Trustworthiness Mechanisms for Long-Distance Networks in Internet of Things

Adrià Mallorquí Campà

http://hdl.handle.net/10803/687704

Data de defensa: 27-01-2023

UNIVERSITAT
RAMON LLULL

DOCTORAL THESIS

Title	Trustworthiness Mechanisms for Long-Distance Networks in Internet of Things
Presented by	Adrià Mallorquí Campà
Centre	La Salle International School of Commerce and Digital Economy
Department	Engineering Department
Directed by	Dr. Agustín Zaballos Diego

DECLARATION OF AUTHORSHIP

I, Adrià Mallorquí Campà, declare that this thesis entitled, "Trustworthiness Mechanisms for Long-Distance Networks in Internet of Things" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.

- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.

- Where I have consulted the published work of others, this is always clearly attributed.

- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.

- I have acknowledged all main sources of help.

- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

"When wireless is perfectly applied the whole Earth will be converted into a huge brain, which in fact it is, all things being particles of a real and rhythmic whole. We shall be able to communicate with one another instantly, irrespective of distance. […] and the instruments through which we shall be able to do his will be amazingly simple compared with our present telephone."

Nikola Tesla (1926)

ACKNOWLEDGMENTS

I would like to express my most profound appreciation to my thesis supervisor, Prof. Agustín Zaballos. His insightful comments and advice, along with the continuous and clear guidance he has provided over the last years played a decisive role in the completion of this thesis.

I want to extend my sincere thanks to all the other members of the SHETLAND-NET project. Without their support and contributions, it would have been impossible to carry on and accomplish all the project's goals. Thanks to Dr. Joan Lluís Pijoan, Dr. David Badia, Dr. Joan Navarro, Dr. Jaume Anguera, Dr. Joaquim Porté, Dr. Cristina Borda, Jordi Malé, Josep Masó, Alex Ferré, Alejandro Lerma, and Dani Serra. Special thanks to Tomás González, my companion during the Antarctic campaign, without whom it would have not been the same. This trip, with all the struggling moments and funny anecdotes we lived will always remain in our memories.

I would like to acknowledge the help of the rest of the members of the GRITS research group, Dr. Guiomar Corral, Dr. Víctor Caballero, and Julia Sánchez, for their contributions in the growth of my career. Special thanks to Dr. Alan Briones, my mentor in the early stages of my research career, who completely influenced in my interest in science. Without his guidance, I would not be writing these pages.

I'm deeply indebted to my closest family, my parents Quim and Mari, and my uncle Jordi, as well as my partner in life, Marta, who have shown unconditional support during the development of this thesis and during my life process. Without them, I would not have reached this landmark.

Finally, I would like to have a special memory for my grandparents, Jordi, Ana, Quim, and Pilar. All of them passed away during the initial stages of this thesis, in 2020, and it was very tough to continue with this work after these episodes that shaped my reality. I am sure all of them would be very proud of me if I could share this moment with them. *Moltes gràcies per tot, avis i iaios, us estimo molt.*

This research was funded by the "Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya", the European Union (EU) and the European Social Fund (ESF) [2020 FI_B 00448] [2021 FI_B1 00175] [2022 FI_B2 00026]. The research was also funded by the "Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR)" of "Generalitat de Catalunya" [2017 SGR 977], and by the Spanish Ministry on Science, Innovation and University, the Investigation State Agency and the European Regional Development Fund (ERDF) for the project "NVIS Sensor Network for The South Shetland Islands Archipelago" (SHETLAND-NET) [RTI2018-097066-B-I00 (MCIU/AEI/FEDER, UE)].

ABSTRACT

This thesis aims at achieving reliable data exchange over a harsh environment by improving its trustworthiness through the design of a multi-layered model that takes into account the different layers of trustworthiness and through the implementation of the model's associated countermeasures. The thesis focuses on the use case of the SHETLAND-NET project, aiming to deploy a hybrid Internet of Things (IoT) architecture with LoRa and Near Vertical Incidence Skywave (NVIS) communications to offer a telemetry service for permafrost monitoring in Antarctica.

To accomplish the thesis objectives, first, a review of related work in trustworthiness is carried out to propose a definition and scope of the trustworthiness term. From these, a four-layer trustworthiness model is designed, with each layer characterized by its scope, metric for trustworthiness accountability, countermeasures for trustworthiness improvement, and the interdependencies with the other layers. This model enables trustworthiness accountability and assessment of the Antarctic use case.

Given the harsh conditions and the limitations of the technology used in this use case, the model is validated and the telemetry service is evaluated through simulations in Riverbed Modeler. To obtain anticipated values of the expected trustworthiness, the proposal has been modeled and simulated to evaluate the performance with different configurations prior to its deployment in the field. The proposed architecture goes through three major iterations of trustworthiness improvement. In the first iteration, using social trust management and consensus mechanisms is explored to take advantage of sensor redundancy. In the second iteration, the use of modern transport protocols is evaluated for the Antarctic use case. The final iteration of this thesis assesses using a Delay Tolerant Network (DTN) architecture using the Bundle Protocol (BP) to improve the system's trustworthiness.

Finally, a Proof of Concept (PoC) with real hardware that was validated in the 2021-2022 Antarctic campaign is presented, describing the functional tests performed in Antarctica and Catalonia.

Keywords. IoT, WSN, trustworthiness, Antarctica, NVIS, LoRa, DTN, modeling, simulation.

RESUM

Aquesta tesi té com a objectiu aconseguir un intercanvi de dades fiable en un entorn hostil millorant-ne la confiabilitat mitjançant el disseny d'un model multi-capa que tingui en compte les diferents capes de confiabilitat i mitjançant la implementació de les contramesures associades al model. La tesi se centra en el cas d'ús del projecte SHETLAND-NET, amb l'objectiu de desplegar una arquitectura d'Internet de les coses (IoT) híbrida amb comunicacions LoRa i d'ona ionosfèrica d'incidència gairebé vertical (NVIS) per oferir un servei de telemetria per al monitoratge del "permafrost" a l'Antàrtida.

Per complir els objectius de la tesi, en primer lloc, es fa una revisió de l'estat de l'art en confiabilitat per proposar una definició i l'abast del terme de confiança. Partint d'aquí, es dissenya un model de confiabilitat de quatre capes, on cada capa es caracteritza pel seu abast, mètrica per a la quantificació de la confiabilitat, contramesures per a la millora de la confiabilitat i les interdependències amb les altres capes. Aquest model permet el mesurament i l'avaluació de la confiabilitat del cas d'ús a l'Antàrtida.

Donades les condicions hostils i les limitacions de la tecnologia utilitzada en aquest cas d'ús, es valida el model i s'avalua el servei de telemetria a través de simulacions en Riverbed Modeler. Per obtenir valors anticipats de la confiabilitat esperada, l'arquitectura proposada es modela i es simula per avaluar els resultats amb diferents configuracions previ al seu desplegament en proves de camp. L'arquitectura proposada passa per tres principals iteracions de millora de la confiabilitat. A la primera iteració, s'explora l'ús de mecanismes de consens i gestió de la confiança social per aprofitar la redundància de sensors. En la segona iteració, s'avalua l'ús de protocols de transport moderns per al cas d'ús antàrtic. L'última iteració d'aquesta tesi avalua l'ús d'una arquitectura de xarxa tolerant al retard (DTN) utilitzant el Bundle Protocol (BP) per millorar la confiabilitat del sistema.

Finalment, es presenta una prova de concepte (PoC) amb maquinari real que es va validar a la campanya antàrtica 2021-2022, descrivint les proves de camp funcionals realitzades a l'Antàrtida i Catalunya.

Paraules clau. IoT, WSN, confiabilitat, Antàrtida, NVIS, LoRa, DTN, modelatge, simulació.

RESUMEN

Esta tesis tiene como objetivo lograr un intercambio de datos confiable en un entorno hostil mejorando su confiabilidad mediante el diseño de un modelo multicapa que tenga en cuenta las diferentes capas de confiabilidad y mediante la implementación de las contramedidas asociadas al modelo. La tesis se centra en el caso de uso del proyecto SHETLAND-NET, con el objetivo de desplegar una arquitectura de Internet de las cosas (IoT) híbrida con comunicaciones LoRa y de onda ionosférica de incidencia casi vertical (NVIS) para ofrecer un servicio de telemetría para el monitoreo del "permafrost" en la Antártida.

Para cumplir con los objetivos de la tesis, en primer lugar, se realiza una revisión del estado del arte en confiabilidad para proponer una definición y alcance del término confiabilidad. Partiendo de aquí, se diseña un modelo de confiabilidad de cuatro capas, donde cada capa se caracteriza por su alcance, métrica para la cuantificación de la confiabilidad, contramedidas para la mejora de la confiabilidad y las interdependencias con las otras capas. Este modelo permite la medición y evaluación de la confiabilidad del caso de uso en la Antártida.

Dadas las condiciones hostiles y las limitaciones de la tecnología utilizada en este caso de uso, se valida el modelo y se evalúa el servicio de telemetría a través de simulaciones en Riverbed Modeler. Para obtener valores anticipados de la confiabilidad esperada, la propuesta es modelada y simulada para evaluar los resultados con diferentes configuraciones previo a su despliegue en pruebas de campo. La arquitectura propuesta pasa por tres iteraciones principales de mejora de la confiabilidad. En la primera iteración, se explora el uso de mecanismos de consenso y gestión de la confianza social para aprovechar la redundancia de sensores. En la segunda iteración, se evalúa el uso de protocolos de transporte modernos para el caso de uso antártico. La última iteración de esta tesis evalúa el uso de una arquitectura de red tolerante al retardo (DTN) utilizando el Bundle Protocol (BP) para mejorar la confiabilidad del sistema.

Finalmente, se presenta una prueba de concepto (PoC) con hardware real que se validó en la campaña antártica 2021-2022, describiendo las pruebas de campo funcionales realizadas en la Antártida y Cataluña.

Palabras clave. IoT, WSN, confiabilidad, Antártida, NVIS, LoRa, DTN, modelado, simulación.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

6LoWPAN: IPv6 over Low power Wireless Personal Area Networks.

AATP: Adaptive and Aggressive Transport Protocol.

ACK: Acknowledgment.

ALT: Active Layer Thickness.

AODV: Ad hoc On-Demand Distance Vector.

BDP: Bandwidth Delay Product.

BIC-TCP: Binary Increase Congestion control TCP.

BLE: Bluetooth Low Energy.

BP: Bundle Protocol.

BN: Byzantine Nodes.

BNT: Byzantine Node Tolerance.

BW: Bandwidth.

CALM: Circumpolar Active Layer Monitoring.

CLA: Convergence Layer Adapter.

COVID-19: Coronavirus Disease 2019.

CPS: Cyber-Physical System.

CPU: Central Processing Unit.

CR: Code Rate.

CRC: Cyclic Redundancy Check.

DIRSN: DTN for Integrated RFID Sensor Networks.

DIU: DTN Implementation Unit.

DPBFT: Delegated Practical Byzantine Fault Tolerance.

DR: Data Rate.

DSL: Digital Subscriber Line.

DTN: Delay Tolerant Network.

DUACK: Duplicate Acknowledgment.

EAATP: Enhanced Adaptive and Aggressive Transport Protocol.

FANET: Flying Ad-hoc Network.

FEC: Forward Error Correction.

FSM: Finite State Machine.

FSR: Faulty Sensing Ratio.

FSV: False Sensed Values.

GA: General Agreement.

GEO: Geostationary Orbit.

GTN-P: Ground Terrestrial Network-Permafrost.

GWSN: Global Wireless Sensor Network.

HF: High Frequency.

H-TCP: High-speed TCP.

HTTP: Hypertext Transfer Protocol.

IBR-DTN: "Institut für Betriebssysteme und Rechnerverbund" DTN.

ICN: Information-Centric Networking.

ICT: Information and Communications Technology.

IEEE: Institute of Electrical and Electronics Engineers.

IoFT: Internet of Flying Things.

ION: Interplanetary Overlay Network.

IoT. Internet of Things.

IP: Internet Protocol.

IPA: International Permafrost Association.

IPv6: Internet Protocol version 6.

ISAC: Integrated Sensing and Communication.

ISCC: IEEE Symposium on Computers and Communications.

JITEL: “Jornadas de Ingeniería Telemática”.

JSCTP: Jitter Stream Control Transmission Protocol.

LEO: Low Earth Orbit.

LFN: Long Fat Network.

LLN: Low power and Lossy Network.

LoRa: Long Range.

LoS: Line of Sight.

LPWAN: Low Power Wide Area Network.

LQSR: Link Quality Source Routing.

LTE eMTC: Long Term Evolution enhanced Machine Type Communication.

M2M: Machine-to-Machine.

MAAT: Mean Annual Air Temperature.

MAC: Medium Access Control.

MANET: Mobile Ad-hoc Network.

NASA: National Aeronautics and Space Administration.

NB-IoT: NarrowBand IoT.

NFC: Near Field Communication.

NVIS: Near Vertical Incidence Skywave.

O: Objective.

OLSR: Optimized Link State Routing.

OSI: Open Systems Interconnection.

PASR: Prediction Assisted Single-copy Routing.

PBFT: Practical Byzantine Fault Tolerance.

PCC: Performance-oriented Congestion Control.

PDR: Packet Delivery Ratio.

PDU: Protocol Data Unit.

PHY: Physical.

PoC: Proof of Concept.

PRoPHET: Probabilistic Routing Protocol using the History of Encounters and Transitivity.

QUIC: Quick UDP Internet Connections.

RAM: Random Access Memory.

RFC: Request for Comments,

RFID: Radio Frequency Identification.

RMDTN: Reliable Multicast Disruption Tolerant Networking.

RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks.

RQ: Research Question.

RTT: Round-Trip Time.

SACK: Selective Acknowledgment.

SDN: Software-Defined Networking.

SDR: Software Defined Radio.

SF: Spreading Factor.

SIoT: Social Internet of Things.

SPF: Single Point of Failure.

ST: Successful Transactions.

S-TCP: Scalable TCP.

STR: Successful Transaction Rate.

TCP: Transmission Control Protocol.

TCP BBR: TCP Bottleneck Bandwidth and Round-trip propagation time.

TSP: Thermal State of Permafrost.

TSV: Total Sensed Values.

TT: Total Transactions.

UAV: Unmanned Aerial Vehicle.

UDP: User Datagram Protocol.

USB: Universal Serial Bus.

WAN: Wide Area Network.

WSN: Wireless Sensor Network.

WWSN: Wide Wireless Sensor Network.

ZAA: Zero Annual temperature Amplitude.

Chapter 1

INTRODUCTION

1.1. Motivation and background

The Internet of Things (IoT) paradigm has been one of the major research and industrial subjects in the Information and Communications Technology (ICT) area since its conception [1]. IoT-based solutions have leveraged service and process automation in several fields such as Smart Cities [2], Smart Grids [3], Vehicular Networks [4], Industry 4.0 [5], eHealth [6], or Smart Agriculture [7], among others. IoT, alongside other ICT-related fields such as Big Data, has been a key enabler for modern scientific research and business digitalization [8]. However, not all scientific research could be modernized yet. One of the trends in current research is the study of Antarctica's ecosystem from multiple disciplines. To put some perspective, 1.18% of all indexed articles in Google Scholar from January to October 2022 are focused on Antarctica's research.

Scientists state that Antarctica, and by extension, Antarctica's research, play a key role in better understanding the Earth's past, present, and future [9]. However, given the particularities of the Antarctic continent, some challenges appear to enable modern research in this location [10]. One of these challenges is the lack of telecommunication systems, which could play a crucial role in data gathering, exchange, and processing automation for research studies.

Currently, none of the 75 Antarctic base stations is provided with Internet connectivity via cable connection (e.g., fiber optic, DSL, or coaxial) or cellular technology. Satellite communications are the only current alternative [11], which implies high costs and low bandwidth compared to the aforementioned technologies. Moreover, satellite coverage is limited to the research base facilities. However, most of the sample and data gathering is performed outside the facilities without connectivity (even in remote locations kilometers away from the nearest base), which implies that Wireless Sensor Networks (WSNs) or IoT-based solutions are exceptionally complex to implement in terms of time, human effort, and cost.

Thus, most of the data collected by sensors are stored in end devices manually (e.g., data logging through external USB drives), needing human intervention. This handicap limits the coverage area of experiments in Antarctica, given that data loggers are more challenging to physically reach as they are further away from base stations. Moreover, Antarctic campaigns are usually only held during summer due to climatic conditions. When summer ends, scientific bases close, and satellite links are disabled due to their high costs. This fact forces researchers to stop gathering data until the next campaign the following year or leave the data loggers working, but being unable to download these data until the next campaign and assuming the risk of suffering unexpected problems (e.g., sensor malfunction or battery drains) without detecting them.

For these reasons, scientists are increasingly demanding to improve and automate the data collection for research experiments in Antarctica [10], enabling a larger coverage area and a longer sample gathering period (ideally continuous monitoring). So far, experimental initiatives with ZigBee [12] and LoRa [13] communication technologies have been deployed in Antarctica to enable IoT solutions for research studies. Nonetheless, these technologies present two main drawbacks:

1. Their coverage range is limited to tens of kilometers (< 30 km) in best-case conditions. Thus, these solutions are unsuitable when devices are placed in remote locations (e.g., different islands).

2. They need Line-of-Sight (LoS) to offer reliable connectivity, which is difficult to achieve in Antarctica due to its terrain orography (e.g., two locations separated by a hill).

Aiming to modernize Antarctic research, the SHETLAND-NET project [14] from La Salle – Universitat Ramon Llull proposes to build an IoT network based on Near-Vertical Incidence Skywave (NVIS) communications in Antarctica. This communication technique operates in the High Frequency (HF) band. It consists of sending a radio wave near-vertically upwards to the ionosphere that, due to its properties, can reflect this signal back down within a circular range with a radius of 250 km [11], providing long-distance links not affected by LoS (see

Figure 1). Moreover, the NVIS nodes of the SHETLAND-NET project are designed for low power consumption, which enables its placement in remote areas with high autonomy by using batteries. Although previous research achieved promising results with NVIS transmissions, the results also showed that this type of communication struggles with adversarial conditions that causes reliability issues, including intermittent connectivity, long delays, and channel unavailability in NVIS links [11], [15]–[19]. These conditions, added to Antarctica's inherent environmental and geographic constraints, produce a harsh environment for IoT sensor networks that demands further development to accomplish the researchers' requirements.

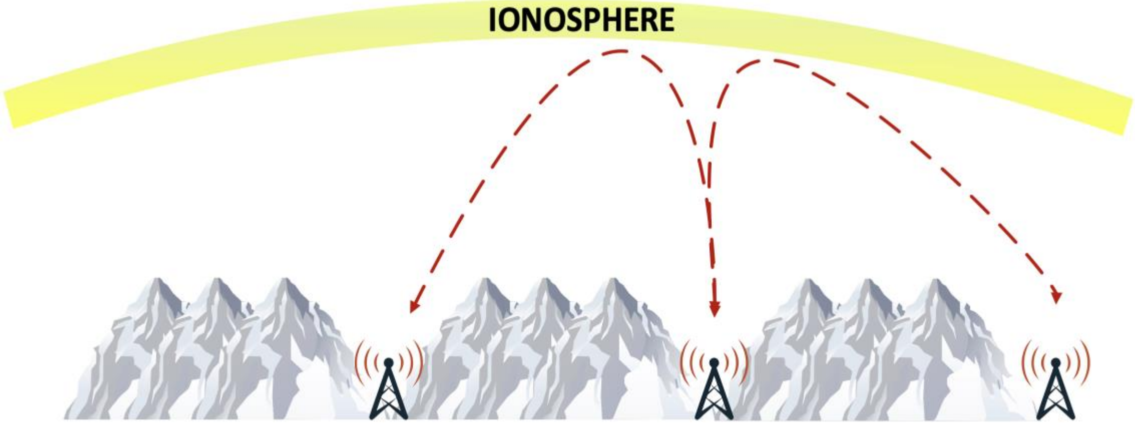


Figure 1. NVIS communication technique.

For this reason, this thesis aims to achieve reliable data exchange in harsh environments by assessing and improving their trustworthiness. Concretely, this thesis focuses on the practical use case of permafrost monitoring in Antarctica [20], in which Ground Terrestrial Network-Permafrost (GTN-P) stations collect data samples to study the ice surface of the Antarctic continent, aiming to offer a trustworthy IoT telemetry service for permafrost monitoring. These data are exchanged through the NVIS-based IoT sensor network designed in the SHETLAND-NET project.

The analysis of multi-annual permafrost thermal regimes allows determining the energy balance between the soil and the atmospheric boundary layer, which depends on climatic variability, the buffer interfacing between the soil and atmosphere, soil thermophysical properties, and the geothermal gradient. In polar regions, where vegetation cover is scarce or absent, seasonal snow is the main factor generating ground thermal insulation, a key factor for the thermal regime of permafrost [21].

The thermal regime of permafrost in Maritime Antarctica has received increasing research attention since the Fourth International Polar Year (2007–2008), following the installation of several boreholes in the Antarctic Peninsula and nearby islands [21]. Those actions are based on the protocol of the Circumpolar Active Layer Monitoring (CALM) and Thermal State of Permafrost (TSP) programs of the International Permafrost Association (IPA). These programs aimed to install a network of boreholes with adequate depths to perform direct measurements of ground temperature and thus determine inter-annual changes in the permafrost thermal regime, reaching, in many cases, the depth of Zero Annual temperature Amplitude (ZAA). The measurements of the temperature gradients from these boreholes feed into a global dataset of permafrost temperature time series (the GTN-P) to evaluate temperature variability across permafrost regions and to analyze the behavior of the Active Layer Thickness (ALT). Records of the ground temperature gradient at near-surface and deep levels of a borehole can be used to extract information about the ALT and ZAA, respectively, which allow estimating the annual heat exchange between the ground and the lower limit of the atmospheric boundary layer [21].

The study area of this use case is the Antarctic South Shetland Islands. This region has witnessed a marked rise in Mean Annual Air Temperature (MAAT) over the past 70 years and is one of the global hot spots of climate warming. MAAT increased by ~+0.56 °C/decade from 1951 to 2000, followed by a statistically significant cooling in the first decade of the 21st century [21], with the series showing a new warming trend after 2015. With MAAT in the South Shetlands at around −2 °C, the region is close to the freezing point of water, and climate change may have a profound effect on the permafrost thermal regime.

TABLE I
GTN-P STATION PARAMETERS

Environmental	**Thermal**	**Properties**	
Timestamp	Device temperature (ºC)	Ice content pack (%)	Water content 30 cm (%)
Air temperature (ºC)	Surface temperature (ºC)	Water content pack (%)	Density 30 cm (kg/m^3)
Air humidity (%)	Ground surface temperature (ºC)	Density pack (kg/m^3)	SWE 30 cm (mm)
Pyranometer up (W/m^2)	Temperature 10 cm (ºC)	SWE pack (mm)	Ice content 50 cm (%)
Pyranometer down (W/m^2)	Temperature 20 cm (ºC)	Ice content 10 cm (%)	Water content 50 cm (%)
Pyrgeometer up (W/m^2)	Temperature 40 cm (ºC)	Water content 10 cm (%)	Density 50 cm (kg/m^3)
Pyrgeometer down (W/m^2)	Temperature 80 cm (ºC)	Density 10 cm (kg/m^3)	SWE 50 cm (mm)
	Temperature 100 cm (ºC)	SWE 10 cm (mm)	Snow weight (kg/m^2)
		Ice content 30 cm (%)	Snow depth (cm)

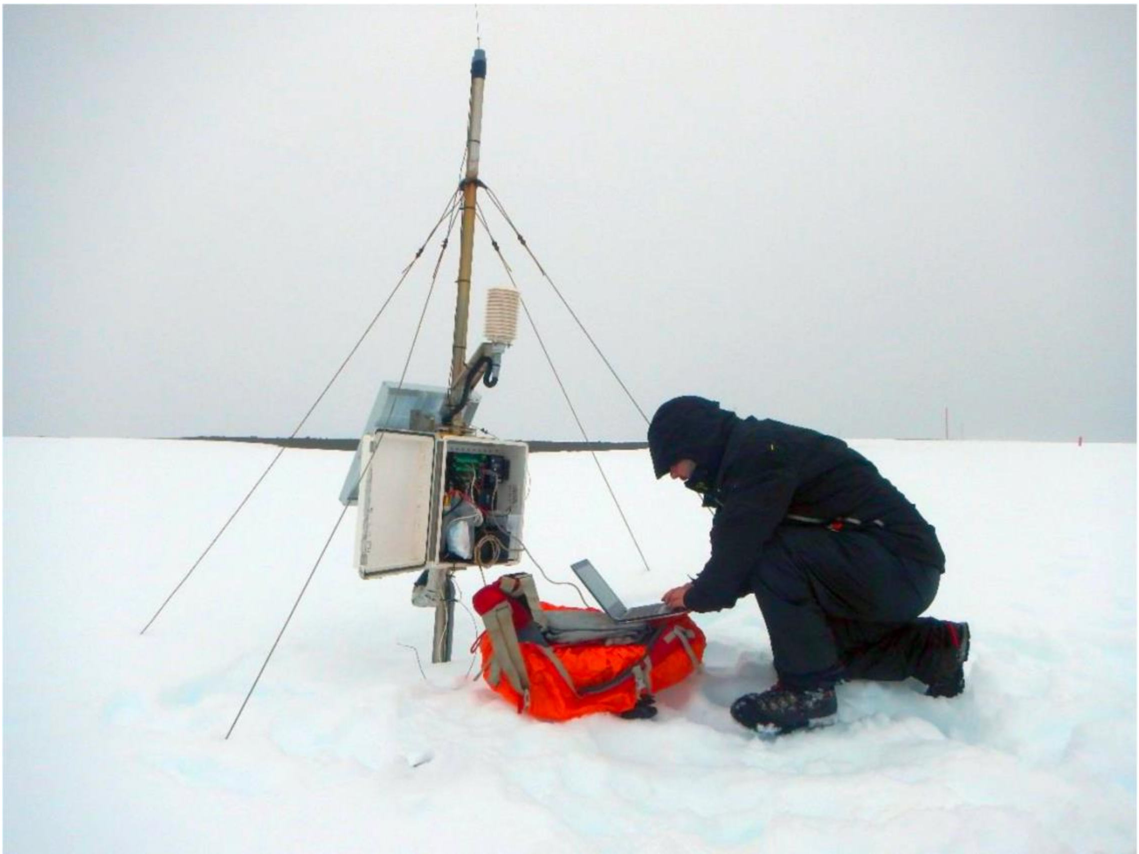


Figure 2. Manual data download from a GTN-P station in Antarctica [22].

As mentioned before, GTN-P stations are responsible for collecting temperature and other data from the permafrost surface. Each GTN-P station collects 32 different parameters and a timestamp periodically (see Table I), and each parameter is 4 bytes long. However, these data must be downloaded from each GTN-P station to a computer manually, as seen in Figure 2. The monitoring stations are operative all over the year, taking measurements and logging these data locally until the research team can retrieve them during the annual Antarctic campaign in the austral summer period. Since Antarctic campaigns are short and very conditioned by meteorology, time is gold in Antarctica. Considering the number of sensors and stations placed in the study area (currently 84), the researcher dedicates most of the time in Antarctica to collecting data and performing maintenance activities [22].

The main goal of the SHETLAND-NET project is to provide a reliable data communication solution that allows scientists to remotely access the data so that valuable time in Antarctica can be dedicated to exploring new areas and performing new research activities. The proposed NVIS-based IoT network can help researchers in several aspects. Before traveling to Antarctica, the status of the measurement sites could be known. This helps in the campaign planning, procuring in advance only those items known to be damaged that require maintenance tasks. This also contributes to decreasing the cost of the campaign and the amount of material to be transported to Antarctica. If, for any reason, the campaign is canceled or the researcher cannot travel to Antarctica, data are still available (the science of the year is not lost). Additionally, this can help to expand the study area by placing more stations, given that it would not be necessary to travel to all the stations periodically. Moreover, the sampling frequency can be increased since sensor memory is no longer a concern (data are not stored locally for a long time, they are sent through the network). Finally, once in Antarctica, the proposed solution prevents the researcher from walking long distances or going into protected areas for those measurement devices that are working correctly.

On the one hand, regarding the reliability issues that affect NVIS links, it was theorized that these could match the properties of challenging networks. Currently, challenging networks problems are coped with Delay Tolerant Network (DTN) architectures and protocols [23], such as the Bundle Protocol (BP) [24], and lightweight implementations have been proposed to leverage DTN architectures in low-resource environments, such as the IoT [25]. For this reason, this thesis explores using a DTN approach as a tool to improve the trustworthiness of the NVIS-based IoT network in Antarctica.

On the other hand, my first steps in research were mainly focused on the design, simulation, and implementation of transport layer mechanisms to optimize bandwidth utilization on data exchange over long-distance networks. In this type of network, standard TCP and other variants grow their congestion window size by one per round trip time (RTT). This made the data transport speed of TCP used in all major operating systems rather sluggish, extremely under-utilizing the network's bandwidth, especially if the length of flows is much shorter than the time TCP grows its windows to the maximum capacity of a path [26]. Aiming to solve this problem, modern TCP variants were defined that tried to occupy the entire bandwidth in a fast, stable, and fair way. From these variants, TCP CUBIC was defined as the default TCP algorithm for the Linux operating system from kernel version 2.6.18 [26], and it is still used in the current version, 5.19.9. Our research evolved towards the proposal of a novel transport protocol, the Adaptive and Aggressive Transport Protocol (AATP) [27], and its evolved version conceived for long-distance heterogeneous networks, the Enhanced AATP (EAATP) [28]. For this reason, this thesis also assesses the use of the EAATP as the transport protocol of the NVIS network, aiming to improve the trustworthiness of the permafrost telemetry service in Antarctica.

The final goal was to validate a Proof-of-Concept (PoC) of the proposed IoT architecture in the field, which was carried out during the Antarctic campaign held from December 2021 until March 2022. This PoC aimed to exchange sensor data between the Spanish Juan Carlos I Base on Livingston Island (Figure 3) and the Uruguayan Artigas Base on King George Island.

Figure 3. View of the Juan Carlos I Base in Livingston Island (South Shetland Islands, Antarctica).

These research bases are separated by approximately 93 km without LoS (see Figure 4). King George Island is expected to be provided with fiber optics in the future, interconnecting the island with the Magallanes region (Chile) through a thousand-kilometer submarine cable [29]. For this reason, the SHETLAND-NET project aims to connect Livingston Island with King George Island through NVIS, achieving permanent Internet connectivity without needing a satellite link, thanks to the fiber optic connection in King George.

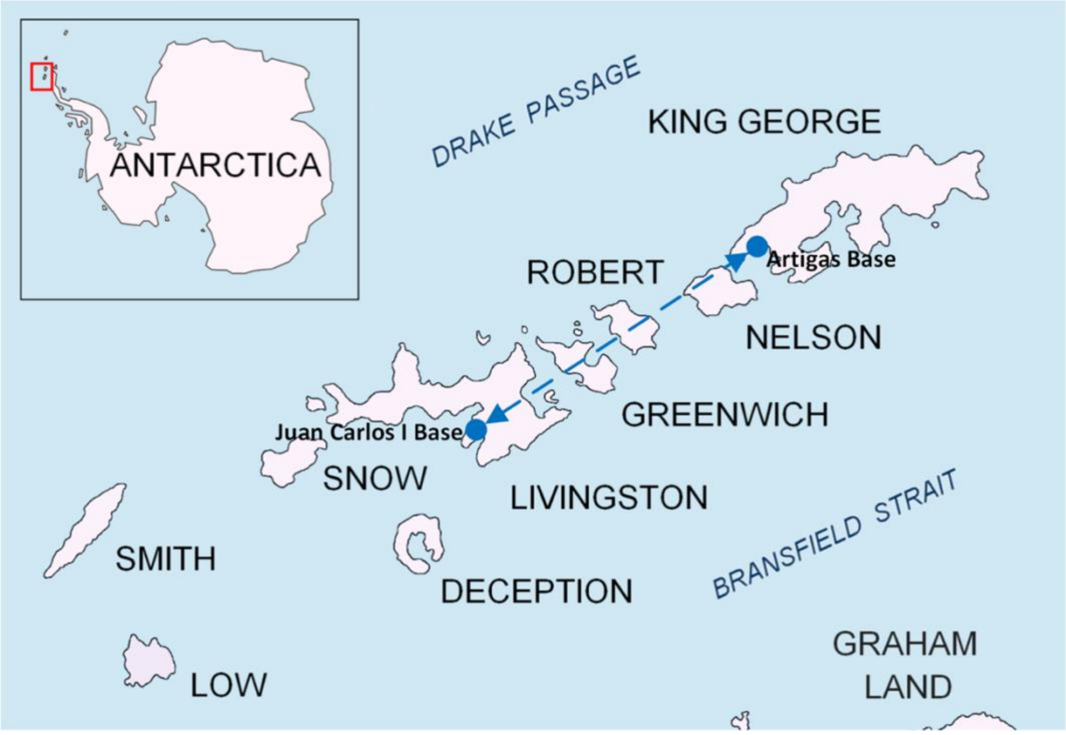


Figure 4. Juan Carlos I and Artigas bases locations in the South Shetland Islands (Antarctica).

During the Antarctic campaign, we traveled to both research bases and deployed NVIS gateways and sensors to evaluate if the PoC worked as expected in that harsh environment. In total, we stayed ten weeks at the Juan Carlos I base and six weeks at the Artigas base, living together with other researchers from multiple disciplines (see Figure 5).

Figure 5. Spanish expedition of the 2021-2022 Antarctic campaign.

1.2. Hypothesis and research questions

Considering the background described in the previous section, the hypothesis of the thesis is the following:

"Achieving reliable data exchange over a harsh environment is possible by improving its trustworthiness through the design of a multi-layered model that takes into account different facets of trustworthiness and through the implementation of the model's associated countermeasures."

That is to say, if we need to improve the trustworthiness of the data exchange over a harsh environment such as the Antarctic permafrost monitoring use case, a multi-layered trustworthiness model must be defined. This model must consider the different facets of trustworthiness (e.g., it must be able to distinguish between the trustworthiness of data and the network transmitting them) to quantify, assess, and detect the reason for unsatisfactory trustworthiness levels. This model also has to propose countermeasures to improve the trustworthiness levels at all layers. For this reason, this thesis does not focus on improving the current specification of NVIS communications by modifying the PHY or MAC layers of NVIS, nor tries to improve standard DTN protocols and architectures. This thesis focuses on using all these tools as mechanisms or countermeasures that can improve the trustworthiness of the overall data exchange service (in this case, the permafrost monitoring in Antarctica through an NVIS-based IoT network) based on the designed trustworthiness model.

To reach this goal and confirm the hypothesis, the following research questions are posed:

RQ1. What is the definition and scope of the trustworthiness term in the field of Cyber Physical Systems (CPS)?

A CPS is defined as a system with integrated computational and physical capabilities [30]. IoT and WSNs are common examples of CPS. In the literature, we can find many interpretations and points of view regarding the trustworthiness term in this field. To improve the trustworthiness of the SHETLAND-NET's service, first, it is necessary to delimit it with an agreed definition and scope.

RQ2. How can trustworthiness be measured?

After setting the scope of the trustworthiness term, it is necessary to establish how to quantify the achieved level of trustworthiness. Without accountability on trustworthiness, it would be harder to compare different solutions to determine which one offers better trustworthiness.

RQ3. How can a model be used to foresee, assess, and improve the achieved trustworthiness in harsh environments?

Once accountability on trustworthiness is achieved, it is necessary to define how the trustworthiness model can be used to identify the weaknesses of the assessed system and how they can be improved by applying the appropriate countermeasures. Aiming to this, a model must be defined, and its goodness must be validated.

RQ4. How does using modern transport protocols affect the achieved trustworthiness in adversarial networks?

