.

At first glance, this radiation pattern might appear sub-optimal for ground coverage because the antenna exhibits a null directly beneath the UAV. However, in practical scenarios this limitation is partially mitigated by relatively short propagation distance between the UAV and ground-based user equipment directly below it, compared to the respective cell edge. The resulting free-space path loss reduction can compensate for the reduced antenna gain in the vertical direction.

To evaluate this effect, the first simulation rounds were conducted for the 3.5 GHz mid-band spectrum, which represents a common 5G deployment band worldwide. The simulated scenario considered a UAV-mounted gNB positioned above the Bondi Junction shopping center area in Sydney, Australia. The UAV altitude in this scenario was modeled to be approximately 500 m above ground level, which illustrates the capability for wide-area coverage for activities for search and rescue, or emergency service deployment in disaster zones. Please note that while regulation limits the maximum altitude of drones, in this specific case, the aim was to understand technical

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

limitations applicable to these special use cases assuming the respective policies may be relaxed.

The initial prediction uses a singular static UAV, with a further prediction introducing an additional two static UAV's (one to the east, the second further south of the initial UAV), to simulate the impacts of interference on predicted downlink throughput. The radio parameters used in this simulation included a 100 MHz channel bandwidth, numerology μ = 1 (corresponding to 30 kHz sub-carrier spacing), and a total transmit power of 5 watts, consistent with currently approved RF power limitations for mid-powered cell equipment, and considered suitable for UAV payload integration. The antenna gain of the simulated dipole was 2.3 dBi in the main radiation directions.

Terrain data used in the simulation had a spatial resolution of approximately one meter, derived from LiDAR point-cloud measurements. Such high-resolution terrain information allows accurate modeling of ground elevation and environmental features affecting signal propagation. As illustrated in Figure 10, the relatively flat terrain in the Bondi Beach direction results in higher received signal strength compared with other directions at similar distances from the UAV. However, the coverage area directly beneath the UAV shows minimal reduction due to the "donut" effect due to the dipole antenna radiation null.

The signal pattern in Figure 10 suggests a relatively even spread of RF Power on the Downlink direction, minimizing "hot-spots" where UE Receiver may experience saturation levels, with signal distortion. When the additional UAV's are included in the prediction, the coverage prediction of Figure 11 shows widespread coverage across the majority of the area, with a central left-to-right band of marginally lower signal, from the lateral separation of UAV's 1 and 3.

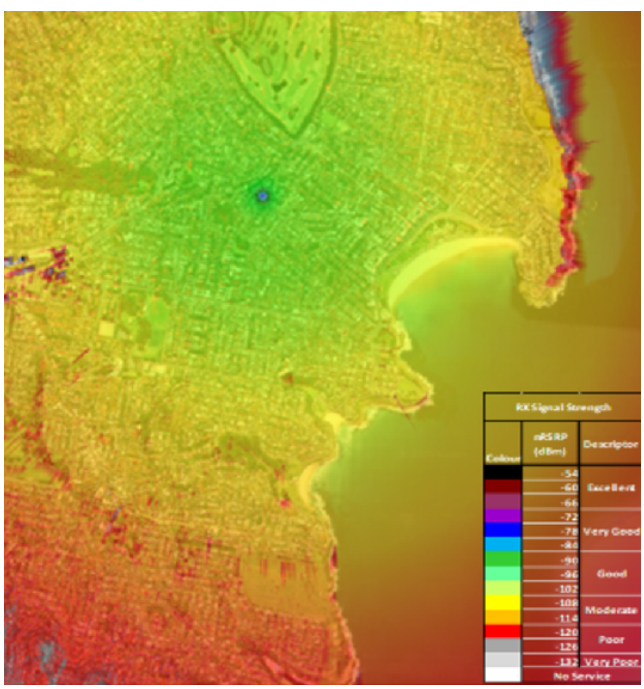


Figure 10. Singular Coverage prediction for UAV-mounted 5G gNB at 500 meters over Sydney area

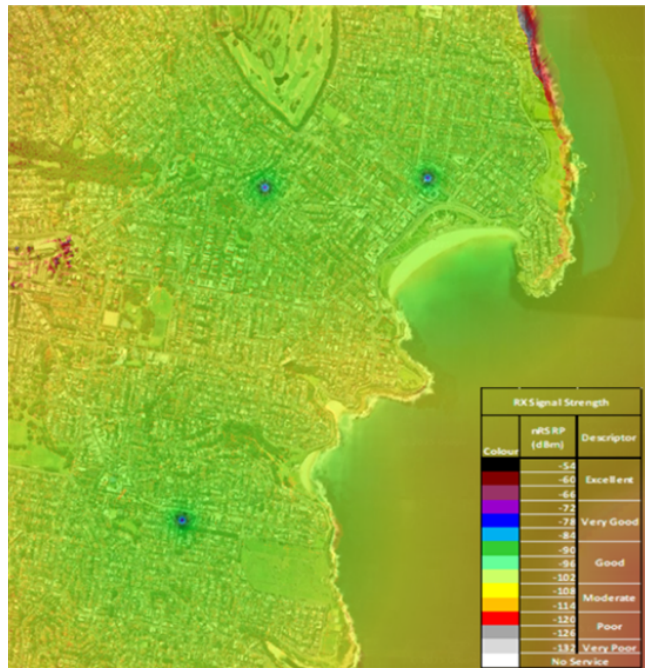


Figure 11. Three-UAV Coverage prediction for UAV-mounted 5G gNB at 500 m over Sydney area

The 5G-SA capacity prediction in Figure 12 illustrates the shortcomings of utilizing small separation between UAV's at a significant operating height, such as in this case, 500 meters.

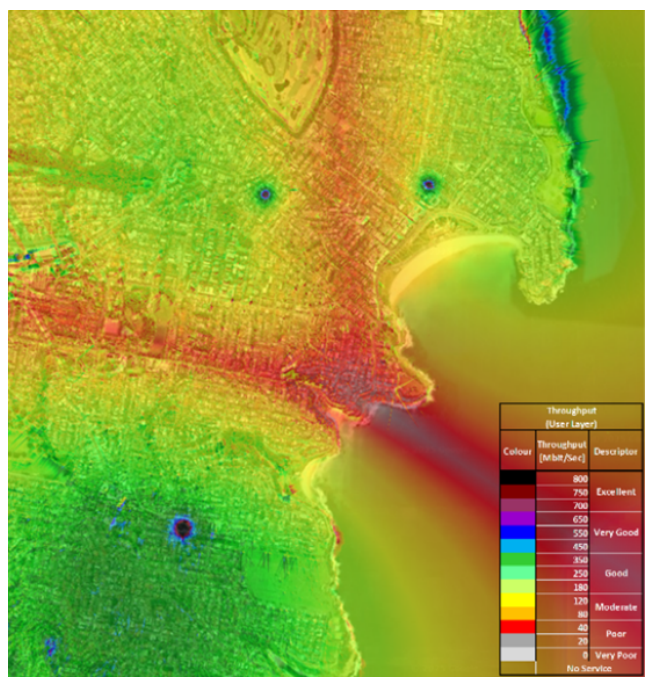


Figure 12. Three-UAV Capacity prediction for UAV-mounted 5G gNB at 500 m over Sydney area

There is significant reduction in the capacity capability of UAVs 1 and 2, due to the excessive overlap of the coverage between the two patterns, with a relatively broad swathe of low-capacity area running North-South as marked in the

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

orange and red colors. A similar pattern is observed at the cell edges between UAV's 1 and 3, however it presents a a sharper delineation of red color, with narrower bands of orange.

This indicates the reduction of inter-cell interference at the cell edges, with a higher propensity for a clean "break" on the handover between the two (reduced cell "ping-ponging"). The additional lateral separation of UAV 3 can also be observed by the much larger area with dark-green, blue, and purple coloring, indicating significantly higher downlink throughput capability compared to that of UAV 1 and UAV 2, which are in proximity.

The second simulation round was conducted for UAV's at an altitude of 200 m. In this case, the antenna gain was assumed to be approximately 6.5 dBi, representing a slightly higher-gain omnidirectional antenna configuration. The effective transmit power was approximately 2 W total cell power, corresponding to small-cell-class gNB equipment designed for portable or aerial deployments.

Compared to the first scenario, the higher altitude increases the overall coverage radius of the UAV cell, although at the expense of reduced received signal strength at longer distances due to increased free-space path loss. This reflects the typical trade-off between coverage area and signal quality when determining the optimal UAV operating altitude.

Figure 13 illustrates the resulting signal strength distribution at the same geographical location. For the selected parameter values, i.e., 100 MHz bandwidth, numerology $\mu = 1$ (30 kHz subcarrier spacing), and approximately 2 W total cell transmit power, the simulated data rate distribution is consistent with expectations for a lightweight aerial small-cell system operating in the 3.5 GHz band.

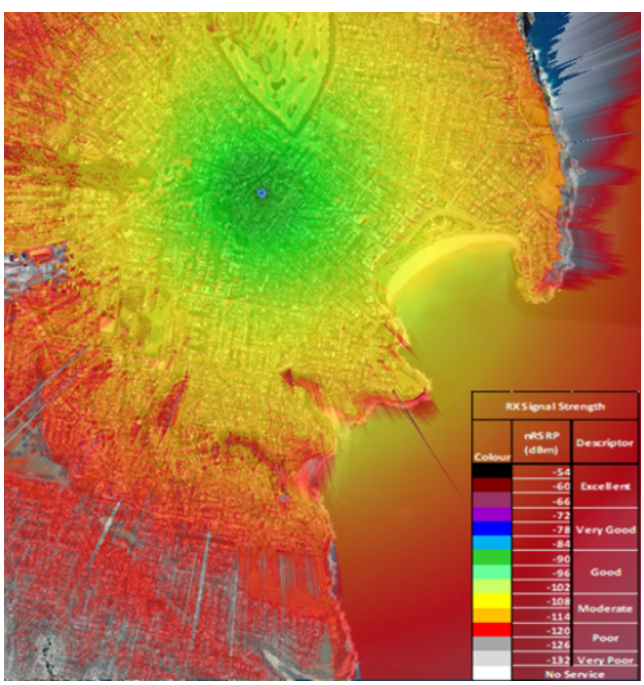


Figure 13. Singular Coverage for UAV-mounted 5G gNB over Sydney area at 200 m altitude

Figures 14 and 15 illustrate how lower altitude, combined with lower transmit power and higher-gain antennae assist in providing a tighter coverage performance, with significant improvements in cell-delineation, greater areas with high capacity coverage, and increases in all UAV zones of areas where high Downlink Throughput can be expected.

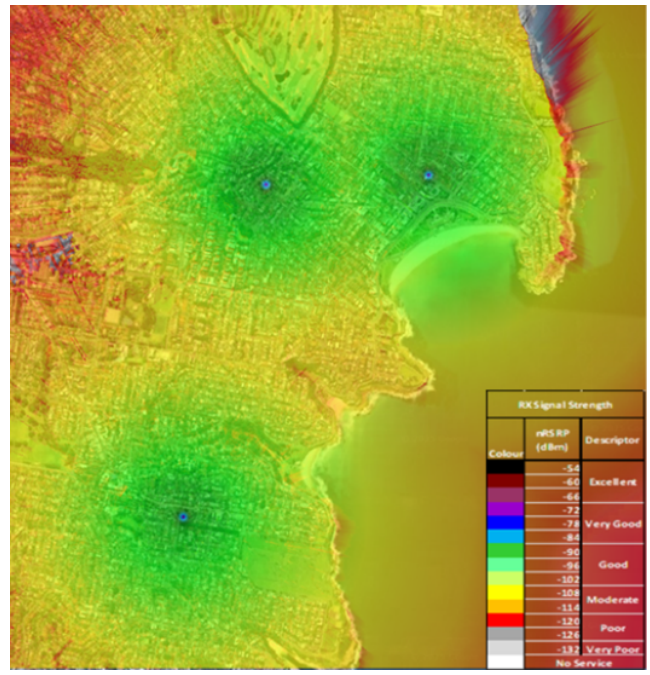


Figure 14. Three-UAV Coverage prediction for UAV-mounted 5G gNB at 200 m over Sydney area

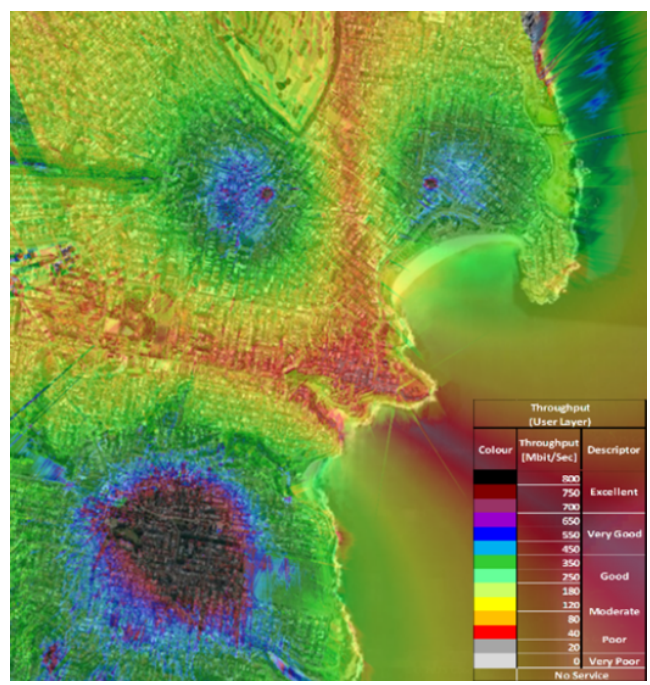


Figure 15. Three-UAV Capacity prediction for UAV-mounted 5G gNB at 200 m over Sydney area

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Comparing the two scenarios highlights several important observations. At lower altitudes, the UAV provides stronger signal levels and higher data rates within a tighter coverage footprint. At higher altitudes, the coverage area expands significantly, but the average throughput per user decreases due to increased path loss. These results are consistent with findings reported in several UAV-base-station studies, which indicate that there exists optimal UAV altitudes and spatial separation that balances coverage radius and signal quality depending on the deployment environment and service requirements.

IX. Summary

This study demonstrates that a single-UAV 5G SNPN assembled from COTS components can provide rapid, standards-based connectivity for both eMBB and IoT use cases, particularly in open or rural terrain environments.

Using the ITU-R P.1411 LOS propagation formulation as the baseline model, we analyzed how operating frequency (1–28 GHz) and UAV altitude (50–400 m) influence the trade-off between coverage area and achievable throughput. The observed altitude–distance relationships and off-nadir propagation characteristics are primarily driven by FSPL. At lower altitudes, coverage is limited primarily by geometric footprint constraints, while at higher altitudes the achievable range becomes increasingly dominated by path loss.

Millimeter-wave bands offer the highest short-range data rates but degrade rapidly with increasing distance due to higher path loss and sensitivity to blockage. In contrast, mid-band frequencies such as 3.5–6 GHz provide a balanced compromise between coverage radius and throughput, making them particularly suitable for UAV-mounted cellular nodes. Lower frequencies near 1 GHz can maximize coverage area at higher altitudes but offer reduced overall system capacity.

Link-budget templates developed for eMBB and IoT traffic profiles further show that IoT connectivity typically benefits from much larger link margins due to narrower bandwidths and lower signal-to-noise ratio requirements. In many IoT scenarios, the coverage limit is determined more by geometric line-of-sight constraints than by radio link budget limitations.

From a practical deployment perspective, the simulations confirm that many principles used in conventional radio network planning tools can be adapted for UAV-mounted gNB systems. However, several additional considerations must be incorporated when modeling aerial cellular nodes related to adequate antenna selection, understanding power and payload constraints, impact of altitude, differences in 3D modeling in terms of mobility and positioning.

Lightweight passive omnidirectional antennas are often preferred for UAV platforms due to their low power consumption and reduced system complexity. Active beamforming arrays can provide higher gain but significantly increase energy consumption and payload weight, so the passive antenna approach may be critical in the lowest power consumption scenarios. Of the power and payload constraints in general, the onboard power budget and payload capacity of the UAV impose strict limits on radio hardware selection and transmission power levels. This study shows that optimal altitude selection is one of the key tasks as UAV altitude must balance coverage area, path loss, regulatory restrictions, and flight endurance.

For the 3D coverage aspects, unlike is the case in terrestrial networks, aerial base stations require modeling of coverage both horizontally and vertically to account for ground users and aerial nodes so that UAV movement can be exploited to dynamically optimize coverage, capacity, and interference conditions. Modeling of interference interactions between dynamic SNPN gNBs and co-channel terrestrial gNBs need to be explored, too, in detailed 3D radio network planning.

This presented experimentation shows significant promise toward developing clear guidelines for complex environments where intelligent decision-making for gNB positioning, flight-paths and use of terrain features are critical for real-time interference management. These results provide useful baseline parameters for future studies involving multi-UAV aerial networks, cooperative UAV base station deployments, and AI-assisted network optimization.

X. Future Research

Building upon the presented research, our future work will expand the evaluation of UAV-mounted 5G networks across a broader range of terrain environments, including low-rise suburban areas and dense urban environments with high-rise buildings with respective radio propagation models capable of capturing complex air-to-ground channel characteristics. Our further research will also aim to explore multi-UAV network architectures where multiple drone-mounted gNBs operate collaboratively to form aerial cellular networks. Such systems may employ wireless backhaul links between UAVs or IAB architectures to extend network coverage dynamically and algorithms to optimize the inter-UAV distances as a function of the UAV altitude, considering the balancing of provided coverage and capacity. These characteristics will support a range of more complex use cases particularly directed toward geo-location capabilities for search and rescue and similar activities.

Another important, potential research direction involves the use of artificial intelligence and machine learning methods for UAV network coordination. AI-assisted positioning algorithms could enable autonomous UAV placement based on real-time user distribution, traffic demand, and interference conditions. Recent research suggests that intelligent UAV placement and trajectory optimization can significantly improve network performance, coverage fairness, and energy efficiency in aerial communication systems.

Finally, advanced sensing technologies and environmental awareness may allow UAV-mounted gNB systems to dynamically adapt their position, altitude, and transmission parameters in response to changing radio conditions, interference patterns, and user mobility. Such capabilities, derived from, e.g., 3GPP's ISAC (Integrated Sensing and Communication) work item could enable fully autonomous aerial communication networks capable of rapid deployment during emergency situations, temporary events, or infrastructure outages.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

ABBREVIATIONS

3GPP 3rd Generation Partnership Project
5G Fifth Generation mobile communications
6G Sixth Generation mobile communications
AGL Above Ground Level
B5G Beyond 5G
CN Core Network
COTS Commercial Off-the-Shelf
CoW Cell on Wings
CSV Comma Separated Variable
EASA European Union Aviation Safety Agency
EIRP Effective Isotropic Radiated Power
eMBB evolved Mobile Broadband
FDD Frequency Division Duplex
FSPL Free Space Path Loss
gNB Next Generation Node B
GPU Graphics Processing Unit
GSMA GSM Association
HARQ Hybrid Automatic Repeat Request
IAB Integrated Access and Backhaul
IoT Internet of Things
ISAC Integrated Sensing and Communication
LiDAR Light Detection and Ranging
LiPo Lithium Polymer
LOS Line of Sight
LTE Long-Term Evolution
LTE-M LTE for Machines
MIMO Multiple-In Multiple Out
mMTC Massive Machine Type Communications
MNO Mobile Network Operator
NB-IoT Narrow-Band IoT
NF Network Function
NFV Network Functions Virtualization
NPN Non-Public Network
NR New Radio
NS Network Slice
NSI Network Slice Instance
NSP Network Slice Provider
NTN Non-Terrestrial Network
PBCCH Physical Broadcast Channel
PLMN Public Land Mobile Network
PNI Public Network Integrated
QoS Quality of Service
RAN Radio Access Network
RB Resource Block
RedCap Reduced Capabilities
RF Radio Frequency
RS Reference Signal
RSRP Reference Signal Received Power
RX Receiver
SBA Service-Based Architecture
SBI Service-Based Interface
SINR Signal-to-Interference-plus-Noise Ratio
SNPN Standalone Non-Public Network
SNR Signal-to-Noise Ratio
TDD Time Division Duplex
TIP Telecom Infra Project
TX Transmitter
UAS Uncrewed Aerial Vehicle
UE User Equipment

ACKNOWLEDGEMENT

We gratefully acknowledge the feedback of Professors M. Reisslein, G. Trichopoulos, S. Zeinolabedinzadeh, and A. Alkhateeb of Arizona State University on the link budget considerations of this work.