In prior research, it was evidenced that modern transport protocols, such as TCP CUBIC or EAATP, could improve the performance of data transmission over long-distance networks in which classical TCP struggles to perform well. For this reason, this thesis also aims to assess how transport protocols affect the trustworthiness of adversarial data networks, evaluating if modern transport protocols improve the achieved trustworthiness and how it can be taken advantage of.

RQ5. How does a DTN architecture affect the trustworthiness of adversarial networks?

Given that the literature shows that reliability problems regarding long delays, lack of end-to-end connectivity, and intermittent disconnections can be coped with DTN-based solutions, this thesis also aims to study the effect of modeling and deploying a DTN architecture into harsh environments with adversarial networks to achieve how the DTN affects to the overall achieved trustworthiness.

RQ6. Are the reached levels of trustworthiness in real implementations match the ones derived from the assessment through the trustworthiness model?

Once a use case and its architecture have been evaluated through the trustworthiness model and an expected level of trustworthiness is derived from it, it is necessary to validate if the physical implementation of the assessed use case in the real world matches the expected levels of trustworthiness in order to confirm the goodness of the model.

1.3. Thesis objectives

This section describes the objectives of the thesis. Since this thesis is strongly related to the SHETLAND-NET project, the thesis' objectives are focused on the project's permafrost telemetry use case. Five objectives are set to answer the six research questions exposed in the previous section. O1, O2, and O3 are directly related to RQ1, RQ2, and RQ3, respectively. O4 is set to answer RQ4 and RQ5. Finally, O5 is set to answer RQ6. The description of each objective is detailed below:

O1. Establish a scope for the trustworthiness term.

We can find diverse approaches to trustworthiness from different points of view in the field of CPS. This objective aims to perform research on the related work in trustworthiness in CPS, classifying the reviewed works based on their approach to trustworthiness. After this review, a general definition of trustworthiness and its scope for this thesis needs to be set.

O2. Define a model that enables trustworthiness accountability to detect weaknesses and propose appropriate countermeasures for trustworthiness improvement.

After limiting the scope and defining the meaning of trustworthiness, this objective aims to propose a model that enables quantifying the trustworthiness level for a given service architecture. Moreover, the model must also help to identify the weak points of a given architecture to propose accurate and appropriate countermeasures to improve its trustworthiness.

O3. Assess the trustworthiness of the Antarctic use case.

As mentioned in section 1.1, Antarctic campaigns are short in time, usually only last a few months, and typically research projects only have one or two campaigns to perform their experiments. For this reason, before deploying the physical system in the field during the Antarctic campaign, the project needs to foresee the expected trustworthiness to assess if the current proposal performs good enough and, therefore, be able to propose and analyze new solutions if needed, to guarantee the desired trustworthiness level during the campaign. This objective aims to use the trustworthiness model from O2 and simulation tools to assess the expected trustworthiness level of the Antarctic use case before its deployment in the field during the campaign and, thus, determine the best architecture to be deployed during the time-constrained campaign. The Riverbed Modeler Simulator [31] was chosen to perform the tests to assess the trustworthiness of the Antarctic permafrost telemetry use case, given its modeling and simulating powerful capabilities and the personal knowledge and experience with this simulation tool from previous research [27], [28].

O4. Propose new mechanisms to improve the trustworthiness of the Antarctic use case.

This objective aims to improve the trustworthiness of the initial service architecture proposed by the SHETLAND-NET project to achieve a reliable service architecture for the Antarctic telemetry use case. To improve it, the use of modern transport protocols and DTNs is evaluated (derived from RQ4 and RQ5), as well as consensus and social trust management mechanisms that can take advantage of sensor redundancy. All these proposals are evaluated through the assessment tests of O3. From these assessments, a reliable service architecture should be proposed for the validation tests in the field.

O5. Deploy a PoC of the proposed service architecture in the field during the 2021-2022 Antarctic campaign.

The final proposal for the SHETLAND-NET's IoT network to provide the permafrost telemetry service should be validated as a PoC in the field during the Antarctic campaign 2021-2022. The results from the tests with real devices should match and confirm the expected trustworthiness level extracted from the simulation tests to validate their goodness. From this deployment, conclusions and future development lines should be extracted to deploy a permanent IoT network in a future campaign.

1.4. Related work

This section summarizes the related work that has been reviewed throughout the thesis. Three main categories were researched for this work: trustworthiness in CPS, transport protocols, and DTNs. The following subsections synthesize the related work in these categories. Further details can be found in the related work sections in Chapter 2, Chapter 3, and Chapter 4.

1.4.1. Trustworthiness in Cyber Physical Systems

A CPS is defined as a system with integrated computational and physical capabilities, such as smart vehicles, WSNs, Industry 4.0 devices, smart grids, and the IoT [30]. In general terms, it is generally agreed to define the trustworthiness of CPS as the property of behaving as expected under adversarial conditions [32]. Nonetheless, many kinds of adversarial conditions can affect a system's trustworthiness, such as faulty nodes, byzantine errors, malicious behaviors, and network malfunction, among others [33]. For this reason, many different approaches are proposed in the literature to measure or provide trustworthiness that refer to disparate facets of a CPS. In this thesis, it is proposed to classify them into the following four categories:

1. Data Trustworthiness: it is defined as the possibility to ascertain the correctness of the data provided by the source [34]. Data trustworthiness studies mainly focus on faulty-node detection and false data detection and correction [33]. For instance, authors in [35] present a distributed Bayesian algorithm to detect faulty nodes, while authors in [36] use a fog computing architecture to detect, filter, and correct abnormal sensed data. Also, the authors in [37] present a Data Intrusion Detection System to trigger false data from malicious attacks.

2. Network trustworthiness: it can be defined as the probability for a packet to reach its destination unaltered despite the adversities (e.g., link failure, link saturation, or malicious attacks, among others) [38]. Network trustworthiness and performance have been studied from different approaches, such as load balancing and redundancy protocols [39], transport protocols [40], dynamic routing and topology control protocols [37,38], cybersecurity mechanisms [43], and Delay Tolerant Network (DTN) architectures and protocols [25]. Studies using modern transport protocols, such as TCP CUBIC [26] and EAATP [28], evidenced an improvement in bandwidth utilization and packet loss prevention over long-distance networks. Moreover, DTN protocols, such as the Bundle Protocol [25], also proved to mitigate the adversarial conditions from challenging networks. These technologies seemed to cope with similar challenges to the Antarctic use case, so transport and DTN protocols were further studied in this thesis.

3. Social Trustworthiness: this facet of trustworthiness became a trend with the emergence of the Social Internet of Things (SIoT) [44]. This area studies the capability of objects to establish autonomous social relationships to define better trust and reputation models that introduce several input parameters not considered before. Subjective models (each sensor builds its own vision of trust in the system) [45] are proposed to improve data gathering and service delivery. These models consider factors such as the computational capabilities of the nodes, the type of relationship between them, the total number of transactions, the credibility of a node, and the feedback provided by other nodes, among others. Similarly, objective models have also been proposed [46], where all nodes share the same vision of the system's trust, which has the advantage of achieving faster convergence times but is less prone to detect nodes that behave maliciously only to selected targets. Other models [47] propose to base trust computation on input parameters such as the expected gain on success, the expected damage on a failure, the expected cost, the expected result, and the goal. Also, a decentralized self-enforcing trust management system [48] is proposed for multiparty systems based on a feedback system and reputational secure multiparty calculations to ensure the privacy of each party's provided data.

4. Consensus: trustworthiness through consensus aims to reach a state where all participants of the same distributed system agree on the same data values [49]. In general, two main blocks of consensus algorithms and protocols have been studied: Proof-based consensus and byzantine consensus. The first group is mainly used as the consensus algorithms for blockchain technologies, where all participants compete against each other to mine a block. The most well-known protocols in this category are Proof-of-Work, Proof-of-Stake, and its variants [50]. The main drawback of these algorithms for low-resource devices is that their simpler hardware specifications and low processing power cause poor performance of the mining tasks [51]. The second group (byzantine fault consensus) is based on collaborative algorithms. These kinds of algorithms implement voting-based mechanisms to reach an agreement, which generally imply less resource consumption. The drawback of these mechanisms is the number of messages that need to be delivered through the network to reach an

agreement. The most well-known protocols in this category are Practical Byzantine Fault Tolerance, RAFT, PaXoS, and Ripple [50].

1.4.2. Transport protocols

Transport protocols have been under continuous development to improve data transmission performance since the early stages of the Internet [27]. The first extensions of the original TCP were the fast retransmit, fast recovery, packet pair link estimation, duplicated acknowledgment (DUACK), and selective acknowledgment (SACK) mechanisms. These mechanisms were introduced in new TCP flavors [52]: TCP Tahoe, TCP Reno, TCP New-Reno, TCP SACK, and TCP-Vegas.

Nonetheless, these flavors, currently considered legacy [26], could not perform well over long-distance networks, especially in the so-called Long Fat Networks (LFNs). The LFN concept and its effects on TCP performance are currently described in the Request For Comments (RFC) 7323. Some TCP variants and other transport protocols developed during the last decade were focused on improving their performance over LFNs [27]. Some of these are Scalable TCP (S-TCP) [53], FAST TCP [54], High-Speed TCP (H-TCP) [55], Binary Increase Control TCP (BIC-TCP) [56], and its evolution: TCP CUBIC [26]. TCP CUBIC (RFC 8312) is currently the most used TCP flavor, given that it is the TCP version used by default on most operating systems.

Currently, a new transport protocol called Quick UDP Internet Connections (QUIC) [57] is in the final stages of its completion and is starting to be used alongside the new Hypertext Transfer Protocol (HTTP) version, HTTP/3. However, various parties still implement QUIC with their own source code, competing to be established as the standard implementation of QUIC in the operating systems, which causes a large heterogeneity that provokes uncovered bugs and compatibility problems [57]. In addition, other modern transport protocols, such as TCP BBR [58], Copa [59], Indigo [60], and Verus [61], can achieve high performance, as validated in several tests performed in Stanford University's testbed called Pantheon [60]. TCP BBR is one of the top-performance protocols, managing the maximum bandwidth with the minimum RTT.

Nevertheless, most of these protocols consider that packet loss always occurs due to network congestion, reducing the congestion window. This assumption is false for wireless links, where random packet losses can also occur due to other reasons, such as fading or channel interference [28]. Under these circumstances, reducing the congestion windows might also degrade the transmission performance, achieving lower throughput [28]. For this reason, other transport protocols focus on implementing mechanisms to differentiate between network congestion losses and random losses, such as Performance-oriented Congestion Control (PCC) [62], TCP Veno [63], TCP Westwood+ [64], Dynamic TCP [65], Jitter TCP [66], and Jitter Stream Control Transmission Protocol (JSCTP) [67]. These protocols reduce their congestion window only in the event of packet drops due to congestion, improving their performance over wireless and heterogeneous links [28].

Moreover, given that the protocols mentioned before did not meet the performance requirements of our cloud data-sharing use case from previous work [28], in previous work we presented the Adaptive and Aggressive Transport Protocol (AATP) [27] and its evolution, the Enhanced AATP (EAATP) [28], which incorporates mechanisms to differentiate the packet losses' cause, fairly adapting its sending rate accordingly to the network circumstances.

1.4.3. Delay Tolerant Networks

A "challenging network" was defined as a network characterized by high latency, bandwidth limitations, error probability, node longevity, or path instability [23]. In this type of network, typical TCP/IP applications cannot perform reliably because they assume that end-to-end connectivity is persistent and stable. To cope with these problems, an architecture called DTN was proposed [23], [68]. This architecture is mainly based on a store-carry-and-forward philosophy, in which end-to-end connectivity is not needed, and intermediate nodes can keep custody of exchanged data for long periods [69].

The proposed DTN architecture in [23], [68] evolved into a DTN protocol defined in the RFC 5050, the BP [24], which aims to build an overlay network composed of the nodes capable of keeping the custody of data, called bundles (see Figure 6). Interplanetary networks were the first application for DTNs and BP [70]. Nonetheless, the use of DTNs rapidly grew in other fields, such as underwater networks, wildlife tracking networks, sparse WSNs, vehicular networks, and military networks, among others [69].

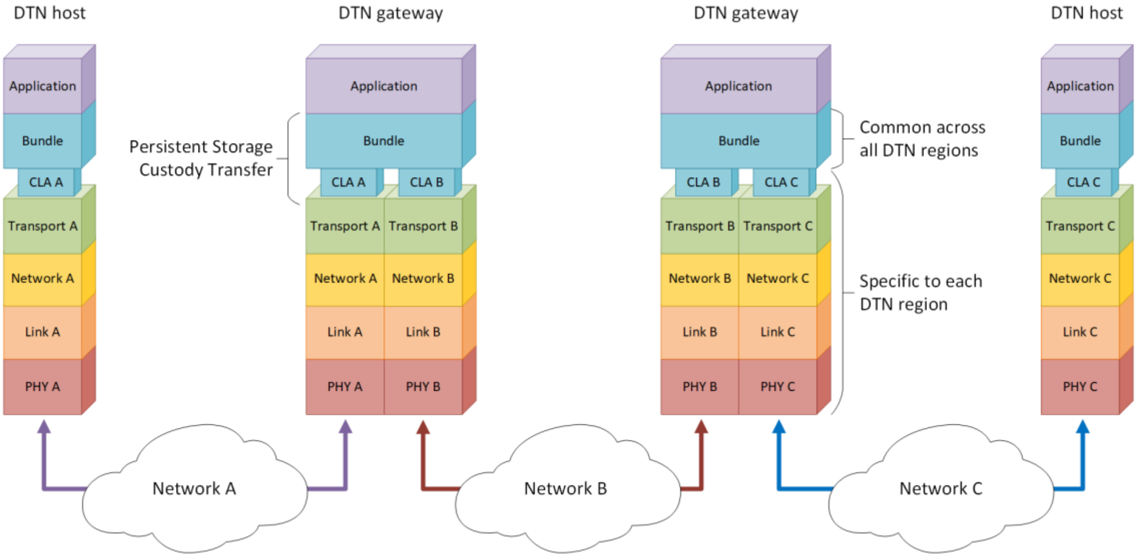


Figure 6. DTN architecture with Bundle Protocol.

Underwater networks use the DTN paradigm to cope with the problems caused by intermittent connectivity, mobility, sparse deployment, high propagation delay, high transmission cost, low asymmetric data rate, and poor transmission reliability (due to positioning inaccuracy and high attenuation). DTNs enable applications for oceanographic data collection, pollution monitoring, offshore exploration, disaster prevention, assisted navigation, and tactical surveillance applications [71], [72].

Wildlife tracking networks, designed for biology research, may consider a DTN approach to face the problems resulting from intermittent connectivity, mobility, sparsity, energy constraints, large end-to-end delay, and asymmetric data rate. These networks allow monitoring of the long-term behaviors of wild animals sparsely distributed over a large area [73], [74].

Sparse WSNs (e.g., space, terrestrial, and airborne) can also apply DTN technology to deal with the problems caused by intermittent connectivity, sparse deployment, limited power (and also limited memory and CPU capability), and low and asymmetric data rate. These networks are usually employed to monitor science and hazard events, like earthquakes, volcanoes, flooding, forest fire, sea ice formation and breakup, lake freezing and thawing, and environmental monitoring [75], [76].

For vehicular networks, store-carry-and-forward DTN concepts have been proposed for maintaining communications in sparsely connected environments, enabling the development of emerging vehicular applications such as, but not limited to, road safety, traffic monitoring, driving assistance, and infotainment. Transmissions over these networks are subject to frequent and unpredictable disconnection caused by a dynamic topology, high-speed mobility, variable node density, short contact durations, limited transmission ranges, radio obstacles, and interferences [77], [78].

In the military field, integrating DTN concepts into military tactical networks can ease communications in hostile environments (battlefields) where network infrastructure is unavailable. These networks suffer from problems of high intermittent connectivity, mobility, destruction, noise, attack, interference, low transmission reliability (due to position inaccuracy and limited visibility), and low data rate [79], [80].

Although the RFC 5050 defines BP operation and architecture, there is no standard implementation. However, popular BP implementations are in use nowadays [25]. For instance, the Interplanetary Overlay Network (ION) is NASA's implementation for outer space communications. DTN2 is also a widespread open-source implementation. BP version 7 also has its implementation, DTN7. Moreover, aiming to enable DTN solutions for IoT applications, many lightweight implementations of BP have emerged during the last few years [25], such as IBR-DTN, µDTN, µD3TN, and NanoDTN, among others.

1.5. Roadmap, methodology, and contributions

Once enrolled in the doctorate program, I continued my research in transport protocol mechanisms to optimize bandwidth usage in bulk data transfers over long-distance networks, linked to the project VSNoIPv6. This research led to publications [27], [28] (see Figure 7 and Table II), in which I was involved in the protocol design and results analysis stages, and was in charge of the modeling, simulation, test definition, and data curation. In [27], we presented

AATP, a transport protocol conceived for long-distance wired networks. The protocol's performance was validated in simulation tests with Riverbed Modeler and in a physical testbed with a Wide Area Network (WAN) emulator called WANem [81]. Results showed that AATP could outperform TCP and UDP in terms of bandwidth usage due to its precise bandwidth estimation and congestion control mechanisms. However, results also evidenced fairness conflicts when various AATP flows competed for the same bandwidth, compromising the trustworthiness of this transport protocol. In [28], we introduced a fairness mechanism to fix this problem, and we also modified the other protocol's mechanisms to adapt them to wired and wireless networks. The new version of the protocol was called EAATP, and its performance was compared with other modern transport protocols and the newest TCP variants with the same tests as the Pantheon platform [60]. Results showed that EAATP could achieve better throughput and Packet Delivery Ratio (PDR) than its competitors while fixing the fairness issues from its predecessor, improving its reliability.

This research track made me realize that even though we were improving the performance and trustworthiness of the bulk data exchange use case of the VSNoIPv6 project, we were approaching the trustworthiness improvement task from a network perspective only. To improve the bandwidth usage (the use case requirement), it was proposed to use a new congestion control mechanism (countermeasure), and the performance of the proposed solutions was assessed with quantitative metrics to choose the most appropriate (accountability). However, we did not approach this use case from other points of view, such as the data trustworthiness, and maybe we could have improved the overall data exchange service trustworthiness even more if we had done it. This was the initial spark of this thesis, thinking that it could be possible to improve the trustworthiness of other cases from different perspectives, approaching all of them together if an appropriate trustworthiness model was used.

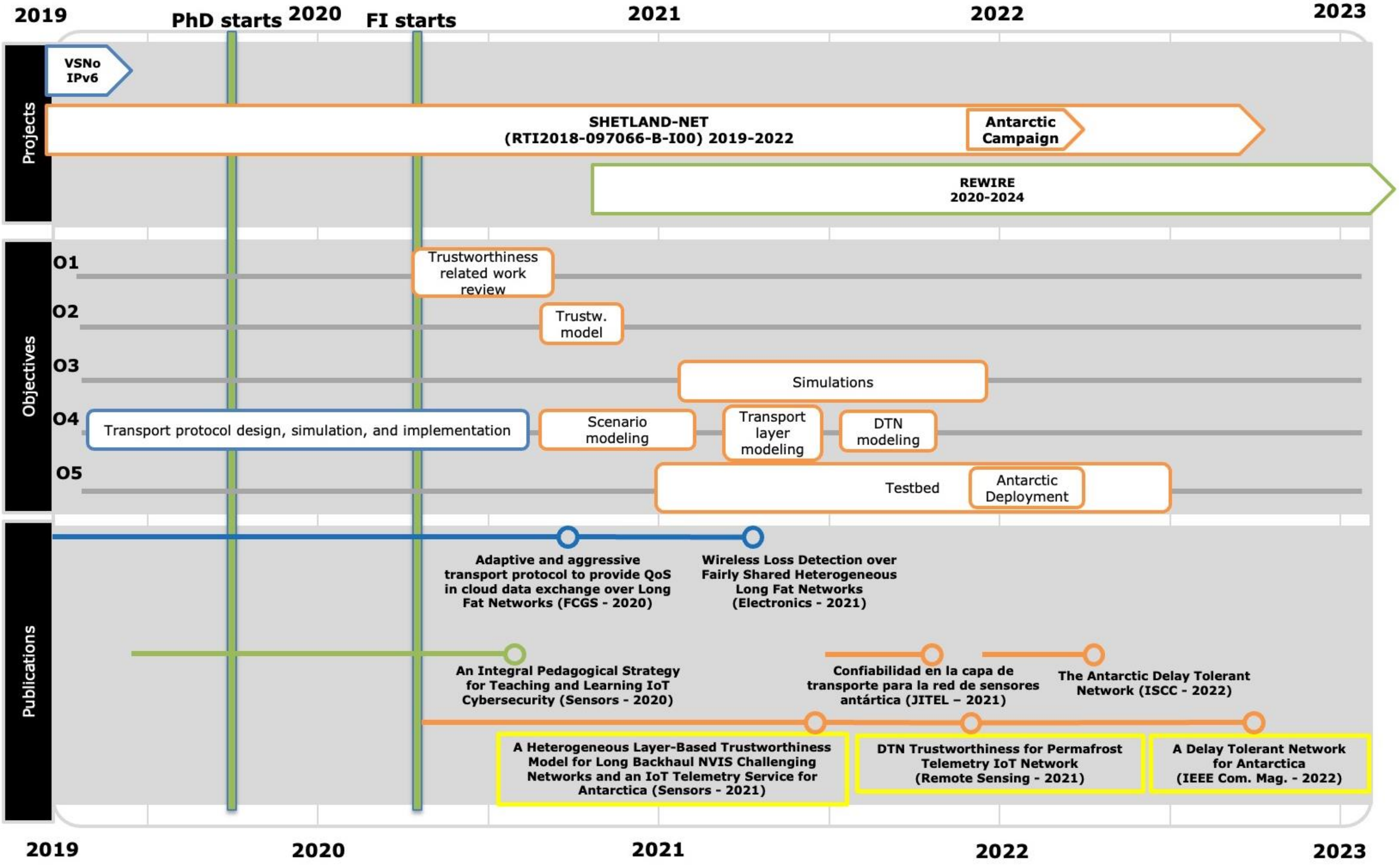

2019
PhD starts
2020
FI starts
2021
2022
2023
Projects
VSNo IPv6
SHETLAND-NET (RTI2018-097066-B-I00) 2019-2022
Antarctic Campaign
REWIRE 2020-2024
Objectives
O1
O2
O3
O4
O5
Trustworthiness related work review
Trustw. model
Simulations
Transport protocol design, simulation, and implementation
Scenario modeling
Transport layer modeling
DTN modeling
Testbed
Antarctic Deployment
Publications
Adaptive and aggressive transport protocol to provide QoS in cloud data exchange over Long Fat Networks (FCGS - 2020)
Wireless Loss Detection over Fairly Shared Heterogeneous Long Fat Networks (Electronics - 2021)
An Integral Pedagogical Strategy for Teaching and Learning IoT Cybersecurity (Sensors - 2020)
Confiabilidad en la capa de transporte para la red de sensores antártica (JITEL – 2021)
The Antarctic Delay Tolerant Network (ISCC - 2022)
A Heterogeneous Layer-Based Trustworthiness Model for Long Backhaul NVIS Challenging Networks and an IoT Telemetry Service for Antarctica (Sensors - 2021)
DTN Trustworthiness for Permafrost Telemetry IoT Network (Remote Sensing - 2021)
A Delay Tolerant Network for Antarctica (IEEE Com. Mag. - 2022)
2019
2020
2021
2022
2023

Figure 7. Thesis roadmap

TABLE II
THESIS CONTRIBUTIONS

Authors	**Title**	**Journal**	**Quartile**	**Reference**	**Contributions**	**Used for compendium**
J. Sánchez, A. Mallorquí, A. Briones, A. Zaballos, G. Corral	An Integral Pedagogical Strategy for Teaching and Learning IoT Cybersecurity	Sensors (MDPI)	Q1	[82]	Methodology description, data curation and results analysis of the Networking Laboratory subject.	-
A. Briones, A. Mallorquí, A. Zaballos, R. Martin de Pozuelo	Adaptive and Aggressive Transport Protocol to Provide QoS in Cloud Data Exchange over Long Fat Networks	Future Generation Computer Systems (Elsevier)	Q1	[27]	Participated in protocol design and results analysis. In charge of modeling, simulation, test definition, and data curation.	-
A. Briones, A. Mallorquí, A. Zaballos, R. Martin de Pozuelo	Wireless Loss Detection Over Fairly Shared Heterogeneous Long Fat Networks	Electronics (MDPI)	Q3	[28]	Participated in protocol design and results analysis. In charge of modeling, simulation, test definition, and data curation.	-
A. Mallorquí, A. Zaballos	A Heterogeneous Layer-Based Trustworthiness Model for Long Backhaul NVIS Challenging Networks and an IoT Telemetry Service for Antarctica	Sensors (MDPI)	Q2	[83]	In charge of article writing, related work review, trustworthiness model definition, architecture proposal, use case modeling, simulations, tests definition, data curation, results analysis.	√
A. Mallorquí, A. Zaballos, A. Briones, G. Corral	Confiabilidad en la Capa de Transporte para la Red de Sensores Antártica	JITEL 21 (National Conference)	N/A	[84]	In charge of article writing, related work, architecture proposal, use case modeling, simulation, tests definition, data curation, results analysis.	-
A. Mallorquí, A. Zaballos, A. Briones	DTN Trustworthiness for Permafrost Telemetry IoT Network	Remote Sensing (MDPI)	Q1	[85]	In charge of article writing, related work review, architecture proposal, use case modeling, simulation, tests definition, data curation, results analysis.	√
A. Mallorquí, A. Zaballos, D. Serra	The Antarctic Delay Tolerant Network	ISCC 2022 (International Conference)	N/A	[86]	In charge of article writing, related work review, architecture proposal, use case modeling, simulation, tests definition, data curation, results analysis. Supervision of the BP implementation.	-
A. Mallorquí, A. Zaballos, D. Serra	A Delay Tolerant Network for Antarctica	IEEE Communications Magazine (IEEE)	Q1	[87]	In charge of article writing, related work review, architecture proposal, use case modeling, simulation, tests definition, data curation, results analysis. Supervision of the BP implementation. Participated in the testbed deployment.	√

In parallel, I also participated in analyzing and divulgating the pedagogical strategy followed in La Salle Universitat Ramon Llull to teach IoT cybersecurity [82]. In this work, I contributed by describing the methodology we follow in the subjects taught in the bachelor's degrees in IT engineering and by curating and analyzing the grades and their evolution over several academic courses. The knowledge in the cybersecurity field and the experience with other Erasmus+ projects ([88], [89]) led to the start of a new Erasmus+ project, REWIRE [90], which is currently ongoing. REWIRE aims to build a blueprint for the cybersecurity industry and a concrete European Cybersecurity Skills Strategy. Its work focuses on delivering concrete recommendations and sustainable solutions that will lead to the reduction of skill gaps between industry requirements and sectoral training provision and contribute to the growth, innovation, and competitiveness of the cybersecurity sector.

From this research track, it was extracted that cybersecurity could be seen as one of the perspectives (i.e., layers) of the trustworthiness model. Cybersecurity measures (e.g., authentication, encryption) can leverage data confidentiality, integrity, availability, authenticity, or non-repudiation and thus improve the trustworthiness of a system or service. However, this thesis considers that the cybersecurity layer of trustworthiness should only be analyzed when the use case entails a concern about cybersecurity. Given that the Antarctic permafrost monitoring service is to be deployed in isolated and remote locations, with a highly improbable interference of malicious users or third parties, the cybersecurity track was left out of the scope of this thesis.

In 2020, I was granted a predoctoral contract called FI by the European Social Fund and the Generalitat de Catalunya. With this contract, I started participating in the SHETLAND-NET project, the cornerstone of this thesis. As mentioned before, the project proposed an NVIS-based IoT network to automatize data collection and exchange for Antarctic research. However, given the performance issues of NVIS communications, improvements had to be made to offer a reliable service for other research projects. While other colleagues focused on improving NVIS PHY and MAC layers' robustness, my contribution was to propose an integrated architecture that could improve the service's trustworthiness with the current NVIS specification, focusing on the use case of a permafrost monitoring service [20], [21], which led to the setting of the objectives described in the previous section.

To accomplish O1, firstly, the related work in trustworthiness was reviewed to build a picture of how the trustworthiness term is interpreted and managed from different points of view. This review of the related work let build an own vision of what trustworthiness is, trying to bind the different perspectives studied in the literature. Starting from this point, a trustworthiness model composed of four layers was proposed (O2), each centered on a different field of the trustworthiness perspective, with its own metric to enable trustworthiness quantification, its countermeasures to improve it, and the interdependencies between the layers.

After defining the model, the first version of a hybrid IoT network architecture that used NVIS and LoRa as communication technologies to give service to the permafrost monitoring process was proposed. To assess the trustworthiness of this proposed architecture before deploying it in the Antarctic campaign (O3), the use case was modeled into the Riverbed Modeler simulator to perform several tests. This simulator was chosen due to the personal experience working with it from previous research [27], [28].

Aiming to improve the trustworthiness of the permafrost monitoring service (O4) until it reached the requirements to perform reliably, the processes of proposing an architecture with possible improvements, modeling it into the simulator, performing the tests, and evaluating the results went through three major iterations. In the first iteration, it was introduced to add consensus and social trust management mechanisms to take advantage of sensor redundancy when it is available and deployable, given that these mechanisms proved to improve the reliability of acquired data in other use cases [46], [50]. Moreover, it was also proposed to use an opportunistic network scheme to send all lost data from nighttime (when NVIS links are unavailable) as a bulk data transfer when the network becomes available again in the daytime. The results and conclusions from this first iteration, along with the related work review on trustworthiness, the trustworthiness model definition, and the use case modeling, were presented in a journal publication [83]. This paper is the first of this compendium thesis, and I was in charge of the article writing, the state-of-the-art review, the trustworthiness model definition, the architecture proposal, the use case modeling, the simulations, the tests definition, the data curation, and the analysis of the results.

In the second iteration of O4, EAATP was proposed to be introduced into the architecture as the transport protocol of the NVIS long-distance links, aiming to improve bandwidth usage and minimize packet loss in congestion situations (e.g., the bulk data transfer moment). EAATP's performance was compared with other modern transport protocols. A preliminary paper was published at a national conference [84], which evolved into an extended journal article [85]. It reviews the trustworthiness model, the use case modeling into the simulator, and the proposed architecture. It also presents the results and conclusions from this second round of tests. This paper is the second of the compendium thesis. In both papers, I was in charge of the article writing, the state-of-the-art review, the new architecture proposal, the use case modeling, the simulations, the tests definition, the data curation, and the analysis of the results.

The third and last iteration of proposed improvements consisted in replacing the opportunistic networking scheme that could provoke network congestion and packet losses. Instead, a DTN architecture would be implemented using BP as the DTN protocol. This solution would help us mitigate predicted losses (nighttime unavailability of NVIS) and unpredicted losses (long delays and disruptions during daytime due to the ionosphere's variating properties). The results from these tests were published in an international conference paper [86]. In this paper, I was in charge of the article writing, the state-of-the-art review, the new architecture proposal, the use case modeling, the simulations, the tests definition, the data curation, and the analysis of the results, and I supervised the BP implementation in the hardware to be used in Antarctica.

The architecture resulting from these iterations achieved the reliability requirements for the permafrost telemetry service. Thus, the last stage would be the deployment and testing of it in a physical testbed with real devices (O5). This testbed or PoC was deployed in Catalonia, between two rural areas in Hostalric and Cambrils, distanced by 150 km, and in Antarctica, between Uruguay's Artigas Base in King George Island and Spain's Juan Carlos I Base in Livingston Island, distanced by 93 km. An extended version of [86] was published as a journal article [87], showing the results from both the physical testbeds and DTN simulations. This paper is the third and last one from this compendium thesis. I was in charge of the article writing, the state-of-the-art review, the new architecture proposal, the use case modeling, the simulations, the tests definition, the data curation, and the analysis of the results. I also supervised the BP implementation in the hardware of the Antarctic campaign, and I also participated in the deployment of the physical testbed.

1.6. Outline

This work is presented in a compendium thesis form[1]. A compendium thesis presents a minimum of three articles already published or accepted in indexed journals, and all articles follow a coherent and cohesive research track. This compendium thesis is structured as follows. After the introduction (Chapter 1), the three published articles [83], [85], [87] follow and correspond to Chapter 2, Chapter 3, and Chapter 4, respectively. Chapter 2 ([83]) describes the proposed trustworthiness model in detail and validates it by simulating and evaluating the Antarctic use case. Chapter 3 ([85]) assesses the use of different transport protocols for the same use case, analyzing their achieved trustworthiness through the model described in Chapter 2. Chapter 4 ([87]) continues the trustworthiness analysis by evaluating the use of BP and proposes and tests a DTN architecture that joins the Antarctic permafrost telemetry use case with this protocol. This chapter also describes the practical tests in the field performed during the past Antarctic campaign. Chapter 5 summarizes the results obtained in each article and discusses them. Finally, Chapter 6 concludes this document and presents future work. A copy of the articles as published is available in the Appendix.

[1] According to the general rules for the organization of the doctorate studies at the Universitat Ramon Llull (URL), approved by the Governing Board of the URL on July 19th, 2018, and modified at the Governing Board meeting of June 17th, 2021, and at the meeting of November 17th, 2022. (https://www.url.edu/sites/default/files/content/file/2022/11/22/68/vri-normes-generals-organitzacio-doctorat-url-2022.pdf)

Chapter 2

A HETEROGENEOUS LAYER-BASED TRUSTWORTHINESS MODEL FOR LONG BACKHAUL NVIS CHALLENGE NETWORKS AND AN IOT TELEMETRY SERVICE FOR ANTARCTICA

Antarctica is a key location for many research fields. The lack of telecommunication systems that interconnect remote base camps hardens the possibility of building synergies among different polar research studies. This paper defines a network architecture to deploy a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service. Long backhaul NVIS links were used to interconnect remote networks. This architecture presents some properties from challenging networks that require evaluating the viability of the solution. A heterogeneous layer- based model to measure and improve the trustworthiness of the service was defined and presented. The model was validated and the trustworthiness of the system was measured using the Riverbed Model simulator.[2]

Keywords: Trustworthiness; Model; Telemetry; IoT; NVIS; Challenge Network; Antarctica.

2.1. Introduction

During the last half-century, Antarctica has been a key location for many research studies in several fields, such as oceanography, bioscience, geoscience, physical sciences, and other environmental studies [9]. Although many bases have been settled down in the peripheral areas of the Antarctic continent [91], the environmental and terrain difficulties provoke numerous challenges when it comes to implementing new operational services for modern studies. One of these challenges is the lack of telecommunication systems in Antarctica [10], especially wireless sensor networks (WSNs). Without WSNs, new research studies tend to use non-automatized ways of gathering data, which are more complex logistically, less scalable, and more error-prone. Moreover, most Antarctic bases are not interconnected between them. This

[2] The work reported in this chapter was published as the paper entitled "A Heterogeneous Layer-based Trustworthiness Model for Long Backhaul NVIS Challenge Networks and an IoT Telemetry Service for Antarctica" in the Sensors journal, 2021, 21(10), 3446, https://doi.org/10.3390/s21103446 Authors: Adrià Mallorquí, Agustín Zaballos.

fact lowers the possibilities for different research groups to collaborate in similar studies, and the advantages of providing synoptic region-wide observations and building synergies are lost [10].

The lack of conventional telecommunication services in Antarctica leverages the use of Satellite communications or other systems such as High Frequency (HF) links to build a network of interconnected remote WSNs. The first option is commonly discarded because of budget reasons, given the high costs of subscribing to this service type. Furthermore, the degree of coverage offered by satellite constellations in Antarctic latitudes is not the desirable [15]. To overcome these difficulties, the SHETLAND-NET [14] project aims to expand the use of communications in HF (3-30 MHz) by ionospheric reflection to the establishment of a low consumption communications system that allows the collection of sensor data distributed throughout the archipelago of the South Shetland Islands. This system, called Near Vertical Incidence Skywave (NVIS), does not require direct vision and is totally independent of the satellite since the signal is transmitted upwards and allows to overcome any geographical feature [11], [15], [16]. The long backhaul NVIS link has a coverage range of up to 250 km, and its reliability is very dependant on the ionospheric conditions and solar activity. Researchers from our University have previously participated in research campaigns in Livingston Island, studying and verifying the NVIS communication system's viability. A new campaign is planned to be carried out between December 2021 and March 2022, with the goal to test the new improvements of the NVIS link [18] and deploy an IoT network for three different use cases: a telemetry service for light data (e.g., penguin tracking [92]), a telemetry service for fat data (e.g., lichen observation [93]), and a distributed computing service to map the ionosphere along Antarctica.

However, a network deployed with long backhaul NVIS links may present some situations typical from challenge networks [94], such as intermittent connectivity, end-to-end disconnection, and variable error rates, difficulting the implementation of the aforementioned services. For this reason, it is necessary to study the viability and the expected trustworthiness of implementing this kind of network before its deployment in the field.

This paper focuses on the use case of the telemetry service for light data. Many Antarctic studies could be helped by automating the data gathering of their research (e.g., geomagnetic studies [95], blowing snow monitoring [96], climate change [97], biological monitoring [92], or permafrost analysis [20], among others). Most of the data for these studies are currently gathered manually, and some zones might be challenging to reach, even with special vehicles such as snow motorbikes. For these reasons, the studies are focused on small areas of the Antarctic region. Thus, a WSN that provided a broader coverage area and the interconnection of remote areas could increase the results' relevance. Moreover, the long backhaul links in charge of communicating remote WSNs could also be used to interconnect different Antarctic bases [15].

The paper has two main objectives. First, it is necessary to define which architecture, technologies, and protocols the telemetry service will use. As mentioned before, the drawbacks of challenge networks added to the extreme conditions sensors and other equipment need to work within that environment can provoke the service to reach low levels of performance and trustworthiness in front of adversities. Thus, the paper's second objective is to propose and validate a model for visualizing, understanding, and measuring the trustworthiness of the overall service. With this model, the service's weaknesses could be detected, and countermeasures could be proposed to improve its trustworthiness. We will use the Riverbed Modeler simulator [31] to validate the model and measure the service trustworthiness. To concrete the results, the tests will be performed by modeling the permafrost use case of [20], where Ground Terrestrial Network-Permafrost (GTN-P) stations are used to measure 32 different parameters. These tests can be replicated to other concrete telemetry use cases by modeling them too.

The rest of the paper is organized as follows. In sections 2.2 and 2.3, the background and related work are described, respectively. Section 2.4 defines the use case's service architecture. Section 2.5 presents the trustworthiness model. In section 2.6, the performed simulations are described, and the extracted results are analyzed in section 2.7. Finally, the conclusions of the paper and future work are detailed in section 2.8.

2.2. Background work

This section presents mature IoT and WSN technologies that can help to define the network architecture of our telemetry use case for remote regions. In terms of network architecture for WSNs, it is necessary to differentiate between the access network and the core network. On the one hand, the access network provides connectivity to the IoT sensors in a variable coverage range, depending on the technology. On the other hand, the backbone is in charge of interconnecting the access networks to build a global WSN. The backbone network can use long backhaul links to reach remote areas and broader coverage than access network technologies.

2.2.1. IoT access network

The access network technologies for WSNs are commonly known as the IoT communication protocols [98] or IoT MAC layer protocols [99]. These protocols are commonly classified, depending on the size of the coverage area, as short-range coverage protocols and long-range protocols. Networks built on the latter kind of protocols are commonly known as Low-Power Wide Area Networks (LPWANs)[98], [99].

For short-range networks, the most common technologies are RFID, NFC, Bluetooth Low Energy (BLE), Zigbee, 6LoWPAN, and Z-Wave [98]. For LPWANs, the most used communication protoocols are Narrow-Band IoT (NB-IoT), Long Term Evolution enhanced Machine-Type Communication (LTE eMTC), Sigfox, and LoRa. In the specific case of Antarctica, short-range communications are rarely used to deploy WSN applications. One example is the SNOWWEB project [12], where a network of weather stations was build using Zigbee transceivers. LPWANs seem to be more suitable options since the coverage area for deploying the WSN is more extended. For that reason, authors in [13] studied the applicability of Lora in Antarctic regions by characterizing its channel in the field, achieving a coverage area of up to 30 km radius. Despite that, it seems feasible that some sensors of the WSN can be located out of range of the gateway due to geographic conditions. In this case, there is the need to use mobile gateways and deploy Mobile Ad-hoc Networks (MANETs) [15]. To deploy this kind of opportunistic mobile network, the concept of Internet of Flying Things (IoFT) [100] with the utilization of devices such as Unmanned Aerial Vehicles (UAVs) has gained attention lately [101], building a Flying Ad-hoc Network (FANET) [102].

2.2.2. IoT backbone network

On the other hand, the backbone network is in charge of interconnecting remote WSNs to build a single major network. For this purpose, LPWAN communications are not valid because the links that need to be established must have a broader range (several tenths of kilometers). Moreover, since the Antarctic region has many terrain variations, it is expected that two nodes separated by several kilometers do not have Line of Sight (LoS) [11]. Satellite communications are a solution to overcome these problems. However, Geostationary Earth Orbit (GEO) satellites do not cover Antarctica's latitudes adequately, and current Low-altitude Earth Orbit (LEO) satellites provide partial or no coverage in deep polar regions [103]. Authors in [103] studied the possibility to cover the whole Antarctic continent with a three-satellite constellation in elliptical orbits, but it has not been implemented. A significantly lower cost solution suitable for WSNs in remote areas is the use of High Frequency (HF) communications. Specifically, the Near Vertical Incidence Skywave (NVIS) technique has already been tested in Antarctica [11], [15], [16]. Results show that this kind of long backhaul link can reach a throughput of up to 4.6 Kbps and a coverage radius of up to 250 km without the need for Line of Sight (LoS). The applicability of NVIS has also been analyzed for natural disasters and emergency situations when common telecommunication infrastructure turns inoperative [17]. The main drawback of NVIS is the considerable variation of the transmitting channel's characteristics, the ionosphere, which can lead to some periods of no connectivity, becoming a challenge network [94]. Thus, it is necessary to test and measure NVIS networks' trustworthiness when used to transport data from actual use cases.

2.3. Related work

This section describes the related work from other authors to define and measure the trustworthiness of Cyber Physical Systems (CPS). A CPS is defined as a system with integrated computational and physical capabilities. Common examples of CPSs include industrial control systems, computerized vehicle and aircraft controls, wireless sensor networks, smart grids, and almost all devices typically encompassed by the Internet of Things [30], [104]. The trustworthiness of CPS is defined in the literature, in general terms, as the property of behaving as expected under adversarial conditions [32]. However, these adversarial conditions can come from different reasons, e.g., faulty nodes, byzantine errors, malicious behaviors, and network malfunction, among others [33]. For this reason, in the literature, there can be found many different approaches to measure or provide trustworthiness that refer to disparate elements. We propose to classify them into the following four categories:

1. Data Trustworthiness: it is defined as the possibility to ascertain the correctness of the data provided by the source [34]. Many methods try to detect faulty nodes, false alarms, and sensor misreadings using different approaches [33]. For instance, authors in [35] present a distributed Bayesian algorithm to detect faulty nodes, while authors in [36] use a fog computing architecture to detect, filter, and correct abnormal sensed data. Also, authors in [37] present a Data Intrusion Detection System to trigger false data from malicious attacks.

2. Network trustworthiness: it can be defined as the probability for a packet to reach its destination unaltered despite the adversities (e.g., link failure, link saturation, or malicious attacks, among others) [38], and it is a crucial factor of Low-Power and Lossy Networks (LLNs). Improving the network trustworthiness and performance is a challenge that has been addressed from different perspectives, such as load balancing and redundancy protocols [39], transport protocols [40], dynamic routing and topology control protocols [41], [42], cybersecurity mechanisms [43], and Delay Tolerant Network (DTN) architectures and protocols [25]. In the case of routing, both proactive routing protocols (e.g., IPv6 Routing Protocol for Low Power and Lossy Networks (RPL) and Optimized Link State Routing (OLSR)) and reactive routing protocols (e.g., Ad-hoc On-Demand Distance Vector (AODV) and Link-Quality Source Routing (LQSR)) have been proposed in the literature to solve the drawbacks of LLNs and MANETs. For DTN architectures, the Bundle Protocol (BP) specification is used as a store-carry-forward overlay mechanism where bundles are stored locally at each node and forwarded when the network link is available. Some implementations of the BP are DTN2, IBR-DTN, and DTN7 [25].

3. Social Trustworthiness: this trend has gained more attention since the irruption of the Social Internet of Things (SIoT) concept [44]. In SIoT trustworthiness, the capability of the objects to establish social relationships autonomously between them is leveraged to define more complex trust and reputation models that take into account several input parameters. Authors in [45] define a subjective model that considers factors as the computational capabilities of the nodes, the type of relationship between them, the total number of transactions, the credibility of a node, and the feedback provided by other nodes, among others. In [46], they evolved their previous model and based it on more parameters, such as the neighborhood of nodes, and presented a new objective model with a faster transitory response. Authors in [47] propose another model that defines the input parameters as the expected gain on success, the expected damage on a failure, the expected cost, the expected result, and the goal. Authors in [48] define a decentralized self-enforcing trust management system based on a feedback system and reputational secure multiparty calculations to ensure the privacy of each party's provided data.

4. Consensus: it represents a state where all participants of the same distributed system agree on the same data values [49]. Consensus protocols can be divided into two general blocks: Proof-based consensus and byzantine-consensus. The first group is oriented to blockchain technology, where all participants compete with each other to mine a block, and the most used protocols are Proof-of-Work, Proof-of-Stake, and its variants [50], [105]. The main drawback of these protocols for IoT is that most devices have simple hardware specifications and low processing power, being incapable of performing the mining tasks of blockchain [51]. The second major group of consensus protocols is the more classical byzantine-based. These kinds of protocols implement voting-based mechanisms to reach an agreement rather than competing among them, which generally results in less resource consumption. The drawback of these mechanisms is the number of messages that need to be delivered through the network to reach an agreement. The most well-known protocols in this category are Practical Byzantine Fault Tolerance, RAFT, PaXoS, and Ripple, although several variants have emerged year-by-year [50].

2.4. Network and service architecture

Prior to applying the model of trustworthiness, the first goal is to define the architecture of the telemetry use case. As mentioned before, the concrete case is the improvement of permafrost studies by automating the data gathering from the GTN-P stations (the sensors), which measure 32 different parameters. Currently, data is gathered only once a day, and authors from [20] leave the complete automation of the GTN-P stations as an open challenge, given that their approach suffers from a lack of connectivity. It is important to remark that the architecture described below applies to any telemetry use case. However, we will use the example of the GTN-P stations' permafrost research for better understanding.

We propose to use the architecture defined for the SHETLAND-NET project [14]. In our approach, we aim to interconnect all remote sensors to a control center, building a Global Wireless Sensor Network (GWSN) composed of several Wide Wireless Sensor Networks (WWSN), able to gather data more frequently. The first approach to designing a remote sensing system for the Antarctic region was described in [4] during the SHETLAND-NET project's early stages, describing how sensors could reach and use NVIS as a long backhaul link. However, it was mostly centered on designing the characteristics of the OSI model Layer 1 of the NVIS (backbone) network. In this paper, a more detailed description of the overall network architecture is presented. The network diagram is detailed in Figure 8.

The access network (WWSN) will be in charge of providing connectivity to the remote sensors, transporting the gathered data from the sensors (GTN-P stations connected to a low-consumption board) to the gateways (e.g., a Raspberry Pi). The main gateway of each WWSN will be located near the research base, with the GTN-P stations located around it in a few-kilometer radius. For redundancy reasons, groups of GTN-P stations can be clustered and placed close enough to interpret that they measure the same permafrost values. These stations will sense the data and send it to the gateway once per hour. For this use case in Antarctica, it is logical to think that the wider the area can be covered by the access communication technology the better, because more sensors will be able to be placed far from the research base so researchers will have access to sensors placed farther away while saving valuable time in collecting the data. For this reason, short-range communications are less suitable, and

LPWAN communications are preferred. The lack of telecommunication operators providing service in Antarctica forces operator-dependent communication services such as Sigfox, NB-IoT, or LTE eMTC to be discarded. This leads LoRa as the main candidate to deploy the access network. LoRa transceivers will be placed in each GTN-P station, responsible for sending the measured data to the LoRa gateway. As explained in [15], this gateway has been implemented with a Raspberry Pi 3B+ in previous Antarctic campaigns of the SHETLAND-NET project, and it is responsible for storing the gathered data from all the sensors it is giving service to, ready to send all these data through the backbone network.

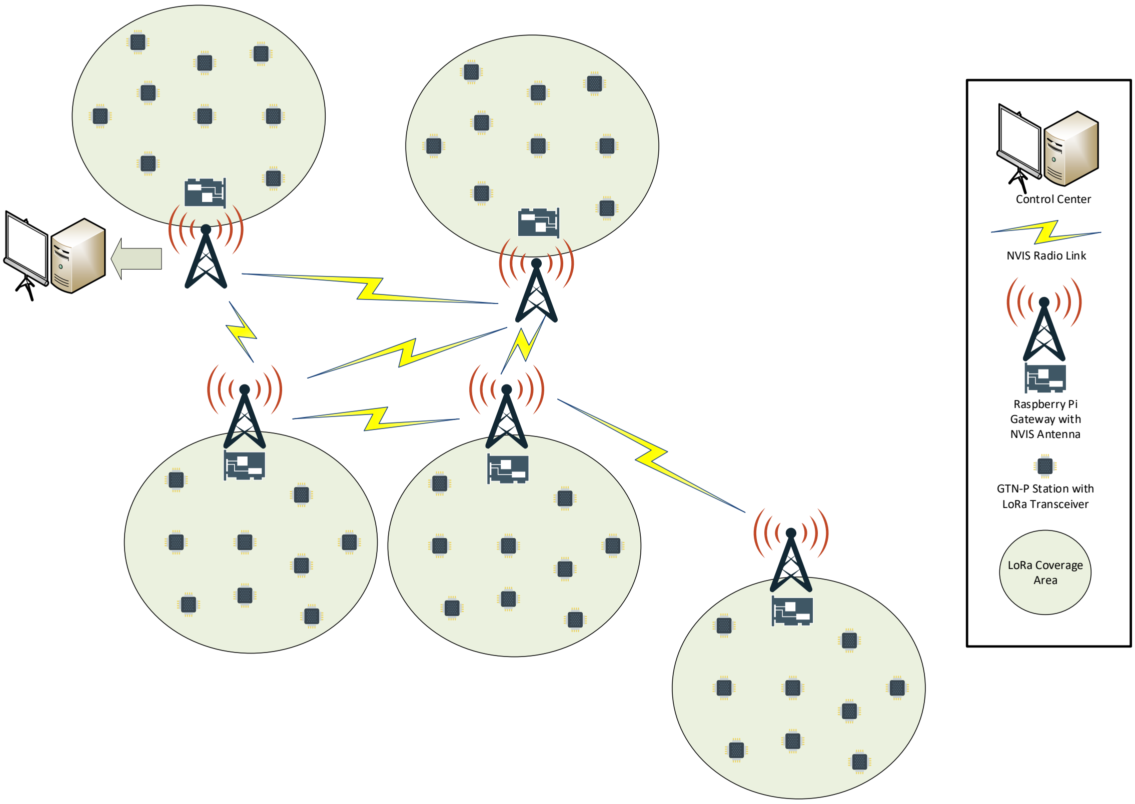

Control Center
NVIS Radio Link
Raspberry Pi Gateway with NVIS Antenna
GTN-P Station with LoRa Transceiver
LoRa Coverage Area

Figure 8. Network diagram of the SHETLAND-NET project telemetry service.

The backbone network will be composed of all NVIS nodes, which will interconnect remote WWSNs through the long backhaul links to form the GWSN. Each NVIS node mainly consists of a Red Pitaya, a Raspberry Pi 3B+, and an HF antenna [11]. The link that can be established between two NVIS nodes has a range of up to 250 km. In order to interconnect all the WWSNs and reach all remote areas, a multi-hop network will need to be deployed. Thus, some of the NVIS nodes will have to act as repeaters. At least one NVIS node will need to be connected to the control center, achieving to send all the data to it. To avoid a Single Point of Failure (SPF), having more than one NVIS node connected to the control center is recommended. The possibility of having multiple paths to reach one destination demands the need for a routing protocol able to find the best possible loop-free path in the network [17].

The operation of the backbone network can be summarized as follows. Each NVIS node will act as a concentrator, gathering the data from every GTN-P station inside their LoRa coverage area. Once all possible data is collected, the NVIS node will forward it to the node connected to the control center, following the path determined by the routing protocol through the backbone network.

However, we can find three main issues that can provoke this architecture to become a challenge network:

1. Due to Antarctica's extreme weather and environmental conditions, both sensors and gateways could experience temporary or persistent malfunctioning.

2. The irregular elevations of the Antarctic terrain might provoke that sensors do not have a Line of Sight path through the gateway [13]. This fact degrades the performance of LoRa communications considerably.

3. Depending on the ionosphere's state and the solar activity, NVIS links may become unavailable temporally or intermittently.

For this reason, our primary goal is to establish a model to measure the trustworthiness of a CPS, with which the performance of the proposed architecture can be evaluated, and its weaknesses can be detected and improved.

2.5. Trustworthiness model definition

Our proposal to measure the trustworthiness and evaluate a CPS's performance (in our case, a group of interconnected remote Antarctic Wireless Sensor Networks providing an IoT telemetry service) is a layer-based model. This model is characterized by two base layers (Data Trustworthiness Layer and Network Trustworthiness Layer), two extension layers (Social Trustworthiness Layer and Consensus Layer), and the interaction between all of them. The Data Trustworthiness, Network Trustworthiness, Social Trustworthiness, and Consensus Layers can collectively define a system's trustworthiness. A graphic representation of the layered model is shown in Figure 9.

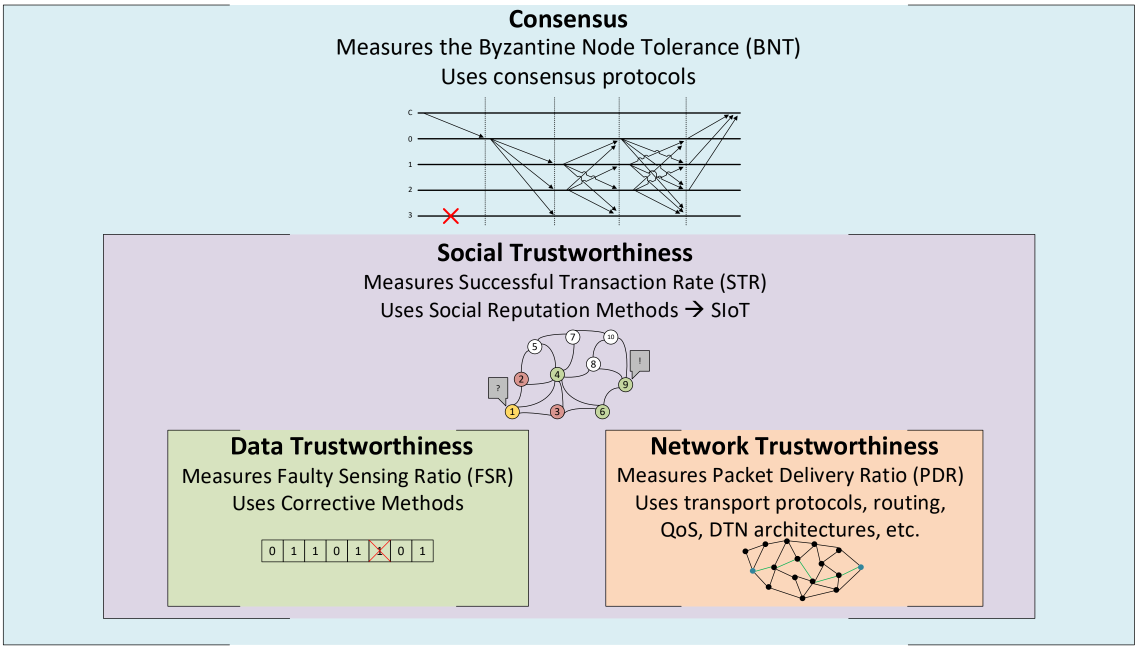

Consensus
Measures the Byzantine Node Tolerance (BNT)
Uses consensus protocols
c
0
1
2
3
Social Trustworthiness
Measures Successful Transaction Rate (STR)
Uses Social Reputation Methods → SIoT
?
!
1
2
3
4
5
6
7
8
9
10
Data Trustworthiness
Measures Faulty Sensing Ratio (FSR)
Uses Corrective Methods
0 1 1 0 1 1 0 1
Network Trustworthiness
Measures Packet Delivery Ratio (PDR)
Uses transport protocols, routing,
QoS, DTN architectures, etc.

Figure 9. Trustworthiness model layers.

We postulate that each layer is characterized by its definition (scope), how the trustworthiness of that layer is measured (metric), and how the value of this metric can be improved (correction).

2.5.1. Trustworthiness layers' definitions

We propose the following definitions for each layer:

1. Data Trustworthiness Layer: is the layer responsible for ascertaining the correctness of the data provided by the source.

2. Network Trustworthiness Layer: is the layer responsible for assuring that a packet reaches its destination on time and unaltered despite the adversities (e.g., link failure, link saturation, or malicious attacks, among others).

3. Social Trustworthiness Layer: is the layer responsible for leveraging the capability of the objects to establish social relationships autonomously between them to improve the trust between them and the correctness of gathered data.

4. Consensus Layer: is the layer responsible for reaching a state where all participants of a group agree on the same response or result.

On the one hand, Data and Network Trustworthiness are the base layers of our model because the system that we want to measure is meaningless if we do not have some data to be exchanged between nodes through a network. On the other hand, Social Trustworthiness and Consensus are the extension layers because they include functionalities that are not needed in the service architecture but are optional to implement.

2.5.2. Trustworthiness layers' definitions

Managing the trustworthiness of a system is possible when the different layers are separately understood. This way, objectives and metrics can be defined to measure the level of trustworthiness. In order to measure the four layers of trustworthiness, we have defined a quantitative metric for each layer. Once metrics are defined, a trustworthiness target can be determined, which is a quantitative objective given to a trustworthiness metric. If a trustworthiness characteristic does not meet its target, a change factor is needed to revert the situation. The combination of all change factors defined to meet the trustworthiness targets is called the trustworthiness implementation.

We propose that the trustworthiness model will use the normalized metrics defined in Table III to quantify and measure trustworthiness.

TABLE III
TRUSTWORTHINESS METRICS

Layer	**Metric**	**Range**
Data	Faulty Sensing Ratio	[0-1]
Network	Packet Delivery Ratio	[0-1]
Social	Successful Transaction Rate	[0-1]
Consensus	Byzantine Node Tolerance	[0-1]

Faulty Sensing Ratio (FSR) is defined as the proportion of false sensed values (FSV) by all nodes and total sensed values (TSV) in a defined time period, as stated in Eq. 1.

$$FSR = \frac{FSV}{TSV} \tag{1}$$

We consider that a sensed value is every independent and semantically significant measured data that a sensor stores in its memory (e.g., RAM, Flash, hard-drive, among others). Suppose no corrective methods are used in the system. In that case, sensed data (e.g., temperature, humidity, position, ice content) is considered to be faultily sensed if the value stored in the sensing (source) node's memory is different from the value that the sensor should have correctly read (within a tolerance percentage). In real implementations, the number of FSV can only be measured if the sensed data's value is known a priori (ground truth) [106]. Otherwise, only in simulations it is possible to quantify the number of FSV. FSV and TSV are parameters that must be gathered within the same time slot to calculate the ratio correctly. The lower the FSR is, the better the data trustworthiness.

Packet Delivery Ratio (PDR) is calculated as the quotient between the total number of packets received (Pr) by all nodes and the total number of packets sent (Ps) by all nodes in the same time slot, as stated in Eq. 2. A packet is considered received if and only if the reception time (T_{rx}) is less or equal to the transmission time (T_{tx}) plus a defined threshold offset η ($T_{rx} \leq T_{tx} + \eta$), and the packet content is not altered. The higher the PDR is, the better the network trustworthiness. In our proposal, retransmitted packets and original packets are counted separately to compute the metric value.

$$PDR = \frac{Pr}{Ps} \tag{2}$$

The Successful Transaction Rate (STR) is the proportion between the number of Successful Transactions (ST) and the total number of transactions (TT) in a defined time slot, as stated in Eq. 3. We define a transaction *l* as a sensed value *v* that a node *j* expects to receive from a node or group of nodes *i*. Retransmitted or duplicated packets for the same value *v* are considered part of a single transaction *l*. A transaction *l* is considered successful when a node *j* expects to get some information or data (*v*) from node *i* before a defined maximum reception time (Trx_{max}) and receives it as expected, thus providing good feedback ($f_{ij}^{l} = 1$) for that transaction to node *i*. ST and TT are parameters that must be gathered within the same time slot to calculate the ratio correctly. The higher the STR is, the better the social trustworthiness.

$$STR = \frac{ST}{TT} \tag{3}$$

The Byzantine Node Tolerance (BNT) is defined as the proportion of supported byzantine nodes (Nb) that can participate in the consensus system without affecting the correctness of the general agreement and the total number of nodes (Nt) that participate in the consensus system, as stated in Eq. 4. A node is considered to be byzantine if it experiences a crash or soft fault that incapacitates it to behave as expected, or if it does not behave as expected on purpose (malicious node). The higher the BNT is, the higher are the probabilities to reach a correct General Agreement. Although theoretically, the BNT value range is between 0 and 1, in practice, it is not possible to reach a correct consensus with a BNT >= 0.5.

$$BNT = \frac{Nb}{Nt} \tag{4}$$

2.5.3. Trustworthiness improvement

Now that we have defined the four trustworthiness layers and its associated metrics, we can give some examples of techniques and protocols that can be used at each layer to improve the metrics' values.

At the Data Trustworthiness Layer, corrective methods can be applied which try to detect abnormal data (False Sensed Values) stored in the source node due to a sensor malfunctioning, a misreading of the sensed data, or erratic writing to the node's memory. Corrective methods can be used to detect and correct these abnormal values by comparing them to the values sensed by the same node previously and other mechanisms such as hashes, checksums, and parity bits, among others. If these corrections are performed at the post-processing stage by the receiving server or gateway, the nodes' malicious data manipulation can also be detected. However, our model assumes that corrections are only made by the own node (source node). Otherwise, errors that originated during the data transport through the network, which are out of our scope definition of the Data Trustworthiness Layer, could be misinterpreted as source node errors. The drawback of this assumption is that only non-malicious errors are likely to be corrected at this layer because malicious nodes might not correct data on purpose. Our model specifies that other layers of the model are responsible for mitigating malicious behaviors (e.g., the Network Trustworthiness Layer).

The method presented in [36] is a suitable example of a corrective method for data trustworthiness. This value-level corrective method defines thresholds to detect potential abnormal data (e.g., a lower-value limit t_{low}, an upper-value limit t_{up}, and an abrupt change threshold t_{ch}). When a potential abnormal value is detected, it is compared with the values sensed from the node’s neighbors, computing the group value similarity (G). Since this breaks our model's assumption, this value similarity could be computed with the historical values from the own sensor. If the similarity is lower than a threshold t_{sim} then the abnormal data is confirmed and corrected (e.g., interpolation with previous and posterior correct values sensed by the own node). This method might experience false positives (by detecting a correct value as abnormal and modifying it), and false negatives (by not detecting an abnormal value), which can be grouped into Faulty Sensed Values (FSV). If the thresholds are too strict, the number of false positives will increase, while the number of false negatives increases if the thresholds are too lax. The lesser the number of FSV, the better the data trustworthiness, so an optimal trade-off value for the thresholds must be found to minimize the overall number of FSV. This number is easy to gather in simulation scenarios, but in real implementations, it is only possible if the values are well-known a priori (ground truth values).

At the Network Trustworthiness Layer, routing protocols and QoS mechanisms are used to find the best path from a source to a destination by quantifying the quality or performance of each link in the network. For each destination, more than one path can be determined as feasible, thus providing load balancing. Many metrics exist to calculate the best path, such as the number of hops, the bandwidth of the link, the delay, and the expected number of retransmissions, among others. These routing protocols can be classified under different categories, such as proactive/reactive, link-state/distance-vector, or monometric/multimetric [42]. Selecting the best path for a traffic flow will eventually improve network statistics, such as throughput, delay, jitter, or Packet Delivery Ratio (PDR). In the case of challenge networks, DTN overlay architectures and protocols, such as the Bundle Protocol [107], is also a solution that can be used to improve the network trustworthiness.

Another relevant element to take into account in this layer is the data security through the network. While traveling from the source to the destination, data should remain private, available, and unaltered, preventing it from cyberattacks. For this purpose, network elements such as Next-Generation Firewalls or Intrusion Detection Systems and security mechanisms such as data encryption, authentication, anti-spoofing techniques, and network filters are used in the network.

At the Social Trustworthiness Layer, most solutions tend to use reputational mechanisms to determine which nodes to trust when exchanging information. This reputation is commonly based on previous transactions' feedback to build an opinion for the node's trustworthiness [108]. More complex and robust mechanisms also incorporate parameters such as the indirect opinion of other nodes, the relevance (weight) of each transaction, the node's centrality, the node's computational capacity, and the type of relationships between the nodes [45].

The model of [46] provides two different ways for computing the reputation of a node. On the one hand, a subjective model of social trustworthiness is presented to compute the reputation of node *i* under the perspective of every other node (R_{ij}), being these reputations different from each other because the experience of interaction with node *i* for two different nodes can be different. Moreover, reputations are asymmetric, meaning that the reputation that node *j* calculates from node *i* can be different from the reputation that node *i* calculates for node *j* ($R_{ij} \neq R_{ji}$). Thus, the system's overall trustworthiness can be represented as an NxN matrix for the reputation that each node calculates for all the other nodes, where N is the total number of nodes. On the other hand, objective models calculate one single reputation for each node (R_i), representing the trustworthiness that the system as a whole perceives from node *i*. This reputation takes into account the opinion and the feedback from all the other nodes. Thus, the system's overall trustworthiness is represented as an N-size vector with the reputation that the whole network perceives for each node.

Both the subjective and objective approaches aim to leverage the transactions between trustful nodes and isolate those with bad reputations, which are considered more faulty or malicious prone. Thus, their goal is to maximize the number of Successful Transactions (ST).

At the Consensus Layer, several mechanisms can be used to reach a decentralized General Agreement (GA) that all nodes of the group consider to be true. Theoretically, if the number of byzantine nodes is more than 50% of the total number of participating nodes, every consensus mechanism will fail to reach a benevolent agreement. Consensus mechanisms aim to reach the GA while tolerating a percentage of byzantine nodes. Consensus protocols are generally classified into competing mechanisms (proof-based) and voting-based mechanisms. The latter are more suitable for IoT devices because they consume fewer resources from the node. These protocols commonly consist of various voting phases to reach the GA, and their goal is to maximize the number of tolerated Byzantine Nodes (BN). A drawback of these mechanisms is that they need participating nodes to exchange a large number of messages between them to reach a consensus, which can be a problem in low-bandwidth networks, consuming most of this bandwidth. Some protocols look for a trade-off between the number of tolerated BNs, throughput, and scalability.

2.5.4. Trustworthiness layer's dependencies

Trustworthiness layers' dependencies must also be understood. For instance, how will the data trustworthiness affect the consensus? Can a robust consensus protocol lower the trustworthiness of the network because it is causing bottleneck congestion? The trade-offs between these layers need to be carefully analyzed in order to obtain the optimal overall trustworthiness level. Our model dependencies proposal is exhibited in Figure 10.

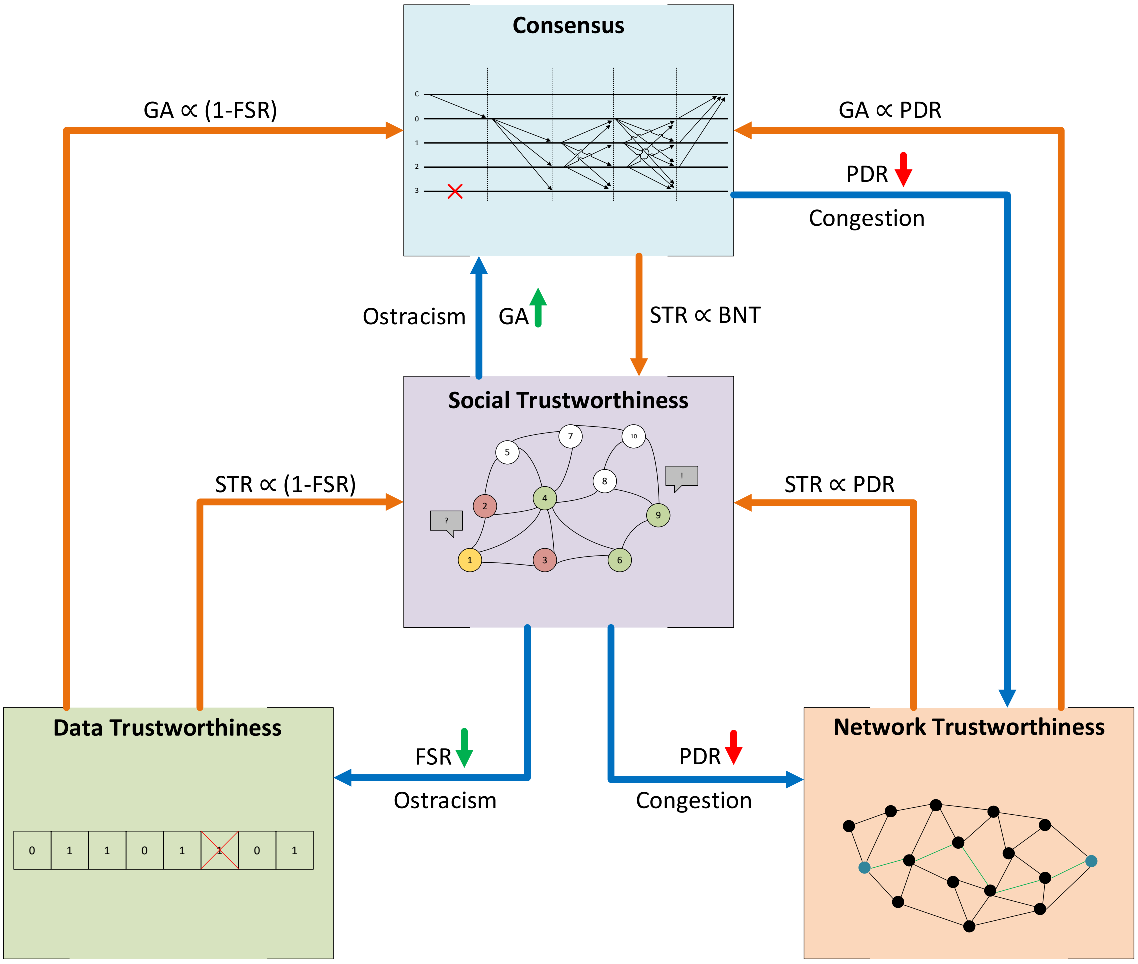


Figure 10. Dependency diagram between trustworthiness layers.

The Consensus Layer is affected by the other three layers. If FSV (Data Layer) is closer to 0, it means that nodes tend to measure the sensed values correctly, so they will be more prone to reach a correct general agreement. From the Social Layer, it is possible to ostracize those nodes with a lower reputation (which should be the ones with more false sensed values) if the application can afford to lose the data from them. In this case, if nodes with the worst reputation were omitted, it should be more probable to reach a correct general agreement for the rest of the nodes. Finally, suppose the PDR (network trustworthiness) is closer to 1. In that case, it means that the whole network delivers most packets unaltered and on time, so fewer nodes will be considered byzantine due to network issues, and reaching correct general agreements will be more feasible. It is important to notice that all these dependencies do not affect the Consensus Layer metric, the Byzantine Node Tolerance, which depends only on the consensus algorithm used and the total number of nodes participating in the consensus group.