REFERENCES

[1] J. Penttinen, "Design Feasibility and Link Budget Assessment of Aerial 5G IoT and eMBB Connectivity," The First International Conference on AI-enabled Unmanned Autonomous Vehicles and Internet of Things for Critical Services, IARIA AIVTS 25, Barcelona, Spain, pp 1-7, 2025.
[2] 3GPP, "TR 38.865: Study on further NR RedCap UE complexity reduction," 2022.
[3] GSMA, "Mobile economy report 2025," GSMA Intelligence, London, UK, 2025.
[4] J. Penttinen, "On 6G visions and requirements," Journal of ICT Standardization, vol. 9, no. 3, pp. 311-325, 2021.
[5] P. Jain, "3GPP SA 6G planning and progress," 3GPP, 2024.
[6] G. Wikstrom and S. Thorson, "Explore the impact of 6G," Ericsson, 19 December 2024. https://www.ericsson.com/en/blog/2024/12/explore-the-impact-of-6g-top-use-cases-you-need-to-know. Accessed 6 May 2026.
[7] O-RAN Alliance, "Who we are," https://www.o-ran.org/who-we-are. Accessed 6 May 2026.
[8] Telecom Infra Project (TIP), "How TIP Works," https://telecominfraproject.com/how-we-work/. Accessed 6 May 2026.
[9] Ericsson, "Industrializing Open RAN," https://www.ericsson.com/en/openness-innovation/open-ran-explained. Accessed 5 May 2026.
[10] M. Mushi, Y. Liu, S. Sreenivasa, O. Ozdemir, I. Guvenc et al., "Open RAN testbeds with controlled air mobility," Computer Communications, vol. 228, Article 107955, 2024.
[11] M. Saym, M. Mahbub, and F. Ahmed, "Coverage maximization by optimal positioning and transmission planning for UAV-assisted wireless communications," in International Conference on Science & Contemporary Technologies (ICSCT), Dhaka, Bangladesh, pp. 294-297, 2021.
[12] Y. Aydin, G. Karabulut, E. Ozdemir, and H. Yanikomeroglu, "Group handover for drone base stations," IEEE Internet of Things Journal, vol. 8, no. 18, pp. 13876-13887, 2021.
[13] M. Alamgir and B. Kelley, "Fixed Wing UAV-based Non-Terrestrial Networks for 5G millimeter wave Connected Vehicles," in IEEE 13th Annual Computing and Communication Workshop and Conference (CCWC), pp. 1167-1173, 2023.
[14] A. Fouda, A. Ibrahim, and I. Guvenc, "UAV-Based in-band Integrated Access and Backhaul," in IEEE 88th Vehicular Technology Conference (VTC-Fall), Chicago, IL, USA, pp. 1-5, 2018.
[15] S. Howe, "ATT's flying COW transmits 5G network by tethered drone," Commercial UAV News, 29 June 2022. https://www.commercialuavnews.com/public-safety/at-t-s-flying-cow-transmits-5g-network-by-tethered-drone. Accessed 6 May 2026.
[16] N. Tafintsev, A. Chiumento, O. Vikhrova, M. Valkama, and S. Andreev, "Utilization of UAVs as flying base stations in urban environments," in 15th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Ghent, Belgium, pp. 7-11, 2023.
[17] P. Valente, A. Teixeira, M. Luis, D. Raposo, P. Rito, and S. Sargento, "Easy to deploy UAV-based non-public open-source 5G network," in 15th International Conference on Network of the Future (NoF 2024), Barcelona, Spain, pp. 72-80, 2024.
[18] J. Penttinen, J. Collin, and J. Pellikka, "On Techno-Economic Optimization of Non-Public Networks for Industrial 5G Applications," in The Nineteenth Advanced International Conference on Telecommunications (AICT 2023), Nice, France, pp. 1-7, 2023.
[19] 3GPP, "TS 23.501: System architecture for the 5G System (5GS)," 2025.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

[20] 3GPP, "TS 23.548: 5G System Enhancements for Edge Computing; Stage 2," 2025.
[21] 3GPP, "TS 22.261: Service requirements for the 5G system," 2025.
[22] DJI, "Support for Matrice 300 RTK," https://www.dji.com/support/product/matrice-300. Accessed 6 May 2026.
[23] "Drone power feed through ground tethering," 17 April 2025. https://yle.fi/a/74-20156841. Accessed 6 May 2026.
[24] ITU, "Recommendation P.1411-12 (08/2023)," ITU-R, August 2023. https://www.itu.int/rec/R-REC-P.1411-12-202308-I/en. Accessed 6 May 2026.
[25] 3GPP, "TR 38.901: Study on channel model for frequencies from 0.5 to 100 GHz (Release 19)," 2025.
[26] M. Gapeyenko, I. Bor-Yaliniz, S. Andreev, H. Yanikomeroglu, and Y. Koucheryavy, "Effects of Blockage in Deploying mmWave Drone Base Stations for 5G Networks and Beyond," in IEEE International Conference on Communications Workshops, ICC, Kansas City, MO, USA, pp. 1-6, 2018.
[27] 3GPP, "TS 38.214 (V19.0): NR physical layer procedures for data (Release 19)," 2025.
[28] 3GPP, "TS 36.141: E-UTRA; Base Station conformance testing," 2025.
[29] 3GPP, "TS 36.101: E-UTRA; User Equipment radio transmission and reception," 2025.
[30] Y. Worku,C. Christodoulou, and M. Devetsikiotis, "UAV-Mounted Base Station Coverage and Trajectory Optimization Using LSTM-A2C with Attention," in Drones 2025, 9(11), 787, pp. 1-25.
[31] A. Abubakar, M. Mollel, O. Onireti, M. Ozturk, I. Ahmad et al. "Coverage and throughput analysis of an energy efficient UAV base station positioning scheme," Computer Networks, Vol 232, August 2023, 109854.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Spatio-Temporal and Context-Aware Dataset for Mobile Network Forecasting

Mathias Gerstner, Sebastian Fischer, Rudolf Hackenberg
Dept. Informatics and Mathematics, OTH Regensburg
Regensburg, Germany
e-mail: {mathias.gerstner,|sebastian.fischer,|rudolf.hackenberg}@oth-regensburg.de

***Abstract*—As autonomous driving becomes increasingly feasible, the German government has introduced a legal framework to enable the operation of vehicles with Level 4 automated driving functionality. A key requirement is the maintenance of a continuous connection between such vehicles and a remote technical supervisor. If this link is lost, the vehicle must transition into a safe state by bringing itself to a controlled stop. To mitigate the risk of connection loss, accurate forecasting of mobile network availability along routes is essential. This paper describes a spatio-temporal analysis of mobile network signal quality metrics along a fixed rural route based on 48 repeated measurement drives. The route was segmented into 320 spatial reference sections to enable consistent cross-trip comparison and stability assessment. A classification into usable and non-usable connectivity states reveals that more than half of the sections show pronounced trip-to-trip variability. The results show that mobile network quality is not only dependent on geographic location but is also influenced by environmental conditions. While global correlations between weather parameters and raw signal metrics remain weak, moderate relationships can be seen when analyzing the share of functionally usable connectivity in spatially unstable sections. A walk-forward forecasting model demonstrates that the inclusion of temperature in the model reduces prediction error compared to a purely historical baseline in 73.7% of evaluated trips. The findings show the importance of feature selection and also the limitations of linear modeling approaches. Rather than acting as deterministic predictors of signal strength, contextual parameters primarily modulate connectivity uncertainty in spatially unstable regions. These findings underscore the importance of contextual information and localized modeling to predict network availability for safety-critical systems, such as autonomous vehicles.**

Keywords*-mobile network stability; connectivity forecasting; vehicular communication; autonomous driving; signal quality indicators.

I. Introduction

This extended study builds upon a previous contribution [1], presented at The Second International Conference on Vehicular Systems (VEHICULAR ANALYTICS 2025). While the previous study focuses on the context-aware prediction of mobile networks, this study extends the data analysis and presents one additional forecasting model approach. It also investigates the spatial stability and trip-to-trip variability of mobile network quality along the selected rural route.

Even though recent forecasts are more cautious regarding the future proliferation of automated driving vehicles, it is evident that they will gradually become part of everyday life as technology evolves [2]. In this context, emerging automated driving functions and Vehicle-to-Everything (V2X) communication systems highlight the growing importance of robust and reliable vehicular connectivity [3].

Automated driving functions provided by future vehicles at Society of Automotive Engineers (SAE) Level 4 depend on reliable mobile network connectivity. While the core driving decisions must be performed locally inside the vehicle, the mobile network is considered a requirement to enable remote supervision and instructions in case of uncertainties. Therefore, it is important to assess the connectivity along a route before such an automated vehicle drives into an area without connection. While static cellular coverage maps provide a coarse indication of network availability, they fail to capture variations in signal quality that may occur at many spatial locations [4]. Although mobile networks are generally planned for wide geographical coverage, this must be distinguished from functional connectivity. A geographically covered location may still have insufficient signal quality under real driving conditions. Therefore, this study focuses on the stability and functional usability of connectivity rather than on theoretical coverage stated by mobile network operators. Variations in mobile network quality are particularly critical for safety-relevant automated driving applications, where transient connectivity loss or severe degradation may trigger fallback behaviors, such as bringing the vehicle to a safe stop [5].

Network connection quality is influenced by various dynamic factors, including the environment and time of day [6][7]. Open areas near base stations usually offer strong coverage, while remote or forested areas do not. To investigate this variability, 48 test drives along a predefined 64 km rural route, recording passive mobile network parameters and Global Navigation Satellite System (GNSS) data, were conducted. During post-processing, additional context, such as weather conditions and metadata, was integrated.

From a system-level perspective, such pre-trip connectivity information can be integrated into the route planning and data transmission schedules of automated vehicles. By identifying route segments with unstable or unreliable connectivity in advance, vehicles may adapt operational strategies, adjust communication-intensive data transmission, or initiate alternative routes.

The main extensions of this contribution are:

- A data cleaning and preprocessing methodology tailored to vehicular mobile network measurements.
- An approach for the functional classification of mobile network quality into usable and non-usable connectivity states.
- A spatial stability analysis revealing route segments with persistent connectivity, persistent coverage holes, and pronounced trip-to-trip variability.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

- An investigation of contextual influences on connectivity variability.

This study is based on measurements collected along a fixed route over the year 2025. The data is restricted to linear correlations between contextual and network parameters. Urban environments and larger datasets, beyond the 48 measurement drives used here, remain for future work and are expected to further improve prediction performance.

A. Motivation

The key motivation for this work comes from recent regulatory developments in Germany. According to legislation, automated vehicles classified under SAE Level 4 (high driving automation [8]) must maintain a permanent connection to a technical supervisor (§ 1e(2) no. 10 StVG [5]). This supervisor must receive real-time status information and can intervene, for example, by enforcing a safety stop or assisting in complex scenarios [5]. One example of such a scenario is near construction zones, where the vehicle may need to cross continuous lane markings, an action typically prohibited by internal rules. If the connection to the technical supervisor is lost, the vehicle has to bring itself to a controlled stop. Driving under Level 4 automation must not resume until a network connection is reestablished. According to the law, the technical supervisor has to be a natural person with specific knowledge about vehicular systems (§ 1d(3) no. 3 StVG [5]. The qualification of the supervisor must be demonstrated by a diploma, bachelor's, master's, or technical degree in mechanical engineering, automotive engineering, electrical engineering, or a related field (§ 14(1) no. 1 AFGBV [9].

Another motivation is the observation that static coverage representations are insufficient for characterizing mobile network availability in safety-critical automated driving scenarios. Even if average signal metrics indicate acceptable coverage at a given location, individual trips may experience severe degradation or complete loss of connectivity (shown in Section VI). Such trip-to-trip variability cannot be captured by static maps or historical averages, yet it directly impacts the operational reliability of Level 4 automated vehicles.

Unlike prior studies that focused on static coverage maps, short-term prediction, or urban areas, this work presents a long-term measurement campaign on a rural route and also integrates contextual factors (e.g., time of day, ambient temperature). To our knowledge, this is among the first studies to target pre-trip forecasting of Long-Term-Evolution (LTE)/ 5G quality for Level 4 driving scenarios.

B. Structure

The remainder of this paper is organized as follows. Section II discusses the state of the art and related work. Section III introduces the fundamentals of mobile networks and describes the target parameters. Section IV outlines the data collection followed by the data cleaning and preprocessing methodology in Section V. Next, Section VI presents a spatial stability and trip-to-trip variability analysis of mobile network connectivity. Section VII investigates the influence of contextual factors. Section VIII introduces an initial connectivity forecasting model. Finally, Sections IX and X discuss the results and limitations of the study. In the end, Section XI concludes the paper and outlines directions for future work.

II. Related Work

Several studies have addressed the prediction and analysis of mobile network quality, to some extent, in vehicular contexts. These efforts vary in scope, methodology, and focus, often targeting real-time diagnostics, network-side optimization, or static coverage mapping rather than pre-trip forecasting.

Torres et al. [10] propose a data-driven approach to forecast congestion in LTE networks using analytics and Machine Learning (ML) techniques. Their model is designed to support Self-Organizing Network (SON) optimization and real-time traffic management from a network operator perspective. However, it does not incorporate contextual factors such as weather or time of day and is not intended for vehicle-centric or route-specific pre-trip predictions.

Madariaga et al. [6] adopt a time-series perspective and demonstrate that meteorological conditions significantly affect mobile Quality of Service (QoS). By combining ML models with temperature, humidity, and rainfall data derived from crowdsourced Android measurements, they show that context-aware approaches outperform purely historical averages, particularly in dense urban environments. In a related study, they address the spatial aggregation of signal strength measurements using advanced interpolation techniques [11]. While these works provide important insights into context-aware modeling and spatial estimation, they are restricted to stationary urban settings and do not investigate long-term variability or repeated measurements along fixed routes.

Sahin and Sathya [12] present a lightweight classification approach that estimates mobile network quality in a given area using only Global Positioning System (GPS) data. Although computationally efficient, their method omits dynamic environmental influences such as weather or time and does not explicitly address predictive uncertainty. Similarly, Rahman et al. [13] employ reinforcement learning in combination with Unmanned Aerial Vehicles (UAVs) to detect mobile coverage holes in urban environments. Their work focuses on infrastructure-centric optimization and coverage assessment rather than vehicle-specific planning or pre-trip prediction.

Other studies have also investigated connectivity-aware navigation and planning. Hultman et al. [14] introduce a route planning approach based on static mobile signal coverage maps derived from publicly available data. While such methods are useful for coarse connectivity-aware routing, they lack the temporal and contextual sensitivity required to capture the trip-to-trip variability experienced during real-world driving.

More recently, large-scale measurement campaigns and open-access datasets have become available. Schippers et al. [7] introduced the DoNext dataset, a comprehensive multi-region 5G measurement campaign designed to support data-driven mobile network research. The dataset provides exten-

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

sive spatial and temporal coverage and enables the analysis of general network behavior across different contexts. However, its primary focus lies on general-purpose analysis rather than route-specific forecasting or the investigation of connectivity stability along repeated vehicular trajectories. Their results nevertheless confirm that signal quality at a given location can vary significantly over time, for example, due to congestion effects during peak hours.

Beyond dataset-centric contributions, recent work has emphasized methodological challenges in predictive mobile networking. Blasco et al. [4] discuss the concept of predictive QoS and highlight fundamental issues such as non-stationarity, feature selection, and evaluation bias in cellular networks. Similarly, Palaios et al. [15] provide an extensive analysis of machine learning-based QoS prediction for vehicular communication, focusing on robustness, data splitting strategies, and explainability. While these studies offer valuable guidance for model design and evaluation, they primarily treat variability as a modeling challenge rather than as an inherent spatial and temporal property of the environment.

Overall, prior work has addressed many important aspects of mobile network analysis, including data aggregation, spatial modeling, and urban QoS prediction. However, most existing approaches implicitly assume that mobile network conditions at a given location can be sufficiently described by average or aggregated metrics. Such assumptions are insufficient for autonomous driving applications, where transient connectivity loss or severe degradation can have immediate operational consequences.

The present study focuses on real-world measurements along a rural route and explicitly investigates the stability of mobile network connectivity across multiple trips. Rather than treating variability solely as noise or uncertainty to be mitigated by predictive models, it is additionally analyzed as a characteristic property of specific route segments. Furthermore, a functional distinction between usable and non-usable connectivity states is introduced that directly reflects the operational requirements of Level 4 autonomous driving and supports pre-trip connectivity assessment under uncertainty.

III. Background and Fundamentals

This section outlines the technical background necessary to understand the mobile network measurements analyzed in this study. In addition to a general overview of LTE and 5G principles, it explains the radio signal metrics that are particularly relevant for vehicular communication. The relationships between these Key Performance Indicators (KPIs) and their impact on the functional connectivity of autonomous driving systems are also described. Furthermore, the physical propagation effects caused by atmospheric conditions are briefly discussed to motivate the investigation of contextual influences in later sections.

A. Principles of Mobile Networks

Mobile communication networks consist of wireless links between User Equipments (UEs) and base stations, which connect to a core network. The transition from LTE to 5G enhances bandwidth, latency, and reliability [16], which are critical for autonomous driving applications [17]. LTE provides high data rates and reduced latency through techniques such as Multiple Input Multiple Output (MIMO), handovers, and advanced radio resource management [18]. 5G builds on LTE by adding capabilities such as beamforming and network slicing. It also specifies dedicated QoS mechanisms, including profiles for specific service classes.

5G can be deployed in Non-Standalone (NSA) or Standalone (SA) architectures. In NSA, 5G is operated with existing LTE infrastructure, whereas in SA deployments use a dedicated 5G core [19]. While German mobile network operators are introducing an increasing amount of 5G SA services, the public network still involves LTE and 5G NSA, depending on location and operator. As a result, robustness against environmental influences and coverage gaps is still strongly tied to LTE radio characteristics, particularly in rural areas where dense 5G deployments are uncommon.

5G operates across low-, mid-, and high-frequency bands, including millimeter waves [20]. Research has also investigated the fundamental performance limits of millimeter-wave systems for cooperative localization, underlining the relevance of advanced 5G [21]. Moreover, 5G offers even lower latency and increased capacity supporting safety-critical automotive applications [17]. In both LTE and 5G networks, signal quality plays a central role in determining the performance and reliability of a connection [22].

In the context of remote supervision for automated driving applications, the data volume of status messages like the Cooperative Awareness Message (CAM) is generally low (a size of 200 to 800 bytes and a frequency between 1 Hertz (Hz) and 10 Hz [23]). For this use case, connection robustness is more important than peak data throughput. This motivates a closer examination of signal stability rather than instantaneous performance metrics.

B. Mobile Network Signal Quality Metrics

Radio signal propagation in vehicular environments is influenced by various physical and atmospheric factors. Temperature and humidity affect air density, while fog, rain, and precipitation can introduce additional attenuation and scattering effects, particularly at higher carrier frequencies [24]. Although such effects are often secondary compared to geometric factors and network load, they can contribute to signal variability and are therefore considered as contextual parameters in this study.

Signal quality in LTE and 5G networks is assessed using a set of key radio frequency indicators. These metrics help to evaluate the reliability and performance of the wireless link between the UE and a cellular base station [25]. The most commonly used parameters are as follows:

- Received Signal Strength Indicator (RSSI) represents the total received power observed by the device within the channel bandwidth [22]. This value includes not only the desired signal but also background noise and interference

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

from neighboring cells. Measured in decibel-milliwatt (dBm), RSSI serves as an indicator of signal presence but lacks specificity for signal quality [26][27].

- Reference Signal Received Power (RSRP) quantifies the average power of reference signals transmitted by a serving cell [22]. Unlike RSSI, it excludes interference and noise, offering a more accurate measure of usable signal strength. RSRP is expressed in dBm and is used by UEs for tasks such as cell selection, handover decisions, and radio link monitoring [26].
- Reference Signal Received Quality (RSRQ) is the ratio of RSRP to RSSI, normalized by the number of resource blocks [22]. It reflects both the signal strength and the level of interference within the channel. It is particularly useful in scenarios with heavy network load or high interference levels, where a strong signal may still result in poor quality due to congestion. RSRQ is commonly measured in decibel (dB) [26].
- Signal-to-Interference-plus-Noise Ratio (SINR) indicates the quality of the wireless communication channel by comparing the power of the desired signal to the sum of interference and background noise [22]. It directly impacts achievable data rates and overall link performance. SINR is also measured in dB and is a parameter for evaluating network efficiency and optimizing system performance [26].

TABLE I. SIGNAL QUALITY METRICS FOR LTE AND 5G NETWORKS [22]

Parameter	Unit	Range	Quality Interpretation
RSSI	dBm	-120 to -30	> -65: Excellent -65 to -75: Good -75 to -85: Fair < -85: Poor
RSRP	dBm	-140 to -60	> -80: Excellent -80 to -90: Good -90 to -100: Fair < -100: Poor
RSRQ	dB	-20 to -3	> -10: Excellent -10 to -15: Good -15 to -20: Fair < -20: Poor
SINR	dB	-20 to +30	> 15: Excellent 10 to 15: Good 5 to 10: Fair < 5: Poor

Table I summarizes the value ranges for each of the four metrics based on industry standards and field experiments, including recommendations provided by the applied router's manufacturer [22].

In general, higher values of these metrics correspond to better network quality and more stable connectivity. However, for autonomous driving applications, the relevance of these metrics lies not only in their absolute values but also in their temporal stability and their ability to indicate whether a connection is functionally usable.

IV. DATASET

This section describes the measurement campaign, the data acquisition setup, and the resulting dataset used for the subsequent analysis.

A. Measurement Campaign

The dataset used in this study was collected during a series of 48 real-world measurement drives along a 64 km segment of the road B16 in southern Germany. The route was selected to cover different environmental conditions. It includes suburban areas near the cities of Ingolstadt and Regensburg, smaller villages, extended rural sections, and forested segments. In addition, the route contains various topological properties such as small hills, bridges, and roadside structures, which are known to influence mobile radio propagation [28]. This diversity was expected to induce a broad variance in mobile network conditions and connectivity behavior, making the route suitable for studying spatial variability and stability of network quality. The measurement campaign extended over 10 months (March to December 2025) to capture a broad range of temporal, seasonal, and environmental variability. In total, 92,210 measurement points along the route were collected.

For the data acquisition, a custom-built mobile measurement unit was used. An overview of the setup, including its hardware components and their interconnections, is illustrated in Figure 1.

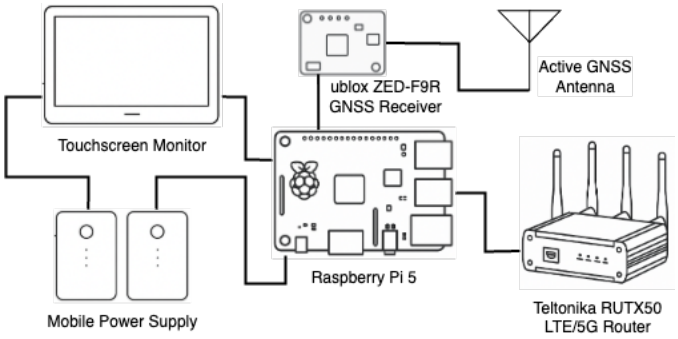


Figure 1. Setup of the custom-built mobile measurement unit

In detail, the setup consists of the following components:

- Raspberry Pi: As the central processing unit, a Raspberry Pi 5 with 8 GB of Random-Access Memory (RAM) is used. It is running Raspberry Pi OS in the Graphical User Interface (GUI) version. For data processing, Python 3.12 is used on this device.
- Router: For the connection to the mobile network, a Teltonika RUTX50 LTE/5G router is used. It is connected to the Raspberry Pi via a Local Area Network (LAN) cable. The router is capable of measuring the passive mobile network parameters needed for the study. Those parameters are transferred to the Raspberry Pi over Simple Network Management Protocol (SNMP).

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

The router does not use dedicated 5G QoS profiles for this measurement campaign.

- GNSS receiver: A u-blox ZED-F9R GNSS receiver is included in the setup for determining a precise geographic location while driving along the route. The data is transferred to the Raspberry Pi via a serial connection using a Universal Serial Bus (USB) port. The receiver uses the National Marine Electronics Association (NMEA) standard commonly used in navigation devices. These messages are in a human-readable format and contain not only the location but also information about time, date, current speed, heading direction, visible satellites, and precision of location determination like Position Dilution of Precision (PDOP).
- GNSS antenna: The active antenna u-blox ANN-MB-00-00 is connected to the receiver. During measurement drives, it is placed in a fixed position under the vehicle's windshield.
- Monitor: A 7-inch display from Waveshare is built into the setup. It shows the current state of the GNSS receiver and the LTE/5G router, as well as the currently measured values. With its capacitive touchscreen, the measurements can be started and stopped via two buttons. The screen is connected to the Raspberry Pi via High-Definition Multimedia Interface (HDMI).
- Power supply: Two rechargeable mobile batteries are included in the setup to provide the power needed. One battery powers the Raspberry Pi while the other powers the display. Both batteries provide 3 Ampere (A) at 5 Volt (V). The router is powered by the vehicle's electrical system to ensure stable power availability for this critical device.

The collected mobile network parameters, as presented in Section III-B were logged continuously throughout each drive. In addition, the GNSS receiver provided geographical coordinates and motion-related parameters, enabling precise spatial alignment of network measurements along the route.

B. Dataset Description

The initial dataset comprises time-stamped mobile network measurements, geographic location information, and motion-related parameters. Recorded network metrics include RSRP, RSRQ, RSSI, and SINR, as well as cell identifiers and connection state information. Spatial context is provided by latitude, longitude, altitude, and GNSS-derived quality indicators such as PDOP, Horizontal Dilution of Precision (HDOP), and number of visible satellites. Vehicle speed and heading were recorded to capture the motion context of each measurement.

In total, 92,210 measurement points were collected with a temporal resolution of approximately 2–3 seconds. Not all measurements are located on the target route, as recordings also include short segments at the beginning and end of each drive. These off-route measurements are excluded during preprocessing, as described in Section V.

C. Relationship Between Signal Metrics and Data Rate

To empirically motivate the relevance of the selected passive mobile network parameters, we analyze an independent open-access dataset collected in the Salzburg region, which provides both passive radio signal metrics and measured data rates [29].

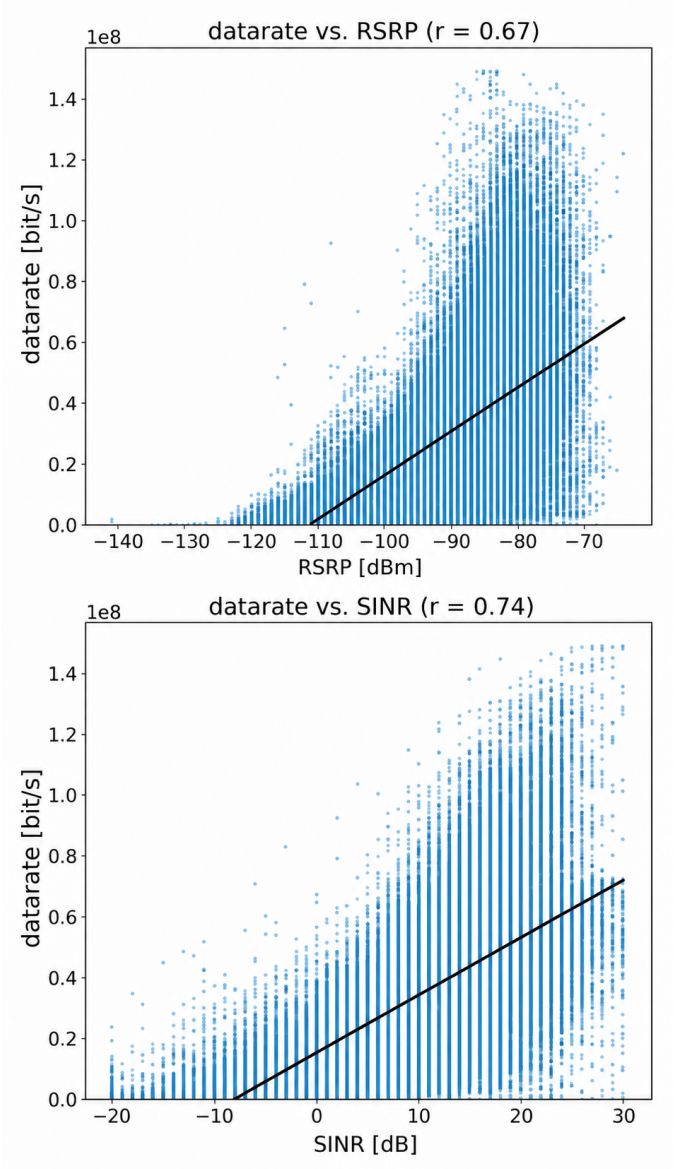


Figure 2. Relationship between measured data rate and passive mobile network parameters based on an open-access dataset [29]. Linear correlation coefficients are indicated for each parameter.

Figure 2 illustrates the relationship between data rate and RSRP and SINR. The corresponding Pearson correlation coefficients are $R = 0.67$ for RSRP and $R = 0.74$ for SINR, indicating a positive linear association between signal quality and achievable data rate. The correlation R-value of the two other metrics is $R = 0.58$ for RSRQ and $R = 0.58$ for RSSI as well. The Pearson correlation coefficient (R-value) quantifies the strength and direction of a linear relationship between two variables on a scale from -1 (perfect negative correlation) to

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$+1$ (perfect positive correlation), with values closer to zero indicating weaker linear dependence.

Across all parameters, a clear positive correlation trend can be observed, with SINR exhibiting the strongest linear correlation. These results are consistent with prior studies on QoS prediction in cellular networks and confirm that passive signal metrics provide meaningful information about the achievable data rate.

V. Data Cleaning and Preprocessing

While the dataset provides rich spatial and temporal information, the raw measurements include missing values, default placeholders, and off-route samples. The following section describes the data cleaning and preprocessing pipeline applied prior to analysis. All steps were implemented in Python using the libraries pandas and numpy and are designed to preserve variability while removing invalid measurements.

The first preprocessing step was the validation of the GNSS data. A total of 339 measurements (0.37%) contained invalid location information, either due to missing coordinates or fallback values with latitude or longitude equal to zero. Notably, 215 of the zero-valued GNSS samples originated from a single measurement run, where the GNSS receiver did not yet obtain a valid fix at the beginning of the drive. As these samples occurred in extended consecutive blocks, no spatial interpolation was applied. Instead, all measurements without valid GNSS coordinates were excluded from spatial analysis. This issue could be avoided in future campaigns by explicitly verifying a valid GNSS fix prior to initiating measurements.

The next cleaning step focused on the identification of placeholders used by the router when no network connectivity is available at all. Radio quality indicators were first analyzed without applying any filtering or correction. By comparing the observed minimum, maximum, and average values of RSSI, RSRP, RSRQ, and SINR with the plausible ranges defined in Table I, fallback values were identified. Specifically, the values -32768 (corresponding to the minimum signed 16-bit integer), -255, and 0 were found. Based on this observation, explicit placeholder values (-32768 and -255) were consistently replaced by Not a Number (NaN) for all radio quality indicators. This served as the basis for all subsequent cleaning steps. An additional column was included in the dataset that marks the locations at which placeholders were replaced.

Handling zero-valued SINR measurements required special attention. Because it lies within the normal range of SINR, a value of 0 can also represent a valid signal condition. But it was also used by the device as a fallback value when no SINR calculation was possible. To distinguish between valid and invalid zero-values, the occurrence of SINR values equal to zero and anomalies in other radio KPIs were analyzed. Many SINR samples with a value of $0\,\mathrm{dB}$ were found to coincide with missing or implausible values in the other radio quality indicators. These samples were therefore classified as invalid values and excluded from the cleaned SINR distribution. Figures 3 and 4 illustrate the SINR value distributions before and after removing those invalid fallback values. The comparison shows the impact of poorly chosen device fallbacks on the observed SINR statistics.

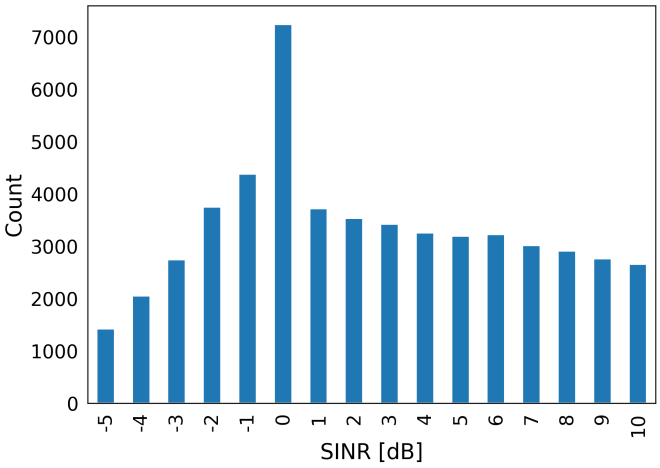


Figure 3. Frequency distribution of SINR values after removing explicit sentinel values. Zero-valued SINR samples include both physical measurements and implicit device fallback values.

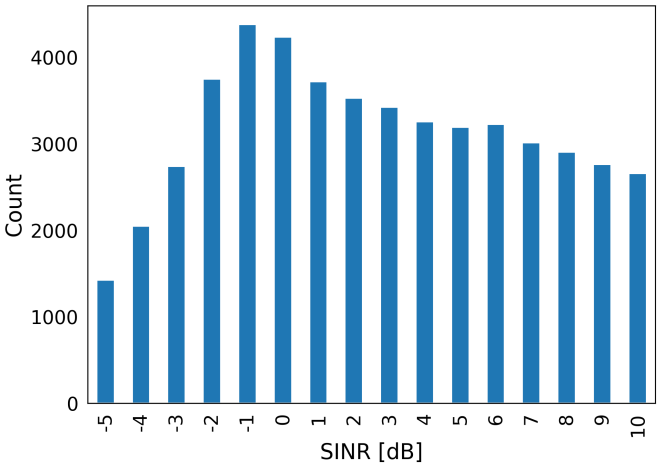


Figure 4. Frequency distribution of SINR values after removing both explicit sentinel values and implicit zero-valued fallback measurements.

After removing the placeholders, all mobile network KPIs follow physically plausible ranges and statistically consistent distributions. Missing radio measurements were not treated as noise but explicitly encoded as a separate indicator variable. This allows the prediction models to distinguish between absent connectivity and weak but valid signal conditions. There are no indications of further systematic invalid values. Figure 5 shows an example of the distribution of the frequency of RSRP values. The other KPIs follow a similar pattern.

Table II summarizes basic descriptive statistics of the cleaned measurements. All KPIs exhibit value ranges and central tendencies consistent with expected cellular radio conditions in vehicular environments, indicating that no further systematic invalid values remain after the applied filtering steps.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

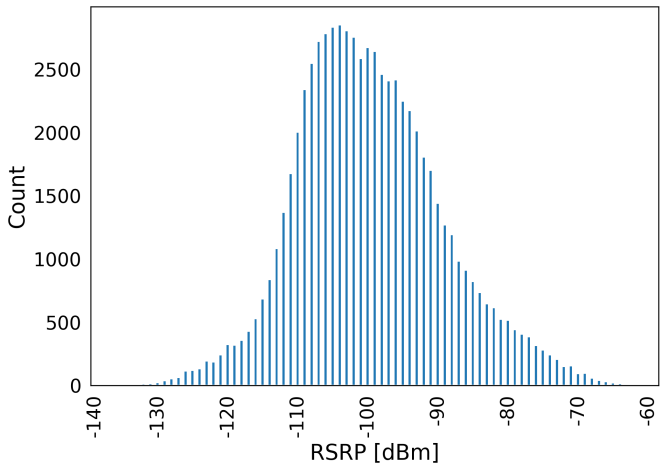


Figure 5. Frequency distribution of RSRP after cleaning the dataset.

TABLE II. SUMMARY STATISTICS OF CLEANED RADIO KPI MEASUREMENTS AFTER REMOVAL OF EXPLICIT AND IMPLICIT PLACEHOLDER VALUES.

KPI	Count	Min	Max	Mean	Median
RSRP [dBm]	72 606	-140	-59	-99.6	-100
RSRQ [dB]	72 627	-20	-3	-13.7	-13
RSSI [dBm]	72 594	-115	-26	-65.8	-67
SINR [dB]	72 538	-20	30	4.7	4

In addition to cleaning, the dataset was enriched with contextual information describing the environment. Base station proximity was incorporated by computing the distance between each measurement point and the serving base station. The base station locations were obtained from the open-source platform CellMapper [30], and distances were computed using haversine-based spatial calculations. Also, historical weather data were added to each measurement point using the Open-Meteo Application Programming Interface (API) [31]. The weather attributes (e.g., temperature, wind speed, precipitation, ground fog) were matched based on timestamp and geographical coordinates, allowing for the inclusion of environmental context in the analysis process.

As a result of this process, a spatiotemporally tagged dataset that combines passive network parameters, location, and environmental conditions was created. This dataset forms the basis for the subsequent spatial stability and variability analysis.

VI. SPATIAL NETWORK VARIABILITY

To gain insights into the structure and variability of the collected dataset, an Exploratory Data Analysis (EDA) was conducted. This analysis aims to find spatial and contextual patterns in the signal measurements and to identify the most relevant features for subsequent modeling tasks.

A. Route Segmentation

As shown in Figure 6, the measurement route was segmented into Reference Points (RPs) along the road. This resulted in a total of 320 RPs covering the test route. The segmentation enables consistent spatial aggregation by grouping measurements recorded at the same physical location across repeated drives. For each RP, the mean, median, minimum, and maximum values of the target parameters were calculated. This helps to mitigate the influence of measurement noise and improves the robustness of comparisons between different measurement runs.

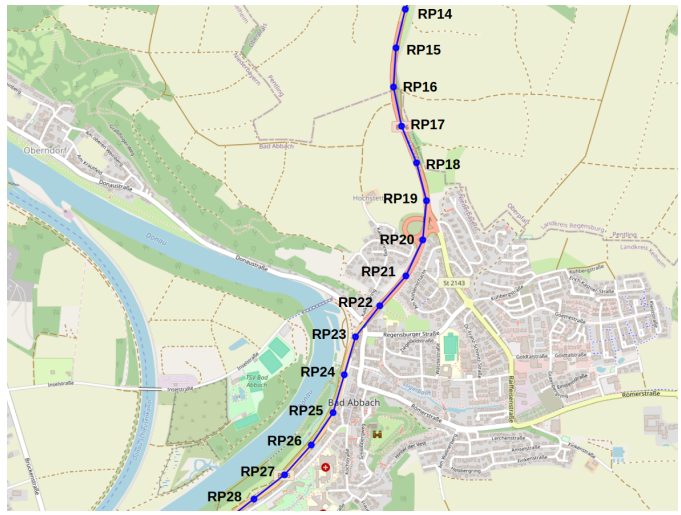


Figure 6. Extract of the route clustered into reference points with a distance of 200 m.

To limit the analysis to measurements recorded on the predefined test route, each sample was assigned to its nearest RP. The RPs were placed on the road centerline with a fixed spacing of 200 m. Consequently, measurements located less than half of the spacing away from their nearest RP originate from the interior of the route. Measurements that exceed the 100 m distance to a RP are likely associated with boundary regions or could also be off-route segments.

TABLE III. VALIDATION OF DISTANCE CUTOFF VALUES BASED ON REFERENCE POINT EXCEEDANCES.

Cutoff [m]	Reference Point ID	Count
100	1	13 932
	11	1
	19	1
	97	1
	123	1
	152	2
	319	1
	320	1 883
101	1	13 931
	152	2
	320	1 882
102	1	13 930
	320	1 880

To determine an appropriate distance threshold, increasing

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

cutoff values were tested, and the occurrence of threshold exceedances across the RPs was inspected. Table III summarizes the number of samples that exceed different distance thresholds and their associated RP identifiers. For cutoff values of 100 m and 101 m, a small number of on-route measurements were still observed. This could plausibly be due to GNSS jitter. At a cutoff of 102 m, all exceedances occurred only at the first and last RPs, which are at the beginning and the end of the route. No interior RPs exhibited distances exceeding this threshold. Based on this observation, a distance cutoff of 102 m was selected as the smallest value that reliably excludes off-route measurements while preserving all valid interior route samples.

B. Variability Across Repeated Drives

After associating the measurements with their nearest RPs, the next step is to analyze how mobile network connectivity differs across repeated drives. While the vehicle's location is a dominant factor influencing signal quality, an initial analysis confirms that connectivity conditions at the same location can vary substantially from trip to trip, as also suggested by prior work discussed in Section II.

Figure 7 illustrates the relationship between the mean probability of usable connectivity and its trip-to-trip variability for each RP.