We propose that the Social Layer can also be affected by the other layers. On the one hand, FSV and STR are inversely related. If the FSV is close to 0, a transaction coming from that node is less probable to have a false sensed value, meaning that it will become a Successful Transaction if the network delivers it properly to the destination. Also, the source node will obtain good feedback from the receiving node, increasing its reputation. On the other hand, PDR and STR are directly related. As the PDR decreases, it is more feasible that packets targeted to a node are lost in the network, decreasing the STR. Thus, the receiver would evaluate the transaction as a failure, providing bad feedback and decreasing the sender's reputation. Finally, if the Consensus Layer is implemented, the negative effect of some false sensed values from byzantine nodes and lost packets can be masked thanks to the consensus algorithm. Nodes could still reach a correct general agreement, marking that transaction as successful and increasing the STR.

The network layer can be affected by the Social and Consensus Layers in terms of congestion [56,57]. Depending on the application, if nodes with lowest reputation are ostracized, the system might tend to concentrate the majority of traffic to the network links that lead to the highest reputation nodes. Thus, these nodes' paths will be more congested and prone to packet drops, lowering the PDR. Similarly, as mentioned before, the use of a consensus mechanism introduces a considerable amount of network traffic. Also, the number of messages exchanged between a group of nodes is directly proportional to the number of nodes in the group. Thus, if the network bandwidth is not enough to support this extra traffic, the network is more prone to be congested and drop packets, decreasing the PDR.

Finally, it is intuitive to think that the Data Layer should not be affected by the other layers. The variability of the FSV should depend on the error probability of the sensors and the node itself (equipment quality, battery degradation), which could also be affected by external factors (i.e.., environmental characteristics). However, we propose the Data Layer can be affected by the Social Layer. Suppose the Social Layer is implemented and is being used to ostracize the lowest reputation nodes. In that case, we consider that sensed values from omitted nodes must not be counted for the FSR computation. Thus, if the lowest reputation nodes were the ones with more false sensed values, the overall FSR should increase.

It is important to see that Data and Network Layers (the base layers, which are always present) are entirely independent, given that the correctness of data is always measured on the source node, never on the destination. This way, data loss or alteration caused by the network does not affect the data correctness measure.

Notice that Social and Consensus Layers (the extension layers, which are optional) are the ones affected by the rest of the layers. However, the way they are affected is different. On the one hand, the dependencies from other layers to the Social Layer directly affect the value of its trustworthiness metric, the STR. On the other hand, the Consensus Layer metric, the BNT, is not affected by other layers, but these dependencies can improve the probability of reaching a correct general agreement, which in final terms improves the Social Layer metric, the STR.

In that sense, we consider that the system's overall trustworthiness can be measured with the STR metric, which is the one affected by the four layers of our model, and intrinsically incorporates the effects of the other three metric values (FSR, PDR, and BNT). Moreover, notice that without implementing the extension layers, the STR can still be computed, which will combine the effects of the base layers (Data and Network Trustworthiness).

Although the dependencies between the layers and metrics of our model have been identified, it is still challenging to quantify the effect of looped dependencies on the system's trustworthiness. We identify two actions that can provoke a looped dependency. First, if Social Trustworthiness Layer is used to ostracize the lowest reputation nodes, their sensed values will be omitted, decreasing the FSR and eventually increasing the STR. However, suppose more traffic than supported by the network is concentrated on the links that lead to most reputation nodes. In that case, it is possible to create network congestion that will decrease the PDR and eventually decrease the STR. Second, implementing a consensus mechanism might help tolerate byzantine nodes and faulty network links, which eventually increases the STR. Nonetheless, suppose the network bandwidth is not large enough to allocate the extra traffic introduced by the consensus mechanism. In that case, the network may suffer from congestion, decreasing its PDR and eventually decreasing the STR.

To quantify the effects and trade-off points between these dependencies, it is essential to test the model's applicability with a use case and measure the trustworthiness metrics under different circumstances and several times. Given the complexity and cost of performing such an amount of tests in the field, we opted to use simulation tests, which provides more flexibility.

2.6. Simulation tests

To validate the trustworthiness model, it is necessary to measure the metrics values for the use case scenario several times under different circumstances. For this purpose, the use case scenario has been represented and evaluated in the Riverbed Modeler Simulator [31]. As a reminder, our use case scenario is a group of interconnected remote Antarctic Wireless Sensor Networks providing an IoT telemetry service. Concretely, the telemetry service will be used to automatize the data gathering of GTN-P stations to study the permafrost of the Antarctic region. The remote sensors of WSNs will be connected to a concentrator gateway through LoRa (access network), and these gateways will be interconnected between them and a control center through long backhaul NVIS links (backbone network). The extreme conditions GTN-P stations need to work with, added to the challenges of NVIS links and a LoRa network without LoS, might degrade the overall system's trustworthiness. In order to measure and evaluate it, our proposed trustworthiness model will be applied in this use case.

The first step is the modeling of the network, the nodes, and the application. Once the model is designed and implemented in the simulator, the set of tests and the simulation parameters must be defined. After that, the simulations are run, and results are collected and evaluated.

2.6.1. Network models

For the use case scenario, the backbone network (NVIS) and the access network (LoRa) have been modeled separately. On the one hand, the NVIS channel has been modeled following the characteristics described in [11]. The transmission frequency is 4.3 MHz with a channel bandwidth of 2.3 kHz and a bitrate of 4.6 kbps. The range of the HF link is up to 250 km. Moreover, given that the ionosphere characteristics vary considerably during one day, we have also modeled the probability of a packet being correctly delivered through an NVIS link hour by hour, following the results in [18]. These results show that the NVIS links are unlikely to be available from 17:00h until 6:00h, while the channel availability from 6:00h to 17:00h varies from 70% to 100% when both the ordinary and extraordinary waves received are combined, as shown in Figure 11.

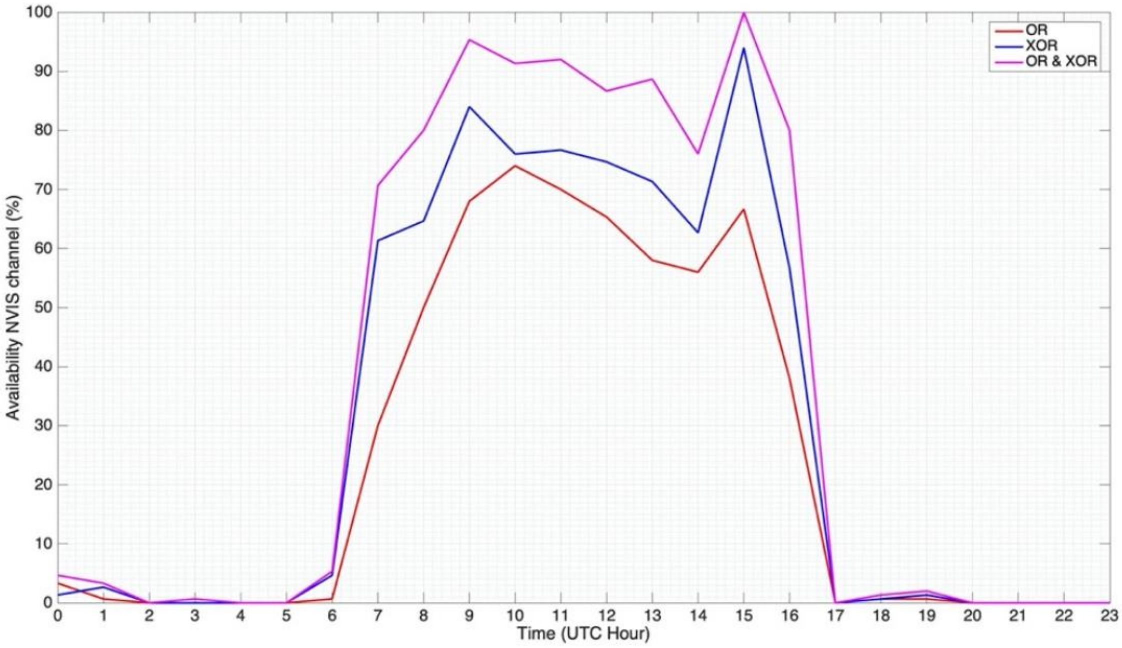


Figure 11. NVIS link availability depending on solar activity and the ionosphere's state [18]. The graph's legend is defined as follows: OR refers to the performance of the ordinary wave, XOR refers to the performance of the extraordinary wave received. OR and XOR refers to the total performance between both ordinary and extraordinary modes.

On the other hand, the LoRa channel has been modeled based on the results shown in [111] and [13]. The transmission frequency is 868 MHz, with a channel bandwidth that varies depending on the chosen Data Rate (DR) and the Rate Spreading Factor (SF). In our case, we chose DR3 and SF7, resulting in a channel bandwidth of 125 kHz and a bitrate of 5.47 kbps. The range of the link is up to 30 km. In the Line of Sight (LoS) case, the channel is always available with a Packet Loss of 0%. Otherwise, with no LoS, the packet loss varies from 0% to 98% depending on the signal reflections, with an average value of 72%. Due to the Antarctic surface's irregularities, we can not assume that GTN-P stations will be located in LoS with the gateway. For this reason, we will consider that 25% of sensors will have LoS to the LoRa gateway, while the remaining 75% will not have LoS. No LoS case. Table IV summarizes the characteristics of our network models.

TABLE IV
NETWORK PARAMETERS TO MODEL THE SCENARIO

Parameter	**NVIS**	**LoRa**
Transmission Band	4.3 MHz	868 MHz
Channel Bandwidth	2.3 kHz	125 kHz
Channel Bitrate	4.6 kbps	5.47 kbps
Coverage range	Up to 250 km	Up to 30 km
Daytime Availability (6am-5pm)	70%-100%	100% (LoS), 2%-100% (No LoS)
Night Availability (5pm-6am)	0%	100% (LoS), 2%-100% (No LoS)
Maximum Payload Size	242 bytes	140 bytes

2.6.2. Node model

In the case of the node, both the GTN-P station and the gateway will use the same finite state machine model, as shown in Figure 12. The "INIT" state initializes the model and its attributes. The "IDLE" state is used when the node is waiting for a packet to arrive, transitioning to the "PROCESS" state, or a self-interruption to send the sensed data values, transitioning to the "SEND" state.

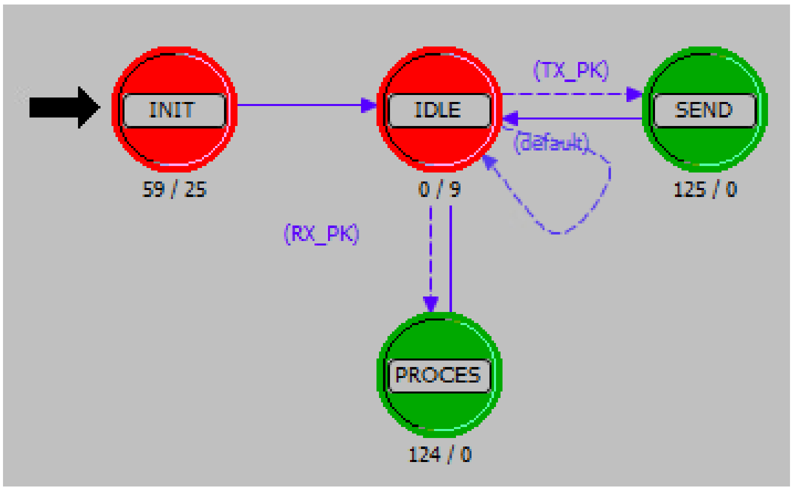


Figure 12. Node model finite state machine.

2.6.3. Application model

Pseudocode algorithms for the application modeling are shown in Appendix A (section 2.9). The application consists of the telemetry service to gather data from measured values by sensors and send them to the control center.

Each measured value *v* is considered a transaction *l* that must reach the control center. The application can be run without implementing any of the extension layers of the proposed trustworthiness model (standard mode), or can implement the Social and Consensus Layers of the model (redundancy mode). In standard mode, each value *v* is measured by a single GTN-P station, while in redundancy mode, the implementation of a reputational or consensus mechanism leverages the creation of clusters (groups of GTN-P stations) that measure the same value *v*.

GTN-P stations will send data packets once per hour, simulating the moment when the 32 values are gathered from the GTN-P station sensors, stored in memory, and delivered to the gateway. In this process, if no consensus mechanism is performed, a hardcoded value *v* for each parameter will be inserted into a 132-byte payload (32 values and a timestamp, 4 bytes each). With a probability *Pb*, the value *v* will be modified to another value out of an acceptable range [v_{min}, v_{max}], and the total number of FSV will be increased by one. This payload will then be inserted into the packet to be sent to the gateway. If an ACK packet is not received from the gateway before a timeout T_{out}, the data packet will be retransmitted up to a maximum of 3 times. In the case of implementing a consensus mechanism, all the GTN-P stations participating in the cluster (which are measuring the same *v*) will start the process to reach a general agreement. Once they have reached it, only the cluster leader will send the payload to the gateway with the agreed value *v*. During the consensus process, if the Social Layer is also implemented, each packet exchanged between the nodes participating in the consensus group will be used to compute the reputation *Ri* of nodes. The node with the highest reputation will be elected as the group leader. Moreover, a node *i* with a reputation R_i lower than R_{min} will not have the right to vote for the value election. However, it will be allowed to continue participating in the consensus group to increase its reputation until it can be granted the right to vote again. On the contrary, if no reputational mechanism is being used, all group members will always have the right to vote, and the leader will be chosen randomly.

On the other site, gateways will collect the data from the GTN-P stations inside its LoRa coverage area and then forward it through the NVIS backbone network until it reaches the control center. Given that gateways are also nodes, they may experience a byzantine failure with probability *Pb*. In that case, the gateway will modify the payload's content. In standard mode, each value *v* received from node *i* must be forwarded to the control center. In redundancy mode, if no consensus mechanism is being used (only the Social Layer is implemented), the gateway will receive several candidates for the value *v* from every node in the cluster. The gateway will inspect the values from it and check if they are in the acceptable range [v_{min}, v_{max}]. In an affirmative case, gateway *j* will provide positive feedback for that transaction *l* from node *i* (f_{ij}^{l} = 1). Otherwise, the feedback will be negative (f_{ij}^{l} = 0). After providing feedback for every transaction, the reputation *Ri* of the nodes will be updated, and the value provided from the node with the greatest reputation will be chosen as the definitive value *v*. Alternatively, if a consensus mechanism is used in redundancy mode, the gateway will only receive a single value *v* for each cluster, which will have to be forwarded to the gateway.

Due to the NVIS backbone network's unavailability during night hours (from 17:00 until 6:00), values received by the gateway during this period will be stored in the gateway's memory and forwarded to the control center later (when the NVIS links start functioning, at 6:00). On the contrary, values received during daytime (from 6:00 until 17:00) will be forwarded to the control center as soon as the gateway receives and processes them. As GTN-P stations do, the gateways also expect to receive an ACK packet for every payload packet they send to the control center. If an ACK packet is not received from the control center before a timeout T_{out}, the data packet will be retransmitted up to a maximum of 3 times.

Finally, the control center will receive all the transactions that had not been lost through the network. Each value *v* from the received payload by the control center from node or node cluster *i* will be considered a transaction *l*. The control center will compute the STR by comparing the received values for each payload with the hardcoded values.

The probability *Pb* of a node to have a byzantine fault is unlikely to be constant over time. As stated in [112], by associating the battery discharge to the WSN node aging process, the node reliability can be identified and associated with the battery charge level. Thus, following the model in [113], we can assume the impact of aging following a linear form, as defined in Eq. 5:

$$Pb(t) = Pb_0 + kt \, , \tag{5}$$

where Pb_0 is the probability of a node having a byzantine fault at time $t = 0$ and k is the aging factor. Thus, the probability of a node having a byzantine fault will increase hour by hour until its battery is completely drained at $t = t_d$, when it experiences a crash fault and $Pb(t_d) = 1$. However, this model will only be applied to GTN-P stations, which will be powered by batteries. On the contrary, we assume that the gateways will always have a constant power supply in our use case because they will be placed in the research base. Thus, their probability of experiencing a byzantine fault will remain constant over time, as defined in Eq.6:

$$Pb(t) = Pb_0 \tag{6}$$

As explained in section 2.5, the use of corrective methods to improve the data trustworthiness provoke, in practice, that the probability Pb_0 of a node experiencing a byzantine fault decreases, thus reducing the number of FSV. For that reason, different values of Pb_0 will be used in our simulations to emulate the use of different corrective methods.

2.6.4. Social Trustworthiness model

The reputational model for implementing social trustworthiness in our use case is a simplified version of the objective model defined in [46]. Our use case simplification assumes that all transactions will have the same weight, all nodes have the same computational capability, and the relationship factors between them are equal. Thus, the reputation R_i of node *i* can be measured as defined in Eq. 7:

$$R_i = \alpha O_i^{short} + (1 - \alpha) O_i^{long} \, , \tag{7}$$

where O_i^{short} is the short-term opinion of node *i*, O_i^{long} is the long-term opinion of node *i*, and α is a design value between [0, 1] to ponder the importance of short-term and long-term opinions. The short-term opinion of node *i* is measured as stated in Eq. 8:

$$O_i^{short} = \sum_{j=1}^{M} \sum_{l=1}^{L^{short}} C_{ij}\, f_{ij}^{l} / \sum_{j=1}^{M} \sum_{l=1}^{L^{short}} C_{ij} \;, \tag{8}$$

where M is the total number of nodes of the group, excluding node *i*, L^{short} is the number of last *l* transactions considered to be relevant for building the short-term opinion, f_{ij}^{l} is the feedback that node *j* gave to node *i* for transaction *l*, and C_{ij} is the credibility of node *j* to evaluate node *i*.

Analogously, the long-term opinion is calculated as defined in Eq. 9:

$$O_i^{long} = \sum_{j=1}^{M} \sum_{l=1}^{L^{long}} C_{ij}\, f_{ij}^{l} / \sum_{j=1}^{M} \sum_{l=1}^{L^{long}} C_{ij} \;, \tag{9}$$

where L^{long} is the number of last *l* transactions considered to be relevant for building the long-term opinion, and $L^{long} > L^{short}$. The credibility of node *j* to evaluate node *i* is calculated as shown in Eq. 10:

$$C_{ij} = \frac{R_j}{1+\log(N_{ij}+1)}, \tag{10}$$

where N_{ij} is the number of transactions between node *j* and node *i*.

2.6.5. Consensus model

A consensus protocol can be modeled by knowing the background traffic (bps) that introduces to the network and the number of byzantine nodes supported (Nb). In our use case, each group of redundant GTN-P stations will run the Practical Byzantine Fault Tolerance (PBFT) algorithm [114]. From [115] we can assume that the background traffic grows as the number of nodes participating in the consensus group is increased. Moreover, the number of tolerated byzantine nodes Nb is calculated as:

$$Nb = \left\lfloor \frac{Nt-1}{3} \right\rfloor \tag{11}$$

In the simulation, if more than Nb nodes experience a byzantine behavior, the agreement reached will have incorrect values. Otherwise, the resulting payload will contain the correct values.

2.6.6. *Tests definitions*

A summary of the characteristics of the simulation tests is shown in Table V.

TABLE V
SIMULATIONS PARAMETERS

Parameter	**Value**
Number of runs per test	30
Simulation duration	120 hours (5 days)
Simulation step	1 hour
Pb_0	[1x10^{-3}, 2x10^{-3}, 4x10^{-3}, 8x10^{-3}, 1x10^{-2}, 2x10$^{-2,}$, 4x10^{-2}, 8x10^{-2}, 1x10^{-1}]
k	5.7x10^{-5}
Routing protocol	[AODV, OLSR]
Consensus Mechanism	[None, PBFT]
Social Trustworthiness	[True, False]
Number of NVIS gateways	5
Number of GTN-P clusters per gateway	[8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096]
Number of GTN-P stations per cluster	[1-10]

Each different test will be run 30 times, which gives us the total amount of 113400 tests. Each test has a simulation duration of 5 days (120 hours), and the average value of the STR trustworthiness metric will be calculated. The different byzantine probabilities are proposed to simulate scenarios with different corrective methods that can reduce the byzantine probability of a node. On the other hand, two different routing protocols will be used to analyze which kind of method (reactive or proactive) has a better impact on the service trustworthiness. Moreover, the Consensus and Social trustworthiness Layers will be implemented or not to analyze their influence on the service performance. Finally, the impact of the number of nodes connected to each gateway will also be studied by varying them. In standard mode, there is only one GTN-P station per group (because there is no redundancy). That means that each of the 9 Pb_0 values must be tested against each number of GTN-P clusters per gateway, giving us a total of 90 possibilities. In redundancy mode, this number increases to 900 since the number of GTN-P stations per cluster varies from 1 to 10. If we sum the cases of standard mode, redundancy mode with consensus, and redundancy mode with social trustworthiness, we have a total of 1890 different cases, which are doubled to 3780 considering that we want to test the system with two different routing protocols. Considering that each test is repeated 30 times, a total of 113400 simulations have been run in the simulator.

2.7. Simulation results

After performing all the simulations, the average value of the STR has been calculated for every set of 30 runs per test. Three different operational modes for the telemetry service can be clearly identified: the standard mode, the redundancy mode with Social Trustworthiness Layer, and the redundancy mode with Consensus Layer. For every mode, an NxM-dimension grid with all the possible combinations of stimulation parameters is formed, where M is the number of different Pb_0 values (9 in our case), and N is the number of different GTN-P node combinations per gateway (10 in standard mode and 100 in redundancy mode). For every point in this grid, the average value of the trustworthiness STR metric is computed. If we link all the STR values for every neighboring point in the grid, a mesh with all the STR values is formed. We call this mesh the Trustworthiness Mesh. Figure 13 exhibits the Trustworthiness Mesh 3-dimensional graph for all the operational modes. Given that the differences between the AODV and OLSR scenarios' obtained results are negligible, only the results for the AODV scenarios are shown. Figure 14 shows different 2-dimensional perspectives of the Trustworthiness mesh graph to understand and analyze the results better.

From Figure 13 and Figure 14, we can analyze the behavior of the trustworthiness mesh. We can see how without redundancy, the STR is always lower than 0.8. Acceptable STR values (>0.6) are maintained if the number of sensor nodes is relatively low, although it decreases below 0.5 if the number of sensors per gateway is higher. Also, we can notice that the shape of the trustworthiness mesh is practically identical for all three cases in the “1xN sensors” zone (no redundancy). This means that, as was expected, adding the Social or the Consensus Layers does not improve the level of trustworthiness if there is no redundancy.

From Figure 13 and Figure 14(a), we can conclude that adding sensor redundancy and implementing the extension layers of our model improves the trustworthiness of the system, given that STR values greater than 0.8 are achieved. In cases of low redundancy (“2xN sensors” and "3xN sensors"), implementing the consensus mechanism does not improve the trustworthiness of the system when compared to the Social Trustworthiness case (the STR values are very similar). This is because, with two or three redundant nodes, the number of byzantine nodes tolerated by the consensus mechanism is still 0. Starting with four redundant nodes ("4xN sensors"), the consensus mechanism's effects start to be noticed, achieving better STR values than the Social Trustworthiness case.

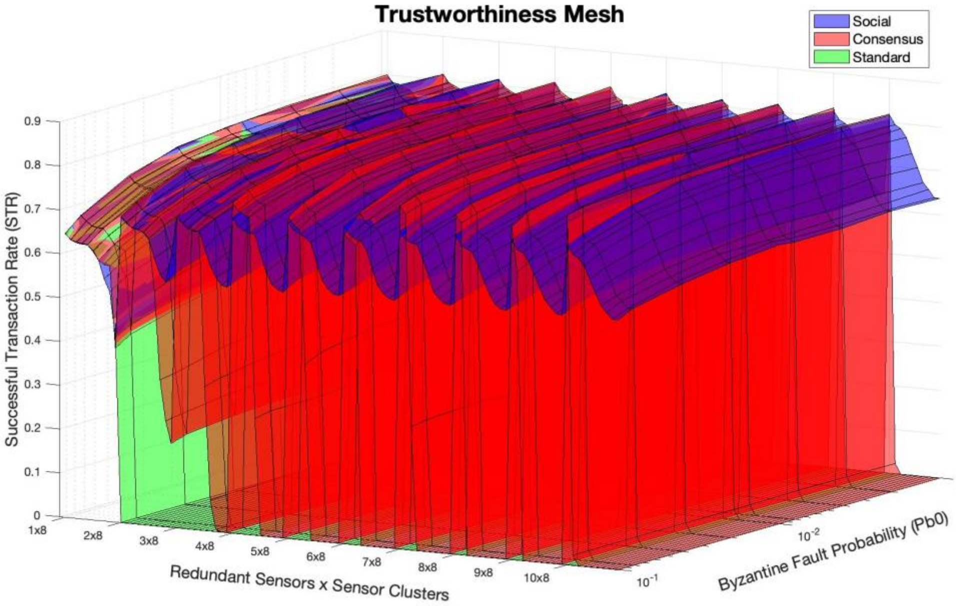
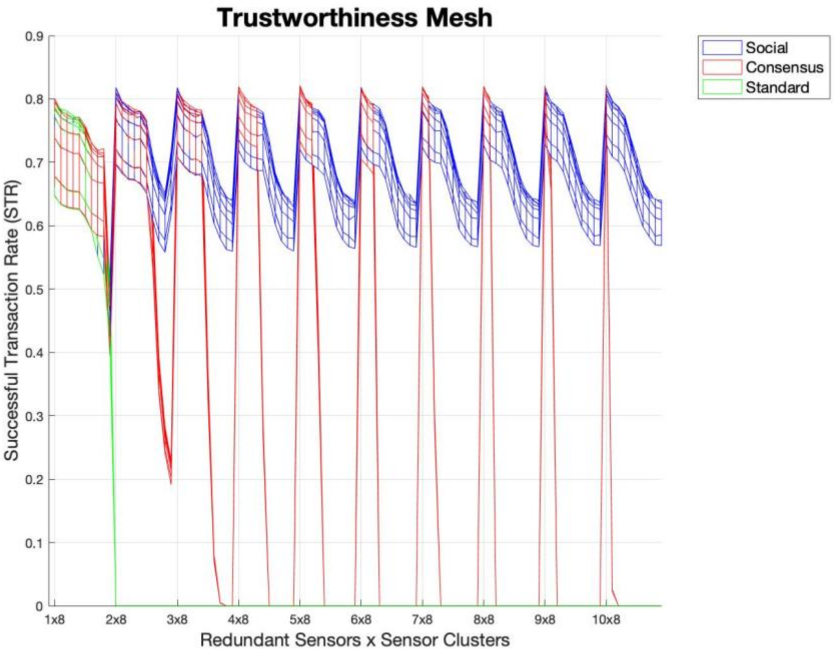
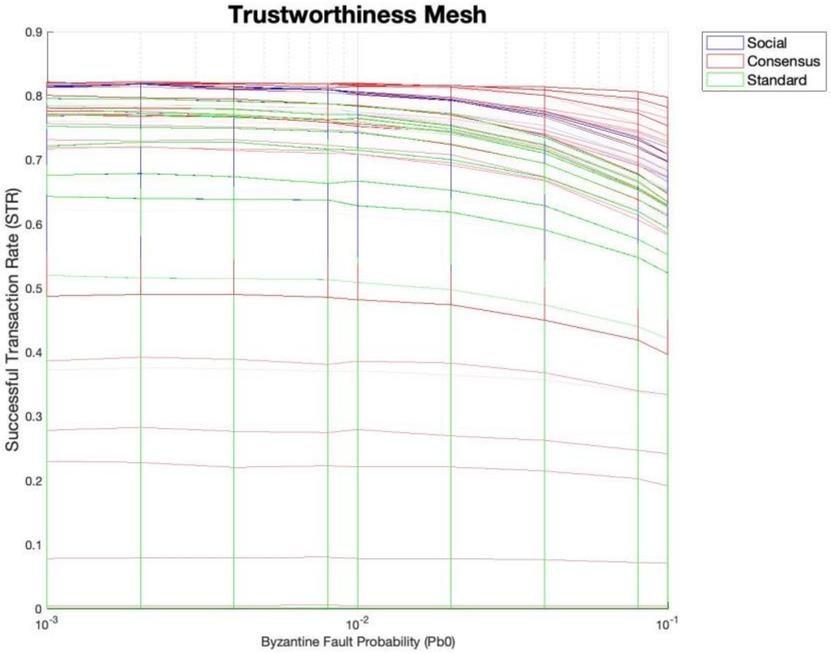


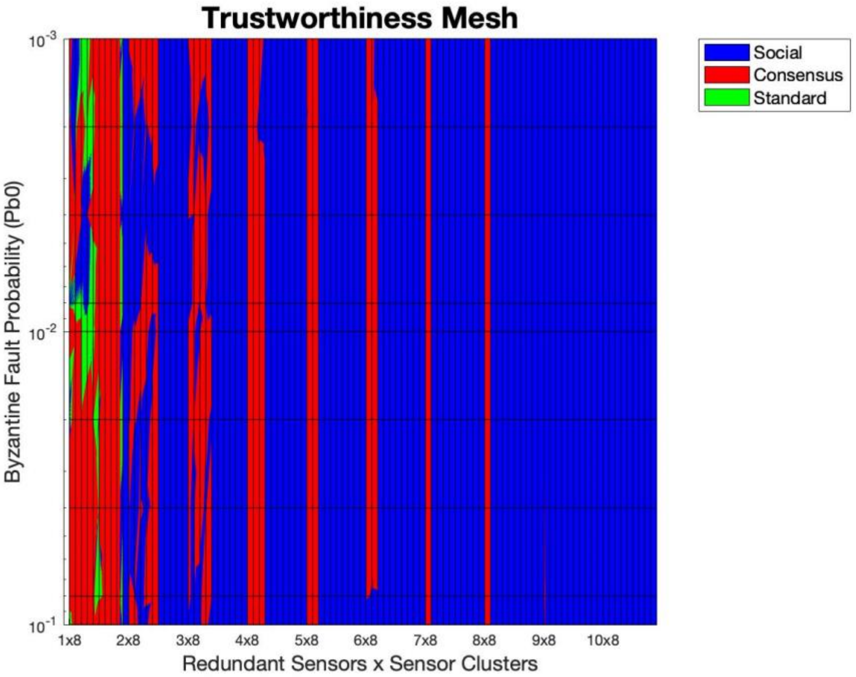
Figure 13. Trustworthiness Mesh graph for the standard operational mode (green), the redundancy mode with Social Trustworthiness (blue), and the redundancy mode with Consensus mechanism (red). The "Byzantine Fault Probability" axis has 9 discrete points, which are [$1\text{x}10^{-3}$, $2\text{x}10^{-3,}$, $4\text{x}10^{-3}$, $8\text{x}10^{-3}$, $1\text{x}10^{-2}$, $2\text{x}10^{-2,}$, $4\text{x}10^{-2}$, $8\text{x}10^{-2}$, $1\text{x}10^{-1}$]. The "Redundant sensors x Sensor Clusters" axis has 100 discrete points, which are [1x8, 1x16, 1x32, …, 1x4096, 2x8, 2x16, 2x32 ,…, 2x4096, …, …, 10x8 , 10x16, 10x32, …, 10x4096].


Trustworthiness Mesh
Social
Consensus
Standard
Successful Transaction Rate (STR)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1x8
2x8
3x8
4x8
5x8
6x8
7x8
8x8
9x8
10x8
Redundant Sensors x Sensor Clusters

(a)


Trustworthiness Mesh
Social
Consensus
Standard
Successful Transaction Rate (STR)
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
10^{-3}
10^{-2}
10^{-1}
Byzantine Fault Probability (Pb0)

(b)



(c)

Figure 14. 2-dimensional views of the Trustworthiness Mesh graph for the standard operational mode (green), the redundancy mode with Social Trustworthiness (blue), and the redundancy mode with Consensus mechanism (red). (a) Frontal view of the trustworthiness mesh (STR vs Number of nodes). (b) Profile view of the trustworthiness mesh (STR vs Byzantine node probability). (c) Top view of the trustworthiness mesh (Byzantine node probability vs Number of nodes).

However, as the Byzantine Fault Probability of the nodes gets lower (meaning the FSR is lower), the difference between the STR values from the Consensus mechanisms case and the Social Trustworthiness case gets smaller. This means that implementing a consensus mechanism is more appropriate when the probability of nodes experiencing byzantine behaviors is relatively high, and it is not necessary when this probability is low. In our cases, differences between STR values from both cases are not relevant from a $Pb_0 = 0.01$.

Moreover, the quantity of network traffic that the consensus mechanism adds, combined with the LoRa and NVIS networks' low bandwidth, provokes low scalability for this solution. We can see that by looking at the evolution of the Consensus Trustworthiness Mesh's STR values (red). We notice that as the number of sensors clusters increase, the STR values decreases until it drops to 0. This is because the nodes generate more traffics than the network supports. Thus, the network is congested, and the PDR rapidly decreases. Furthermore, the higher the number of redundant sensors per cluster, the sooner the STR dropping point (network saturation) happens. This resolves one of the looped dependencies postulated in section 2.5.4.

On the contrary, it seems that implementing a Social Trustworthiness approach is more robust to these variations. Even it does not achieve the same levels of STR as the Consensus mechanism case when the number of sensor clusters is low, its STR values never drop below 0.55, even in the scenario with more sensors and worse FSR. It is clear that the trustworthiness of the Social case is also affected if the number of sensor nodes increase (which implies more network load and lower PDR), but its STR does not drop drastically and can maintain acceptable values. Due to our use case's nature, the Social Trustworthiness implementation does not ostracize the nodes so that paths to most reputation nodes congest. That is because the reputational mechanism is used to rate nodes that measure the same value, which implies that they share the same network domain. Thus, the behavior of the other looped dependency postulated in section 2.5.4 remains uncertain.

From Figure 14(b), we can also conclude, as expected, that Data Trustworthiness has a direct affection to the overall system's trustworthiness. In all cases, as the Byzantine Fault Probability Pb_0 increases (meaning that more values are faulty sensed, increasing the FSR), the STR decreases.

Finally, Figure 14(c) shows, for each of the 900 possible scenarios, which is the most trustworthy option to implement the service. From this view, we can clearly see the robustness of the Social Trustworthiness case, showing how it gains ground as the number of sensors in the network increases.

2.8. Conclusions

This paper continues the SHETLAND-NET project's task to design a remote WSN for the Antarctic region using NVIS technology. The article focuses on the use case of deploying a group of interconnected remote Antarctic Wireless Sensor Networks providing an IoT telemetry service. A system and network architecture to implement the telemetry service has been defined, which uses LoRa at the access network and NVIS long backhaul links at the core network. The extreme conditions remote sensors need to work with, added to the challenges of NVIS links and a LoRa network without LoS, can provoke a degradation of the overall system's trustworthiness. In order to study the viability of the service to be implemented, we have proposed a model to measure and evaluate the trustworthiness of the system proposed. This trustworthiness model consists of four layers (two base layers and two extension layers) that can affect the Successful Transaction Rate (STR) trustworthiness metric.

The trustworthiness model and the system architecture has been validated using the Riverbed Modeler simulator. A total of 113400 tests have run under 3780 different scenarios. The results show that the defined system architecture can reach acceptable levels of STR (>0.6) in case a relatively low number of sensors are deployed, although it drops too much with a large number of sensors. Adding redundancy to the measured values with multiple sensors and applying a Social reputational mechanism improves the robustness of the system's trustworthiness, reaching higher STR values (>0.8) and never dropping below 0.55 even in high sensor-density scenarios. On the contrary, applying a consensus mechanism improves the trustworthiness when a low number of sensors is deployed. However, the STR values abruptly decrease as the number of deployed sensors increases.

Future work aims to study the influence of implementing a DTN architecture at the NVIS backbone network, given that it has characteristics of challenge networks. The authors also plan to study the viability of deploying a FANET in the access network to provide connectivity to sensors placed outside the coverage area of the current LoRa network.

2.9. Appendix A

Sensor node application pseudocode

```
int t, gateway_id, own_id, pk_id, tx_time, num_retries;
int data_values[32];
int node_reputations[N]; //N: number of redundnant nodes
float Pb0, Pb, k;
boolean consensus, social, is_leader, ack_received;
initializeVariables(Pb0,k,consensus,social, gateway_id, own_id, is_leader, ack_received);
for (t=0; t++; t<T_MAX){
   num_retries = 0;
   Pb = Pb0 + k * t;
   data_values = gatherData(Pb);
   if (consensus==TRUE){
      data_values = reachGeneralAgreement(data_values);
      if (social==TRUE){
       node_reputations = computeReputations();
       is_leader = checkLeader(node_reputations);
      }else{
       is_leader = chooseRandomLeader();
      }
      if (is_leader==TRUE)
         [tx_time, pk_id] = sendPacket(data_values, gateway_id, own_id);
      }else{
         pk_id = NULL;
      }
   }else{
      [tx_time, pk_id] = sendPacket(data_values, gateway_id, own_id);
   }
   if (pk_id != NULL){
      ack_received = checkAck(pk_id);
      while(ack_received==FALSE && num_retries<MAX_RETRIES){
         if (currentTime() >= tx_time + MAX_TIMEOUT){
            [tx_time, pk_id] = sendPacket(data_values, gateway_id, own_id);
            num_retries++;
         }
         ack_received = checkAck(pk_id);
      }
   }
   pk_id = NULL;
}
```

Gateway node application pseudocode

```
int own_id, sensor_id; control_ctr_id, pk_id, tx_time, num_retries;
int data_values[32];
int stored_values[N][32];
int node_reputations[N]; //N: number of redundnant nodes
float Pb;
boolean social, ack_received, data_pk_received;
initializeVariables(Pb, social, sensor_id, pk_id, own_id, control_ctr_id, ack_received, data_pk_received);
while(TRUE){
   num_retries = 0;
   if (dataPkReceived()==TRUE){
      [sensor_id, pk_id, data_values] = retrievePkData();
    if (social==FALSE){
      sendAck(pk_id, own_id, sensor_id);
      [tx_time, pk_id] = forwardDataPk(pk_id, data_values, gateway_id, sensor_id, control_ctr_id);
      ack_received = checkAck(pk_id);
        while(ack_received==FALSE && num_retries<MAX_RETRIES){
         if (currentTime() >= tx_time + MAX_TIMEOUT){
          [tx_time, pk_id] = forwardDataPk(pk_id, data_values, gateway_id, sensor_id, control_ctr_id);
          num_retries++;
         }
         ack_received = checkAck(pk_id);
        }
      ack_received=FALSE;
      num_retries=0;
    }else{
      stored_values[sensor_id] = data_values;
        node_reputations[sensor_id] = computeReputation(data_values);
        if (roundIsFinished()==TRUE){
       [data_values, sensor_id] = chooseData(node_reputations, stored_values);
       [tx_time, pk_id] = forwardDataPk(pk_id, data_values, gateway_id, sensor_id, control_ctr_id);
         ack_received = checkAck(pk_id);
           while(ack_received==FALSE && num_retries<MAX_RETRIES){
              if (currentTime() >= tx_time + MAX_TIMEOUT){
                 [tx_time, pk_id] = forwardDataPk(pk_id, data_values, gateway_id, sensor_id,
                                          control_ctr_id);
            num_retries++;
            }
           ack_received = checkAck(pk_id);
          }
         ack_received=FALSE;
       num_retries=0;
       }
      }
    }
  }
```

Control center application pseudocode

```
int own_id, sensor_id; gateway_id, pk_id;
int data_values[32];
boolean data_pk_received;
initializeVariables(sensor_id, pk_id, own_id, gateway_id, data_pk_received);
while(TRUE){
    if (dataPkReceived()==TRUE){
        [sensor_id, gateway_id, pk_id, data_values] = retrievePkData();
        storeData(data_values, sensor_id);
        computeSTR(data_values);
        sendAck(pk_id, gateway_id);
    }
}
```

Chapter 3

DTN TRUSTWORTHINESS FOR PERMAFROST TELEMETRY IOT NETWORK

The SHETLAND-NET research project aims to build an Internet of Things (IoT) telemetry service in Antarctica to automatize the data collection of permafrost research studies on interconnecting remote wireless sensor networks (WSNs) through near vertical incidence skywave (NVIS) long fat networks (LFN). The proposed architecture presents some properties from challenging networks that require the use of delay tolerant networking (DTN) opportunistic techniques that send the collected data during the night as a bulk data transfer whenever a link comes available. This process might result in network congestion and packet loss. This is a complex architecture that demands a thorough assessment of the solution's viability and an analysis of the transport protocols in order to find the option which best suits the use case to achieve superior trustworthiness in network congestion situations. A heterogeneous layer-based model is used to measure and improve the trustworthiness of the service. The scenario and different transport protocols are modeled to be compared, and the system's trustworthiness is assessed through simulations.[3]

Keywords: Transport Protocols; Trustworthiness; Antarctica; IoT; NVIS; remote WSN; LFN.

3.1. Introduction

Research studies from multiple disciplines are carried out every year in Antarctica [10]. Researchers are temporarily placed in Antarctic base stations, normally located in the peripheral areas of the continent. One of the main challenges in Antarctica is its lack of conventional telecommunication systems [10], which hinders the deployment of wireless sensor networks

[3] The work reported in this chapter was published as the paper entitled "DTN Trustworthiness for Permafrost Telemetry IoT Network" in the Remote Sensing journal, 2021, 13(22), 4493, https://doi.org/10.3390/rs13224493 Authors: Adrià Mallorquí, Agustín Zaballos, Alan Briones.

(WSNs). This fact reduces the possibilities of carrying out research studies (e.g., automation of data collection and remote bases interconnection).

To overcome these difficulties, our research project, the SHETLAND-NET, proposes the use of near vertical incidence skywave (NVIS) high-frequency (HF) radio links to provide low-consumption Antarctic communications, continuing previous research on ionospheric communications [116]. The ionosphere reflects this signal, providing a long backhaul link of a 250 km radius coverage area [11], [18]. Networks using this type of links can be classified as long fat networks (LFNs), which are characterized by having long links with a bandwidth delay product (BDP) greater than 1 × 105 bits (12,500 bytes) [27], following Eq. 12, where the link bandwidth (BW) is expressed in bits per second (bps) and the round-trip time (*RTT*) in seconds (s).

$$BDP = BW \times RTT \tag{12}$$

The NVIS technology can be used to interconnect remote base stations [19]. Our final goal is to deploy a telemetry service by interconnecting remote WSNs [83], which will help in the automatization of data gathering for Antarctic research studies. This deployment will be carried out during the next Antarctic campaign in the field. However, this communication technique can be error-prone due to the variant properties of the ionosphere. It may present typical challenging network issues [25], such as intermittent connectivity, end-to-end disconnection, and variable error rates, which could degrade the performance of the overall offered IoT service.

Therefore, before the deployment phase of our project, we had to study and try to anticipate the expected trustworthiness of the IoT telemetry service we want to deploy. For this reason, we defined a model to assess the trustworthiness of our proposed system [83]. This enabled us to foresee the possible trustworthiness issues that might arise during the campaign in the field and decide on the respective countermeasures.

For our work, we focus on the use case of automating the monitoring of Ground Terrestrial Network-Permafrost (GTN-P) stations [20], which are used in permafrost research studies.

Each of these GTN-P stations senses 32 different values hourly, which need to be remotely monitored from a control center. During the Antarctic campaign, we will deploy a test scenario. WSNs will be placed in two locations: the Spanish Juan Carlos I Base in Livingston Island, and the Uruguayan Artigas Base in King George Island, both part of the South Shetland Islands. The Artigas Base will provide Internet connectivity, so data gathered from the WSNs can be reached remotely. However, sensors in the Juan Carlos I Base will not have direct Internet connectivity, and the data from these sensors will need to be sent through an NVIS link to the Artigas base in order to reach the Internet. Figure 15 shows the test scenario in Antarctica.

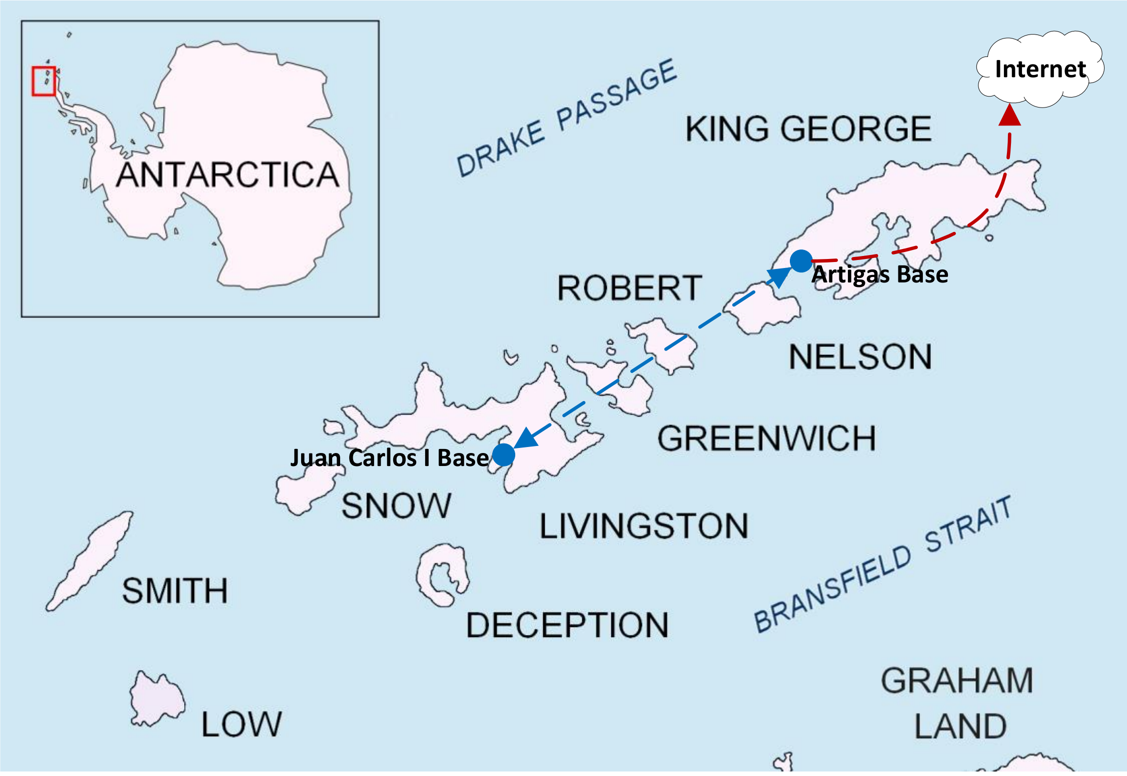


Figure 15. Map of the South Shetland Islands in Antarctica [117], showing the position of the WSNs (blue circles) during the test scenario of the campaign. The NVIS link is represented with the discontinuous blue line, and the Internet connectivity is represented with the discontinuous red line. The reproduction of the image was slightly modified under a Creative Commons License (CC BY-SA 3.0).

As seen in previous research [18], the main drawback of the NVIS link is its unavailability during the night, given that the ionosphere's characteristics vary drastically due to solar activity. For this reason, we decided to adopt a delay tolerant network (DTN) technique to opportunistically send all the data collected during the night as a bulk data transfer when the NVIS link becomes available in the morning. This complex scenario required a trustworthiness assessment to analyze its feasibility to be deployed in Antarctica before the campaign [83]. As shown in our first round of simulations, performing this opportunistic bulk data transfer in an LFN that presents network challenges could degrade the system's performance (packet losses) due to network congestion caused by the large quantity of data sent. On the other hand, in prior work, we also analyzed the suitability of different transport protocols for LFNs and designed a new one, the Enhanced Adaptive and Aggressive Transport Protocol [27], [28]. Given that the NVIS links can also be considered as LFNs and given the strong performance that some modern transport protocols showed in our tests, we believed that it was crucial to assess how the use of modern transport protocols could improve or affect the performance and trustworthiness of the service, especially in this congestion situation provoked by the DTN technique. Having collected the initial results and analyzed the system's trustworthiness in previous work with the standard transport protocols of the devices' operative systems, this paper studies the trustworthiness and compares the usage of different transport protocols by modeling the scenario in the Riverbed Modeler. The paper contributions are as follows:

1. The definition and concretion of the remote sensor network architecture that will be deployed in Antarctica, detailing the type of nodes, protocol stack, and communication techniques that will be used.

2. The modeling of the Antarctic scenario in the simulator. To perform the simulation tests, we modeled the communication media (LoRa and NVIS), the telemetry application, the faulty behavior of Byzantine nodes, the social trust management and the consensus algorithms, the DTN technique, and the tested transport protocols.

3. The assessment and analysis of the results using our proposed trustworthiness model. From this analysis, we conclude which transport protocol best suits our use case and propose a modification of the scenario to be deployed in Antarctica.

The rest of this paper is structured as follows. Section 3.2 describes the related work in DTNs, transport protocols, and a system's trustworthiness. Section 3.3 defines our use case's network architecture. Section 3.4 reminds our proposed model to measure and evaluate a system's trustworthiness. Section 3.5 describes the simulation tests. Sections 3.6 and 3.7 present and discuss the obtained results, respectively. Finally, section 3.8 concludes the paper.

3.2. Related work

3.2.1. Delay Tolerant Networks

The DTN was first presented as an alternative network architecture designed for challenging networks [25] which suffer from high bit error rates, lack of end-to-end connectivity, and long delays [69]. It was initially designed for interplanetary communications in space [68], given the number of disconnections that this network suffers. However, over the years, many other types of terrestrial networks have emerged in response to similar problems (e.g., underwater networks [72], wildlife tracking networks [118], sparse wireless sensor networks [119], and vehicular networks [120]).

Conventional TCP/IP protocols are not suitable for these kinds of environments. In contrast, the RFC 5050 presented a DTN protocol, the Bundle Protocol (BP) [24], which enabled message delivery to cope with all the issues of challenging networks, even if the source and the destination were never connected to the network simultaneously. The BP is based on a store–carry–forward overlay network, where "bundles" are transported through endpoints on top of the transport layer of the OSI model when a connection opportunity is present between two endpoints. The BP version 7 draft was recently released [107], which introduces new features, such as optional CRCs for nonprimary blocks, and proposes other changes to make it simpler, more capable, and easier to use. Many implementations of the Bundle Protocol adapted to the constraints of IoT and WSNs exist nowadays, such as IBR-DTN [121], μDTN [122], and DTN7 [107], among others.

However, other DTN approaches are not based on the BP but use their own routing protocol designed to be disruption and delay-tolerant [25]. DIRSN [123], PASR [124], RMDTN [125], and PRoPHET [126] are some examples of this kind of approach. Moreover, we can find other schemes that mix DTN with other kinds of technologies, such as opportunistic networking [127], [128], machine to machine (M2M) communications [129], information-centric networking (ICN) [130], and fog computing [131].

As stated before, in our use case, we will use an opportunistic networking technique to send all the data collected during the night in the morning, when the NVIS link comes available, as a bulk data transfer. This kind of approach is possible because our research group has studied the behavior of the ionosphere and NVIS links in prior research [18], and were aware that the link is down at nighttime and becomes available at sunrise. However, we also know this bulk data transfer provokes network congestion, degrading the system's performance with packet losses. For this reason, it is crucial to study how modern transport protocols can help improve this performance, especially in LFNs such as the NVIS links.

3.2.2. Transport Protocols

The performance of transport protocols for network communications has been a topic under discussion and development since the Internet was conceived [27]. The first extensions of the original Transmission Control Protocol (TCP) were [52] TCP Tahoe, TCP Reno, TCP New-Reno, TCP SACK, and TCP-Vegas, which included new mechanisms such as the fast retransmit, the fast recovery, the packet pair link estimation, the duplicated acknowledgment (DUACK), and the selective acknowledgment (SACK).

However, these legacy transport protocols suffered performance degradation over some types of networks, including LFNs. The LFN concept and its effects on TCP performance were firstly defined and detailed in the Request For Comments (RFC) 1072, which was obsoleted by the RFC 1323 to finally become the standard RFC 7323. Some TCP variants and other transport protocols developed during the last decade have improved their performance over LFNs [27]. Some of these are Scalable TCP (S-TCP) [53], FAST TCP [54], High-Speed TCP (H-TCP) [55], Binary Increase Control TCP (BIC-TCP) [56], and its evolution: TCP CUBIC [26]. TCP CUBIC (RFC 8312) is the most commonly used transport protocol nowadays, given that it is the TCP variant used by default on most operating systems. However, most of these protocols consider that packet loss always occurs due to network congestion, reducing the congestion window. This assumption is false for wireless links, where packets can also be dropped for other reasons (e.g., fading, channel interference) [28]. Under these circumstances, reducing the congestion windows might also degrade the transmission performance, achieving lower throughput [28].

For this reason, other transport protocols, such as Performance-oriented Congestion Control (PCC) [62], TCP Veno [63], TCP Westwood+ [64], Dynamic TCP [65], Jitter TCP [66], and Jitter Stream Control Transmission Protocol (JSCTP) [67] are focused on implementing mechanisms to detect if lost packets occur due to network congestion or random channel loss. They only reduce the congestion window in the first case, achieving better performance [28].

In addition, other modern transport protocols, such as TCP BBR [58], Copa [59], Indigo [60], and Verus [61], can achieve high performance, as proven in several physical tests carried out by Stanford University's platform Pantheon [60]. TCP BBR is one of the top-performance protocols, managing the maximum bandwidth with the minimum RTT. Copa is a practical delay-based protocol that fixes an RTT target and adjusts its congestion windows based on the minimum RTT and the standing RTT measured during data transfers. Indigo is a data-driven protocol that uses a machine-learning congestion control scheme that learns from previous performance data. Verus is a transport protocol oriented to cellular networks that relates the congestion windows with delay variations through short-term RTT measurement.

Moreover, given that the aforementioned protocols did not meet the performance requirements of our cloud data-sharing use case from previous work [28], we presented the Adaptive and Aggressive Transport Protocol (AATP) [27] and its evolution, the Enhanced AATP (EAATP) [28], which incorporates mechanisms to differentiate the packet losses' cause, fairly adapting its sending rate accordingly to the network circumstances. The performance in these tests was solid, both in simulations and in a physical testbed with an LFN emulator, showing better results than other protocols, maximizing throughput and minimizing packet losses [27], [28]. Figure 16 shows a summary of the tests' results. However, we did not know how these protocols (including ours) could affect the trustworthiness of a system, especially in the use case of this paper. For this reason, we thought that we needed to assess whether using the EAATP in the remote Antarctic WSN use case could improve the system's performance and trustworthiness, especially in congestion situations.

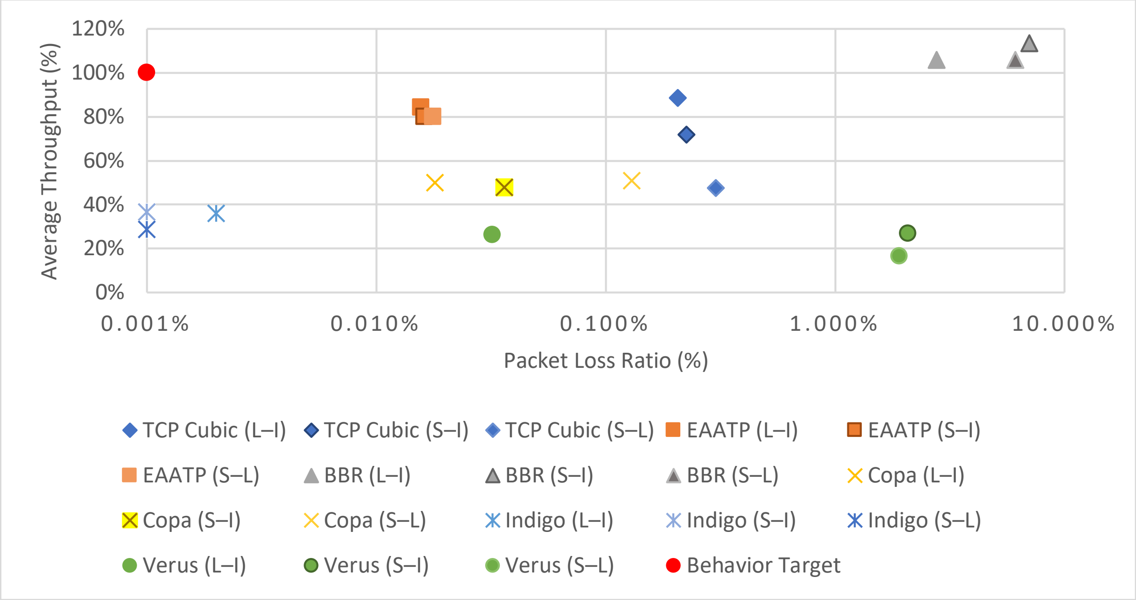


Figure 16. Results of average throughput (%) vs. packet loss ratio (%) of the transport protocols tested in previous work [28]. To represent the graph in semilogarithmic scale, the packet loss ratio values of 0% are represented as 0.001% in the graph. Each transport protocol was tested in three LFN scenarios: London to Iowa (L–I), Sidney to Iowa (S–I), and Sidney to London (S–L).

3.2.3. Trustworthiness in Cyber Physical Systems

A cyber physical system (CPS) is defined as a system with integrated computational and physical capabilities. Wireless sensor networks, smart grids, and some IoT devices are examples of CPSs [32]. Even though there is no consensus in the literature to define the trustworthiness property and its scope [132], we can define a CPS's trustworthiness, in general terms, as the property of behaving as expected under adversarial conditions [32]. Network malfunction, Byzantine errors, and faulty nodes are examples of adverse conditions that can affect a system's trustworthiness. Some authors limit this definition to system security issues only [133], while others propose a broader scope and relate trustworthiness with other terms such as resilience, availability, reliability, scalability, maintainability, heterogeneity, data quality, hardware resources, and fault management policies [132]. We can find many approaches to measuring or providing trustworthiness in literature, referring to different elements. We classify them into four main categories [83]:

1. Data trustworthiness: It is defined as the possibility to ascertain the correctness of the data provided by the source [34]. Many methods use different approaches that try to detect faulty nodes, false alarms, and sensor misreading. For instance, authors in [36] use a fog computing architecture to detect, filter, and correct abnormal sensed data. In addition, authors in [37] present a data intrusion detection system to trigger false data from malicious attacks.

2. Network trustworthiness: Defined as the likelihood of a packet to reach its destination unaltered despite the adversities (e.g., link failure, link saturation, or malicious attacks, among others), it is a relevant aspect to consider in challenging networks [38], such as the use case we propose. The network's performance and trustworthiness have been addressed from several perspectives, such as channel coding [134], transport protocols [28], dynamic routing and topology control protocols [41], [135], and DTN architectures and protocols [25].

3. Social trustworthiness: This field has become more popular since the appearance of the Social Internet of Things (SIoT) [44], [136]. In SIoT trustworthiness, objects or network nodes interact and establish social relationships, which are used to define trust and reputation models that take into account several input parameters. Authors in [46] present a model that considers factors as the computational capabilities of the nodes, the type of relationship between them, the total number of transactions, the credibility of a node, and the feedback provided by other nodes, among others. Authors in [137] present an evolution of the aforementioned trust management model, which applies a machine learning algorithm to calculate novel parameters such as the goodness, usefulness, and perseverance of a node. Thanks to these parameters, this upgraded trust model is resilient to more types of malicious node attacks. Authors in [47] propose another model that defines the input parameters as the expected gain on success, the expected damage on a failure, the expected cost, the expected result, and the goal. Authors in [48] define a decentralized self-enforcing trust management system which is based on a feedback system and reputational secure multiparty calculations to ensure the privacy of each party's provided data.

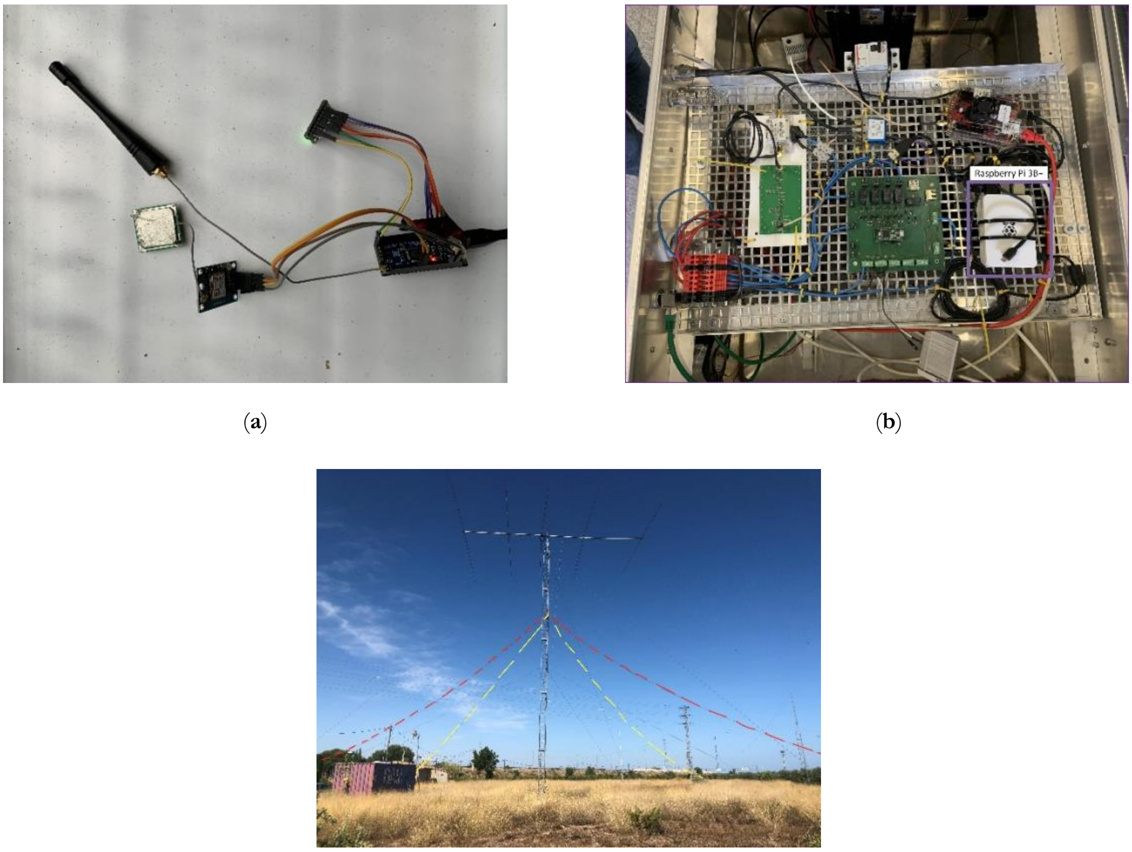
4. Consensus: It represents a state where all the participants of the same distributed system agree on the same data values [50]. Consensus protocols can be classified into two major groups: proof-based consensus and Byzantine consensus. The first group is related to blockchain technology, where all participants compete against each other to mine a block, and the most commonly used protocols are proof-of-work, proof-of-stake, and their variants [50]. The main drawback of these protocols for the IoT is that devices usually have lesser hardware resources and low processing power, which make the mining tasks of blockchain extremely difficult [50]. On the other hand, Byzantine-based protocols implement voting-based mechanisms to reach an agreement rather than competing among them, generating less resource consumption in general. Their main drawback is the number of messages that need to be delivered through the network to reach an agreement. Some well-known protocols from this category are Practical Byzantine Fault Tolerance (PBFT), RAFT, PaXoS, and Ripple, among others [50].

3.3. Remote sensor network architecture

As stated before, the use case of this article is an IoT telemetry service to monitor remote WSNs in Antarctica interconnected through NVIS LFNs. The monitored data are used for permafrost studies and are gathered by GTN-P stations [20], which are the sensors of our network. Each of these GTN-P stations senses 32 different values hourly, and these values must reach the remote control center in Europe.

The GTN-P stations are equipped with a Moteino [138], an Arduino-based board designed for low-power consumption applications. The Moteino will send, through LoRa, its sensed values to a Raspberry Pi 3B+ gateway acting as a concentrator (access network). LoRa was preferred over other alternatives (e.g., Sigfox, NB-IoT) as the access network protocol because of its teleoperator independence. The LoRa network will be configured with a transmission frequency of 868 MHz, a code rate CR3 (4/7), and a spreading factor SF7, obtaining a 125 kHz channel bandwidth with a bit rate of 5.47 kbps. As proved in [13], this configuration can offer a coverage range of up to 30 km in Antarctica. Figure 17(a) shows the Moteino board with the LoRa transceiver that will be used during the campaign to collect and forward the data from the GTN-P stations.

The Raspberry Pi 3B+ gateway will forward these data through NVIS links (backbone network) to the Internet edge router in the Uruguayan Artigas Base in Antarctica. NVIS was preferred over satellite communication because the latter presents coverage issues in polar zones and has a higher economic cost [11]. The NVIS nodes will be configured to transmit at the 4.3 MHz transmission band, with a channel bandwidth of 2.3 kHz and a bit rate of 4.6 kbps. As in [11], we will increase the NVIS transmission reliability with an FEC convolutional code (1/2 rate code) and interleaving. With this configuration, an NVIS link range is up to 250 km. Figure 17(b) shows the NVIS node with the Raspberry Pi 3B+ gateway, and Figure 17(c) shows the NVIS antenna (inverted vee antenna).



(**a**)

(**b**)

(**c**)

Figure 17. Antarctic WSN Hardware. (**a**) Moteino node with LoRa transceiver. (**b**) NVIS node with Raspberry Pi 3B+ gateway. (**c**) NVIS inverted vee antenna.

From the closest NVIS node to the Internet edge router (the one with Internet connectivity), data will be pushed to the Internet. From this moment, data monitoring and gathering will be available remotely from the control center. Figure 18 shows the network architecture diagram of the remote WSN.

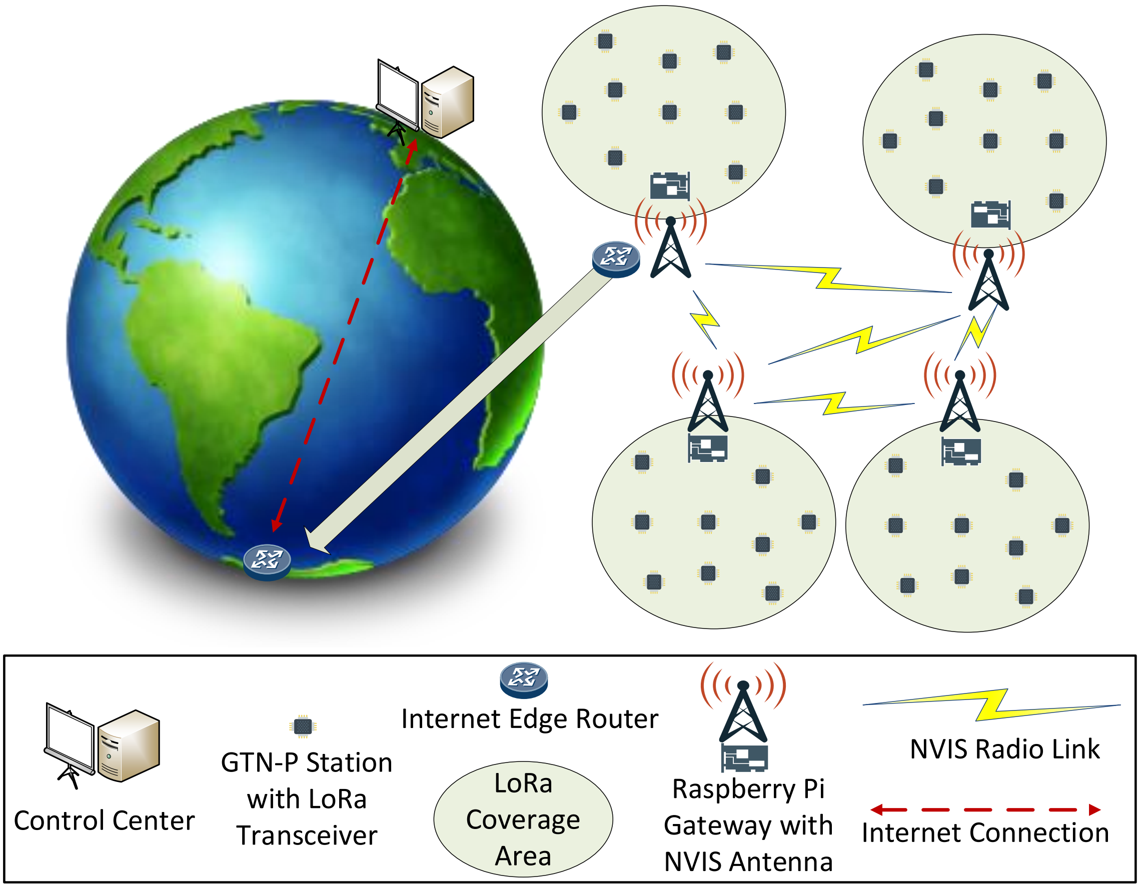


Figure 18. Network architecture of the remote WSN providing the IoT telemetry service.

The Artigas Base's Internet connectivity is supposed to have high reliability, so our trustworthiness assessment is focused on the access network (LoRa) and the backbone network (NVIS). As mentioned before, the reliability of NVIS links is very dependent on the ionosphere state, so it is not possible to send data during the night as all of it would be lost. For this reason, we believed it was necessary to apply a DTN technique to prevent the loss of data gathered during the night. In our case, we apply the DTN in the backbone network, as it is more likely to suffer from a lack of end-to-end connectivity, long delays, and network disruption.

Given that, in our case, we can predict a specific time slot when the NVIS links do not work (nighttime), we opted to implement a lightweight DTN approach, opportunistically sending the data collected during the whole night as a bulk transfer when the NVIS channel becomes available in the morning. Each concentrator should have collected 13 different sets of sensed values from each GTN-P station during the night. Our project requires that, on average, at least 9 out of the 13 datasets gathered from each station (around 70%) reach the control center correctly [83].

The DTN is usually implemented as an overlay network below the application layer of the Open Systems Interconnection model (OSI model) and needs a convergence layer as an interface to connect to the lower layers of the protocol stack. Figure 19 shows the protocol stack from our use case.

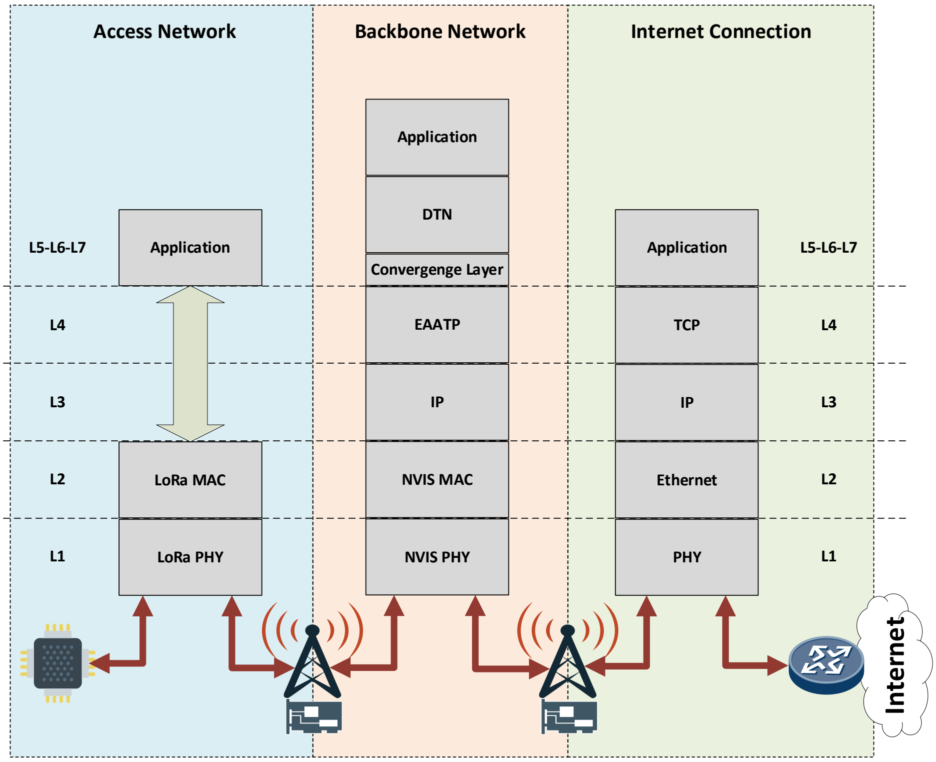


Figure 19. Antarctic IoT network protocol stack.