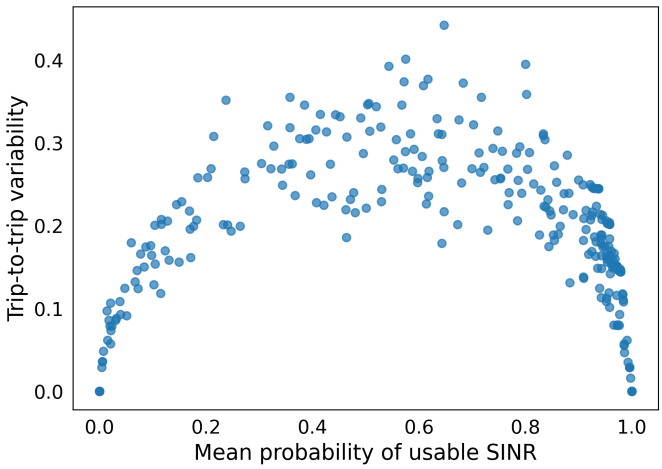


Figure 7. Trip-to-trip variability of functional connectivity based on the SINR. Every point shows one reference point of the route.

For this plot, connectivity was classified as usable if the measured SINR was available and higher than 0. Accordingly, the mean value shows the measurements with functionally usable connectivity across the drives. The standard deviation quantifies how this functional state fluctuates between trips.

Route sections with consistently good connectivity (x-axis close to 1) and sections with persistently poor connectivity (x-axis close to 0) showed very low variability (y-axis), indicating stable connectivity conditions across drives. In contrast, sections with intermediate mean values consistently show trip-to-trip variability. At these locations, the same geographic position on the route alternates between usable and non-usable connectivity depending on the individual drive.

The resulting inverted U-shaped distribution shows that connectivity cannot be sufficiently characterized by static coverage maps or average signal metrics alone [4][15]. The same analysis was conducted for the RSRP, showing the same distribution. These findings underline the necessity of incorporating additional contextual information when assessing mobile network conditions before starting a trip.

C. Stability Classification

Based on the per-trip aggregation of functional connectivity, each route section was classified according to its stability across repeated drives. Four distinct stability classes were defined.

- *Stable good* sections show consistently usable connectivity across almost all trips. These segments represent areas where a good mobile connection can be expected, independent of contextual variations.
- *Stable hole* sections are characterized by persistently non-usable connectivity. Across repeated drives, these segments rarely reach sufficient signal quality for data transmission, indicating stable coverage gaps.
- *Unstable* sections are those in which functional connectivity alternates between usable and non-usable states across different measurement drives. As a result, the same physical location may support reliable data transmission during one drive while being unusable during another. This behavior cannot be captured by static coverage representations or average signal metrics.
- *Mixed* sections denote transitional segments that do not exhibit pronounced trip-to-trip state changes, yet cannot be classified as consistently usable or consistently non-usable. These sections reflect moderate variability without clear state transitions.

TABLE IV. CLASSIFICATION OF ROUTE SECTIONS ACCORDING TO THEIR FUNCTIONAL CONNECTIVITY STABILITY ACROSS REPEATED DRIVES.

Stability Class	Number of Sections	Share [%]
Stable good	25	7.8
Stable hole	102	31.9
Unstable	111	34.7
Mixed	82	25.6
Total	320	100.0

Table IV shows the distribution of those classes along the route. Notably, only a small fraction of the route sections (7.8%) exhibit consistently good connectivity, whereas about one third (31.9%) are classified as stable holes. The remaining 60.3% of the route cannot be uniquely characterized by a single functional connectivity state.

Figure 8 illustrates the spatial distribution of the stability classes along a part of the route. Stable hole sections mostly appear in rural and near forested areas, whereas stable good connectivity is mainly observed near villages. Unstable sections frequently occur in clusters between these extremes. This finding indicates that functional connectivity uncertainty is not

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

randomly distributed but is tied to specific environmental and infrastructural characteristics.

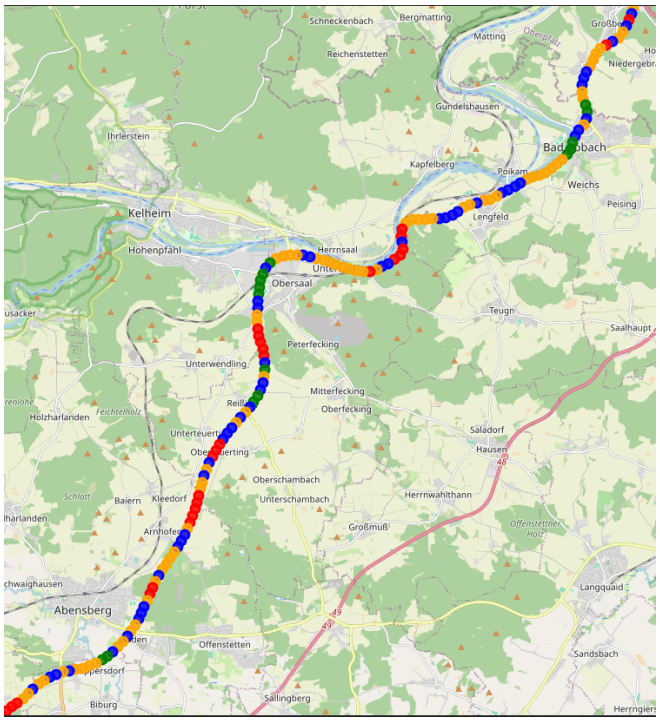


Figure 8. Spatial distribution of route sections classified by functional connectivity stability across repeated drives. The colors show the four classes: stable good (green), stable hole (red), mixed (blue) and unstable (orange).

The stability analysis shows that a significant part of the route cannot be characterized by a single functional connectivity state. Instead, several route sections exhibit pronounced trip-to-trip variability, alternating between usable and non-usable connectivity despite identical spatial locations.

This observation underlines the research question of whether such variability can be partially attributed to contextual factors that differ between individual drives. Environmental conditions, temporal aspects, and geometric relationships to the cellular infrastructure may influence the instantaneous radio conditions and thereby affect the functional connectivity state observed along a given route section.

In the following section, the role of contextual parameters and their relationship with the observed connectivity variability is analyzed. The goal is to assess whether contextual information provides systematic indications that can support pre-trip connectivity prediction under uncertainty.

VII. Influences of Contextual Factors

To improve the prediction of mobile network KPIs, particularly in areas with weak or inconsistent coverage, it is important to account for environmental and temporal influences. Contextual parameters are assumed to influence radio propagation and therefore the network performance [24][28].

The previous study [1], based on 38 measurement drives, investigated correlations between contextual features and the four passive radio quality indicators RSRP, RSRQ, RSSI, and SINR. The analyzed context variables include weather conditions, temporal features, distance to the serving base station, GNSS precision metrics (e.g., PDOP and the number of visible satellites), and vehicle speed. Figure 9 summarizes the correlations reported in that study.

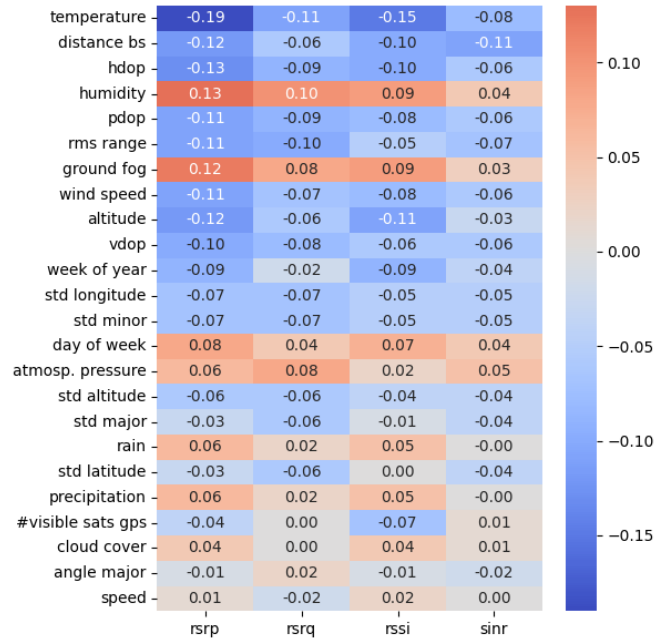


Figure 9. Heatmap showing the correlations between environmental parameters and received signal quality metrics. Based on 38 test drives of the previous study [1].

These first results indicate weak to moderate correlations for the parameters. In particular, ambient temperature showed a negative correlation with RSRP and RSSI, while humidity and ground fog exhibited small positive correlations with multiple signal metrics. Temporal features such as the day of the week and the week of the year also showed minor correlations, potentially reflecting small time-dependent network load. Vehicle speed, in contrast, showed no relevant influence on radio signal quality.

For the pre-trip connectivity forecast assessment in autonomous driving scenarios, it is essential to verify whether such contextual parameters provide explanatory or predictive value beyond spatial location. Therefore, a similar correlation analysis with an extended dataset of 48 measurement drives was conducted. Because weather parameters appeared to be the most promising, the analysis focused on them. As a first step, correlations between weather and passive mobile network indicators were computed over the entire dataset. Missing radio measurements (NaN due to no signal at all) were replaced by worst-case reception values to preserve the distinction between weak signal and missing connectivity. Both Pearson correlation coefficients, which capture linear relationships, and Spearman correlations, which capture monotonic dependencies, were evaluated.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Across the complete route, all correlations between weather parameters and radio KPIs remain close to zero. This observation is consistent for temperature, relative humidity, precipitation, wind speed, and surface pressure. Table V reports the Spearman correlation coefficients for the global dataset.

TABLE V. SPEARMAN CORRELATION BETWEEN WEATHER PARAMETERS AND RADIO KPIS (ENTIRE DATASET, NANS REPLACED BY WORST-CASE VALUES).

Context	RSRP	RSRQ	RSSI	SINR
Temperature	-0.06	-0.00	-0.07	0.00
Relative humidity	0.04	-0.00	0.04	-0.02
Precipitation	0.02	0.01	0.02	-0.01
Wind speed	-0.02	0.00	-0.02	0.01
Surface pressure	0.03	0.02	0.04	0.04

This result indicates that environmental context alone does not explain the dominant spatial structure of mobile network quality. Instead, infrastructure-related factors and geographic location remain the primary determinants on a global scale.

To isolate potentially context-sensitive behavior, the analysis was repeated for route sections classified as *unstable* or *mixed*, as described in Section VI-C. These sections showed high trip-to-trip variability and therefore represent critical candidates for contextual influence. In these selected sections, correlations increase slightly in magnitude compared to the global analysis but remain weak overall. This suggests that weather-related context does not directly determine instantaneous signal strength.

As the next step, the analysis was shifted from numeric signal values to a functional connectivity perspective. Radio KPIs are mapped to binary usability indicators based on the thresholds for poor connectivity as described in Subsection III-B. If the KPI is above the respective threshold, it is defined as usable; otherwise, it is defined as not usable. This transformation reflects whether mobile connectivity is sufficient for data transmission rather than absolute signal strength. The analysis was again restricted to unstable and mixed route sections. For each measurement drive, the share of usable signal conditions was computed and correlated with aggregated contextual parameters.

When evaluating the share of functional connectivity instead of raw signal values, moderate correlations with contextual parameters can be observed. In particular, higher ambient temperatures are associated with a reduced probability of usable connectivity. Additionally, higher relative humidity and precipitation show a small positive association with usable signal conditions. Table VI summarizes these Spearman correlations between the weather parameters and the share of usable connectivity states in unstable and mixed route sections.

The effects are clearer for RSRP and RSSI than for RSRQ and SINR. The observed correlations suggest that environmental factors affect the likelihood of functional connectivity, particularly in spatially unstable regions. However, they are also insufficient to predict connectivity state transitions deterministically on their own. Contextual information provides complementary value by quantifying uncertainty tendencies. This motivates the integration of contextual parameters into higher-level probabilistic models rather than relying on direct signal prediction, as shown in Section VIII-C.

TABLE VI. SPEARMAN CORRELATION BETWEEN CONTEXTUAL PARAMETERS AND THE SHARE OF USABLE SIGNAL CONDITIONS IN UNSTABLE AND MIXED ROUTE SECTIONS.

Context	RSRP	RSRQ	RSSI	SINR
Temperature mean	**-0.53**	-0.10	**-0.62**	-0.01
Humidity mean	**0.32**	0.03	**0.35**	-0.21
Wind speed mean	-0.29	-0.08	-0.27	-0.06
Surface pressure	0.24	0.16	0.16	-0.06
Rain sum	0.12	0.21	0.04	-0.15

A. Correlation with Ambient Temperature

To further investigate the influence of temperature, Figure 10 illustrates the average RSRP values across different temperature bins, separated by connection type. The diagram originates from the previous study with a total of 38 measurement drives. A gradual degradation in signal strength with increasing ambient temperature can be observed. LTE connections appear to be more sensitive to temperature increases than 5G-NSA, which seems to be a little more robust across the temperature range.

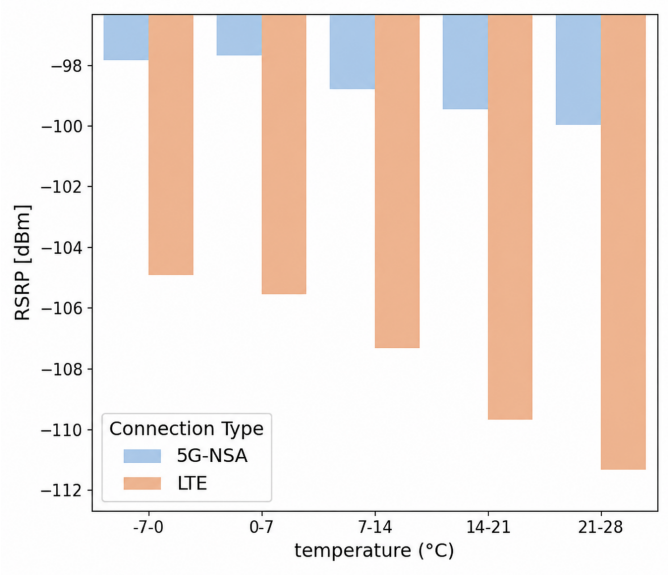


Figure 10. Influence of the ambient temperature on the RSRP value separated into 5G-NSA and LTE measurements. Lower RSRP corresponds with lower network availability.

To show another perspective, Figure 11 evaluates temperature effects from a functional connectivity standpoint, as also used in Section VII. In this case, all 48 measurement drives were included. Instead of analyzing absolute signal values, the figure shows the share of functionally usable RSRP measurements per trip, aggregated over repeated drives and grouped by mean ambient temperature.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

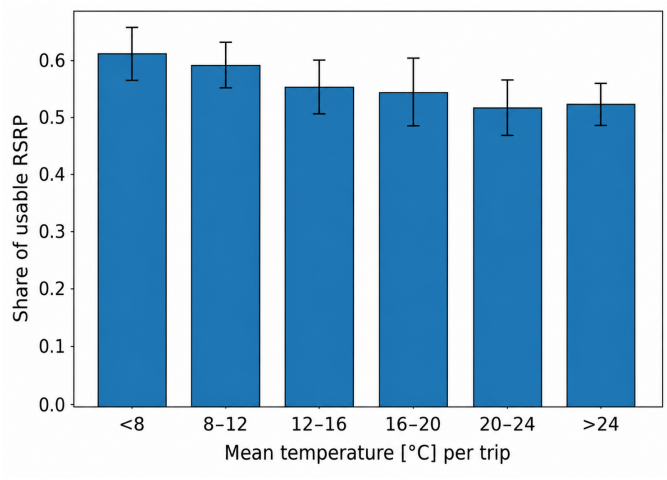


Figure 11. Influence of the ambient temperature on the share of functionally usable RSRP values aggregated per trip. The black lines indicate the value range of the respective temperature bin.

A gradual decrease in the probability of usable RSRP can be observed with increasing temperature up to approximately 24°C. This trend is consistent with the correlation analysis presented in Section VII, which revealed moderate relationships between temperature and functional connectivity metrics. The highest temperature bin (>24°C) contains only three measurement drives and is therefore not considered statistically representative.

The corresponding statistics across all KPIs further support this observation. While the average share of usable connectivity remains above 0.45 for all metrics, the minimum observed shares indicate that higher temperatures coincide with more frequent connectivity degradation events, particularly for RSRP and RSSI. In contrast, SINR shows a smaller sensitivity, suggesting that interference and noise conditions partially compensate for temperature-related signal attenuation.

VIII. Connectivity Forecasting Model

The discussed correlations provide a foundation for feature selection and motivate the inclusion of contextual parameters into a predictive model, which is described in this section. The Mean Absolute Error (MAE) was used to assess the influence of contextual features on the target KPIs. MAE is chosen for its interpretability and robustness to outliers. It provides a direct average of absolute errors in the same units as the target variable, making it suitable for this purpose. The MAE is defined as:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{1}$$

where y_i denotes the true value, $\hat{y}_i$ the predicted value, and n the number of samples [32].

As mentioned in Section III-B, the target KPIs are expressed in dB, respectively dBm, which are both logarithmically scaled. Therefore, even small changes in the MAE can have a significant impact on the effective signal quality [26].

A. Context Parameter Scaling

To investigate how external context parameters affect mobile network signal quality, a modeling approach that adjusts the section-wise signal predictions based on environmental deviations was developed. The baseline model predicts the expected target parameters for every RP along the route using only the mean value from historical measurements. This section explores whether incorporating individual context parameters can improve the prediction by reducing the MAE.

For each context parameter, a linear model was trained using the deviation between the parameter's current value and its historical section average. This deviation was multiplied by a scale factor to model the resulting shift in signal behavior. The prediction was then computed by adding this adjustment to the baseline section average. To identify the optimal scale factor, values ranging from –5.0 to 15.0 in increments of 0.1 for each context parameter were tested. The scale factor that minimized the MAE on the test measurement was selected. The goal was to quantify how strongly each context parameter contributes to prediction accuracy and whether any of them provided consistent improvements over the baseline.

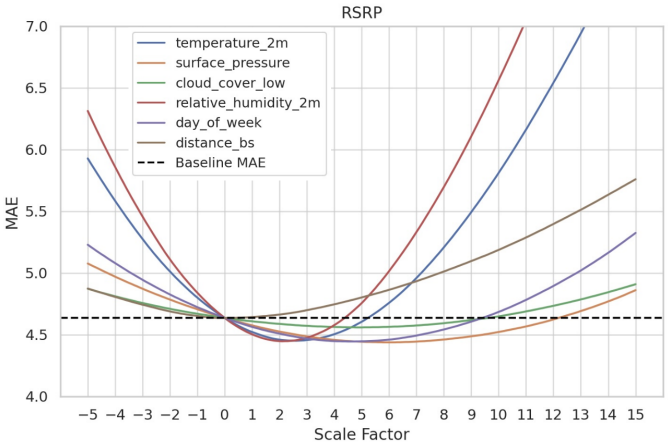


Figure 12. Influence of the context parameters with different scale factors on the MAE compared to the baseline MAE without context

Figure 12 shows the results of the scale factor optimization for the signal parameter RSRP. Each curve in the plot represents the MAE progression as a function of the scale factor applied to the respective context parameter. In all cases, the MAE initially decreases with increasing scale factor, reaches a minimum, and then increases again. The findings show that most context parameters can improve prediction accuracy compared to the baseline. However, the improvement varies between the parameters. The distance to the base station could not improve the MAE. This behavior was expected because the base station for every RP is the same for most drives, and therefore, the distance is a fixed parameter.

B. Combining Context Parameters

The aim of this subsection is to investigate whether the prediction accuracy of mobile network parameters could be further improved by simultaneously incorporating multiple

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

context parameters. Building on the results of the individual analysis, it was examined whether combining two context parameters, each with its own scale factor, could reduce the MAE even further. Therefore, all possible pairs of the five most relevant context parameters (temperature, surface pressure, relative humidity, ground fog, and day of week) were systematically evaluated. For each pair, a linear model was trained using the deviation of each context value from its corresponding section average. A grid search over scale factor combinations in the range -5 to 15 (in steps of 0.1) was conducted. The goal was to identify the combination that minimized the MAE for a separate test measurement.

However, the results in Table VII show that none of the tested parameter combinations yielded significantly better performance than the single context parameter. For RSRP and SINR, the MAE even increased when using two parameters instead of one.

TABLE VII. MAE COMPARISON OF BASELINE, BEST SINGLE FEATURE, AND BEST FEATURE COMBINATION

Metric	Baseline	Best Feature	Best Combination
RSSI	3.684	3.556 Surface Pressure	3.548 Surface Pressure Humidity
RSRP	4.635	4.437 Surface Pressure	4.440 Surface Pressure Humidity
RSRQ	2.158	2.065 Surface Pressure	2.063 Surface Pressure Ground Fog
SINR	3.041	3.017 Temperature	3.024 Temperature Surface Pressure

This finding suggests that the influence of context parameters on mobile network measurements cannot be adequately described by a simple linear superposition. Another explanation may be the multicollinearity among weather variables, which adds redundancy and inflates linear model variance. For example, temperature and relative humidity are often negatively correlated [24]. It is likely that complex interactions or nonlinear relationships exist between the context features that are not captured by the linear model.

C. Pre-Trip Temperature-Conditioned Forecasting

Since the previous subsections used 38 measurement drives, another approach was conducted using data from 48 drives. The extended experiment focuses on the influence of environmental temperature on the RSRP because this combination is the most examined in Section VII. Under real-world conditions, only historical measurements are available before the start of a trip. Therefore, a walk-forward approach was implemented in which each measurement campaign was predicted using only past trips. A minimum of ten historical trips was required before a forecast was generated.

The baseline model in this experiment predicts the expected signal level for each route section using only the historical section mean value. Building on the correlation analysis in Section VII, temperature was taken as a global adjustment factor. Instead of performing scale-factor optimization, a linear regression [33] was trained on historical averages per trip:

$$\bar{y}_{trip} = \alpha + \gamma \bar{T}_{trip} \tag{2}$$

where $\bar{y}_{trip}$ denotes the mean signal level of a historical trip, and $\bar{T}_{trip}$ denotes its corresponding mean temperature. The resulting temperature shift applied to a future trip is then defined as:

$$\Delta y = \gamma(\bar{T}_{trip} - \bar{T}) \tag{3}$$

where $\bar{T}$ is the historical mean trip temperature. The final prediction becomes:

$$\hat{y}_s^{\text{temp}} = \mu_s + \Delta y \tag{4}$$

where μ_s denotes the historical mean signal level of section s. The results of this simple model using the extended set of measurements are as follows. For the RSRP, the baseline achieved the following MAE:

$$\text{MAE}_{\text{baseline}} = 4.443 \pm 0.233 \text{ dB}$$

Incorporating the global temperature adjustment reduced the error to:

$$\text{MAE}_{\text{temp}} = 4.415 \pm 0.211 \text{ dB}$$

corresponding to an average improvement of:

$$\Delta\text{MAE} = 0.028 \text{ dB}$$

The absolute improvement is small, but it must be interpreted in the context of logarithmic signal scaling. The model, taking into account the ambient temperature, reached a lower error in 73.7% of all trips. This shows that temperature can help predict signal metrics, but the influences seem to be more complex than a simple linear model can represent. The estimated temperature sensitivity was:

$$\gamma = -0.109 \pm 0.021 \text{ dB/°C}$$

indicating that higher temperatures are associated with systematically reduced expected RSRP levels. The uncertainty term ± 0.021 dB/°C is the standard error of the estimated coefficient.

An additional analysis examined only route sections with strong trip-to-trip variability. Sections exhibiting a standard deviation of 6 dB across historical trips were classified as variable. In total, 37 of the 320 sections met this criterion. When the forecasting model was restricted only to these variable sections, the temperature adjustment produced an absolute improvement of:

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

$$\Delta \mathrm{MAE}_{\mathrm{variable}} = 0.071 \text{ dB}$$

This confirms that contextual information is particularly relevant in dynamically fluctuating route segments, whereas permanently stable sections are already well captured by the historical mean.

IX. Discussion

The study demonstrates that location alone is insufficient to characterize mobile network availability along a route. The stability analysis in Section VI revealed that more than half of the sections on the specific route cannot be described by a functional connectivity state (stable good or stable hole). In particular, sections classified as unstable show high trip-to-trip variability, indicating that connectivity uncertainty is a property of certain spatial environments.

The forecasting experiments show that including contextual information yields moderate but consistent reductions in prediction error. Although the MAE improvements are small in magnitude, the win-rate across trips suggests that contextual conditioning captures variability rather than noise. The results further indicate that contextual effects are not globally uniform but are concentrated in spatially unstable sections. This suggests that context influences mobile network uncertainty rather than shifting signal levels in a linear way.

The weak global correlations between weather parameters and radio KPIs contrast with the moderate correlations observed when analyzing functional usability in unstable sections. This indicates that contextual variables do not directly determine instantaneous signal metrics but influence the likelihood of a route section's current state between usable and non-usable. Rather than treating variability as noise, the findings suggest that spatial instability zones represent structurally sensitive regions where environmental conditions influence existing propagation weaknesses.

The limited improvements achieved by the linear integration of multiple context parameters indicate that the relationship between environmental factors and radio propagation may be nonlinear. Multicollinearity among weather variables further limits interpretability within linear models. The findings support the need for nonlinear modeling approaches capable of capturing conditional dependencies and interaction effects, particularly within spatially unstable sections.

A. Implications for Vehicle Connectivity

The findings of the analysis have different implications for the operation of modern (automated) vehicles. Pre-trip forecasts of route-section-wise connectivity quality could be used to reschedule the path between start and finish for cars for which a continuous network connection is required (SAE Level 4). In addition, such forecasts can enable strategic scheduling of communication-intensive but non-safety-critical data transfers. For example, the upload of high-resolution sensor logs, map updates, or cloud-based analytics can be scheduled accordingly. By allocating large data packets to route segments with predicted stable and high-quality connectivity, vehicles can reduce the probability of transmission interruption, preserve bandwidth for safety-relevant remote supervision, and minimize fallback maneuvers triggered by temporary degradation. Conversely, predicted unstable or low-quality segments could be internally marked as communication-restricted zones in which only safety-critical data streams are maintained. These use cases can make connectivity forecasts an important component of active communication management for Level 4 automated vehicles.

The practical integration of the proposed forecasting framework into automated vehicle architectures will be part of future work. In such an implementation, the connectivity forecasting module could act as an additional network planning and communication-management layer. Before a trip, the module would combine the planned route, historical route-section-wise connectivity data, map information, network operator information, and contextual parameters such as weather conditions. During operation, it could be updated with live radio measurements from the vehicle's communication unit. The resulting outputs could include route-section-wise connectivity risk scores, predicted signal quality, and warnings for unstable or communication-restricted segments.

X. Limitations

The dataset is restricted to a single rural route and one mobile network operator. Urban environments were not considered but are often analyzed in other studies [6][7][11]. All measurements were conducted using a single hardware configuration, which limits generalizability across device types. Temporal coverage is limited to the measurement period of one year, and some meteorological events are not represented.

Functional usability was defined using fixed KPI thresholds derived from industry recommendations. No direct throughput, latency, or packet-loss measurements were incorporated into the usability classification. Therefore, the functional connectivity state represents a proxy for application-level performance rather than a validated QoS.

The forecasting models rely on linear mechanisms and section-wise historical averaging. Nonlinear relationships and temporal dependencies between consecutive route sections were not modeled. The walk-forward validation approach requires a minimum number of historical trips before generating forecasts. While this ensures temporal realism, it limits applicability in early deployment stages where only limited historical data is available.

XI. Conclusion and Future Work

This paper presents a spatio-temporal and context-aware analysis of mobile network stability along a fixed route based on 48 repeated measurement drives. The stability classification revealed that a substantial portion of the route exhibits trip-to-trip variability that cannot be captured by static coverage representations. Contextual conditioning yields consistent reductions in prediction error and improves forecasting performance, especially in unstable geographic locations. The results indicate that context parameters act as uncertainty modulators

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

rather than deterministic predictors of signal quality. The study demonstrates that connectivity forecasting can support not only connectivity-aware route planning but also vehicle communication management strategies.

Future work will focus on extending the modeling framework. Nonlinear ML approaches (e.g., gradient boosting, random forests, or neural networks) will be investigated to capture more complex conditional dependencies. The dataset will also be expanded to include an additional route and another measurement platform to improve generalizability and robustness. Future measurement campaigns will include connectivity performance indicators, such as throughput and latency, alongside passive radio KPIs. Through these extensions, an advanced framework for context-aware mobile network prediction for automated driving environments will be created. Such capabilities represent important improvements for meeting the safety and reliability requirements of modern and future vehicles, as well as for mobile data exchange in general.

References

[1] M. Gerstner and R. Hackenberg, "Context-aware forecasting of mobile network quality for autonomous vehicle connectivity", in *Proc. Vehicular Analytics 2025, The Second International Conference on Vehicular Systems*, IARIA, 2025.

[2] Goldman Sachs, *Partially autonomous cars forecast to comprise 10% of new vehicle sales by 2030*, [Online]. Available: https://www.goldmansachs.com/insights/articles/partially-autonomous-cars-forecast-to-comprise-10-percent-of-new-vehicle-sales-by-2030. Accessed: Feb. 19, 2026.

[3] A. Takacs and T. Haidegger, "A method for mapping V2X communication requirements to highly automated and autonomous vehicle functions", *Future Internet*, vol. 16, no. 4, p. 108, 2024. DOI: 10.3390/fi16040108.

[4] R. Blasco, G. C. Ferrante, C. Watermann, and A. Palaios, "Predictive quality of service in cellular networks: Challenges, framework, and application in vehicular communications", *IEEE Commun. Mag.*, vol. 61, no. 3, pp. 44–49, 2023. DOI: 10.1109/MCOM.004.2200180.

[5] Federal Ministry of Justice and Consumer Protection, *Act to amend the road traffic act and the compulsory insurance act – act on autonomous driving*, [Online]. Available: http://www.bgbl.de/xaver/bgbl/start.xav?startbk=Bundesanzeiger_BGBl&jumpTo=bgbl121s3108.pdf. Accessed: Mai. 21, 2026.

[6] D. Madariaga, M. Panza, and J. Bustos-Jimenez, "I'm only unhappy when it rains: Forecasting mobile QoS with weather conditions", in *Proc. 2018 Network Traffic Measurement and Analysis Conf. (TMA)*, 2018, pp. 1–6. DOI: 10.23919/tma.2018.8506509.

[7] H. Schippers, M. Geis, S. Böcker, and C. Wietfeld, "DoNext: An open-access measurement dataset for machine learning-driven 5G mobile network analysis", *IEEE Trans. Mach. Learn. Commun. Netw.*, 2025. DOI: 10.1109/TMLCN.2025.3564239.

[8] Federal Motor Transport Authority, *Levels of automation*, [Online]. Available: https://www.kba.de/EN/Themen_en/Marktueberwachung_en/Produktpruefungen_en/AutomatisiertesAutonomesFahren_en/Automatisierungsstufen_en/Automatisierungsstufen_node_en.html. Accessed: Mai. 21, 2026.

[9] Federal Republic of Germany, *Ordinance on the approval and operation of motor vehicles with autonomous driving functions in defined operating areas – AFGBV*, [Online]. Available: https://www.buzer.de/AFGBV.htm. Accessed: Mai. 20, 2026.

[10] P. Torres et al., "Data analytics for forecasting cell congestion on LTE networks", in *Proc. 2017 Network Traffic Measurement and Analysis Conf. (TMA)*, 2017, pp. 1–6. DOI: 10.23919/tma.2017.8002917.

[11] D. Madariaga, J. Madariaga, J. Bustos-Jiménez, and B. Bustos, "Improving signal-strength aggregation for mobile crowdsourcing scenarios", *Sensors*, vol. 21, no. 4, p. 1084, 2021. DOI: 10.3390/s21041084.

[12] O. Sahin and V. Sathya, *Network quality prediction in a designated area using gps data*, SSRN preprint, 2023. DOI: 10.2139/ssrn.4525388.

[13] M. Rahman, I. Guvenc, D. Ramirez, and C.-W. Wong, *Uav-assisted coverage hole detection using reinforcement learning in urban cellular networks*, 2025. DOI: 10.48550/arXiv.2503.06494.

[14] T. Hultman, A. Boudjadar, and M. Asplund, "Connectivity-optimal shortest paths using crowdsourced data", in *Proc. IEEE Int. Conf. on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2016, pp. 1–6. DOI: 10.1109/percomw.2016.7457106.

[15] A. Palaios et al., "Machine learning for QoS prediction in vehicular communication: Challenges and solution approaches", *IEEE Access*, vol. 11, pp. 92 459–92 477, 2023. DOI: 10.1109/ACCESS.2023.3303528.

[16] G. Soos, D. Ficzere, P. Varga, and Z. Szalay, "Practical 5G KPI measurement results on a non-standalone architecture", in *Proc. NOMS 2020*, 2020, pp. 1–5. DOI: 10.1109/noms47738.2020.9110457.

[17] Z. Szalay et al., "5G-enabled autonomous driving demonstration with a V2X scenario-in-the-loop approach", *Sensors*, vol. 20, no. 24, p. 7344, 2020. DOI: 10.3390/s20247344.

[18] C. Inc., *LTE category guide: Enhancing wireless communication*, [Online]. Available: https://www.cavliwireless.com/blog/nerdiest-of-things/lte-category-wireless-communication. Accessed: Mai. 21, 2026.

[19] H. Fehmi, M. F. Amr, A. Bahnasse, and M. Talea, "5G network: Analysis and compare 5G NSA / 5G SA", *Procedia Computer Science*, vol. 203, pp. 594–598, 2022. DOI: 10.1016/j.procs.2022.07.085.

[20] I. Maskulainen et al., "Performance evaluation of adaptive beamforming in 5G-V2X networks", in *Proc. European Conf. on Networks and Communications (EuCNC)*, 2017, pp. 1–5. DOI: 10.1109/EuCNC.2017.7980728.

[21] V. A. Reddy et al., "Fundamental performance limits of mm-wave cooperative localization in linear topologies", *IEEE Wireless Commun. Lett.*, vol. 9, no. 11, pp. 1899–1903, 2020. DOI: 10.1109/LWC.2020.3007417.

[22] Teltonika Networks, *Mobile signal strength recommendations*, [Online]. Available: https://wiki.teltonika-networks.com/view/Mobile_Signal_Strength_Recommendations. Accessed: Mai. 21, 2026.

[23] R. Molina-Masegosa, M. Sepulcre, J. Gozalvez, F. Berens, and V. Martinez, "Empirical models for the realistic generation of cooperative awareness messages in vehicular networks", *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5713–5717, 2020. DOI: 10.1109/TVT.2020.2979232.

[24] R. H. Ratul et al., *Atmospheric influence on the path loss at high frequencies for deployment of 5g cellular communication networks*, arXiv:2306.01894, 2023.

[25] A. L. Imoize, S. O. Tofade, G. U. Ughegbe, F. I. Anyasi, and J. Isabona, "Updating analysis of key performance indicators of 4G LTE network with the prediction of missing values of critical network parameters based on experimental data from a

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

dense urban environment", *Data in Brief*, vol. 42, p. 108 240, 2022. DOI: 10.1016/j.dib.2022.108240.

[26] F. Afroz, R. Subramanian, R. Heidary, K. Sandrasegaran, and S. Ahmed, "SINR, RSRP, RSSI and RSRQ measurements in long term evolution networks", *Int. J. Wireless & Mobile Networks*, vol. 7, no. 4, pp. 113–123, 2015. DOI: 10.5121/ijwmn.2015.7409.

[27] Techlteworld, *RSSI (received signal strength indicator) in LTE*, [Online]. Available: https://techlteworld.com/rssi-received-signal-strength-indicator-in-lte/. Accessed: Mai. 21, 2026.

[28] A. Molisch, F. Tufvesson, J. Karedal, and C. Mecklenbrauker, "A survey on vehicle-to-vehicle propagation channels", *IEEE Wireless Commun.*, vol. 16, no. 6, pp. 12–22, 2009. DOI: 10.1109/MWC.2009.5361174.

[29] S. Farthofer et al., "An open mobile communications drive test data set and its use for machine learning", *IEEE Open J. Commun. Soc.*, vol. 3, pp. 1688–1701, 2022. DOI: 10.1109/OJCOMS.2022.3210289.

[30] CellMapper, *Cellular coverage and tower map*, [Online]. Available: https://www.cellmapper.net/. Accessed: Jan. 18, 2026.

[31] Open-Meteo, *Free open-source weather API*, [Online]. Available: https://open-meteo.com/. Accessed: Sep. 18, 2025.

[32] C. J. Willmott and K. Matsuura, "Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance", *Climate Research*, vol. 30, no. 1, pp. 79–82, 2005. DOI: 10.3354/cr030079.

[33] O. Zeigermann and C. N. Nguyen, *Machine Learning: A Brief Introduction with Python, Scikit-Learn, and TensorFlow*, 3rd. Heidelberg, Germany: O'Reilly, 2024.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

Some Deficiencies of TCP BBR Implementation in Ns-3 Network Simulator Detected under High Speed and Heavy Load Conditions

Toshihiko Kato†, Takahiko Kato‡, Ryo Yamamoto†, and Satoshi Ohzahata†

† University of Electro-Communications
Tokyo, Japan

‡ University of Fukui
Fukui, Japan

e-mail: kato@net.lab.uec.ac.jp, t-kato@u-fukui.ac.jp, ryo-yamamoto@uec.ac.jp, ohzahata@uec.ac.jp

***Abstract*— TCP Bottleneck Bandwidth and Round-trip propagation time (BBR) is a congestion control algorithm that was introduced by Google relatively recently, and it is widely used currently. BBR is based on the congestion-based congestion control which is different from the conventional algorithms. So, a lot of researches have been reported on the performance evaluation. In the early stage, it is done using the BBR software in the Linux operating system over a physical network. Nowadays, a network simulator, such as ns-3 is used for the evaluation. BBR was officially introduced in ns-3 version 3.34 released in 2021. However, the BBR software in ns-3 is different from that in Linux, and so it may be possible it behaves differently. This paper reports some results of BBR performance evaluation where sixteen flows share one 1 Gbps bottleneck link with a limited router buffer. The results were different when we used ns-3 versions 3.40 and 3.44. We also found some problems that the BBR state transition does not work correctly over a high speed link. We analyzed the behaviors of BBR program implemented in the ns-3 network simulator and examined the TCP BBR source code to find the reasons for those deficiencies. This paper describes the results of those investigations.**

Keywords- TCP; Congestion Control; BBR; Ns-3 Network Simulator.

I. INTRODUCTION

This paper is an extension of our previous paper [1], which was presented at the IARIA conference named AFIN 2025.

Along with the proliferation of various types of networks and applications, a lot of TCP variants with different congestion control algorithms are designed, implemented, and widely spread [2]. The congestion control manages the congestion window size, called cwnd according to a network congestion situation. The conventional algorithms are categorized into the loss-based, the delay-based and the hybrid approaches. The loss-based algorithm detects a network congestion by packet losses, and this category includes TCP NewReno [3], HighSpeed TCP [4], CUBIC TCP [5], and Hamilton TCP [6]. The delay-based algorithm detects a congestion by an increase of the Round-Trip Time (RTT), and an example of this category is TCP Vegas [7]. The hybrid algorithm combines the loss-based and delay-based approaches. In TCP Veno [8], the congestion is detected by packet losses and the change of cwnd is controlled according to RTT. In Compound TCP [9], cwnd is the sum of a window size following HighSpeed TCP and one following TCP Vegas, and it behaves in a way such that the former is effective when RTT is small and vice versa.

TCP BBR (Bottleneck Bandwidth and Round-trip propagation time) [10] was recently proposed by Google. It is a new type of congestion control algorithm called congestion-based congestion control. A data sender tries to send data segments according to the Bandwidth Delay Product (BDP) of the TCP connection. A sender measures the bottleneck bandwidth and the minimum RTT independently and estimates the BDP. It does not take care of retransmissions for the congestion control. Another feature of TCP BBR is that it follows the rate-based data transmission. Most of TCP variants adopt the window-based data transmission, where a sender sends out data segments in a burst within the limitation of window size (congestion window and advertised window). Instead, TCP BBR sends data segments in the pace determined by the estimated BDP and the segment size. As a result, data segments are transmitted sparsely and it is highly possible to avoid congestions in a multiple TCP flow condition.

TCP BBR came to be used widely since its proposal. Sawada et al. reported that TCP BBR was used by 30% of the major web servers in 2020. There are a lot of study results on the performance of TCP BBR [11]-[18]. In early stages, the performance evaluation was done using the BBR implementation in the Linux kernel over a physical network or a network emulator, such as Mininet [19]. Recently, TCP BBR was introduced in the network simulator, such as ns-3 [20]. Actually, the TCP BBR was implemented in ns-3 at version 3.34 released in 2021. Some studies performed experiments using ns-3 [15][17][18]. However, TCP BBR in ns-3 is implemented by its original source code, and so it may be possible that the code behaves differently from that in Linux.