In the access network, LoRa uses a reduced protocol stack, thus avoiding layers 3 to 6 of the OSI model. The application data is directly encapsulated into the LoRa data link layer. Once data arrives at the NVIS node, the protocol stack introduces all the OSI model layers and adds the DTN layer below the application layer. The DTN layer needs a convergence layer to adapt to the transport protocol below. Figure 19 shows the EAATP as the transport protocol in the backbone network, although we test diverse transport protocols in our simulations, as discussed in section 3.5. Finally, when the data arrives at the last NVIS node and must be forwarded through the Internet, the DTN and convergence layers are removed. The common, well-known TCP/IP model is used, given that end-to-end connectivity at this zone is assumed.

3.4. Trustworthiness model specification

In this section, we summarize our trustworthiness model. Further details of the model can be found in [83]. To the best of our knowledge, none of the prior analyzed trustworthiness approaches have tried to include all of the four trustworthiness areas but have instead focused on one or some of them without considering the interdependencies between all the four categories. This could lead to assuming incorrect reasons for a lower trustworthiness level and implementing the wrong countermeasures to improve it. For this reason, we believed it necessary to design our model to measure a system's trustworthiness level, which includes the four categories mentioned above and helps us to anticipate and identify the possible weaknesses of our IoT telemetry system.

We propose a layer-based model to measure the trustworthiness and evaluate a system’s performance (in our case, a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service). This model is characterized by 1) two baseline layers (data trustworthiness layer and network trustworthiness layer), 2) two extension layers (social trustworthiness layer and consensus layer) that include optional functionalities, and 3) the interaction between all of them. The data trustworthiness, network trustworthiness, social trustworthiness, and consensus layers can collectively define a system's trustworthiness.

We postulate that each layer is characterized by its definition (scope), how the trustworthiness of that layer is measured (metric), and how the value of this metric can be improved (countermeasures).

3.4.1. Data trustworthiness layer

This layer aims to ascertain the correctness of the source's collected data. We propose the measurement of this layer's trustworthiness with the metric faulty sensing ratio (*FSR*), defined in Eq. 13 as the proportion of false sensed values (*FSV*) by all nodes and total sensed values (*TSV*) in a defined period. The lower the *FSR*, the better the data trustworthiness.

$$FSR = \frac{FSV}{TSV} \tag{13}$$

Corrective methods (e.g., [36], [37]) which try to detect abnormal data (*FSV*) stored in the source node due to a sensor malfunctioning, a misreading of the sensed data, or erratic writing in the node's memory, can be applied. Additional examples of corrective methods are hashes, checksums, and parity bits, among others (see Figure 20).

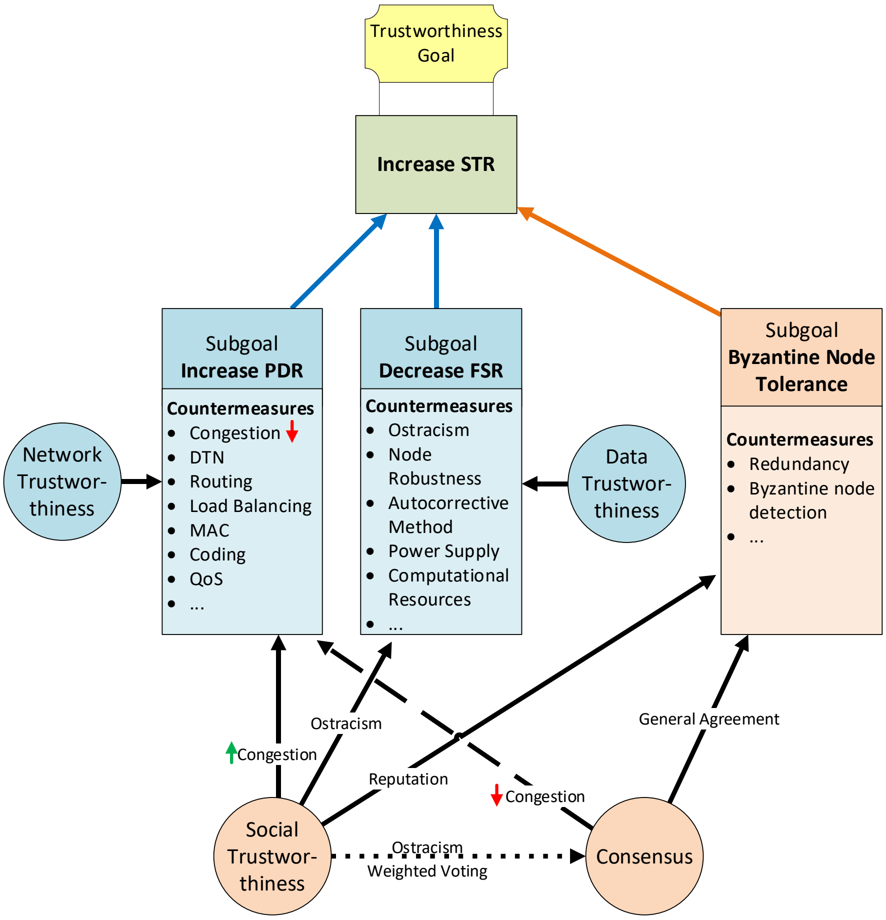


Figure 20. Trustworthiness model goals and countermeasures relationship [83].

3.4.2. Network trustworthiness layer

This layer is responsible for assuring that a packet reaches its destination on time and unaltered despite the adversities (e.g., link failure, network congestion). We measure this layer's trustworthiness with the packet delivery ratio (*PDR*), defined in Eq. 14 as the quotient between the total number of packets correctly received (*Pr*) by all nodes and the total number of packets sent (*Ps*) by all nodes in the same time slot. The higher the *PDR* is, the better the network's trustworthiness.

$$PDR = \frac{Pr}{Ps} \tag{14}$$

At the network trustworthiness layer, transmission coding techniques [139] are used to increase the robustness of the transmitted signal. Routing protocols and quality of service (QoS) mechanisms are used to find the best path from a source to a destination by quantifying the quality or performance of each link in the network [41], [135]. Congestion control algorithms and other mechanisms of transport protocols [28] can also improve network trustworthiness. In the case of challenge networks, DTN overlay architectures and protocols, such as the Bundle Protocol [25], can also improve network trustworthiness (see Figure 20).

3.4.3. Social trustworthiness layer

This layer is responsible for leveraging the capability to autonomously establish social inter-object relationships to improve the trust between them and the correctness of the collected data. We measure this layer's trustworthiness with the successful transaction rate (*STR*), calculated as the proportion between the number of successful transactions (*ST*) and the total number of transactions (*TT*) in a defined time slot, as stated in Eq. 15. A transaction *l* is considered successful when a node *j* expects to obtain some information or data (*v*) from node *i* before a defined maximum reception time (Trx_{max}) and receives it as expected, thus providing good feedback (f_{ij}^{l} = 1) for that transaction to node *i*. The higher the *STR* is, the better the social trustworthiness.

$$STR = \frac{ST}{TT} \tag{15}$$

Most solutions tend to use reputational mechanisms to determine which nodes to trust when exchanging information. This reputation is commonly based on the feedback of previous transactions to build an opinion of the node's trustworthiness [46], [48], [137].

3.4.4. Consensus layer

This layer is responsible for reaching a state where all group participants agree on the same response or result. We measure this layer's trustworthiness with the Byzantine node tolerance (*BNT*), defined as the proportion of supported Byzantine nodes (*Nb*) that can participate in the consensus system without affecting the correctness of the general agreement and the total number of nodes (*Nt*) that participate in the consensus system, as defined in Eq. 16. A node is considered Byzantine if it experiences a crash or soft fault that incapacitates it to behave as expected or if it does not behave as expected on purpose (malicious node). The higher the *BNT* is, the higher the probability of reaching a correct general agreement (GA).

$$BNT = \frac{Nb}{Nt} \tag{16}$$

Several mechanisms can be used to reach a decentralized GA that all group nodes consider to be true. Theoretically, if the number of Byzantine nodes is higher than 50% of the total number of participating nodes, none of the consensus mechanism will reach a benevolent agreement [50]. A drawback of these mechanisms is that participating nodes need to exchange a large quantity of messages between them to reach a consensus, which can degrade the performance of low-bandwidth networks.

3.4.5. Trustworthiness layers relationships

Figure 20 synthesizes our trustworthiness model actors. Blue-colored elements form part of our model baseline layers, and orange-colored elements form part of the extension layers. The primary goal is to increase the *STR* to provide better trustworthiness. Three main factors directly help increase the *STR*: 1) Mitigate/tolerate Byzantine errors; 2) decrease the *FSR*; and 3) increase the *PDR*. These factors can be seen as secondary goals that leverage the success of the final goal to provide trustworthiness. Each of these secondary goals can be accomplished by implementing a set of actions or countermeasures. Each of these countermeasures commonly affects only one of the goals. Moreover, two transversal actions impact more than one secondary goal. These transversal actions implement the extension layers of our model: the social trustworthiness layer and the consensus layer.

In Figure 20, continuous-line arrows indicate a positive outcome, discontinuous-line arrows indicate a negative outcome, and dotted-line arrows indicate an uncertain outcome. On the one hand, the use of social trustworthiness can reduce network congestion thanks to the ostracism of nodes with the worst reputation by only sending the values from nodes with the highest reputation to the control center. In addition, social trustworthiness also helps to reduce the *FSR* thanks to the ostracism of bad reputation nodes. It also leverages the mitigation of Byzantine errors because only values from high reputation nodes (leaders) are trusted. On the other hand, implementing a consensus mechanism mitigates Byzantine errors thanks to the general agreements reached by all nodes from a consensus group. Contrarily, the consensus layer can negatively affect the *PDR*, given that it introduces a considerable amount of extra traffic to the network, which could lead to link congestion.

3.5. Simulation tests

As mentioned before, the first tests we performed to assess the system's trustworthiness in this use case [83] showed that it was possible to have an STR greater than 0.7 in some circumstances. However, we noticed that the DTN approach of using opportunistic bulk data transfers when the NVIS link becomes available produced network congestion in these periods. On the other hand, we also compared, evaluated, and designed modern transport protocols for heterogeneous LFNs to improve the performance of data transfers over this type of network. Our tests showed that our protocol, the EAATP, maximized throughput and minimized packet losses in LFNs. However, we did not evaluate how the use of these protocols could affect the trustworthiness of a system. Given that the NVIS links in the remote Antarctic WSN use case can be considered an LFN (with a *BDP* greater than 12,500 bytes, from Eq. 12), we thought that using a particular transport protocol might affect the system's trustworthiness. For this reason, we decided to run a second round of tests and check if the hypothesis was correct.

In order to 1) foresee which problems may occur during the Antarctic campaign, 2) decide which transport protocol to use, and 3) build more accurate expectations of the system's performance and outcomes, we applied our trustworthiness model to measure and evaluate them in this use case. For this purpose, the use case scenario was represented and evaluated in the Riverbed Modeler simulator. The first step is the modeling of the different elements that characterize our use case. More details about the modeling of this scenario and its technologies and protocols can be found in [28], [83].

Firstly, the backbone network (NVIS) and the access network (LoRa) were modeled separately, characterized as stated in Table VI following the aforementioned description of the network architecture (please revisit section 3.3) and the link availability results from [18] and [13]. On the one hand, LoRa does not experience any availability variation between daytime and nighttime, being fully available if there is LoS between the sensor and the gateway, and with partial availability in the case of no LoS. On the other hand, NVIS is not affected by not having LoS. However, its availability varies hour by hour, depending on the ionosphere state, which is highly correlated to solar activity. During nighttime (5 p.m. to 6 a.m.), the NVIS links are not available, while during daytime (6 a.m. to 5 p.m.), their availability varies between 70% and 100%.

TABLE VI
NETWORK PARAMETERS USED TO MODEL THE SCENARIO

Parameter	**NVIS**	**LoRa**
Transmission Band	4.3 MHz	868 MHz
Channel Bandwidth	2.3 kHz	125 kHz
Channel Bitrate	4.6 kbps	5.47 kbps
Coverage range	Up to 250 km	Up to 30 km
Daytime Availability (6 a.m. – 5 p.m.)	70%-100%	100% (LoS), 2%-100% (No LoS)
Night Availability (5 p.m. – 6 a.m.)	0%	100% (LoS), 2%-100% (No LoS)
Maximum Payload Size	242 bytes	140 bytes

Secondly, we modeled the following transport protocols as in our previous work [28]: BBR, Copa, CUBIC, EAATP, Indigo, and Verus. We focused on modern transport protocols that have been proven to perform well [60] and TCP CUBIC, which is the standard transport protocol in most operating systems nowadays. These protocols were modeled according to the results from our previous work in physical testbeds and simulations [27], [28] and the Pantheon tests [60].

Thirdly, we needed to model the Byzantine behavior of nodes. As stated in [113], the probability *Pb* of a node having a Byzantine fault is unlikely to be constant over time. The node reliability can be related to the battery charge level by associating the battery discharge with the WSN node aging process. Following the model in [113], we can assume the impact of aging as following a linear form, as defined in Eq. 17:

$$Pb(t) = Pb_0 + kt \,, \tag{17}$$

where $Pb_0$is the probability of a node having a Byzantine fault at time $t = 0,$ and *k* is the aging factor. This probability *Pb* increases hour by hour until its battery has practically run out at $t = t_d$, when it experiences a crash fault and $Pb(t_d) = 1$. In the simulations, we tested nine different values of Pb_0 to emulate the use of different corrective methods (see Table VII).

As we are in a simulation environment and we can keep track of all collected, sent, and received values by all nodes, we can compute *FSV* and *ST* by comparing the values that the sensor should have collected with the values that the sensor actually sends and the values that the control center receives, respectively. In a testbed environment with real devices, this would only be possible if previously known ground truth values were sent, in order to compare them with the values received by other nodes.

To model the implementation of the social trustworthiness layer, we used a simplified version of the objective reputational model from [46]. Our use case simplification assumes that all transactions will have the same weight, all nodes have the same computational capability, and the relationship factors between them are equal. Finally, a consensus protocol can be modeled by knowing the background traffic (bps) introduced to the network and the number of Byzantine nodes supported (*Nb*). In our use case, each group of redundant GTN-P stations will run the PBFT algorithm [114]. The background traffic grows exponentially as the number of nodes participating in the consensus (*Nt*) group increases. Moreover, the number of tolerated Byzantine nodes *Nb* is calculated as in Eq. 18:

$$Nb = \left\lfloor \frac{Nt-1}{3} \right\rfloor \tag{18}$$

Our scenario has five NVIS gateways, each providing an independent LoRa coverage area (access network) with its own sensors. For each gateway, there are clusters of sensors measuring the same data. In our test on the field during the campaign, we will deploy eight clusters per gateway. However, in the simulations, we also tested larger numbers of clusters (as seen in Table VII) to assess the goodness of our model and the system's scalability. Each cluster will have a specific number of redundant sensors measuring the same data. From our previous tests, we defined that we would set seven redundant sensors (GTN-P stations) in each cluster in the field deployment, so two Byzantine nodes could be tolerated. Despite this, in the simulation tests, we varied this number from 1 to 10 in order to compare the results with different Byzantine node tolerances (from 0 to 4, following Eq. 18) and assess the system's scalability.

The simulations consider three different operational modes: the standard mode (no redundancy), the redundancy mode with social trustworthiness, and the redundancy mode with consensus. In the standard mode, all the values gathered by every sensor are pushed through the backbone network to the remote control center. On the contrary, in redundancy modes, only one value is forwarded to the control center by each cluster. This value is agreed by cluster members with the social or the consensus mechanism. This fact reduces the amount of traffic that has to pass through the NVIS backbone LFN, although, contrarily, it introduces more overload to the LoRa access network due to the messages that need to be exchanged between cluster members.

All these possibilities add up a total amount of 16,200 different scenarios. Each scenario was simulated for 120 hours (5 days) to experience diverse nighttime and daytime cycles, and each test was repeated 30 times to assure results confidence. A summary of the simulation parameters to run our tests is shown in Table VII.

TABLE VII
SIMULATIONS PARAMETERS

Parameter	**Value**
Number of runs per test	30
Simulation duration	120 hours (5 days)
Pb_0	[1x10^{-3}, 2x10^{-3}, 4x10^{-3}, 8x10^{-3}, 1x10^{-2}, 2x10^{-2}, , 4x10^{-2}, 8x10^{-2}, 1x10^{-1}]
k	5.7x10^{-5}
Transport protocol	[BBR, Copa, CUBIC, EAATP, Indigo, Verus]
Redundancy Mode	[None, Social, Consensus (PBFT)]
Number of NVIS gateways	5
Number of GTN-P clusters per gateway	[8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096]
Number of GTN-P stations per cluster	[1-10]

3.6. Results

After performing all the simulations, the average value of the *STR* was calculated for every set of 30 runs per test. The results obtained have a maximum error deviation of 0.68% with a confidence interval of 99%. Three different operational modes for the telemetry service can be identified: the standard mode, the redundancy mode with social trustworthiness layer, and the redundancy mode with consensus layer. For every mode, an N × M-dimension grid with all the possible combinations of stimulation parameters is formed, where M is the number of different Pb_0 values, and N is the number of different GTN-P node combinations per gateway. For every point in this grid and for every transport protocol, the average value of the trustworthiness *STR* metric is computed. If we link all the *STR* values for every neighboring point in the grid, a mesh with all the *STR* values for each transport protocol is formed. We call this mesh the trustworthiness mesh.

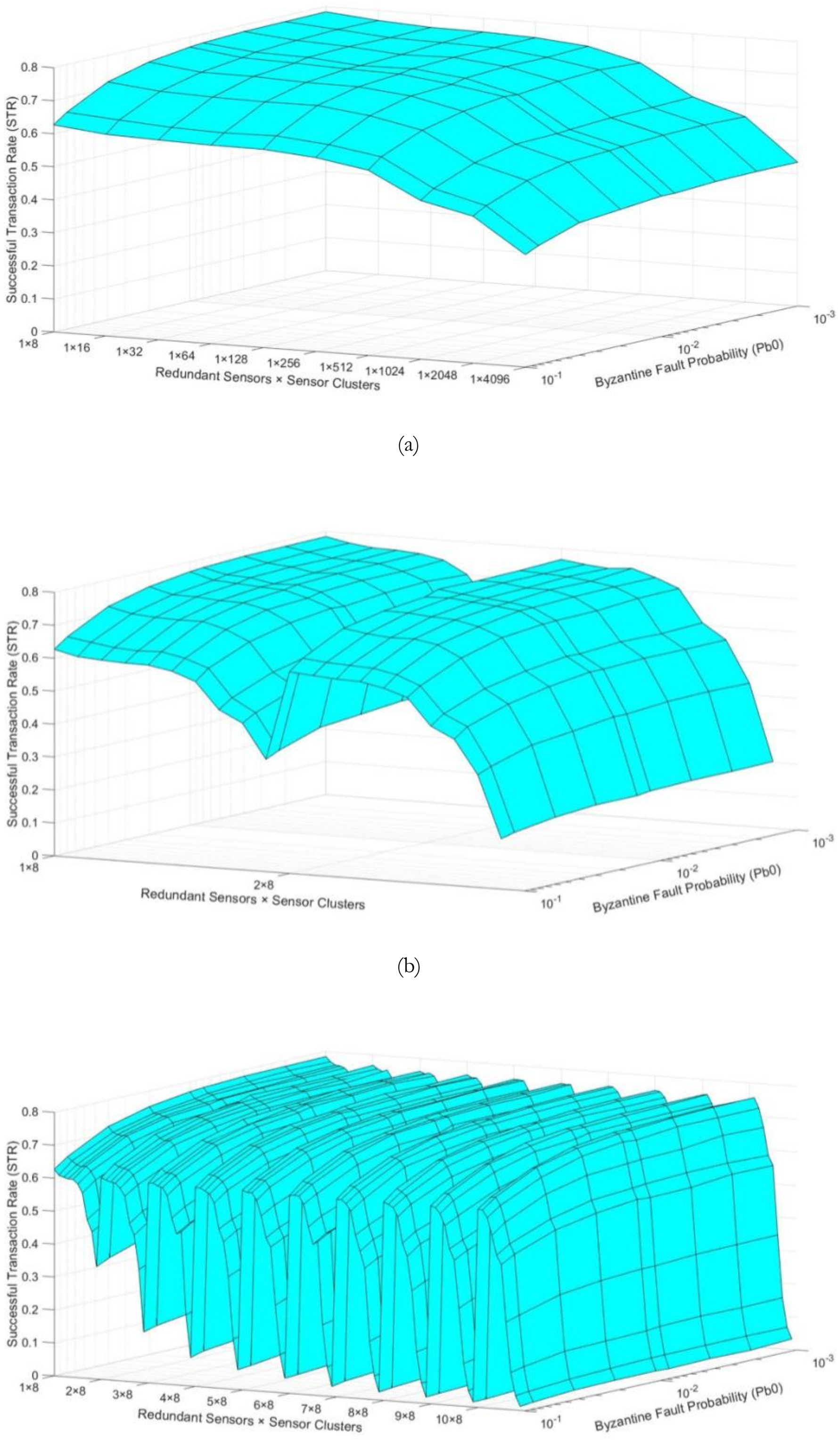
Given that it is complex to understand the trustworthiness mesh results, we first use an example to describe how the results are visualized. If we wanted to represent the results for only one transport protocol, when the number of redundant sensors per cluster is 1, and the number of clusters varies from 8 to 4096 (Table VII, row 9) we could obtain a mesh similar to Figure 21(a). The "Byzantine Fault Probability" axis has nine discrete points, corresponding to the nine different Pb_0 values shown in Table VII, row 4. The "Redundant Sensors × Sensor Clusters" axis has 10 discrete points, which are 1×2^N, where N = [3, 4, …, 12], according to the values shown in Table VII, row 9. Figure 21(a) shows the general behavior that *STR* values will follow in the actual results. On the one hand, across the "Byzantine Fault Probability" axis, the *STR* decreases as the Pb_0 increases, given that more values are faulty sensed when the Pb_0

is higher. On the other hand, across the “Redundant Sensors × Sensor Clusters” axis, the *STR* decreases as the number of clusters increases, given that more devices are introduced to the network, provoking more packet losses caused by network congestion.

Similarly, suppose we wanted to show, in a single mesh, the results from the same scenario, but the number of redundant sensors per cluster varied between 1 and 2. In that case, we could obtain a mesh similar to Figure 21(b). In this case, the “Byzantine Fault Probability” axis remains the same. In contrast, now the “Redundant Sensors × Sensor Clusters” axis has 20 discrete points, which are [1 × 2^N, 2 × 2^N] where N = [3, 4, …, 12]. If all the discrete points of this axis were labeled, it could be too congested. For this reason, we only label the beginning of each “redundant sensors” series, i.e., the “1 × 8” and the “2 × 8” discrete points. The same behavior as before is observed, but now the *STR* values recover when we jump from the “1 × 4096” to the “2 × 8” discrete point, given that much fewer nodes are introduced to the network, i.e., fewer packets are dropped due to network congestion.

Analogously, Figure 21(c) shows the trustworthiness mesh if we wanted to visualize all the results simultaneously, varying the number of redundant sensors from 1 to 10 (Table VII, row 10). In this case, the “Redundant Sensors × Sensor Clusters” axis has 100 discrete points, which are [1 × 2^N, 2 × 2^N, …, 10 × 2^N] where N = [3, 4, …, 12]. In this case, we observe the same general behavior again. However, now we can also detect that, if we compare the discrete points with the same number of clusters, the *STR* also decreases as the number of redundant sensors per each cluster increases, i.e., more packet losses are caused by network congestion as more nodes are introduced to the network.

Figure 22 shows the frontal view of the trustworthiness mesh from Figure 21(c). From this view, we can observe how the *STR* varies across the “Redundant Sensors × Sensor Clusters” axis without showing the variance, depending on the Pb_0 of the nodes.



(a)

(b)

(c)

Figure 21. Trustworthiness mesh examples: (a) only one redundant sensor per cluster; (b) one or two redundant sensors per cluster; (c) one to ten redundant sensors per cluster.

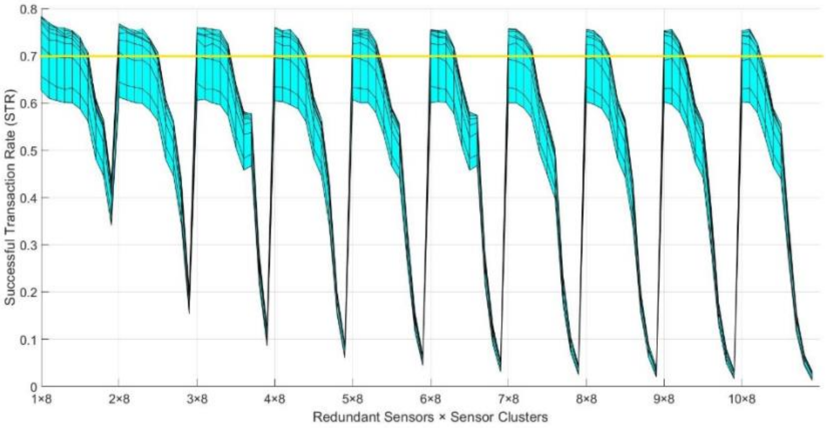


Figure 22. Example of frontal view of the trustworthiness mesh, corresponding to Figure 21(c). The yellow line is used to construct the trustworthiness working domain shown in Figure 23.

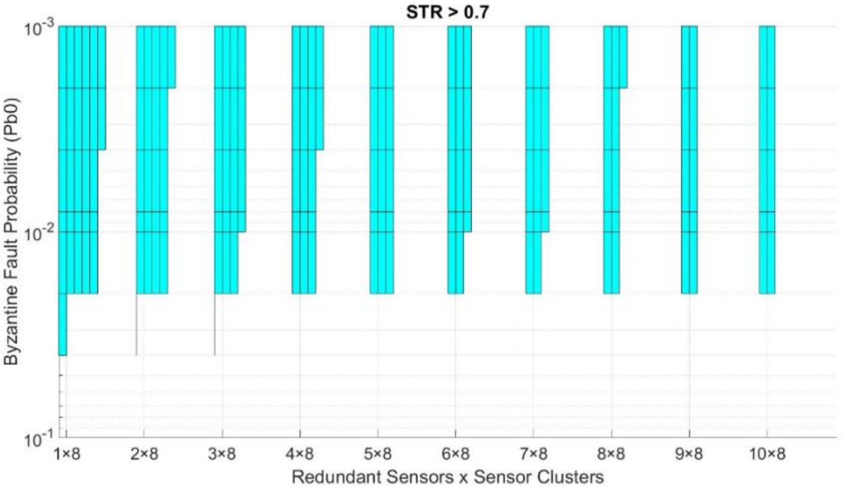


Figure 23. Example of the trustworthiness working domain corresponding to Figure 21(c) and Figure 22 with a minimum *STR* required of 0.7.

Our model can also be used to visualize the working domain in which to implement our service, given a desired minimum trustworthiness level. As stated before, our use case requires a minimum *STR* of 0.7, so an average of 9 out of 13 sensed values per night reach the control center correctly to meet the objective of [20]. Figure 23 shows the working domain of the example trustworthiness mesh presented in Figure 21(c) and Figure 22, requiring an *STR* higher than 0.7. For every point in the grid, if no solution provides an *STR* higher than the desired minimum value, the surface for that area is white-colored, meaning we cannot deploy the service with those conditions. On the contrary, if one or more solutions achieve an *STR* higher than the desired minimum value, the surface is painted with the color of the solution with the highest *STR*. This representation is achieved by "cutting" Figure 22 along the yellow line, which represents the minimum *STR* level that must be achieved. The part of the trustworthiness mesh above the yellow line meets the criteria and is part of the working domain, while the part below does not.

After clarifying how to visualize the data shown in these graphs, we present the tests' results in the following graphs. Figure 24, Figure 25, and Figure 26 show the trustworthiness mesh for the standard mode, the redundancy mode with social trustworthiness, and the redundancy mode with consensus, respectively. In each graph, the trustworthiness mesh of each transport protocol is superposed with the others in order to visualize which one achieves the highest *STR*. Moreover, Figure 27 shows the trustworthiness working domain of our telemetry service for an *STR* higher than 0.7.

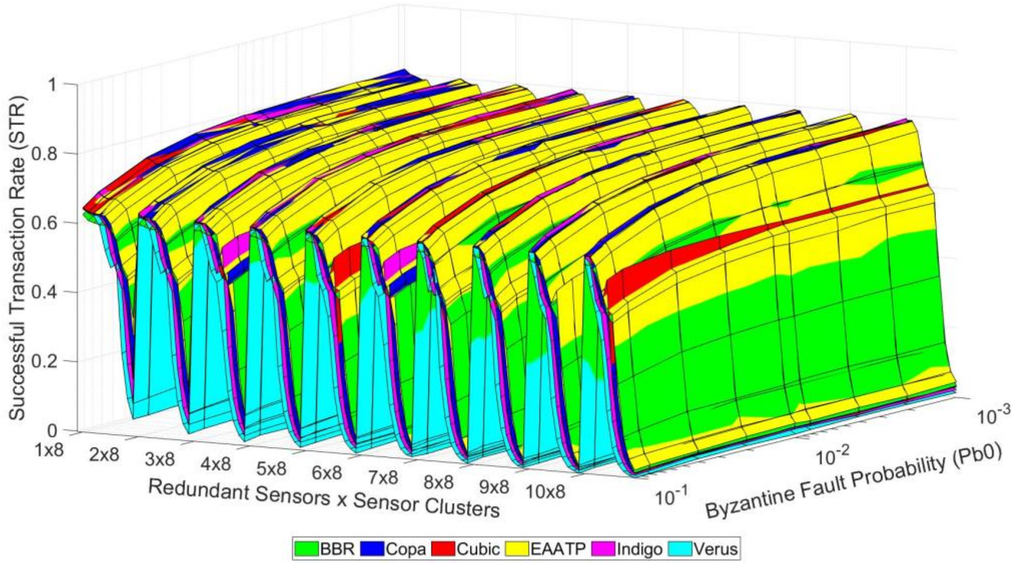


Figure 24. Trustworthiness mesh (standard mode).

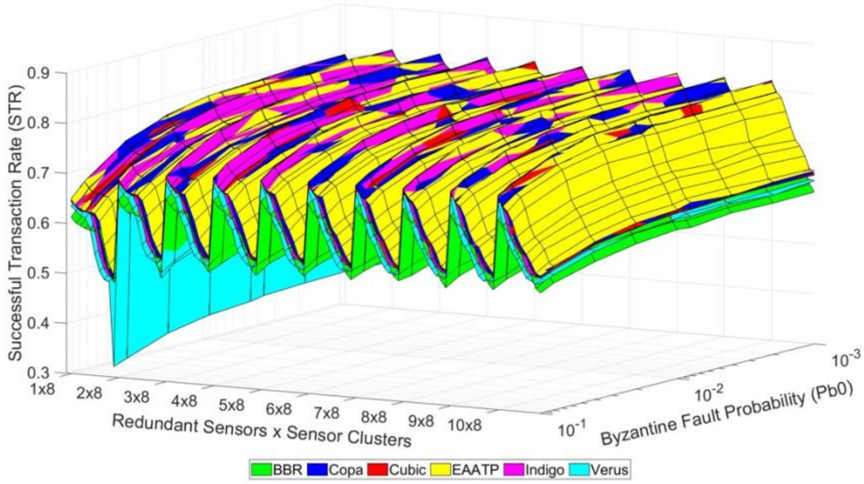


Figure 25. Trustworthiness mesh (social trustworthiness).

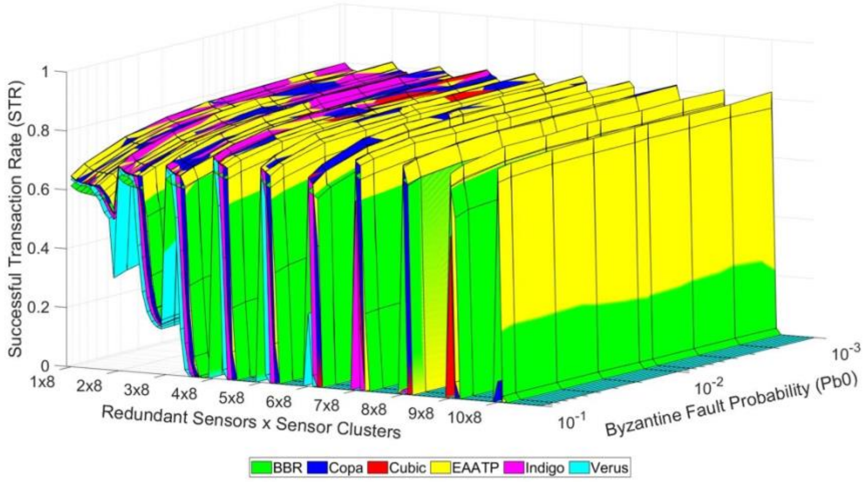


Figure 26. Trustworthiness mesh (consensus).

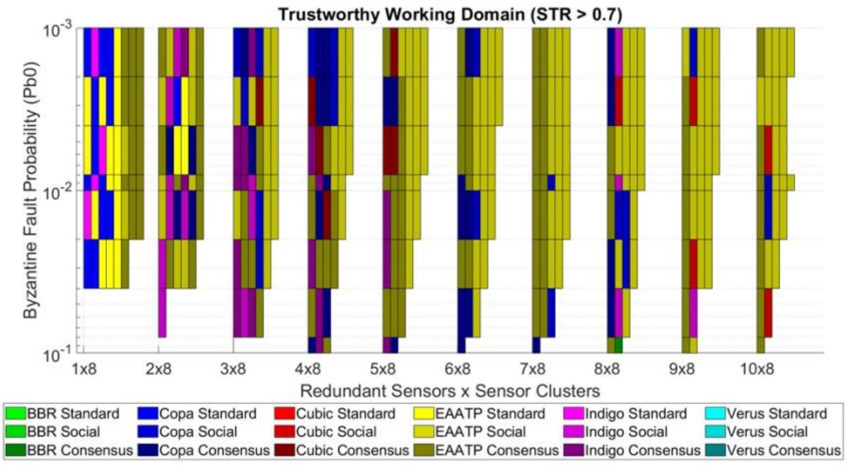


Figure 27. Trustworthiness working domain requiring *STR* > 0.7.

3.7. Discussion

On the one hand, Figure 24, Figure 25, and Figure 26 show that the levels of trustworthiness achieved are similar for all the studied transport protocols with low network load (left side of the mesh and cases with fewer sensor clusters). This fact seems reasonable because we already selected the most suitable and top-performance transport protocols to perform our tests, discarding those that do not adapt well in LFNs. We believe that if other transport protocols less suitable for this kind of network had been tested, the difference in the results would be more evident. However, 1) the levels of BBR and Verus are slightly lower than their competitors, and 2) Copa, Indigo, and EAATP share the highest *STR* values in the case of low network load, although the predominance of EAATP grows as the network load increases (the yellow mesh is more visible than the others).