We performed the TCP BBR throughput test under a high speed and heavy load condition where sixteen BBR flows share a 1Gbps bottleneck link [1]. When the output buffer of the router connected to the bottleneck link is limited, the throughput of some BBR flows becomes extremely low. Although Hock et al. reported the intra-protocol unfairness of TCP BBR due to small bottleneck buffer [11], the results of our experiment are much worse than that. We conducted this experiment using ns-3 whose version is 3.40. We performed the same experiment using ns-3 version 3.44, which was the newest as of writing our previous paper. The experimental results were different from that using ns-3 version 3.40, and all of the sixteen BBR

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

flows provided similar throughput. In this experiment, we also found that, in some cases, TCP BBR in ns-3 does not perform the BBR state transition for measuring the bandwidth and minimum RTT independently [21].

We analyzed the behaviors of TCP BBR in ns-3 in detail and consulted its BBR source code, from the two points of views, i.e. focusing on the difference of versions and on the BBR state transitions. This paper describes the results of our experiments and analysis of BBR behaviors.

The rest of this paper is organized as follows. Section II gives some background information including the overview of TCP BBR and the performance evaluation studies reported so far. Section III describes the details of the performance evaluation experiments focusing on the differences of ns-3 versions. Section IV compares the BBR codes implemented in ns-3 version 3.40 and 3.44 in detail. Section V discusses the detailed analysis focusing on the BBR state transitions. In the end, Section VI concludes this paper.

II. BACKGROUNDS

A. Overview of TCP BBR

TCP BBR uses the estimation for the available bottleneck bandwidth, *BtlBw*, and the minimal round-trip propagation time, *RTprop*, to calculate a path's available BDP. A BBR sends data segments with the pacing rate given by pacing_gain × BltBw × RTprop with the help of the rate-based control. The parameter pacing_gain, which is set to 1 most of the time, is used to control the actual pacing rate.

The BBR algorithm has four different states specific to BBR: Startup, Drain, Probe Bandwidth (ProbeBW), and Probe RTT (ProbeRTT).

The first phase starts in the Startup state in order to adapt the exponential startup behavior where the pacing_gain is set to 2/ln2 (2.885). Once the measured bandwidth does not increase further, BBR assumes to have reached the bottleneck bandwidth (maximum throughput), and shifts to the Drain state. Here, BBR reduces the pacing_gain to ln2/2.

Afterwards, BBR enters the ProbeBW state in which it probes for more available bandwidth. This is performed in eight cycles, each of which takes RTprop. The pacing_gain of each cycle is 5/4, 3/4, 1, 1, 1, 1, 1, and 1. In this state, BBR continuously samples the bandwidth and uses the maximum as the BtlBw estimator, whereby values are valid for the timespan of ten RTprop.

After not measuring a new RTprop value for time window W_R (typically, ten seconds), BBR stops probing for bandwidth and enters the ProbeRTT state. In this state, the bandwidth is reduced to four packets per sec. to drain any possible queue and get a real estimation of the RTT. This state is kept for 200 ms plus one RTT. If a new minimum value is measured, RTprop is updated and valid for W_R.

B. Studies on Performance Evaluation of TCP BBR

Table I summarizes the performance evaluation studies of TCP BBR. It shows the BBR source codes and the network environment used by the evaluation, the target TCP congestion control algorithms, and the maximum numbers of TCP flows examined in the experiment. TCP BBR was proposed in 2016, and introduced to the Linux operating system. So the evaluations in the early stage used the BBR code in Linux [11]-[14][16]. The BBR software was introduced to the ns-3 network simulator by Vivek Jain et al. [22] in 2018. This software was implemented over ns-3 version 3.27, which was released in 2017. References [15] and [17] used this version of TCP BBR. TCP BBR was introduced into ns-3 officially in the version 3.34 released in 2021. It is considered that [18] used this or later version of ns-3. The performance evaluation experiments were done for single BBR, multiple BBRs and mixture of multiple BBRs and other congestion control algorithms.

TABLE I. PERFORMANCE EVALUATION STUDIES OF TCP BBR.

reference	BBR code	network (link speed)	evaluation target	max number of flows
Hoch et al., 2017 [11]	Linux	physical (1, 10Gbps)	BBR, BBR + CUBIC	6 BBRs, 1 BBR + 1 CUBIC
Miyazawa et al., 2018 [12]	Linux	physical (1Gbps)	BBR + CUBIC	10 BBRs + 10 CUBICs
Sholz et al., 2018 [13]	Linux	Mininet (10Mbps)	BBR, BBR + CUBIC	6 BBRs, 10 BBRs + 10 CUBICs
Claypool et al., 2019 [14]	Linux	physical (40, 80, 120Mbps)	BBR, BBR + CUBIC	8 BBRs, 3 BBRs + 3 CUBICs
Zhang et al., 2019 [15]	ns-3 (Jain's version)	ns-3 (1, 10Gbps)	BBR, BBR + BIC, BBR + BIC + NewReno + Vegas	2 BBRs, 2 BBRs + 2 BIC, 1 BBR + 1 BIC + 1 NewReno + 1 Vegas
Kim et al., 2020 [16]	Linux	Mininet (100Mbps)	BBR, (Delay-aware BBR)	2 BBRs
Lanigan, 2020 [17]	ns-3 (Jain's version)	ns-3 (5, 10, 40Mbps)	BBR, (other BBR variants)	4 BBRs
Tang, 2024 [18]	ns-3	ns-3 (10, 100Mbps)	BBR, BBR + CUBIC	8 BBRs, 2 BBRs + 2 CUBICs
Ours, 2025 [1]	ns-3	ns-3 (1Gbps)	BBR	16 BBRs

As for the simulation studies shown by gray-colored boxes, the network bandwidth and the number of BBR flows are limited. Other than our result, only Zhang et al. use a Gbps level bandwidth, but this result evaluates the BBR performance only 10 sec., which is not enough to analyze the detailed BBR behaviors. The number of flows were limited, eight BBR flows at the maximum by Tang. In contrast with them, our performance evaluation used sixteen BBR flows using 1 Gbps links.

III. PERFORMANCE EVALUATION FOCUSING ON NS-3 VERSIONS

A. Experimental Setup

Figure 1 shows the network configuration used in this evaluation, which is a dumbbell configuration where sixteen data senders (S1 through S16) and the corresponding receivers (R1 through R16) are connected through two routers (N1 and N2). The bandwidth and transmission delay

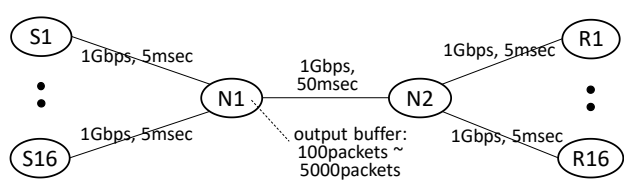


Figure 1. Performance evaluation network.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

are specified in the figure, where the backbone link between N1 and N2 has a longer delay, 50 msec, than the other links. The output buffer in N1 is set to 100 packets through 5,000 packets. The maximum segment size is 1,420 bytes. Since RTT is 120 msec and the bottleneck bandwidth is 1 Gbps, the BDP in this experiment is 15 MB, i.e., around 10,000 packets.

The details of ns-3 parameters are as follows.

- Queue discipline: First-In First-Out queue discipline following the drop tail policy [23].
- TCP loss recovery algorithm: TCP classic recovery [24].
- DelAckCount (Number of packets to wait before sending a TCP Ack): 1.
- Explicit congestion notification functionality: not used.
- SACK: used.

In the evaluation, all data senders start data transfer at the same time and continue the communication until time 50 sec. We used the ns-3 version 3.40, released on Sep. 27, 2023, and version 3.44 released Mar. 9, 2025.

B. *Evaluation Results Using Ns-3 Version 3.40*

This subsection describes the results using ns-3 version 3.40. In the configuration shown in Figure 1, we set the output buffer of N1 to 100, 500, 1,000, 2,000, and 5,000 packets. As described above, the BDP in this configuration is around 10,000 packets, and so the buffer is not large enough, especially the value of 100 packets is extremely small.

Figure 2 shows the throughput of individual TCP BBR flows for five cases of the N1 output buffer. The bars are sorted in the descending order of throughput.

When the N1 output buffer is 100 packets, ten BBR flows provided high throughput more than 50 Mbps, and on the other hand, the other six flows provided very low throughput less than 1 Mbps. When the N1 output buffer is 500 and 1,000 packets, the results were similar. Nine or ten BBR flows out of the sixteen provided high throughput but the throughput of the rest was extremely low. But, for the cases that the N1 output buffer is 2,000 and 5,000 packets, the results were different, and almost all BBR flows provided high performance. Figure 3 shows the box-and-whisker plots where the whiskers show the minimum and maximum values, and the boxes give the first quartile, median and third quartile. The cross marks in the figure give the average. Table II shows the average, standard deviation, and total of sixteen flows' throughput. For the cases that the N1 output buffer is 100, 500, and 1,000 packets, the distribution of the throughput is large, and the total throughput is around 600 Mbps. Since the transmission rate of the bottleneck link is 1 Gbps, the efficiency is around 60 %. In contrast, in the cases that the N1 output buffer is 2,000 and 5,000 packets, the distribution becomes small and the efficiency is as high as 85 % or 90%.

Figure 4 shows the time variation of the pacing rate of the 6th BBR flow (flow 6) and the 12th BBR flow (flow 12) when the N1 output buffer is 100 packets. In the beginning, the pacing rate increases similarly and some packet losses following timeout retransmissions. After that, the pacing rate of flow 12 keeps a very low value, such as 0.1 and 0.01 Mbps. In contrast, that of 6th flow recovers to the value of 87 Mbps. This situation continues to the end of experiment. Other flows showed similar behaviors. This is the reason for the high and low throughput.

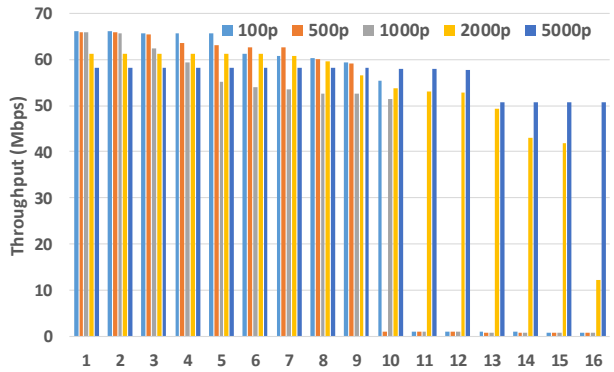


Figure 2. Throughput of individual BBR flows.

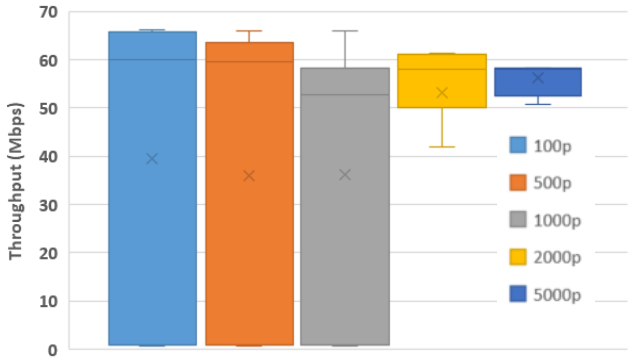


Figure 3. Box-and-whisker plots of BBR flow throughput.

TABLE II. AVERAGE, STANDARD DEVIATION, AND TOTAL OF THROUGHPUT OF BBR FLOWS (MBPS).

buffer size	average	std. dev.	total
100 pkts.	39.5	31.0	632.0
500 pkts.	35.9	32.0	574.8
1000 pkts.	36.1	28.5	577.9
2000 pkts.	53.1	12.7	850.3
5000 pkts.	56.3	3.3	900.2

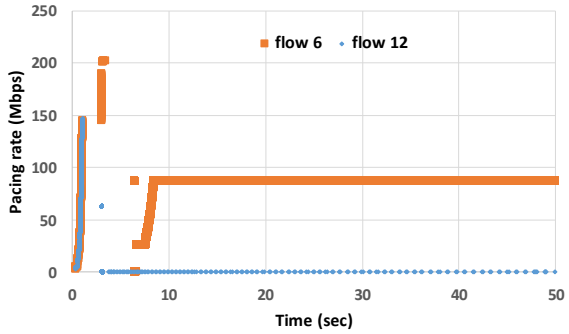


Figure 4. Pacing rate vs. time for flow 6 and 12 when the N1 output buffer is 100 packets.

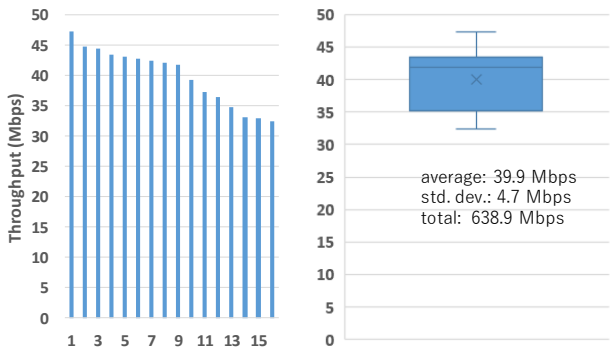
2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

C. Evaluation Results Using Ns-3 Version 3.44

Next, we performed the same experiment using ns-3 version 3.44. Actually, we used the same script file as that for ns-3 version 3.40. Figure 5 shows the results for the case that the N1 output buffer is 100 packets. Figure 5 (a) gives the individual throughput of sixteen BBR flows and Figure 5 (b) is its box-and-whisker plot. The distribution becomes small and all flows provide high performance, between 32 Mbps and 47 Mbps. It should be noted that the total throughput is similar to that in ns-3 version 3.40. This means that the behaviors of flows is fair compared with the former experiment.

Figure 6 shows the time variation of the pacing rate of flow 5. The behavior in the first 10 seconds is similar to that of ns-3 version 3.40 depicted in Figure 4. After that, however, the pacing rate changes among 50 Mbps, 70 Mbps, and 85 Mbps. For other flows, the behavior of pacing rate is similar, and as a result, the throughput of individual flows provided similar value in the experiment using ns-3 version 3.44.

Both results seem to be explainable. As mentioned above, it is possible that TCP BBR shows the intra-protocol unfairness when the bottleneck buffer is limited [11]. So, the results by using ns-3 version 3.40 are not necessarily wrong. The problem is that the behavior changed for ns-3 version 3.40 and 3.44. The next section discusses this issue in more detail.



(a) Individual throughput (b) Box-and-whisker plots

Figure 5. Results by ns-3 3.44 for 100 packet N1 output buffer.

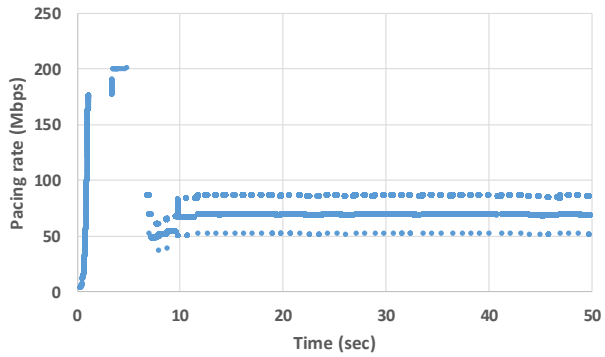


Figure 6. Pacing rate vs. time for flow 5 by ns-3 3.44 when the N1 output buffer is 100 packets.

IV. Detailed Analysis of Ns-3 BBR Source Code Focusing on Versions

A. Program Strucsture of Ns-3 BBR Code

In ns-3, the main procedures of TCP are implemented in the tcp-socket-base.{h, cc} files. They give the functions for handling data send requests from the upper layer and for handling TCP segments received from the remote node. The control variables, such as `m_nextTxSequence` and `m_cWnd` are implemented in tcp-socket-state.h. The congestion control algorithms are implemented in other files independently. TCP BBR is implemented in the tcp-bbr.{h, cc} files.

Figure 7 depicts the function call graph of the TCP BBR congestion control (function `CongControl()`). It calls two main functions, `UpdateModelAndState()` and `UpdateControlParameters()`. The former calls the functions changing the states corresponding to BtlBw and RTprop. The latter calls the functions updating the related parameters, such as `m_pacingRate` and `m_cWnd`. The overviews of those functions are listed in Table III.

Figure 8 depicts the function call graph in tcp-socket-base.cc when an ACK segment is received. First, `ReceivedAck()` is called. As for the ACK processing, it calls `ProcessAck()`, and `CongControl()` in tcp-bbr.cc. `ProcessAck()` manages the ACK related parameters, and if the received ACK is a duplicate ACK, `DupAck()` is called and the retransmission is processed if necessary. After `CongControl()`, `SendPendingData()` is called to send the following data when the received ACK opens a window. In BBR, the rate-based data transfer is adopted and this is implemented by the TCP Pacing mechanism [25]. A timer called `m_pacingTimer` is used, which expires according to a rate to send data segments according to `m_pacingRate`, calculated in `SetPacingRate()`.

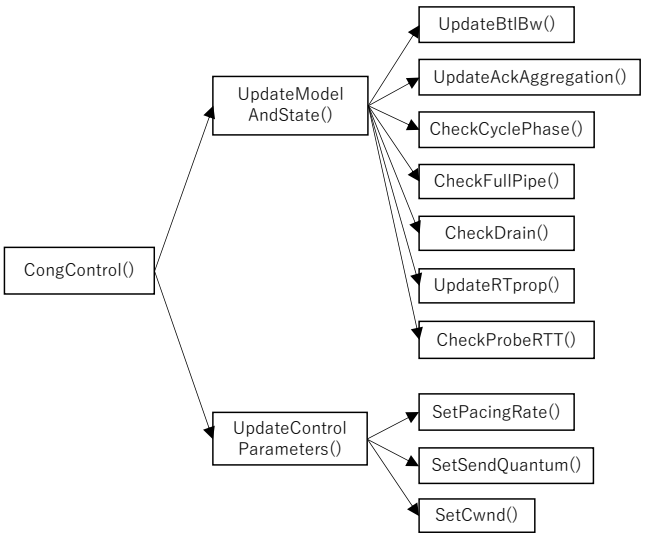


Figure 7. Function calls in TCP BBR Code in ns-3 version 3.40.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

TABLE III. OVERVIEW OF BBR FUNCTIONS IN FIG. 7.

function	behavior
UpdateBtlBw()	Updates maxBwFilter if a new delivery rate is larger than prior value.
UpdateAck Aggregation()	Takes account of delayed ACK.
CheckCyclePhase()	If in ProbeBW phase and a RTT elapsed, advance cycle.
CheckFullPipe()	If delivery rate becomes large enough, set m_isPipeFilled = true
CheckDrain()	If in Startup phase and m_isPipeFilled is true, enter Drain phase. If in Drain phase and inflight data decreased, enter ProbeBW phase.
UpdateRTprop()	If RTT didn't increase or W_R (10 sec) elapsed (m_rtPropExpired = true), update m_rtProp.
CheckProbeRTT()	If not in ProbeRTT phase and m_rtPropExpierd, enter ProbeRTT phase and save cwnd. If in ProbeRTT phase and 200 msec elapsed, enter ProbeBW if m_isPipeFilled or Startup otherwise.
SetPacingRate()	Set rate = Max value in maxBwFilter × pacing_gain. If m_isPipeFilled or rate > m_pacingRate, set m_pacingRate = rate.
SetSendQuantum()	Set m_sendQuantum = MSS.
SetCwnd()	Determine cwnd considering targetCwnd and packet losses. If in ProbeRTT phase, set cwnd to 4 MSS.

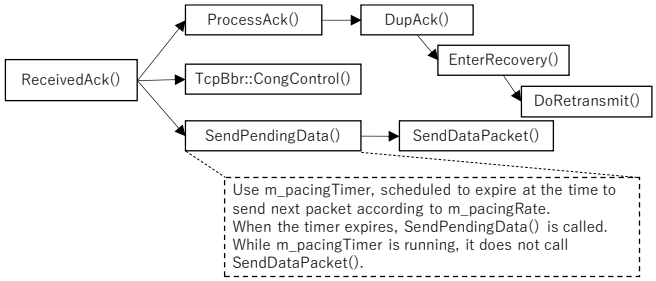


Figure 8. Function calls in TCP when receiving ACK in ns-3 version 3.40.

These are described based on the program structure of ns-3 version 3.40. The structure itself is the same in version 3.44.