On the other hand, we can also see that the redundancy mode with social trustworthiness (Figure 25) is the most robust scenario, given that its *STR* decrease in high-load situations is less accentuated compared to the other cases (Figure 24 and Figure 26), always maintaining *STR* values greater than 0.5. Furthermore, it is confirmed that, in general, as the probability of a node experiencing a Byzantine error decreases, the achieved *STR* values accordingly increase.

From the trustworthiness working domain (Figure 27), we can see the aforementioned predominance of the EAATP. As mentioned in section 3.5, the scenario intended to deploy in the next Antarctic campaign was the "7 redundant sensors × 8 sensor clusters". Concretely, we can check that this case reaches the *STR* requirement of 0.7 for any Pb_0 value.

If we focus on this case, in Figure 27, we can see that the EAATP is the most trustworthy protocol except for the $Pb_0 = 1 \times 10^{-1}$ and $Pb_0 = 8 \times 10^{-2}$ cases, in which Copa performs better. Table VIII shows, in detail, the results for the "7 redundant sensors × 8 clusters" case. For each protocol and each Pb_0, we show the best *STR* achieved from the three possible operational modes (standard, social, and consensus). Although Copa, CUBIC, and EAATP have similar results, the latter can outperform Copa and CUBIC between 0.1% and 0.5% better in terms of *STR* in most cases, and also outperforms up to 7% more than its other competitors. These results confirm our hypothesis, i.e., using a particular transport protocol can directly affect the system's trustworthiness in our use case.

TABLE VIII
BEST *STR* ACHIEVED BY EACH TRANSPORT PROTOCOL IN THE "7 REDUNDANT SENSORS × 8 CLUSTERS" CASE. THE BEST *STR* FOR EACH Pb_0 IS HIGHLIGHTED IN BOLD.

Pb_0	BBR	Copa	CUBIC	EAATP	Indigo	Verus
1×10^{-3}	0.767	0.818	0.817	**0.818**	0.814	0.801
2×10^{-3}	0.767	0.814	0.814	**0.819**	0.817	0.802
4×10^{-3}	0.772	0.819	0.819	**0.819**	0.811	0.795
8×10^{-3}	0.768	0.816	0.814	**0.817**	0.807	0.797
1×10^{-2}	0.767	0.818	0.817	**0.820**	0.805	0.794
2×10^{-2}	0.767	0.814	0.813	**0.815**	0.799	0.782
4×10^{-2}	0.762	0.811	0.809	**0.813**	0.777	0.765
8×10^{-2}	0.750	**0.796**	0.795	0.794	0.741	0.727
1×10^{-1}	0.731	**0.785**	0.781	0.779	0.724	0.710

We believe that the EAATP's superior trustworthiness is caused by the fact that it incorporates a fairness mechanism to share the network bandwidth, which reduces congestion and packet losses. Moreover, EAATP's congestion control tries to occupy the entire network bandwidth rapidly, and its mechanism to differentiate between random channel losses and congestion losses optimizes its achieved throughput in heavy congestion situations. These features give the EAATP a competitive advantage in terms of performance in our use case, where the DTN opportunistic scheme we use to send accumulated data during the night as a bulk data transfer congests the network.

For these reasons, we decided to use the EAATP as the backbone network transport protocol for our IoT telemetry service that will be deployed in the field during the next Antarctic campaign. Moreover, we can identify which of the three modes best suits the different scenarios which may arise. In general, the standard mode obtains the highest *STR* values when there is no redundancy (1 × N zone). If redundancy is applied, the consensus solution shows the highest levels of trustworthiness in most cases with a low network load. However, as mentioned before, when the network load increases, the social trustworthiness solution is more robust, achieving the highest *STR* values for those cases.

Finally, we also propose that the scenario to be deployed is reconsidered. In the "7 redundant sensors × 8 clusters" scenario, each gateway has 56 sensors connected, while only eight different values are sensed, which might be an excessive low efficiency. We propose to switch to the "5 redundant sensors × 16 clusters". In this case, increasing the number of sensors by

43% (80 sensors per gateway) results in increasing the number of different sensed values by 100% (16 values). Table IX shows the detailed results for this use case. If we compare the results from Table VIII and Table IX, the latter case achieves slightly worse *STR* values (which seems evident because we decrease the redundancy and increase the total number of sensors). However, Copa, CUBIC, EAATP, and Indigo still meet the required *STR* level of 0.7, providing trustworthiness to the service. In this case, we can also confirm the predominance of the EAATP, being the protocol with the highest *STR* in five of the nine Pb_0cases, while Copa and CUBIC achieve the highest *STR* in two cases each. Moreover, EAATP outperforms its competitors by up to 5.1%, while in the cases where another protocol outperforms the EAATP, it is only by 0.3% at most. Thus, we believe that the EAATP would also be the most suitable transport protocol to be used in this case.

TABLE IX
BEST *STR* ACHIEVED BY EACH TRANSPORT PROTOCOL IN THE "5 REDUNDANT SENSORS × 16 CLUSTERS" CASE. THE BEST *STR* FOR EACH Pb_0 IS HIGHLIGHTED IN BOLD.

Pb_0	BBR	Copa	CUBIC	EAATP	Indigo	Verus
1×10^{-3}	0.757	0.797	**0.798**	0.797	0.795	0.783
2×10^{-3}	0.752	**0.799**	0.799	0.798	0.796	0.783
4×10^{-3}	0.748	0.796	**0.798**	0.797	0.792	0.777
8×10^{-3}	0.75	0.794	0.795	**0.801**	0.792	0.775
1×10^{-2}	0.749	0.793	0.795	**0.796**	0.786	0.775
2×10^{-2}	0.74	0.79	0.787	**0.792**	0.779	0.764
4×10^{-2}	0.73	0.776	0.781	**0.781**	0.757	0.747
8×10^{-2}	0.698	**0.74**	0.736	0.737	0.727	0.706
1×10^{-1}	0.672	0.717	0.714	**0.718**	0.704	0.692

3.8. Conclusions

This paper analyzes the applicability of the deployment of a remote WSN for the Antarctic region using NVIS technology and the provision of an IoT telemetry service for permafrost studies. This service will be deployed during the 2021–2022 Antarctic campaign of the SHETLAND-NET project. This work focuses on analyzing and comparing transport protocols' trustworthiness in our remote WSN with DTN use case, which uses LoRa at the access network and NVIS links at the backbone network. Due to certain ionospheric characteristics, NVIS links do not work correctly at night. For this reason, values sensed at night are sent opportunistically to the control center as bulk data when the NVIS channel becomes available, which might cause network congestion. In this situation, the choice to use a particular transport protocol might affect the overall system's trustworthiness. In order to study the viability of the service to be implemented before its deployment in the field during the Antarctic campaign and in an attempt to compare the performance of various transport protocols, we use our model to measure and evaluate the trustworthiness of the proposed system. This trustworthiness model consists of four layers that can affect the *STR* trustworthiness metric.

Three operational modes and six transport protocols were analyzed under different conditions using the Riverbed Modeler simulator. The results show a predominance of the EAATP as the most trustworthy transport protocol, while BBR and Verus have the worst trustworthiness. Adding redundancy to the measured values with multiple sensors and applying a social reputational mechanism improves the robustness of the system's trustworthiness, reaching higher *STR* values and never dropping below 0.5, even in high-load scenarios. On the contrary, a consensus mechanism improves the system's trustworthiness if the number of sensors is kept at a low value.

The research group decided to deploy eight clusters for each NVIS gateway and seven GTN-P redundant stations per cluster in the Antarctic campaign. The collected results confirm that this scenario achieves the minimum *STR* required of 0.7, resulting in a feasible deployment. In this case, the results show that the EAATP can outperform up to 7% of the other analyzed transport protocols in terms of trustworthiness (*STR*). However, we recommend sacrificing some redundancy (i.e., trustworthiness) and increasing the number of different sensed values, implementing the scenario with 16 clusters and five GTN-P redundant stations. In this case, although slightly worse *STR* values are achieved, the requirement of achieving at least an *STR* of 0.7 is met, while more data can be remotely monitored from the control center. The EAATP is also the most trustworthy transport protocol in this case, outperforming its competitors by up to 5.1%. Thus, the research group has decided to use the EAATP as the transport protocol for the offered telemetry service.

Future work aims to 1) study the viability of using the same network architecture to deploy an integrated sensing and communication system (ISAC) capable of using ionosondes as data transmission signals through NVIS; and 2) analyze the implementation of other DTN architectures and protocols to improve the trustworthiness of the entire system in situations when the availability of the NVIS link is not previously known (daytime).

Chapter 4

A DELAY TOLERANT NETWORK FOR ANTARCTICA

Antarctica is the land of science. Every year, many studies are carried out in diverse disciplines. Some of these studies collect relevant data for their research with sensors. However, Antarctica's lack of telecommunication technologies hardens the possibility of automatizing this data collection. In most cases, the collection is done manually, limiting research projects' time and space scopes. Over the last years, some alternatives have been studied to deploy remote Wireless Sensor Networks in Antarctica. Near-Vertical Incidence Skywave (NVIS) communications are an example of these alternatives. However, NVIS presents problems that cannot guarantee persistent end-to-end connectivity. For this reason, this paper assesses adapting a Delay Tolerant Network protocol, the Bundle Protocol, to deliver sensor data reliably through an NVIS network. The scenario is developed and tested in the Riverbed Modeler simulator, and performance is evaluated through a trustworthiness model. A practical testbed is also presented.[4]

Keywords: Antarctica; IoT; WSN; NVIS; Trustworthiness; Reliability; DTN; Bundle Protocol.

[4] The work reported in this chapter was published as the paper entitled "A Delay Tolerant Network for Antarctica" in the IEEE Communications Magazine on August 2nd, 2022 for the 12th issue of Volume 60, https://doi.org/10.1109/MCOM.007.2200147 Authors: Adrià Mallorquí, Agustín Zaballos, Daniel Serra.

Due to copyright reasons, content from pages 95-110 was omitted from this version.

Due to copyright reasons, content from pages 95-110 was omitted from this version.

Due to copyright reasons, content from pages 95-110 was omitted from this version.

Due to copyright reasons, content from pages 95-110 was omitted from this version.

Chapter 5

RESULTS

This chapter presents and summarizes the results obtained in each publication [83], [85], [87]. Then, the results are discussed and related to the objectives posed in the introduction (section 1.3). As a reminder, the thesis objectives are focused on the use case to answer the research questions and validate the proposed trustworthiness model with quantitative results. The objectives are listed below:

- O1. Establish a scope for the trustworthiness term.
- O2. Define a model that enables trustworthiness accountability to detect weaknesses and propose appropriate countermeasures for trustworthiness improvement.
- O3. Assess the trustworthiness of the Antarctic use case.
- O4. Propose new mechanisms to improve the trustworthiness of the Antarctic use case.
- O5. Deploy a PoC of the proposed service architecture in the field during the 2021-2022 Antarctic campaign.

5.1. A Heterogeneous Layer-Based Trustworthiness Model for Long Backhaul NVIS Challenging Networks and An IoT Telemetry Service for Antarctica

This work aimed at objectives O1, O2, O3, and O4. For O1, a related work review was presented, classifying related work in trustworthiness into four categories. From this review, the scope of trustworthiness in CPS was limited to the areas of data trustworthiness, network trustworthiness, social trustworthiness, and consensus for the Antarctic use case. This led to the model definition (O2), which was based on four layers, each of them defined by its scope, its metric, its countermeasures, and the interdependencies between them (see Table XII):

TABLE XII
TRUSTWORTHINESS MODEL'S LAYERS SUMMARY.

Layer	Definition	Metric	Countermeasures examples	Interdependencies
Data	Is the layer responsible for ascertaining the correctness of the data provided by the source.	FSR	Autocorrection methods, node robustness, computational resources improvement, node ostracism.	FSR can be reduced thanks to social trust and consensus mechanisms, relying only on data from trusted nodes or agreed for all group members.
Network	Is the layer responsible for assuring that a packet reaches its destination on time and unaltered despite the adversities.	PDR	Congestion decrease, load balancing, QoS, coding, MAC protocols, transport protocols, routing, DTN.	Social ostracism can remove untrusted nodes from the network, thus decreasing network congestion and increasing PDR. Consensus mechanisms introduce extra traffic to the network, potentially congesting the network and decreasing PDR.
Social	Is the layer responsible for leveraging the capability of the objects to establish social relationships autonomously between them to improve the trust between them and the correctness of gathered data.	STR	Sensor redundancy, node ostracism.	Data and network trustworthiness directly affect social trustworthiness. As less data is faulty-sensed and more packets are correctly delivered, more trusted nodes and successful transactions can be achieved.
Consensus	Is the layer responsible for reaching a state where all group participants agree on the same response or result.	BNT	Increase the number of group participants, use social trust for weighted voting.	Data and network trustworthiness directly affect consensus. As less data is faulty-sensed and more packets are correctly delivered, more general agreements can be achieved. Social trustworthiness can improve general agreements by using weighted voting depending on nodes' social trust.

1. Data Trustworthiness Layer: is the layer responsible for ascertaining the correctness of the data provided by the source. This layer's metric is the FSR. Countermeasures include autocorrection methods to detect faulty nodes, false alarms, and sensor misreading. Data trustworthiness can be affected by social trustworthiness and consensus.

2. Network Trustworthiness Layer: is the layer responsible for assuring that a packet reaches its destination on time and unaltered despite the adversities. This layer's metric is the PDR. Countermeasures include channel coding, transport protocols, dynamic routing and topology control protocols, and DTN architectures and protocols. Network trustworthiness can be affected by social trustworthiness and consensus.

3. Social Trustworthiness Layer: is the layer responsible for leveraging the capability of the objects to establish social relationships autonomously between them to improve the trust between them and the correctness of gathered data. This layer's metric is the STR. Countermeasures for social trustworthiness include increasing sensor redundancy and using node ostracism to remove untrusted nodes. Social trustworthiness is directly affected by data and network trustworthiness.

4. Consensus Layer: is the layer responsible for reaching a state where all group participants agree on the same response or result. This layer's metric is the BNT. Consensus' countermeasures include using social trust to implement weighted voting among the consensus group participants and increasing the total number of participants to tolerate more byzantine nodes. This layer has dependencies with all the other trustworthiness layers.

Given the nature of the Antarctic use case, it was proposed to use the STR as the metric that would quantify the reached level of trustworthiness. The first version of a hybrid IoT architecture was proposed, using LoRa as the access network communications technology and NVIS at the core network. For O4, it was proposed to use social trust management and consensus mechanisms to take advantage of sensor redundancy, defining three operation modes: the Standard mode, the Social mode, and the Consensus mode. For O3, the use case scenario was modeled into Riverbed Modeler, simulating the scenario several times and varying its parameters to validate the model's goodness.

The use case modeling and simulation have been the core and the most time-consuming tasks of this thesis, with continuous development and testing for several months (please revisit the roadmap in Figure 7). The LoRa access network and the NVIS backbone network were modeled separately, each characterized by their transmission frequency, channel bandwidth, throughput, and range, among other parameters. The nodes and their byzantine behavior were also modeled, as well as the social trust and consensus mechanisms. The application (permafrost monitoring service), BP, and EAATP were also modeled as node modules. Each module executes a specific piece of software coded as an FSM and interconnects with other modules from the same node. Figure 12 shows the DTN node model with the modules' architecture.

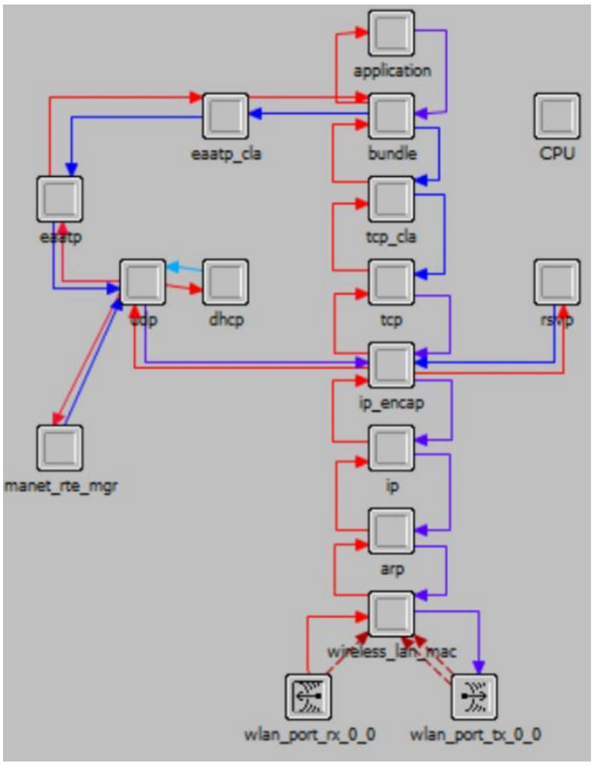


Figure 32. DTN node model in the Riverbed Modeler Simulator.

From the tests results, it could be extracted that:

- All the combinations achieve an STR lower than 0.82 (Figure 13). In all cases, as the Pb_0 increases, the STR decreases. This happens because increasing Pb_0 implies a worse (higher) FSR, which, as seen in Figure 10, is inversely related to the STR.

- The number of sensors affects hugely on the STR, given that NVIS and LoRa networks have reduced bandwidth (especially the LoRa access network). As more sensors send their data, more bandwidth is demanded. This provokes a network congestion that decreases the PDR and, consequently, the STR, as seen in Figure 10. Approximately, if more than 700 sensors per gateway are placed, the STR drops significantly to values lower than 0.65 with the Standard and Social modes and drops to values close to or equal to 0 with the Consensus mode (see Figure 13).

- The Social mode only improves the trustworthiness of the Standard mode if there is sensor redundancy. This happens because the social trust mechanism does not reduce network congestion if there are no redundant sensors, given that all sensed values must reach the control center in any case. Contrarily, the Social mode takes advantage when various sensors collect the same data, but only the most trusted one reaches the control center without increasing network congestion. Thus, the PDR is similar, but the FSR is better (lower), consequently improving the STR.

- In contrast, the Consensus mode only improves the Standard if a minimum of four redundant sensors are used. This happens because, following Eq. 11, at least four participants are needed in the consensus group to tolerate a byzantine error. This way, the FSR decreases thanks to GAs and, consequently, the STR increases.

- With four or greater redundant sensors, the Consensus mode generally achieves higher STR than the Social if the total number of sensors is kept to less than 200, approximately. However, their difference reduces with low Pb_0, negligible if it is equal to or lower than 0.01.

- The Social mode is more robust than Consensus due to its achieved STR never dropping below 0.55. Contrarily, the Consensus mode's STR can drop to 0 due to the congestion that the PBFT mechanism adds to the network. This happens because the social trust mechanism adds minimal extra traffic without affecting network congestion, while the PBFT algorithm's data to be exchanged over the network increases exponentially following Eq. 19, where *N* is the number of participants in a consensus group, *M* is the number of consensus groups in the same network, and *P* is the payload size of the messages exchanged between group participants. Thus, with PBFT, more congestion is introduced to the network as *N* and *M* increase, worsening the PDR and, consequently, decreasing the STR.

$$PBFT_{data}[bits] = P[bits] * M * (2N^2 - N) \tag{19}$$

- Thus, a trade-off between the number of sensors in the network and the congestion their traffic caused was identified for the Antarctic use case. Figure 13 shows a threshold of 700 GTN-P stations for the Social mode and 200 GTN-P stations for the Consensus mode. If more sensors were placed in the network, the network would be excessively congested, dropping its PDR and, consequently, decreasing the STR.

- These results validated the proposed trustworthiness model, given that it could quantify the level of trustworthiness reached and distinguish the configurations with better reliability, recommending the Social mode as the most robust.

5.2. DTN Trustworthiness for Permafrost Telemetry IoT Network

This work aimed at objectives O3 and O4. Given the proposed architecture for the telemetry service, data could not be exchanged during nighttime due to NVIS links' unavailability. Attempting to prevent data loss, it was proposed to keep collected data in the intermediate gateways (NVIS nodes) and opportunistically send the accumulated data as a bulk data transfer when NVIS links became available in the daytime. However, these bulk data transfers provoked network congestion, given the low bandwidth of NVIS links. Therefore, data did not reach the control center due to packet losses. In this work, it was introduced that the use case required that at least 70% of the samples had to reach the control center successfully to guarantee a reliable permafrost monitoring service. Thus, we fixed an STR of 0.7 as our trustworthiness threshold. For this reason, previous results had to be improved (O4).

Using modern transport protocols proved to improve the performance of data exchange in network congestion situations over long-distance links, so it was proposed to test some of these protocols in our use case and validate them through simulations. Concretely, we compared the performance and trustworthiness of BBR, Copa, Indigo, TCP CUBIC, Verus, and our proprietary transport protocol from the VSNoIPv6 project, EAATP (see the roadmap in Figure 7). We also tested the three operational modes (Standard, Social, and Consensus) to validate if the general behavior was still the same as in previous tests despite changing the transport protocol. This was the second major iteration of O4, which was modeled and assessed again (O3) with Riverbed Modeler. From the results of the tests, we extracted the following:

- The general behavior regarding trustworthiness remained the same regarding which transport protocol was used. On the one hand, if no redundancy was used, the Standard mode was the most trustworthy. On the other hand, with redundant sensors, the Social mode was the most robust, keeping its STR higher. Contrarily, the Consensus mode could not achieve good trustworthiness values consistently, even with modern transport protocols, introducing excessive network congestion and provoking too many packet losses. For this reason, the Consensus mode was discarded to implement during the Antarctic campaign.

- In Standard and Social modes, with less than 700 sensors, when the network load was under the maximum capacity (i.e., not provoking congestion), the difference between the tested transport protocols was practically negligible, achieving similar STR values. This happens because, in this situation, packet losses are caused by channel errors or network disruptions, but not by congestion. Thus, the different congestion control algorithms of the studied transport protocols are less significant. However, as the number of sensors increased, more differences could be spotted, with Copa, CUBIC, EAATP, and Indigo capable of reaching the required STR. In contrast, BBR and Verus performed worse and did not reach the trustworthiness threshold. These results proved that the congestion control algorithms of Copa, CUBIC, EAATP, and Indigo managed better the use case congestion situations by minimizing packet losses, thus increasing the PDR and consequently the STR.

- Furthermore, in most situations, the EAATP was the protocol with the highest STR thanks to its fairness mechanism, which balances the throughput of all EAATP flows on the same network and minimizes packet losses even more. For this reason, it was considered the best option to implement in the physical scenario.

- Concretely, in the case with the best trustworthiness (7 redundant sensors × 8 sensor clusters), EAATP could achieve the best STR values in 7 out of 9 cases. It could outperform the other tested transport protocols by up to 7%. However, it only outperformed the other best alternatives (Copa and CUBIC) between 0.1% and 0.5%.

- In another scenario, 5 redundant sensors × 16 sensor clusters, by doubling the total potential samples, the performance of all protocols slightly decreased. Indeed, Copa, CUBIC, EAATP, and Indigo reached the required STR, with EAATP being the best in 5 out of 9 cases and CUBIC performing better in 2 cases.

- These results confirmed that the election of a particular transport protocol is significant and can be a countermeasure for the network trustworthiness layer, given that it impacts the network PDR and, consequently, can affect the overall trustworthiness of a system, answering RQ4.

- Given these results, we considered EAATP and TCP CUBIC for the physical deployment during the Antarctic campaign. EAATP was elected because it showed the best results overall. TCP CUBIC was also elected because its results were also satisfactory for the use case (it could reach an STR greater or equal to 0.7). Additionally, it is the default transport protocol of the NVIS nodes' (Raspberry Pi) operating system. As a reminder, the Raspberry Pi and the Moteino boards were chosen as the hardware for the physical deployment because their ease of use and low power consumption characteristics (among others) met the requirements of the harsh Antarctic environment (revisit Table X). The Raspberry Pi uses the Raspbian operating system, which is Linux-based, so its default TCP variant is CUBIC. Using TCP CUBIC reduces development time and minimizes risks that an experimental protocol such as EAATP implies, which is a relevant element to consider in time-constrained experiments such as the deployment in the Antarctic campaign.

5.3. A Delay Tolerant Network for Antarctica

This work aimed at objectives O3, O4, and O5. Given the congestion problems that the opportunistic networking scheme caused despite using modern transport protocols, a DTN architecture using BP was proposed. In contrast to previous tests, in which the simulated duration was five days, in this third iteration, we wanted to test the use case for 400 days, which is the approximate timespan between two consecutive campaigns in Antarctica. This way, we could foresee if our proposed architecture could provide the telemetry service reliably for the entire campaign duration. Given that the amount of simulated time was increased, it was expected that achieved STR values would decrease. Thus, another mechanism was needed to improve the service's trustworthiness, proposing BP for this purpose.

In this third major iteration for O4, in addition to introducing a DTN architecture with BP and increasing the simulation time, we also discarded BBR, Copa, Indigo, and Verus transport protocols and the Consensus operational mode, given the aforementioned results from previous work. Moreover, we also discarded using a vast number of sensors, limiting the number of measuring spots to 32 or 64 and the number of redundant sensors from 1 to 5, given the constraints we would have in the campaign regarding available time and budget. From the trustworthiness assessment with Riverbed Modeler (O3), we could extract the following:

- When BP was not used, the service could not reach the required STR, given that these tests were performed with a long-term duration instead of the short-term duration (five days) of the previous tests. This happened because the opportunistic networking scheme only solved the disconnections of the NVIS network at nighttime, which were the predictable disruptions. These disruptions were predictable because the previous study on the NVIS channel showed that, due to the ionosphere's properties dependent on solar activity, the working frequency of NVIS differs during daytime and nighttime. Given that the NVIS antenna of the SHETLAND-NET project only works at a single frequency, the daytime frequency was chosen, expecting network disruption at nighttime. However, the intermittent disconnections and long delays of the NVIS network during the daytime were unpredictable. Therefore, the proposed opportunistic networking scheme could not handle these situations, and packets were dropped. Thus, in the long term, all cases using the opportunistic networking scheme suffered a decrease in their PDR, consequently worsening the STR to values lower than 0.7, not meeting the use case's trustworthiness threshold.

- Contrarily, if BP was used, the service could accomplish the required minimum trustworthiness level to guarantee a reliable service, both with the Standard and Social modes, because BP was able to mitigate predicted and unpredicted network disruptions and delays thanks to its store-carry-forward methodology for bundle exchange, improving the overall PDR in the long-term compared to the opportunistic networking scheme, and consequently increasing the STR above 0.7. Thus, it was clear that BP was strictly necessary to deploy the monitoring service in Antarctica. Concretely, BP outperformed the opportunistic networking scheme by up to 14 % in terms of STR, which marked the difference between meeting or not the use case's trustworthiness requirements.

- As in previous tests, the Social mode was the most robust, achieving the highest STR values in all cases due to its advantage of using the social trust management mechanism with sensor redundancy. Nevertheless, the Standard mode with BP could also reach the required trustworthiness threshold without needing sensor redundancy if the Pb_0 was equal to or less than 0.02.

- Using EAATP as the transport protocol was less relevant than BP regarding reaching trustworthiness in the long term. Using BP instead of opportunistic networking could improve STR values by up to 14%, while using EAATP instead of TCP CUBIC could only improve the STR by up to 5%. Moreover, this 5% improvement by EAATP was not decisive in helping reach the required reliability level because all cases without BP could not reach it, and all cases satisfying it were using BP regardless of the transport protocol.

Given the simulation results, it was decided that the physical deployment would include BP. Concretely, we used the IBR-DTN implementation of BP in our devices to validate the simulation results. This implementation was chosen among other alternatives because it met all the use case requirements, including the support for Linux-based operating systems, the compatibility with TCP/IP protocols, its lightweight and ease of use, and its open-source code (see Table X). The source code of IBR-DTN had to be slightly modified to adapt it to the Raspberry Pi, given that the original version presented some bugs due to incompatibilities with the Raspbian operating system.

In addition, it was also decided that we would use TCP CUBIC as the transport protocol of the PoC because to use EAATP, it was required to develop an ad-hoc CLA for any BP implementation, including IBR-DTN. Developing a CLA for EAATP in IBR-DTN would have added more stress in the deployment phase of the project, given that new software had to be designed, coded, and tested in a time-constrained stage. In contrast, the CLA for TCP (and, by extension, TCP CUBIC) was already developed and included in IBR-DTN, allowing more time to perform the other planned tasks of the project. It was considered that using TCP CUBIC instead of EAATP was not critical because CUBIC proved in simulations that it also reached the required reliability level for the telemetry service, despite EAATP slightly improved the obtained STR. The development of the EAATP CLA for IBR-DTN was postponed for future work.

For the PoC in the Antarctic campaign, we presented the DIU. This physical testbed consisted of five NVIS nodes exchanging sensor data across NVIS links. With this testbed, we could validate the functionality of BP over NVIS for the backbone network. Finally, in the Antarctic campaign, we attempted to deploy the PoC between the Uruguayan Artigas Base on King George Island and the Spanish Juan Carlos I Base on Livingston Island. We could validate the correct functioning of the LoRa access networks, exchanging data between sensors and NVIS gateways. In addition, we could also verify the 93 km NVIS backbone link between the two bases, with BP deployed across the backbone network. However, a COVID-19 outbreak in Antarctica [143] caused the modification of the logistic plans from all research projects in Juan Carlos I Base, forcing us to interrupt and cancel our final tests merging the access and core networks, and deploying the permafrost monitoring service. These validation tests had to be continued in Catalonia afterward, as seen in the following section.

5.4. Discussion

The SHETLAND-NET project proposed an NVIS-based IoT network to automatize data collection and exchange for Antarctic research. Due to reliability issues of NVIS communications, this thesis aimed to improve the trustworthiness of this IoT network without modifying NVIS specifications, focusing on the use case of permafrost monitoring. To accomplish it, objectives O1-O5 were set. These objectives were addressed through [27], [28], [83]–[87].

In [83], we presented a review of the related work to build a picture of how the trustworthiness term was interpreted and managed from different points of view. From this review, we proposed an integrated scope for trustworthiness in the Antarctic use case, binding the different perspectives studied in the literature and accomplishing O1. Next, we proposed a trustworthiness model (O2) composed of four layers, each centered on a different trustworthiness field, with its own metric to enable trustworthiness quantification, its countermeasures to improve it, and the interdependencies between the layers. The STR was proposed as the metric to quantify trustworthiness in our use case. We verified the usefulness of this model by modeling all the elements of the Antarctic use case in the Riverbed Modeler simulator. In these tests, we could quantify the expected trustworthiness of the use case under different circumstances by computing the STR and accomplishing O2.

These tests were also the first step toward reaching O3 and O4. In [83], we proposed a hybrid IoT architecture with LoRa and NVIS as the communication technologies for the access and core networks, respectively. Moreover, we proposed using the consensus and social trust management mechanisms to take advantage of sensor redundancy. The results of this first round showed that it could be possible to accomplish the required trustworthiness level (0.7) in the short term if less than 700 or 200 GTN-P stations were used with the Social or

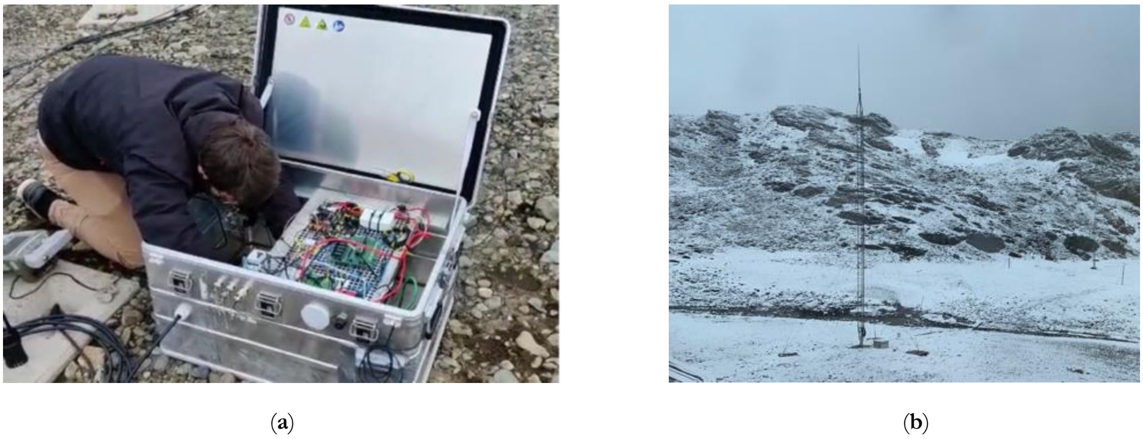
Consensus mode, respectively. Using more sensors caused excessive traffic and provoked congestion in LoRa and NVIS low bandwidth networks. Moreover, we could conclude that the Social mode was the most robust, while the Consensus mode was not recommended because it introduced even more congestion.

This behavior was confirmed in [84], [85], which were the second iteration of trustworthiness improvement to reach the goals of the thesis. We proposed using modern transport protocols we had worked with in other works [27], [28], aiming to maximize bandwidth usage and minimize packet losses in congestion situations. The scenario was tested in five-day (short-term) simulations to observe the difference in the behavior and performance of the assessed transport protocols. The results of the tests showed that EAATP and TCP CUBIC were the most suitable candidates. On the one hand, EAATP was the transport protocol that reached the highest STR values in most cases, surpassing the trustworthiness threshold. In the short term, it could outperform its competitors by up to 7% in terms of obtained STR. It is believed that it is caused by the fact that EAATP incorporates a fairness mechanism to share the network bandwidth, which mitigates congestion and packet losses. Moreover, EAATP's congestion control tries to occupy the entire network bandwidth rapidly, and its mechanism to differentiate between random channel losses and congestion losses optimizes its achieved throughput in heavy congestion situations. On the other hand, TCP CUBIC could also reach the required minimum trustworthiness level. Furthermore, CUBIC is the default transport protocol of the NVIS nodes, simplifying the deployment in the field.

However, in [86], [87], we saw that short-term results could not be extrapolated to the long term (400 days), i.e., the timespan between consecutive Antarctic campaigns. In this third major iteration of O3 and O4, we assessed using a DTN architecture with BP instead of the opportunistic networking scheme. Thanks to it, we could avoid the congestion provoked by the bulk data transfer when NVIS became available in the daytime and packet losses in front of unpredicted NVIS disruptions due to the ionosphere's changing properties. The results showed that it was mandatory to use BP to accomplish the required minimum level of reliability, increasing the obtained STR values by up to 14% compared to the results without using BP. In contrast, the election of EAATP or TCP CUBIC was less relevant, with EAATP slightly improving the results of TCP CUBIC.

With these three iterations, we were able to accomplish O3 and O4, determining that it was possible to implement the permafrost monitoring service in Antarctica and guarantee the required trustworthiness level if we deployed BP for the DTN architecture, the Standard or Social modes, EAATP or TCP CUBIC as the transport protocol, 64 or less measuring spots, and five or less redundant sensors per spot.

We finally attempted to deploy a PoC of the hybrid IoT DTN architecture in the 2021-2022 Antarctic campaign (O5). We aimed to perform validation tests with a DTN composed of two NVIS gateways distanced by 93 km and five sensors per gateway. It was not considered to deploy and test more complex scenarios given the time constraints and the available resources for the campaign, in addition to the harsh conditions of the use case in the Antarctic field. In the PoC deployed in Antarctica, we validated the correct functioning of the LoRa access network, with the Moteino boards sending the data samples collected with the sensors to a Raspberry Pi NVIS gateway with LoS. We could also validate the data exchange across BP nodes over the NVIS backbone network. Figure 33 shows the NVIS node deployed in the Juan Carlos I base. However, due to the campaign's restrictions imposed by a COVID-19 outbreak, we had to discontinue our tests and could not validate the whole permafrost telemetry system in Antarctica.



(**a**) (**b**)

Figure 33. NVIS Hardware deployed in Juan Carlos I base (Antarctica). (**a**) NVIS node. (**b**) NVIS inverted vee antenna.