B. *Analysis on Difference of Ns-3 3.40 and 3.44*

We analyzed the BBR codes in version 3.40 and 3.44 to clarify why the results are different in these two versions. From the homepage of ns-3 [26], the TCP related changes from version 3.40 to 3.44 are as follows.

1) TCP Cubic now supports TCP-friendliness by default, making the congestion window growth somewhat more aggressive. (from 3.40 to 3.41)
2) TcpCubic and TcpLinuxReno will no longer grow their cwnd when application-limited. (from 3.41 to 3.42)
3) Deprecated EventId::IsRunning(). It has been replaced with EventId::IsPending(). (from 3.41 to 3.42)
4) TCP Proportional Rate Reduction (PRR) recovery has been aligned to the updates in draft-ietf-tcpm-prr-rfc6937bis. (from 3.42 to 3.43)
5) A new trace source TcpSocketBase::LastRtt has been added for tracing the last RTT sample observed. The existing trace source TcpSocketBase::Rtt is still providing the smoothed RTT, although it had been incorrectly documented as providing the last RTT. (from 3.42 to 3.43)

Among them, 1), 2), and 4) do not give any impacts to our experiment, because we do not use TCP Cubic, NewReno, nor PRR recovery.

Then we compared the source codes of tcp-bbr.{h, cc} and tcp-socket-base.{h, cc}. The followings are differences that are considered to give some impacts to the BBR behaviors.

tcp-bbr.cc:

- `m_rtProp` in 3.40 is changed to `m_minRtt` in 3.44. This seems to be just a text-level modification.
- After rate is calculated in `SetPacingRate()` as shown in Table III, the following modification is added.
  `rate *= (1.f - m_pacingMargin),`
  `m_pacingMargin` is set to 0.01
  This needs to be considered.
- One `if` sentence is modified in `UpdateBltBw()`.
  3.40: `if (rs.m_deliveryRate == 0)`
  3.44: `if (rs.m_delivered < 0 || interval.IsZero())`
  This needs to be considered.
- `m_rtProp` (`m_minRtt`) is set to `m_lastRtt` in 3.40, but to `m_srtt` in 3.44. This is related to item 5) in the change history. This may need to be considered.

tcp-socket-base.cc

- One else if sentence is modified in `DupAck()`:
  3.40: `else if (m_txBuffer->IsLost(m_highRxAckMark + m_tcb->m_segmentSize)`
  3.44: `else if (m_txBuffer->IsLost(m_highRxAckMark)`
  This needs to be considered.
- Related to item 5) in the change history, `EstimateRtt()` is largely modified. This needs to be considered.

Based on these considerations, we took the following two approaches.

Approach 1: Modify tcp-bbr.cc and tcp-socket-base.cc for the points we decided to be considered.

Approach 2: Use tcp-bbr.cc and tcp-socket-base.cc of version 3.44 in ns-3 version 3.44.

Approach 2 required the following modification in the BBR source code,

tcp-socket-base.cc to be ported:

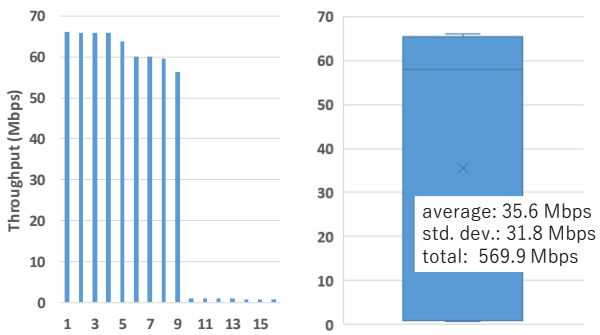
- Change `IsPending()` for `EventId` variables back to `IsRunning()`. This is related to item 3) in the change history.

tcp-socket-state.h in version 3.40:

- Define `TraceValue<Time> m_srtt` and `bool m_isCwndLimited`.

We performed both examination for the case that the bottleneck buffer is 100 packet. The results were similar and did not improve the performance of ns-3 version 3.40. Figure 9 shows the result of Approach 2. Out of the sixteen BBR flows, nine provided high throughput but the throughput of the other seven flows was extremely low. The average and total throughput was even worse than that of the original version 3.40. This result says that the difference of BBR behaviors in ns-3 versions 3.40 and 3.44 may come from some other parts in the simulator. In the current stage, we do not clarify the details.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org



(a) Individual throughput (b) Box-and-whisker plots

Figure 9. Results of Approach 2 porting 3.44 BBR code to 3.40 for 100 packet N1 output buffer.

V. Detailed Analysis of Focusing on BBR State Transitions

A. *State Transition in Ns-3 BBR Implementation*

As stated in Section II, TCP BBR introduces its own state transition to estimate BtlBw and RTprop independently. After the initial state transition from Startup and Drain states, BBR repeatedly takes states ProbeBW and ProbeRTT, each of which is designed for estimating BltBw and RTprop, respectively. Especially, BBR shifts its state to ProbeRTT by a time window basis of W_R, whose value is 10 sec. by default.

Figure 10 shows the state transition log of BBR flows executed by ns-3 version 3.44. In each flow, the state starts from Startup and shifts to Drain. Although the state is changed to ProbeBW immediately, there are no state changes after that. That means there are no state transitions to ProbeRTT. Figure 10 shows the results of five flows. The other even flows showed the similar state transitions. In the case of ns-3 version 3.40, the results were the same.

B. *Analysis oc Ns-3 BBR Source Code Focusing on State Transitions*

Among the functions listed in Table III, the BBR state transition from ProbeBW to ProbeRTT is performed in function `CheckProbeRTT()`. Figure 11 shows the corresponding source code. The state transition itself is performed in function `EnterProbeRTT()` in the figure, and the condition to execute it is that the state is not ProbeRTT, `m_minRttExpired` is true, and `m_idleRestart` is false. The reason for not shifting to ProbeRTT is supposed that `m_minRttExpired` is always false, or that `m_idleRestart` is always true. By examining the values of `m_minRttExpired` and `m_idleRestart` in the experiment using version 3.44 BBR, we found that `m_minRttExpired` is always false in the experiment.

```
flow 1 TcpBbr:SetBbrState(): At 0.12 Set to BBR_STARTUP
flow 1 TcpBbr:SetBbrState(): At 3.6997 Changing from BBR_STARTUP to BBR_DRAIN
flow 1 TcpBbr:SetBbrState(): At 3.6997 Changing from BBR_DRAIN to BBR_PROBE_BW

flow 2 TcpBbr:SetBbrState(): At 0.12 Set to BBR_STARTUP
flow 2 TcpBbr:SetBbrState(): At 3.5903 Changing from BBR_STARTUP to BBR_DRAIN
flow 2 TcpBbr:SetBbrState(): At 3.5983 Changing from BBR_DRAIN to BBR_PROBE_BW

flow 3 TcpBbr:SetBbrState(): At 0.12 Set to BBR_STARTUP
flow 3 TcpBbr:SetBbrState(): At 3.5799 Changing from BBR_STARTUP to BBR_DRAIN
flow 3 TcpBbr:SetBbrState(): At 3.5799 Changing from BBR_DRAIN to BBR_PROBE_BW

flow 4 TcpBbr:SetBbrState(): At 0.12 Set to BBR_STARTUP
flow 4 TcpBbr:SetBbrState(): At 3.5813 Changing from BBR_STARTUP to BBR_DRAIN
flow 4 TcpBbr:SetBbrState(): At 3.5895 Changing from BBR_DRAIN to BBR_PROBE_BW
. . .
flow 16 TcpBbr:SetBbrState(): At 0.12 Set to BBR_STARTUP
flow 16 TcpBbr:SetBbrState(): At 3.4705 Changing from BBR_STARTUP to BBR_DRAIN
flow 16 TcpBbr:SetBbrState(): At 3.4705 Changing from BBR_DRAIN to BBR_PROBE_BW
```

Figure 10. BBR state transition log in version 3.44 BBR program.

Based on this result, we examined the source code for setting `m_minRttExpired`. This is performed in function `UpdateRTprop()` shown in Figure 12. In this function, `m_minRttExpired` is set to a boolean value whether the current time is larger than `m_minRttStamp` + W_R (10 sec.). Here, `m_minRttStamp` is the time when a new value was assigned to `m_minRtt`, and `m_minRtt` corresponds to RTprop.

The latter part of Figure 12 shows how to update `m_minRtt` and `m_minRttStamp`. Variable `tcb->m_lastRtt` used here indicates the RTT value of the last ACKed packet. The condition to update the variables is `tcb->m_lastRtt` is not greater than `m_minRtt`, or `m_minRttExpired` is true.

In the experiment, we obtained the program log shown in Figure 13 as for the update of `m_minRttStamp` for flow 1. As shown in the figure, these updates were performed 89,422 times during the fifty second program run. Ideally, `m_minRttStamp` is updated every ten second along with `m_minRttExpired` being true as described above. However, too many advances of `m_minRttStamp` make the sum of itself and W_R larger than the current time and keep `m_minRttExpired` false.

Since `m_minRttExpired` is always false, the condition that allows `m_minRttStamp` to be updated is supposed to be "`tcb->m_lastRtt <= m_minRtt`." We examined the values of `tcb->m_lastRtt` and

```
if (m_state != BbrMode_t::BBR_PROBE_RTT
   && m_minRttExpired && !m_idleRestart)
   {
      EnterProbeRTT();
      SaveCwnd(tcb);
      m_probeRttDoneStamp = Seconds(0);
   }
```

Figure 11. Process to enter ProbeRTT state in `CheckProbeRTT()`.

```
m_minRttExpired = Simulator::Now() >
        (m_minRttStamp + m_minRttFilterLen);

if (tcb->m_lastRtt.Get().IsPositive()
   && (tcb->m_lastRtt <= m_minRtt ||
        m_minRttExpired))
   {
      m_minRtt = tcb->m_lastRtt;
      m_minRttStamp = Simulator::Now();
   }
```

Figure 12. Process to update `m_minRttExpired` in `UpdateRTprop()`.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

```
TcpBbr:UpdateRTprop(): At 0.048 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0482 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0484 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0486 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0488 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.049 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0721 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0729 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0737 minRttStamp is updated.
TcpBbr:UpdateRTprop(): At 0.0746 minRttStamp is updated.
…
Total number is 89,442.
```

Figure 13. `m_minRttStamp` update log in `UpdateRTprop()` for.flow 1.

```
TcpBbr:UpdateRTprop(): At 0.048 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0482 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0484 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0486 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0488 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.049 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0492 tcb->m_lastRtt is 0.025 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0494 tcb->m_lastRtt is 0.025 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0495 tcb->m_lastRtt is 0.025 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0497 tcb->m_lastRtt is 0.025 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0721 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0729 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0737 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0746 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0754 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
m_lastRtt is 0.024 continuously  …
TcpBbr:UpdateRTprop(): At 0.0862 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.087 tcb->m_lastRtt is 0.025 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0879 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0961 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0969 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0978 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0986 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.0994 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1003 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1011 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1019 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1028 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1036 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1044 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1052 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.106 tcb->m_lastRtt is 0.025 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1068 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1075 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
TcpBbr:UpdateRTprop(): At 0.1082 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
…
```

Figure 14. Log of `tcb->m_lastRtt` and `m_minRttStamp`.for flow 1.

`m_minRtt` when these variables are updated. Figure 14 gives a part of the results for flow 1. As shown by the numbers with the underlined and bold form in this figure, `tcb->m_lastRtt` and `m_minRtt` often take the same value. This introduces the frequent updates of `m_minRttStamp`.

We examined the BBR source code in the Linux operating system. Figure 15 shows the corresponding part of the BBR source code [27]. The ns-3 BBR source code has a similar structure with that in Linux. `UpdateRTprop()` in ns-3 corresponds to `bbr_update_min_rtt()` in Linux. `tcb->m_lastRtt` and `m_minRtt` correspond to `rs->rtt_us` and `bbr->probe_rtt_min_us`, respectively. So, the condition we focus on corresponds to

`rs->rtt_us < bbr->probe_rtt_min_us`,

```
static void bbr_update_min_rtt(struct sock *sk, const struct rate_sample *rs)
{
    expire = bbr->probe_rtt_min_stamp +
             msecs_to_jiffies(bbr_param(sk, probe_rtt_win_ms));
    probe_rtt_expired = after(tcp_jiffies32, expire);
    if (rs->rtt_us >= 0 && (rs->rtt_us < bbr->probe_rtt_min_us ||
                            (probe_rtt_expired && !rs->is_ack_delayed))) {
        bbr->probe_rtt_min_us = rs->rtt_us;
        bbr->probe_rtt_min_stamp = tcp_jiffies32;
    }
```

Figure 15. Corresponding source code in Linux.

highlighted in the figure by the underlined and bold style. From these analysis, it is obvious that the condition we focus on should be

`tcb->m_lastRtt < m_minRtt`,

instead of

`tcb->m_lastRtt <= m_minRtt`.

The discussions above were done for the ns-3 version 3.44 BBR program. The situation was the same for the ns-3 version 3.40 BBR program.

C. Results of Bug Fixing for Ns-3 Ver.3.44 BBR Program

We fixed the bug described above in the version 3.44 BBR program.

The throughput of individual flows is as shown in Figure 16. By comparing this result and that given in Figure 5, we can conclude that the throughput has the similar tendency for the original 3.44 BBR program and the bug-fixed 3.44 BBR program.

Figure 17 shows the BBR state transition logs for the bug-fixed program. Comparing this figure with Figure 10, we can conclude that the BBR state transition is performed well in this result. The state transitions from ProbeBW to ProbeRTT are performed repeatedly.

But there is another problem in this result. It is considered that the state transitions to ProbeRTT need to be observed four times in each flow. However, the count of these transitions in the experiment is less than four. For example, only one transition to ProbeRTT occurs in flows 2 and 3. In flows 1 and 9, the count of the transitions to Probe RTT is two, and in flows 4, 15, and 16 the count is three. Actually, there were no flows with four time transitions among sixteen flows in the experiment.

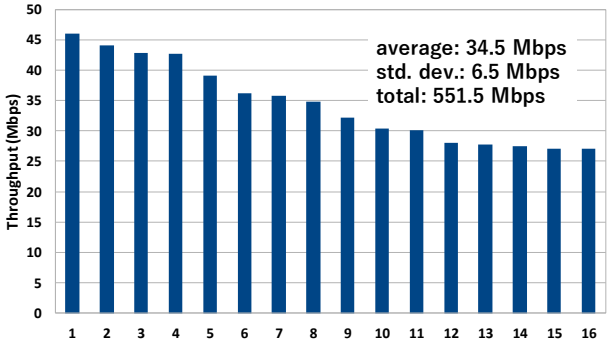


Figure 16. Throughput of individual flows using bug-fixed 3.44 BBR program for 100 packet N1 output buffer.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

```
flow 1 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 1 TcpBbr:SetBbrState(): At 1.1993 Changing from BBR_STARTUP to BBR_DRAIN
flow 1 TcpBbr:SetBbrState(): At 1.1993 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 1 TcpBbr:SetBbrState(): At 11.6166 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 1 TcpBbr:SetBbrState(): At 11.8413 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 1 TcpBbr:SetBbrState(): At 21.8414 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 1 TcpBbr:SetBbrState(): At 22.8356 Changing from BBR_PROBE_RTT to BBR_PROBE_BW

flow 2 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 2 TcpBbr:SetBbrState(): At 0.254 Changing from BBR_STARTUP to BBR_DRAIN
flow 2 TcpBbr:SetBbrState(): At 1.2264 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 2 TcpBbr:SetBbrState(): At 10.639 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 2 TcpBbr:SetBbrState(): At 10.8553 Changing from BBR_PROBE_RTT to BBR_PROBE_BW

flow 3 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 3 TcpBbr:SetBbrState(): At 0.2524 Changing from BBR_STARTUP to BBR_DRAIN
flow 3 TcpBbr:SetBbrState(): At 1.2264 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 3 TcpBbr:SetBbrState(): At 10.8866 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 3 TcpBbr:SetBbrState(): At 11.1029 Changing from BBR_PROBE_RTT to BBR_PROBE_BW

flow 4 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 4 TcpBbr:SetBbrState(): At 0.246 Changing from BBR_STARTUP to BBR_DRAIN
flow 4 TcpBbr:SetBbrState(): At 0.2673 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 4 TcpBbr:SetBbrState(): At 10.8754 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 4 TcpBbr:SetBbrState(): At 11.0918 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 4 TcpBbr:SetBbrState(): At 21.092 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 4 TcpBbr:SetBbrState(): At 21.3396 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 4 TcpBbr:SetBbrState(): At 32.0601 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 4 TcpBbr:SetBbrState(): At 32.2808 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
…
flow 9 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 9 TcpBbr:SetBbrState(): At 0.2533 Changing from BBR_STARTUP to BBR_DRAIN
flow 9 TcpBbr:SetBbrState(): At 1.2302 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 9 TcpBbr:SetBbrState(): At 10.6391 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 9 TcpBbr:SetBbrState(): At 10.8554 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 9 TcpBbr:SetBbrState(): At 31.0323 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 9 TcpBbr:SetBbrState(): At 31.9589 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
…
flow 15 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 15 TcpBbr:SetBbrState(): At 0.2469 Changing from BBR_STARTUP to BBR_DRAIN
flow 15 TcpBbr:SetBbrState(): At 1.2267 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 15 TcpBbr:SetBbrState(): At 10.0241 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 15 TcpBbr:SetBbrState(): At 10.264 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 15 TcpBbr:SetBbrState(): At 20.2643 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 15 TcpBbr:SetBbrState(): At 20.5036 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 15 TcpBbr:SetBbrState(): At 41.262 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 15 TcpBbr:SetBbrState(): At 41.472 Changing from BBR_PROBE_RTT to BBR_PROBE_BW

flow 16 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 16 TcpBbr:SetBbrState(): At 1.1993 Changing from BBR_STARTUP to BBR_DRAIN
flow 16 TcpBbr:SetBbrState(): At 1.1993 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 16 TcpBbr:SetBbrState(): At 10.7939 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 16 TcpBbr:SetBbrState(): At 11.0102 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 16 TcpBbr:SetBbrState(): At 21.0102 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 16 TcpBbr:SetBbrState(): At 21.957 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
flow 16 TcpBbr:SetBbrState(): At 32.0349 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 16 TcpBbr:SetBbrState(): At 33.0316 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
```

Figure 17. BBR state transition log in bug-fixed 3.44 BBR program.

Besides, the timing of the state transition to ProbeRTT is different among flows. In flows 9 and 15, there is twenty second gap before the second or the third transition to ProbeRTT, respectively.

As depicted in Figure 11, the condition to execute `EnterProbeRTT()` includes `m_minRttExpired` is ture and `m_idleRestart` is false. We checked the value of `m_idleRestart` when becomes `m_minRttExpired` true. Figure 18 shows the result. As shown in this figure, `m_idleRestart` sometimes becomes true, and the moment when it is true corresponds to the moment when the BBR state transition skips the transition to RrobeRTT described in Figure 17. The condition that `m_idleRestart` is true is that there are no packets in flight when a TCP sender sends a packet during a congestion avoidance phase. That means the situation that `m_idleRestart` is true will be a normal situation even if the above mentioned bug is fixed.

```
flow 1 TcpBbr:CheckProbeRTT(): At 11.6166 m_idleRestart is false in  BBR_PROBE_BW
flow 1 TcpBbr:CheckProbeRTT(): At 21.8414 m_idleRestart is false in  BBR_PROBE_BW
flow 1 TcpBbr:CheckProbeRTT(): At 33.2152 m_idleRestart is true in  BBR_PROBE_BW
flow 1 TcpBbr:CheckProbeRTT(): At 43.5469 m_idleRestart is true in  BBR_PROBE_BW

flow 2 TcpBbr:CheckProbeRTT(): At 10.639 m_idleRestart is false in  BBR_PROBE_BW
flow 2 TcpBbr:CheckProbeRTT(): At 20.9859 m_idleRestart is true in  BBR_PROBE_BW
flow 2 TcpBbr:CheckProbeRTT(): At 31.9032 m_idleRestart is true in  BBR_PROBE_BW
flow 2 TcpBbr:CheckProbeRTT(): At 42.3683 m_idleRestart is true in  BBR_PROBE_BW

flow 3 TcpBbr:CheckProbeRTT(): At 10.8866 m_idleRestart is false in  BBR_PROBE_BW
flow 3 TcpBbr:CheckProbeRTT(): At 22.0099 m_idleRestart is true in  BBR_PROBE_BW
flow 3 TcpBbr:CheckProbeRTT(): At 32.0372 m_idleRestart is true in  BBR_PROBE_BW
flow 3 TcpBbr:CheckProbeRTT(): At 42.3678 m_idleRestart is true in  BBR_PROBE_BW

flow 4 TcpBbr:CheckProbeRTT(): At 10.8754 m_idleRestart is false in  BBR_PROBE_BW
flow 4 TcpBbr:CheckProbeRTT(): At 21.092 m_idleRestart is false in  BBR_PROBE_BW
flow 4 TcpBbr:CheckProbeRTT(): At 32.0601 m_idleRestart is false in  BBR_PROBE_BW
flow 4 TcpBbr:CheckProbeRTT(): At 42.3687 m_idleRestart is true in  BBR_PROBE_BW

flow 9 TcpBbr:CheckProbeRTT(): At 10.6391 m_idleRestart is false in  BBR_PROBE_BW
flow 9 TcpBbr:CheckProbeRTT(): At 20.9822 m_idleRestart is true in  BBR_PROBE_BW
flow 9 TcpBbr:CheckProbeRTT(): At 31.0323 m_idleRestart is false in  BBR_PROBE_BW
flow 9 TcpBbr:CheckProbeRTT(): At 44.7737 m_idleRestart is true in  BBR_PROBE_BW

flow 15 TcpBbr:CheckProbeRTT(): At 10.0241 m_idleRestart is false in  BBR_PROBE_BW
flow 15 TcpBbr:CheckProbeRTT(): At 20.2643 m_idleRestart is false in  BBR_PROBE_BW
flow 15 TcpBbr:CheckProbeRTT(): At 31.2293 m_idleRestart is true in  BBR_PROBE_BW
flow 15 TcpBbr:CheckProbeRTT(): At 41.262 m_idleRestart is false in  BBR_PROBE_BW

flow 16 TcpBbr:CheckProbeRTT(): At 10.7939 m_idleRestart is false in  BBR_PROBE_BW
flow 16 TcpBbr:CheckProbeRTT(): At 21.0102 m_idleRestart is false in  BBR_PROBE_BW
flow 16 TcpBbr:CheckProbeRTT(): At 32.0349 m_idleRestart is false in  BBR_PROBE_BW
flow 16 TcpBbr:CheckProbeRTT(): At 43.6012 m_idleRestart is true in  BBR_PROBE_BW
```

Figure 18. Log of `m_idleRestart` when `m_minRttExpired` is true in bug-fixed 3.44 BBR program.

D. *Results of Bug Fixing for Ns-3 Ver.3.40 BBR Program*

We applied the bug fixing to the ns-3 version 3.40 BBR program modified according to the Approach 2 described in the previous section.

Figure 19 shows the throughput of individual flows. By comparing with the result given in Figure 9, there is a big difference between the original program and the bug-fixed program. There is no unfairness among the throughput any more, but the throughput itself is extremely low.

Figure 20 shows the BBR state transition log for flows 1

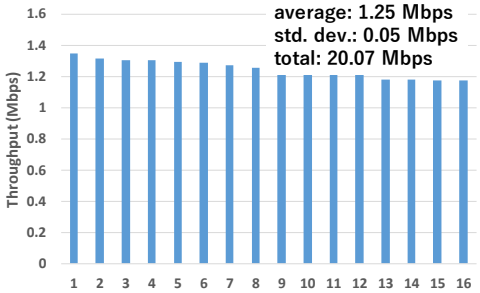


Figure 19. Throughput of individual flows using bug-fixed 3.44 BBR program ported into ns-3 version 3.40 (Approach 2).

```
flow 1 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 1 TcpBbr:SetBbrState(): At 0.1682 Changing from BBR_STARTUP to BBR_DRAIN
flow 1 TcpBbr:SetBbrState(): At 0.1815 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 1 TcpBbr:SetBbrState(): At 10.0272 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 1 TcpBbr:SetBbrState(): At 10.23 Changing from BBR_PROBE_RTT to BBR_PROBE_BW

flow 2 TcpBbr:SetBbrState(): At 0.024 Set to BBR_STARTUP
flow 2 TcpBbr:SetBbrState(): At 0.1682 Changing from BBR_STARTUP to BBR_DRAIN
flow 2 TcpBbr:SetBbrState(): At 0.1815 Changing from BBR_DRAIN to BBR_PROBE_BW
flow 2 TcpBbr:SetBbrState(): At 10.0273 Changing from BBR_PROBE_BW to BBR_PROBE_RTT
flow 2 TcpBbr:SetBbrState(): At 10.23 Changing from BBR_PROBE_RTT to BBR_PROBE_BW
…
```

Figure 20. BBR state transition log for 2 flows in Approach 2.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

and 2. In both flows, the state transition to ProbeRTT is performed only once. After the state is back to ProbeBW, there are no state transitions any more. The other fourteen flows performed the similar state transitions.

In order to find the reason for this situation, we checked the value of `m_idleRestart` when becomes `m_minRttExpired` true, similarly with the case of ns-3 version 3.44. Figure 21 shows the log of the value change of `m_idleRestart`. At the timing around 10 sec., `m_idleRestart` is false and so the state can be shifted to ProbeRTT through function `EnterProbeRTT()` as shown in Figure 11. But, at the other timings, `m_idleRestart` is true and so the state transition does not occur. This is similar with the behaviors described in the previous subsection.

However, the situation in this case is a little different. In the case of the bug-fixed 3.44 BBR program, the log in Figure 18 shows that the event that `m_minRttExpired` becomes true happens regularly with a 10 sec. interval. But, in the case of porting bug-fixed 3.44 BBR program to ns-3 version 3.40, some events that `m_minRttExpired` becomes true are skipped. In flow 1, the ProbeRTT transition check around 45 sec. is skipped, and in flow 2, that around 31 sec. is skipped.

The condition that does not call function `EnterProbeRTT()` may be that `m_minRttExpired` is false, which is derived by the condition `tcb->m_lastRtt < m_minRtt`. Figure 22 shows the log for the cases when

```
flow 1 TcpBbr:CheckProbeRTT(): At 10.0272 m_idleRestart is false in  BBR_PROBE_BW
flow 1 TcpBbr:CheckProbeRTT(): At 20.322 m_idleRestart is true in  BBR_PROBE_BW
flow 1 TcpBbr:CheckProbeRTT(): At 34.6069 m_idleRestart is true in  BBR_PROBE_BW

flow 2 TcpBbr:CheckProbeRTT(): At 10.0273 m_idleRestart is false in  BBR_PROBE_BW
flow 2 TcpBbr:CheckProbeRTT(): At 25.2809 m_idleRestart is true in  BBR_PROBE_BW
flow 2 TcpBbr:CheckProbeRTT(): At 45.5526 m_idleRestart is true in  BBR_PROBE_BW
...
```

Figure 21. Log of `m_idleRestart` when `m_minRttExpired` is true in Approach 2.

```
flow 1 TcpBbr:UpdateRTprop(): At 20.5768 tcb->m_lastRtt is 0.0240235 m_minRtt is 0.0240269
... 12 events that tcb->m_lastRtt is smaller than m_minRtt
flow 1 TcpBbr:UpdateRTprop(): At 24.2652 tcb->m_lastRtt is 0.0240041 m_minRtt is 0.0240047
flow 1 TcpBbr:UpdateRTprop(): At 34.9708 tcb->m_lastRtt is 0.024117 m_minRtt is 0.0241337
flow 1 TcpBbr:UpdateRTprop(): At 35.3346 tcb->m_lastRtt is 0.0241024 m_minRtt is 0.024117
flow 1 TcpBbr:UpdateRTprop(): At 35.6985 tcb->m_lastRtt is 0.0240896 m_minRtt is 0.0241024
flow 1 TcpBbr:UpdateRTprop(): At 36.0624 tcb->m_lastRtt is 0.0240784 m_minRtt is 0.0240896
flow 1 TcpBbr:UpdateRTprop(): At 36.4542 tcb->m_lastRtt is 0.0240686 m_minRtt is 0.0240784
flow 1 TcpBbr:UpdateRTprop(): At 42.8648 tcb->m_lastRtt is 0.0240625 m_minRtt is 0.0240686
flow 1 TcpBbr:UpdateRTprop(): At 43.2849 tcb->m_lastRtt is 0.0240547 m_minRtt is 0.0240625
flow 1 TcpBbr:UpdateRTprop(): At 43.705 tcb->m_lastRtt is 0.0240479 m_minRtt is 0.0240547
flow 1 TcpBbr:UpdateRTprop(): At 44.1251 tcb->m_lastRtt is 0.0240419 m_minRtt is 0.0240479
flow 1 TcpBbr:UpdateRTprop(): At 44.5451 tcb->m_lastRtt is 0.0240367 m_minRtt is 0.0240419
flow 1 TcpBbr:UpdateRTprop(): At 44.9652 tcb->m_lastRtt is 0.0240321 m_minRtt is 0.0240367
flow 1 TcpBbr:UpdateRTprop(): At 45.3853 tcb->m_lastRtt is 0.0240281 m_minRtt is 0.0240321
flow 1 TcpBbr:UpdateRTprop(): At 45.8054 tcb->m_lastRtt is 0.0240246 m_minRtt is 0.0240281

flow 2 TcpBbr:UpdateRTprop(): At 10.0322 tcb->m_lastRtt is 0.024003 m_minRtt is 0.0240034
... 49 events that tcb->m_lastRtt is smaller than m_minRtt
flow 2 TcpBbr:UpdateRTprop(): At 10.912 tcb->m_lastRtt is 0.024 m_minRtt is 0.024
flow 2 TcpBbr:UpdateRTprop(): At 11.2827 tcb->m_lastRtt is 0.024034 m_minRtt is 0.0240388
flow 2 TcpBbr:UpdateRTprop(): At 11.4845 tcb->m_lastRtt is 0.0240297 m_minRtt is 0.024034
flow 2 TcpBbr:UpdateRTprop(): At 11.6864 tcb->m_lastRtt is 0.024026 m_minRtt is 0.0240297
flow 2 TcpBbr:UpdateRTprop(): At 11.8882 tcb->m_lastRtt is 0.0240228 m_minRtt is 0.024026
flow 2 TcpBbr:UpdateRTprop(): At 15.1525 tcb->m_lastRtt is 0.0240223 m_minRtt is 0.0240228
flow 2 TcpBbr:UpdateRTprop(): At 25.3807 tcb->m_lastRtt is 0.0240196 m_minRtt is 0.0240223
... 36 events hat tcb->m_lastRtt is smaller than m_minRtt
flow 2 TcpBbr:UpdateRTprop(): At 35.3603 tcb->m_lastRtt is 0.0240002 m_minRtt is 0.0240002
flow 2 TcpBbr:UpdateRTprop(): At 45.9444 tcb->m_lastRtt is 0.0240902 m_minRtt is 0.0241031
... 9 events hat tcb->m_lastRtt is smaller than m_minRtt
flow 2 TcpBbr:UpdateRTprop(): At 49.9188 tcb->m_lastRtt is 0.0240237 m_minRtt is 0.0240271
```

Figure 22. Log of cases when `tcb->m_lastRtt` is smaller than `m_minRtt` in Approach 2.

`tcb->m_lastRtt` is smaller than `m_minRtt`. In the case of this experiment, there are many cases that `tcb->m_lastRtt` is smaller than `m_minRtt`. In flow 1, there are fourteen such cases from time 20.5768 sec. to time 24.2652 sec. After that, the case that `tcb->m_lastRtt` is smaller than `m_minRtt` happens at time 34.9708 sec. The time gap between 24.2652 sec. and 34.9708 sec. is larger than 10 sec., and so the event that `m_minRttExpired` becomes true happens at time 34.6069 sec. (The state transition to ProbeRTT was not performed due to the values of `m_idleRestart`.) After time 34.9708 sec., several events that `tcb->m_lastRtt` is smaller than `m_minRtt` happened contiguously, and at each of them, the value of `m_minRttStamp` was updated. As a result, `m_minRttExpired` did not become true and the attempt to shift to ProbeRTT did not occur.

As for flow 2, after the `m_minRttExpired` event happens at time 25.2809 sec., the update of `m_minRttStamp` continued thirty-eight times from time 25.3807 sec. to time 35.3603 sec. So, the `m_minRttExpired` event around 25 sec. did not happen.

Those behaviors such that `m_idleRestart` is true and that `tcb->m_lastRtt` is smaller than `m_minRtt` can be feasible, and so the results in this experiment might be reasonable. The problem is that the bug-fixed 3.44 BBR program ported into ns-3 version 3.40 presents different behavior from that in bug-fixed ns-3 version 3.44.

VI. Conclusions

This paper described some deficiencies of the TCP BBR software implemented in the ns-3 network simulator, which are detected when the transmission speed is high and heavy loads are applied to the bottleneck link. The experiment in this paper took a network configuration where sixteen senders and receivers are connected to the corresponding intermediate router through a 1 Gbps link and the intermediate routers are connected through a bottleneck 1 Gbps link. The buffer size in the bottleneck router is limited.

The first deficiency is that the TCP BBR software in different version ns-3 presents different behaviors. BBR in version 3.40 provided an unfairness in the throughput of individual flows. But BBR in version 3.44 provided a fair throughput.

We analyzed the BBR source code in 3.40 ns-3 and 3.44 ns-3, and found some differences. We modified the differences or ported 3.44 BBR code to 3.44 ns-3. But, the unfairness still remained. So far is the result of our previous conference paper [1].

Besides, we found that the BBR software in ns-3 does not perform the BBR state transition in a high speed environment [21]. In our experiment in this paper, this also happens. This is the second deficiency. In our network environment, ns-3 BBR software does not shift its state to ProbeRTT. We analyzed the ns-3 BBR source code and found a simple bug in the condition for entering ProbeRTT.

We fixed this bug and evaluated 3.44 ns-3 and 3.40 ns-3 to which 3.44 BBR program is ported. As for the bug-fixed

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

3.44 ns-3, the behaviors of sixteen BBR flows are reasonable and the state transitions to ProbeRTT are performed.

As for 3.40 ns-3 with 3.44 BBR ported, the situation was completely different. The unfairness of the throughput was resolved but the throughput itself became very low. The BBR state transitions were performed. The behavior itself might be reasonable, but the problem is that the behavior of 3.40 ns-3 with bug-fixed 3.44 BBR ported and that of bug-fixed 3.44 ns-3 are largely different. It may be said that the software in the ns-3 network simulator is not a real protocol program, and so the simulation results need to be examined carefully.

ACKNOWLEDGMENT

This work was supported by JSPS KAKENHI Grant Number JP25K15075.

REFERENCES

[1] T. Kato, T. Kato, R. Yamamoto, and S. Ohzahata, “Detailed Analysis of TCP BBR Implementation in Ns-3 Network Simulator under Heavy Traffic Loads,”The Seventeenth International Conference on Advances in Future Internet (AFIN 2025) IARIA, Oct. 2025, pp. 1-6.

[2] T. Sawada, R. Yamamoto, S. Ohzahata, and T. Kato, “Congestion Control Algorithm Estimation by Deep Recurrent Neural Network and Its Application to Web Servers on Internet,” International Journal on Advances in Networks and Services, vol. 16, no. 1&2, pp. 27-35, 2023.

[3] S. Floyd, T. Henderson, A. Gurtov, and Y. Nishida, “The NewReno Modification to TCP’s Fast Recovery Algorithm,” IETF RFC 6582, 2012.

[4] S. Floyd, “HghSpeed TCP for Large Congestion Windows,” IETF RFC 3649, 2003.

[5] S. Ha, L. Rhee, and L. Xu, “CUBIC: A New TCP-Friendly High-Speed TCP Variant,” ACM SIGOPS Operating Systems Review, vol. 42, no. 5, pp. 64-74, 2008.

[6] D. Leith and R. Shorten, “H-TCP: TCP for high-speed and long distance networks,” The Second International Workshop on Protocols for Fast Long-Distance Networks (PFLDnet 2004), Feb. 2004, pp. 1-16.

[7] L. Brakmo and L. Perterson, “TCP Vegas: End to End Congestion Avoidance on a Global Internet,” IEEE J. Sel. Areas in Commun., vol. 13, no. 8, pp. 1465-1480, 1995.

[8] C. Fu and S. Liew, “TCP Veno: TCP Enhancement for Transmission Over Wireless Access Networks,” IEEE J. Sel. Areas in Commun., vol. 21, no. 2, pp. 216-228, 2003.

[9] K. Tan, J. Song, Q. Zhang, and M. Sridharan, “A Compound TCP Approach for High-Speed and Long Distance Networks," The 25th IEEE International Conference on Computer Communicationsb (INFOCOM 2006), Apr. 2006, pp. 1-12.

[10] N. Cardwell, Y. Cheng, C. S. Gumm, S. H. Yeganeh, and V. Jacobson, V., “BBR: Congestion-Based Congestion Control,” ACM Queue, vol. 14 no. 5, pp. 20-53, 2016.

[11] M. Hock, R. Bless, and M. Zitterbart, “Experimental Evaluation of BBR Congestion Control,” The 25th IEEE International Conference on Network Protocols (ICNP 2017), Oct. 2017, pp. 1-10.

[12] K. Miyazawa, K. Sasaki, N. Oda, and S. Yamaguchi, “Cycle and Divergence of Performance on TCP BBR,” The 7th IEEE International Conference on Cloud Networking (CloudNet 2018), Oct. 2018, pp. 1-6.

[13] D. Scholz et al., “Towards a Deeper Understanding of TCP BBR Congestion Control,” IFIP Networking 2018, May 2018, pp. 1-9.

[14] S. Claypool, M. Claypool, J. Chung, and F. Li, “Sharing but not Caring – Performance of TCP BBR and TCP CUBIC at the Network Bottleneck,” The Fifteenth Advanced International Conference on Telecommunications (AICT 2019) IARIA, Jul. 2019, pp. 74-81.

[15] H. Zhang et al., “Performance Analysis of BBR Congestion Control Protocol Based on NS3,” The Seventh International Conference on Advanced Cloud and Big Data (CBD 2019), Sep. 201, pp. 363-368.

[16] G. Kim and Y. Cho, “Delay-Aware BBR Congestion Control Algorithm for RTT Fairness Improvement,” IEEE Access, vol. 8, pp. 4099-4109, 2020.

[17] K. Lanigan, “An Evaluation of BBRv2 Congestion Control Using NS-3,” Trinity College Dublin, 2020, availabe at https://publications.scss.tcd.ie/theses/diss/2020/TCD-SCSS-DISSERTATION-2020-046.pdf, accessed Jul. 2025.

[18] L. Tang, “BBR Fairness Evaluation Using NS-3,” arXiv:2410.22560v1 [cs.NI], 2024, available at https://arxiv.org/html/2410.22560v1, accessed Jul. 2025.

[19] “Mininet An Instant Virtual Network on your Laptop (or other PC),” https://mininet.org/, accessed Jul. 2025.

[20] “ns-3 Network Simulator,” https://www.nsnam.org/, accessed Jul. 2025.

[21] T. Kato, T. Kato, R. Yamamoto, and S. Ohzahata, “Does TCP BBR in Ns-3 Network Simulator Work Correctly ?” The 14th International Conference on Networks, Communication and Computing (ICNCC 2025), Dec. 2025, pp. 1-6.

[22] V. Jain, V. Mittal, ad M. P. Tahiliani, “Design and Implementation of TCP BBR in ns-3,” Workshp on ns-3 (WNS3 2018), Jun. 2018, pp. 16-22.

[23] “ns-3 document, 32.3. Fifo queue disc,” https://www.nsnam.org/docs/models/html/fifo.html, accessed Jul. 2025.

[24] “ns-3 document, 16.5.2.10. Loss Recovery Algorithm,” https://www.nsnam.org/docs/models/html/tcp.html#loss-recovery-algorithms, accessed Jul. 2025.

[25] A. Aggarwal, S. Savage, and T. Anderson, “Understanding the Performance of TCP Pacing,” The 19th IEEE International Conference on Computer Communicationsb (INFOCOM 2000), Mar. 2000, pp. 1157-1165.

[26] “ns-3: API and model change history,” https://gitlab.com/nsnam/ns-3-dev/blob/ns-3.44/CHANGES.md, accessed Jul. 2025.

[27] A. Philip and N. Cardwell, “net-tcp_bbr: v3,” https://github.com /google/bbr/blob/v3/net/ipv4/tcp_bbr.c, accessed Sep. 2025.

2026, © Copyright by authors, Published under agreement with IARIA - www.iaria.org