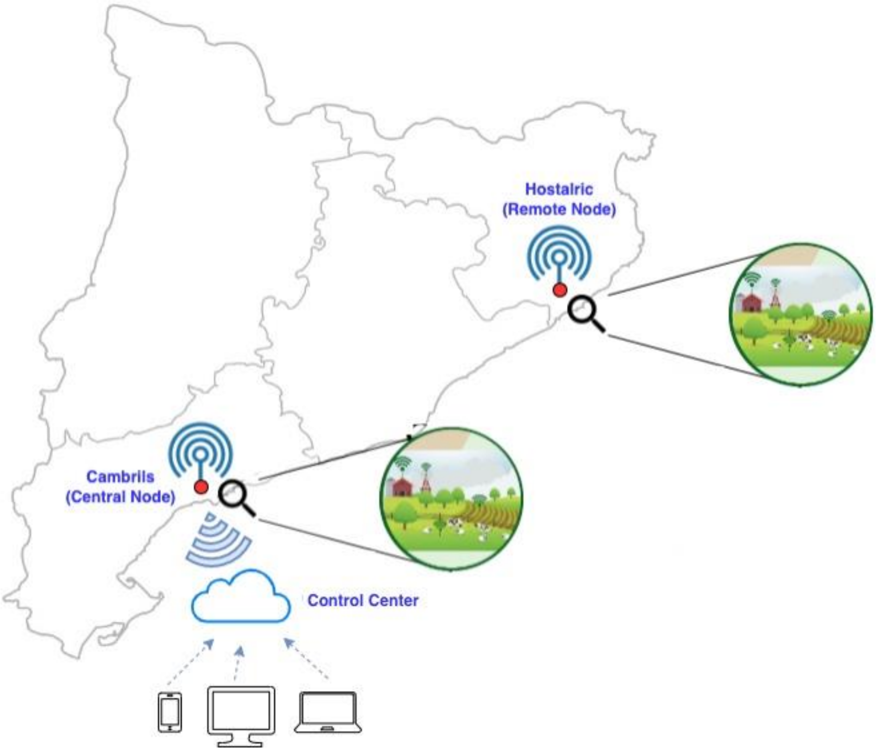
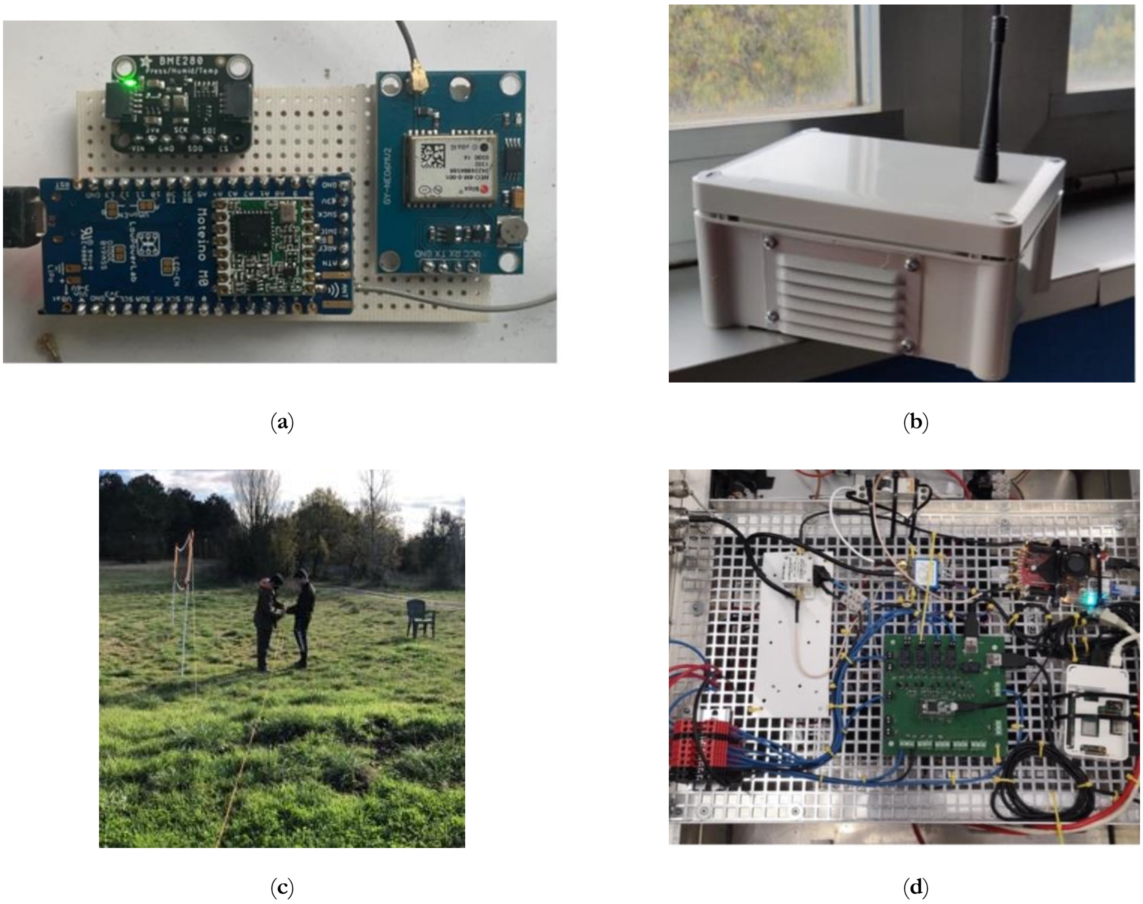


Figure 34. PoC deployment in Catalonia.

To accomplish O5, the validation tests were completed in Catalonia. To emulate with the best possible fidelity the deployment in Antarctica, a 150 km NVIS link was established between two remote rural locations placed in Hostalric and Cambrils. The Cambrils node was in charge of sending the collected data to the control center through the Internet (see Figure 34). Five sensors with a Moteino board were placed in each location, which sent the collected data to the Raspberry Pi NVIS nodes through LoRa. In addition, the Raspberry PI nodes also acted as BP nodes, which exchanged the collected data through the NVIS link. Figure 35 shows the hardware deployed in Hostalric (please revisit Figure 17 to see Cambrils' hardware).



(**a**) (**b**)

(**c**) (**d**)

Figure 35. Hardware deployed in Hostalric. (**a**) Moteino board with attached sensors. (**b**) Sensor and Moteino embedded into protection box. (**c**) Deployment of the NVIS antenna. (**d**) NVIS node with the Raspberry Pi and peripheral elements.

This deployment was tested for an entire week, and the monitoring service could be successfully executed every day, being able to review sensor data collected in both remote locations from the control center. The DTN architecture with BP handled the NVIS network's predicted and unpredicted disruptions, exchanging data bundles between the remote DTN nodes. With the functional tests in Antarctica, which were later completed in Catalonia, O5 was achieved, following the design recommendations extracted from the simulations' assessment and the trustworthiness model, and validating the proposed architecture for the monitoring service.

Chapter 6

FINAL CONCLUSIONS AND FURTHER WORK

This thesis focuses on improving the trustworthiness of harsh IoT environments with long-distance adversarial networks, focusing on the use case of the NVIS-based IoT long-distance network of the SHETLAND-NET project in Antarctica, by proposing and assessing new mechanisms and architectures for the permafrost telemetry use case. Three main contributions can be extracted from this thesis:

1. The definition of the scope of trustworthiness for harsh IoT environments in long-distance challenging networks.

2. A multi-layered trustworthiness model, based on the data, network, social, and consensus layers, used to assess the trustworthiness of a given use case quantitively, detect its weaknesses and apply appropriate countermeasures for trustworthiness improvement.

3. The proposal of new mechanisms and an IoT telemetry service architecture to provide a trustworthy permafrost monitoring service in Antarctica, which was validated through simulations and the deployment of a PoC testbed.

In the initial phase of the thesis, it was hypothesized that:

"Achieving reliable data exchange over a harsh environment is possible by improving its trustworthiness through the design of a multi-layered model that takes into account different facets of trustworthiness and through the implementation of the model's associated countermeasures."

To verify or deny the hypothesis, six research questions were posed:

RQ1. What is the definition and scope of the trustworthiness term in the field of Cyber Physical Systems (CPS)?

RQ2. How can trustworthiness be measured?

RQ3. How can a model be used to foresee, assess, and improve the achieved trustworthiness in harsh environments?

RQ4. How does using modern transport protocols affect the achieved trustworthiness in adversarial networks?

RQ5. How does a DTN architecture affect the trustworthiness of adversarial networks?

RQ6. Are the reached levels of trustworthiness in real implementations match the ones derived from the assessment through the trustworthiness model?

RQ1 was answered by accomplishing O1 (establish a scope for the trustworthiness term.). We set the scope of trustworthiness in CPS into four main areas: data trustworthiness, network trustworthiness, social trustworthiness, and consensus.

RQ2 was answered by accomplishing O2 (define a model that enables trustworthiness accountability to detect weaknesses and propose appropriate countermeasures for trustworthiness improvement). We proposed a four-layer model that provided trustworthiness accountability. We could quantify our use case's trustworthiness by measuring the achieved STR.

RQ3, RQ4, and RQ5 were answered by accomplishing O3 (assess the trustworthiness of the Antarctic use case) and O4 (propose new mechanisms to improve the trustworthiness of the Antarctic use case). The trustworthiness assessment was made using the proposed model and implementing the permafrost use case into the Riverbed Modeler simulator. In summary, three operational modes, six transport protocols, an opportunistic networking scheme, and a DTN scheme were tested, as well as other parameters of the service were modified throughout simulations. Three significant design and testing iterations were performed. The final proposal consisted of a 2-tier IoT network architecture, using LoRa in the access network and BP over NVIS in the backbone network (see Figure 28). EAATP and TCP CUBIC were the recommended transport protocols. This architecture guaranteed the required trustworthiness level for the permafrost telemetry service if 64 or fewer measuring spots and five or fewer redundant sensors were deployed per each NVIS gateway. The simulation results validated the goodness of the trustworthiness model.

Finally, RQ6 was answered through O5 (deploy a PoC of the proposed service architecture in the field during the 2021-2022 Antarctic campaign) and completing the PoC deployment in Catalonia. Unfortunately, during the Antarctic campaign, we had to interrupt and cancel our deployment in the field due to a COVID-19 outbreak in the Spanish delegation. However, we could independently validate the LoRa access network and the BP over the NVIS backbone network, establishing a 93 km link between Spain's Juan Carlos I base on Livingston Island and Uruguay's Artigas base on King George Island. Moreover, to answer RQ6 and finish the PoC deployment, the validation tests were completed in Catalonia, deploying the same scenario as in Antarctica but placing the remote NVIS nodes in rural areas from Hostalric and Cambrils, distanced by 150 km.

Overall, given that RQ1-6 could be answered satisfactorily, it is considered that a reliable data exchange over a harsh IoT environment (in this case, the Antarctic use case) could be achieved by improving its trustworthiness through the design of a model that enabled trustworthiness accountability and its improvement by implementing the model's associated countermeasures. Thus, the thesis hypothesis is confirmed.

Finally, a few future lines of work have been derived from this research:

1. The current model of the use case has some areas for improvement. Currently, the byzantine-fault model is based on a linear approach that associates faults to the battery discharge level and an initial fault probability. However, it is believed that other fault models could better represent the actual behavior of nodes in a harsh environment like Antarctica. As a future line, more complex fault models such as the ones described in [113] could be explored.

In addition, the PBFT algorithm was the only consensus mechanism modeled into the simulator, given that this thesis was focused on other studies, such as transport protocols and DTNs, rather than consensus mechanisms. Given the poor scalability of PBFT, some modern consensus mechanisms such as Delegated Practical Byzantine Fault Tolerance (DPBFT) [144] or RapidChain [145] have been proposed in the literature to solve this issue, which would require more complex modeling into the simulator. Assessing the use of new consensus mechanisms is another future line of this research.

2. The use of drones is growing worldwide, with new applications and usages emerging regularly. In rural areas, drones can conduct many different actions, such as monitoring based on cameras or other sensors, providing connectivity as a mobile gateway, or even transporting products in hard-to-reach areas. Many of these uses can be directly linked with primary sector activity, but their use can be extended to other fields.

   When various drones work together for data gathering and exchange, they form a FANET. It is thought that for the Antarctic use case, the coverage of the LoRa access network could be increased by deploying a FANET. Currently, the placement of GTN-P stations is limited by the coverage area of LoRa, which is strongly related to the presence of LoS. With a FANET, GTN-P stations could be extended to zones without LoS with the gateway. This FANET would be formed by drones capable of downloading sensor data when they enter the contact zone. The DTN architecture would have to be extended to the access network to achieve this, using the BP with LoRa. This would demand the development of a CLA for LoRa. These changes would mean a major iteration in the system's architecture, which would need a new trustworthiness assessment to evaluate if the required level of trustworthiness could still be reached.

3. Ionosondes are radars transmitting vertical pulses of a sweep of frequencies, usually in the range of 0.1–20 MHz. The signals reflected from the ionosphere are given in a graph of the time of reflection as a function of the frequency called the ionogram. These ionograms are used to detect the best operating frequencies for data

transmission. Currently, NVIS channel sounding is performed by sending ionosondes that block channel utilization (sensor data cannot be transmitted) during the sounding period.

As a step beyond simply sharing the spectrum, the codesign of sensing and communication functionalities seems promising. Such an approach allows considerable gains in terms of spectral, energy, hardware, and cost efficiency. This type of research, normally referred to as integrated sensing and communication (ISAC), is a paradigm change where the previously competing sensing and communication operations can be jointly optimized via the shared use of a single hardware platform and a joint signal processing framework. A future line of research would be designing an ISAC device capable of using ionosondes as data transmission signals, maximizing NVIS channel time availability for sounding and data exchange purposes. Given that this hardware should be capable of working at multiple frequencies for channel sounding, the system could also be used to transmit data at all these frequencies, solving the current problem of nighttime unavailability.

4. In challenging networks such as LoRa and NVIS networks, Software-Defined Networking (SDN) [146] might be able to optimize the network's resources to optimize its performance and prevent packet losses, improving the system's trustworthiness. For example, by decoupling the control plane from the data plane, the ISAC system described below could be used to adjust the operating frequencies on-the-go to reconfigurable antennas based on the information received from channel soundings, optimizing the transmitting channel, and minimizing packet losses. Other SDN examples that could apply to this use case are the implementation of adaptive QoS and routing policies defined by software in multipath NVIS networks, which could optimize bandwidth allocation and increase the effective network usage in congestion or packet-loss situations in these capacity-constrained networks.

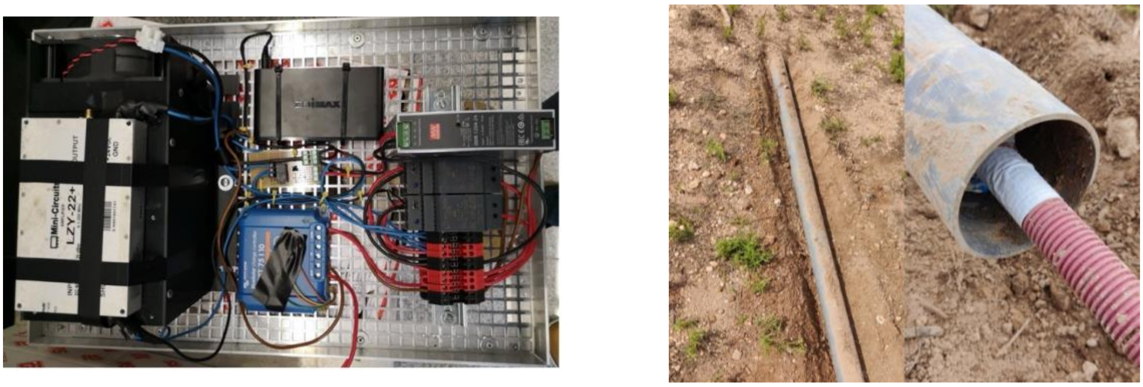
5. An important goal of a DTN architecture is to accommodate a wide range of networking technologies and environments. BP requires the services of a CLA to send and receive bundles using the service of some "native" link, network, or internet

protocol. The CLA must define the syntax and format of the messages exchanged between BP and the native protocol, acting as an interface between both layers.

Currently, CLAs have been specified for well-known internet protocols, such as TCP [147], UDP [148], or HTTP [149]. However, given that EAATP is a novel proprietary transport protocol, a CLA of it does not exist. Thus, to deploy the proposed DTN architecture with EAATP in a deployment with real hardware, it is necessary to design, develop, and implement a CLA for this transport protocol in physical hardware. Given the results of the simulations, it is believed that using EAATP could improve the overall system's trustworthiness of the physical deployment.

6. Obviously, a future objective is to deploy in Antarctica a permanent architecture that can provide service to other research proposes. The research group aims to accomplish it in future campaigns. To do so, and given the harsh conditions in Antarctica, improvements to the current NVIS hardware are needed to guarantee its autonomy and trustworthiness for an entire timespan between consecutive campaigns.

   First, for NVIS nodes placed far from research bases, a power supply system that provided enough autonomy for a whole year would be needed. This system would also be necessary for nodes placed in research bases that only operate during the Antarctic summer, given that, in this case, all systems and power generators are shut down during the winter season. Currently, a prototype of an NVIS node powered by high-capacity batteries is being tested in Hostalric. Secondly, the current NVIS hardware is vulnerable to the harsh climatic conditions of Antarctica, especially to the strong gusts of wind that could damage the antenna of big dimensions exposed outdoors. For this reason, underground antennas buried in the soil have been proposed. Currently, an underground antenna is being tested to provide NVIS communications in Cambrils. However, the possible effects of snow covering in Antarctica need to be assessed for this type of antenna. Figure 36 shows these next-generation prototypes.



(**a**) (**b**)

Figure 36. Next-generation NVIS hardware prototypes. (**a**) Battery-based power supply tested in Hostalric. (**b**) Underground antenna tested in Cambrils.

7. It is also desirable to expand the usage of the proposed architecture to other use cases so that Antarctic research projects can be modernized. Currently, we started working with the use case of Antarctic volcanoes' monitoring on Deception Island (South Shetland Islands) [150]. Deception Island is one of the most active volcanoes in Antarctica (revisit Figure 4 to see its location). In the 1988–1989 austral summer, after the most recent eruptive process on the island (1967–1970), monitoring of volcanic activity through geophysical and geodetic techniques was resumed by Spanish and Argentinean scientists. A geodetic network was deployed to monitor the island's tectonic and volcanic behavior. Currently, this network consists of 15 geodetic benchmarks located around Port Foster, Deception's inner bay open to the sea.

   However, current studies can only analyze data once it is manually exported from the geodetic sensors. Thus, it is not possible to monitor the status of the Deception volcano in real time. Live monitoring would be a major advance in the surveillance of this volcano, given that it could allow a fast response to unexpected situations. For instance, a new eruption like the one that happened in the late 1960s could potentially vanish the entire Spanish Gabriel de Castilla research base on that island. Currently, if a catastrophe like this happened during the winter season, the Spanish delegacy could only acknowledge it once they arrived on the island, being forced to cancel the entire campaign and wasting many resources unnecessarily. Contrarily, if live monitoring was enabled, a fast response could be provided, canceling the campaign before wasting the resources and dedicating all of them to rebuilding the base station.

Given that the Gabriel de Castilla base closes during winter, it is proposed to deploy a similar architecture to the used one in this thesis, providing coverage to the geodetic network through LoRa and interconnecting the Gabriel de Castilla base to another one with permanent Internet connectivity through an NVIS link.

8. The analyzed consensus mechanisms bring a considerable overhead to the communication network that degrades the overall system throughput limiting the data processing capabilities. This is especially concerning in remote sensing scenarios with low bandwidth capacities. Recently, quantum technologies have emerged as an appealing alternative to building prospective quantum Internet by maximizing the benefits of inherent privacy and instantaneous coordination. In the long term, we aim to apply quantum networking techniques to the architecture. Concretely, we propose to use a quantum consensus mechanism [151], which eliminates the problem of added network congestion. If this was accomplished, the quantum consensus mechanisms could help improve the overall system's trustworthiness.

BIBLIOGRAPHY

[1] L. Atzori, A. Iera, and G. Morabito, "Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm," *Ad Hoc Networks*, vol. 56, pp. 122–140, 2017, doi: 10.1016/J.ADHOC.2016.12.004.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, 2014, doi: 10.1109/JIOT.2014.2306328.

[3] S. Chen *et al.*, "Internet of Things Based Smart Grids Supported by Intelligent Edge Computing," *IEEE Access*, vol. 7, pp. 74089–74102, 2019, doi: 10.1109/ACCESS.2019.2920488.

[4] Z. Lu, G. Qu, and Z. Liu, "A Survey on Recent Advances in Vehicular Network Security, Trust, and Privacy," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 760–776, 2019, doi: 10.1109/TITS.2018.2818888.

[5] P. K. Malik *et al.*, "Industrial Internet of Things and its Applications in Industry 4.0: State of The Art," *Computer Communications*, vol. 166, pp. 125–139, 2021, doi: 10.1016/J.COMCOM.2020.11.016.

[6] G. Sebestyen, A. Hangan, S. Oniga, and Z. Gal, "eHealth solutions in the context of internet of things," in *Proceedings of 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, AQTR 2014*, 2014, pp. 1–6. doi: 10.1109/AQTR.2014.6857876.

[7] O. Friha *et al.*, "Internet of Things for the Future of Smart Agriculture: A Comprehensive Survey of Emerging Technologies," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 4, pp. 718–752, 2021, doi: 10.1109/JAS.2021.1003925.

[8] A. Sestino, M. I. Prete, L. Piper, and G. Guido, "Internet of Things and Big Data as enablers for business digitalization strategies," *Technovation*, vol. 98, p. 102173, 2020, doi: 10.1016/J.TECHNOVATION.2020.102173.

[9] M. Kanao, T. Genti, and M.-Y. Yamamoto, *Antarctica - A Key to Global Change*. IntechOpen, 2019. doi: 10.5772/intechopen.75265.

[10] M. C. Kennicutt *et al.*, "Delivering 21st century Antarctic and Southern Ocean science," *Antarctic Science*, vol. 28, no. 6, pp. 407–423, 2016, doi: 10.1017/S0954102016000481.

[11] J. Porte, J. M. Maso, J. L. Pijoan, and D. Badia, "Sensing System for Remote Areas in Antarctica," *Radio Science*, vol. 55, no. 3, pp. 1–12, 2020, doi: 10.1029/2019RS006920.

[12] B. Jolly, A. Willig, A. McDonald, M. Pannell, and G. Plank, "SNOWWEB - Wirelessly connected weather stations in Antarctica," in *38th Annual IEEE Conference on Local Computer Networks - Workshops*, 2013, pp. 194–202. doi: 10.1109/LCNW.2013.6758519.

[13] J. Gaelens, P. van Torre, J. Verhaevert, and H. Rogier, "Lora mobile-to-base-station channel characterization in the Antarctic," *Sensors*, vol. 17, no. 8, p. 1903, 2017, doi: 10.3390/s17081903.

[14] "NVIS sensors network for the South Shetland Islands." https://www.salleurl.edu/en/research/research-lines-and-institutes/antarctica-project/summary (accessed Oct. 10, 2022).

[15] J. Porté, J. L. Pijoan, J. Masó, D. Badia, A. Zaballos, and R. M. Alsina-pagès, "Advanced HF Communications for Remote Sensors in Antarctica," *Antarctica-A Key To Global Change*, pp. 21–39, 2018, doi: 10.5772/intechopen.81108.

[16] J. Maso, J. Porte, J. L. Pijoan, and D. Badia, "Internet of things communications for remote sensors in Antarctica using NVIS," 2019.

[17] J. Porte, A. Briones, J. M. Maso, C. Pares, A. Zaballos, and J. L. Pijoan, "Heterogeneous wireless IoT architecture for natural disaster monitorization," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, pp. 1–27, 2020, doi: 10.1186/s13638-020-01793-3.

[18] J. Male, J. Porte, T. Gonzalez, J. M. Maso, J. L. Pijoan, and D. Badia, "Analysis of the Ordinary and Extraordinary Ionospheric Modes for NVIS Digital Communications Channels," *Sensors*, vol. 21, no. 6, p. 2210, 2021, doi: 10.3390/s21062210.

[19] T. Gonzalez *et al.*, "SC-FDE Layer for Sensor Networks in Remote Areas Using NVIS Communications," *Electronics*, vol. 10, no. 14, p. 1636, 2021, doi: 10.3390/electronics10141636.

[20] M. Á. de Pablo Hernández *et al.*, "Frozen ground and snow cover monitoring in livingston and deception islands, antarctica: Preliminary results of the 2015-2019 PERMASNOW project," *Geographical Research Letters*, vol. 46, no. 1, pp. 187–222, 2020, doi: 10.18172/cig.4381.

[21] M. Ramos, G. Vieira, M. A. de Pablo, A. Molina, and J. J. Jimenez, "Transition from a Subaerial to a Subnival Permafrost Temperature Regime Following Increased Snow Cover (Livingston Island, Maritime Antarctic)," *Atmosphere*, vol. 11, no. 12, p. 1332, 2020, doi: 10.3390/ATMOS11121332.

[22] M. Prieto, M. A. de Pablo, M. Ramos, and J. J. Jimenez, "Experimental Tests and Performance Evaluation of a VHF Data Transceiver Prototype for Operation in the Antarctic Regions," *Radioengineering*, vol. 29, no. 1, pp. 132–139, 2020, doi: 10.13164/re.2020.0132.

[23] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications - SIGCOMM '03*, 2003, pp. 27–34. doi: 10.1145/863956.863960.

[24] K. L. Scott and S. Burleigh, "Bundle Protocol Specification," *RFC 5050*, 2007. https://tools.ietf.org/html/rfc5050 (accessed Oct. 10, 2022).

[25] S. Bounsiar, F. Z. Benhamida, A. Henni, D. L. de Ipiña, and D. C. Mansilla, "How to Enable Delay Tolerant Network Solutions for Internet of Things: From Taxonomy to Open Challenges," *Proceedings*, vol. 31, no. 1, p. 24, 2019, doi: 10.3390/proceedings2019031024.

[26] S. Ha, I. 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, doi: 10.1145/1400097.1400105.

[27] A. Briones, A. Mallorquí, A. Zaballos, and R. M. de Pozuelo, "Adaptive and aggressive transport protocol to provide QoS in cloud data exchange over Long Fat Networks,"

Future Generation Computer Systems, vol. 115, pp. 34–44, 2021, doi: 10.1016/j.future.2020.08.043.

[28] A. Briones, A. Mallorquí, A. Zaballos, and R. M. de Pozuelo, "Wireless loss detection over fairly shared heterogeneous long fat networks," *Electronics*, vol. 10, no. 9, p. 987, 2021, doi: 10.3390/electronics10090987.

[29] "Chile to Build Submarine Cable to Antarctica | Fiber Optic Cables | NewsFeed." https://www.subcableworld.com/newsfeed/fiber-optic-cables/chile-to-build-submarine-cable-to-antarctica (accessed Oct. 10, 2022).

[30] S. Zanero, "Cyber-Physical Systems," *Computer*, vol. 50, no. 4, pp. 14–16, 2017, doi: 10.1109/MC.2017.105.

[31] "Riverbed Modeler." https://www.riverbed.com/gb/products/npm/riverbed-modeler.html (accessed Oct. 10, 2022).

[32] M. Crawford and E. Liongosary, *IIC Journal of Innovation*, vol. 9. The Industrial Internet of Things Consortium, 2018.

[33] L. A. Tang *et al.*, "Trustworthiness analysis of sensor data in cyber-physical systems," *Journal of Computer and System Sciences*, vol. 79, no. 3, pp. 383–401, 2013, doi: 10.1016/j.jcss.2012.09.012.

[34] N. Haron, J. Jaafar, I. A. Aziz, M. H. Hassan, and M. I. Shapiai, "Data trustworthiness in Internet of Things: A taxonomy and future directions," in *2017 IEEE Conference on Big Data and Analytics (ICBDA)*, 2017, pp. 25–30. doi: 10.1109/ICBDAA.2017.8284102.

[35] H. Yuan, X. Zhao, and L. Yu, "A Distributed Bayesian Algorithm for data fault detection in wireless sensor networks," in *2015 International Conference on Information Networking (ICOIN)*, 2015, pp. 63–68. doi: 10.1109/ICOIN.2015.7057858.

[36] G. Zhang and R. Li, "Fog computing architecture-based data acquisition for WSN applications," *China Communications*, vol. 14, no. 11, pp. 69–81, 2017, doi: 10.1109/CC.2017.8233652.

[37] R. Fantacci, F. Nizzi, T. Pecorella, L. Pierucci, and M. Roveri, "False Data Detection for Fog and Internet of Things Networks," *Sensors*, vol. 19, no. 19, p. 4235, 2019, doi: 10.3390/s19194235.

[38] M. M. Hassan, A. Gumaei, S. Huda, and A. Almogren, "Increasing the Trustworthiness in the Industrial IoT Networks through a Reliable Cyberattack Detection Model," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 9, pp. 6154–6162, 2020, doi: 10.1109/TII.2020.2970074.

[39] T. Semong *et al.*, "Intelligent load balancing techniques in software defined networks: A survey," *Electronics*, vol. 9, no. 7, p. 1091, 2020, doi: 10.3390/electronics9071091.

[40] M. Polese, F. Chiariotti, E. Bonetto, F. Rigotto, A. Zanella, and M. Zorzi, "A Survey on Recent Advances in Transport Layer Protocols," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 4, pp. 3584–3608, 2019, doi: 10.1109/COMST.2019.2932905.

[41] H. Prasad and S. Babu, "A Survey on Network Routing Protocols in Internet of Things (IOT)," *International Journal of Computer Applications*, vol. 160, no. 2, pp. 18–22, 2017, doi: 10.5120/ijca2017912973.

[42] D. E. M. Ahmed and O. O. Khalifa, "A Comprehensive Classification of MANETs Routing Protocols," *International Journal of Computer Applications Technology and Research*, vol. 6, no. 3, pp. 141–158, 2017, doi: 10.7753/ijcatr0603.1004.

[43] H. Mrabet, S. Belguith, A. Alhomoud, and A. Jemai, "A survey of IoT security based on a layered architecture of sensing and data analysis," *Sensors*, vol. 20, no. 13, p. 3625, 2020, doi: 10.3390/s20133625.

[44] L. Atzori, A. Iera, and G. Morabito, "SIoT: Giving a Social Structure to the Internet of Things," *IEEE Communications Letters*, vol. 15, no. 11, pp. 1193–1195, 2011, doi: 10.1109/LCOMM.2011.090911.111340.

[45] M. Nitti, R. Girau, L. Atzori, A. Iera, and G. Morabito, "A subjective model for trustworthiness evaluation in the social Internet of Things," in *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications - (PIMRC)*, 2012, pp. 18–23. doi: 10.1109/PIMRC.2012.6362662.

[46] M. Nitti, R. Girau, and L. Atzori, "Trustworthiness Management in the Social Internet of Things," *IEEE Transactions on knowledge and data engineering*, vol. 26, no. 5, pp. 1253–1266, 2013, doi: 10.1007/s11277-017-4319-8.

[47] Z. Lin and L. Dong, "Clarifying Trust in Social Internet of Things," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 2, pp. 234–248, 2018, doi: 10.1109/TKDE.2017.2762678.

[48] M. A. Azad, S. Bag, F. Hao, and A. Shalaginov, "Decentralized Self-Enforcing Trust Management System for Social Internet of Things," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2690–2703, 2020, doi: 10.1109/JIOT.2019.2962282.

[49] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, "A Survey of Distributed Consensus Protocols for Blockchain Networks," *IEEE Communications Surveys and Tutorials*, vol. 22, no. 2, pp. 1432–1465, 2020, doi: 10.1109/COMST.2020.2969706.

[50] U. Bodkhe, D. Mehta, S. Tanwar, P. Bhattacharya, P. K. Singh, and W. C. Hong, "A survey on decentralized consensus mechanisms for cyber physical systems," *IEEE Access*, vol. 8, pp. 54371–54401, 2020, doi: 10.1109/ACCESS.2020.2981415.

[51] L. Lao, Z. Li, S. Hou, B. Xiao, S. Guo, and Y. Yang, "A survey of IoT applications in blockchain systems: Architecture, consensus, and traffic modeling," *ACM Computing Surveys*, vol. 53, no. 1, pp. 1–32, 2020, doi: 10.1145/3372136.

[52] M. Kazmi, A. Shamim, N. Wahab, and F. Anwar, "Comparison of TCP Tahoe, Reno, New Reno, Sack and Vegas in IP and MPLS Networks under Constant Bit Rate Traffic," in *International Conference on Advanced Computational Technology and Creative Media (ICACTCM)*, 2014, pp. 33–38. doi: 10.15242/iie.e0814537.

[53] T. Kelly, "Scalable TCP: improving performance in highspeed wide area networks," *ACM SIGCOMM computer communication Review*, vol. 33, no. 2, pp. 83–91, 2003, doi: 10.1145/956981.956989.

[54] C. Jin, D. X. Wei, and S. H. Low, "FAST TCP: Motivation, Architecture, Algorithms, Performance," *IEEE/ACM transactions on Networking*, vol. 14, no. 6, pp. 1246–1259, 2006, doi: 10.1109/INFCOM.2004.1354670.

[55] D. Leith and R. Shorten, "H-TCP protocol for high-speed long-distance networks," 2004.

[56] Lisong Xu, K. Harfoush, and Injong Rhee, "Binary increase congestion control (BIC) for fast long-distance networks," in *IEEE INFOCOM 2004*, 2004, vol. 4, pp. 2514–2524. doi: 10.1109/INFCOM.2004.1354672.

[57] R. Marx, J. Herbots, W. Lamotte, and P. Quax, "Same Standards, Different Decisions: A Study of QUIC and HTTP/3 Implementation Diversity," in *EPIQ 2020 - Proceedings of the 2020 Workshop on the Evolution, Performance, and Interoperability of QUIC*, 2020, pp. 14–20. doi: 10.1145/3405796.3405828.

[58] N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, "BBR: Congestion-Based Congestion Control," *Queue*, vol. 14, no. 5, pp. 20–53, 2016, doi: 10.1145/3012426.3022184.

[59] V. Arun, H. Balakrishnan, M. I. T. Csail, S. Design, and I. Nsdi, "Copa : Practical Delay-Based Congestion Control for the Internet," in *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, 2018, pp. 329–342.

[60] F. Y. Yan *et al.*, "Pantheon: The training ground for internet congestion-control research," in *Proceedings of the 2018 USENIX Annual Technical Conference, USENIX ATC 2018*, 2018, pp. 731–743.

[61] Y. Zaki, T. Pötsch, J. Chen, L. Subramanian, and C. Görg, "Adaptive Congestion Control for Unpredictable Cellular Networks," *Computer Communication Review*, vol. 45, no. 4, pp. 509–522, 2015, doi: 10.1145/2785956.2787498.

[62] M. Dong, Q. Li, D. Zarchy, P. B. Godfrey, and M. Schapira, "PCC: Re-architecting congestion control for consistent high performance," in *Proceedings of the 12th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2015*, 2015, pp. 395–408.

[63] C. P. Fu and S. C. Liew, "TCP Veno: TCP Enhancement for Transmission Over Wireless Access Networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 2, pp. 216–228, 2003, doi: 10.1109/JSAC.2002.807336.

[64] L. a. Grieco and S. Mascolo, "Performance evaluation and comparison of Westwood+, New Reno, and Vegas TCP congestion control," *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 2, pp. 25–38, 2004, doi: 10.1145/997150.997155.

[65] M. R. Kanagarathinam, S. Singh, I. Sandeep, A. Roy, and N. Saxena, "D-TCP: Dynamic TCP congestion control algorithm for next generation mobile networks," in *CCNC 2018 - 2018 15th IEEE Annual Consumer Communications and Networking Conference*, 2018, pp. 1–6. doi: 10.1109/CCNC.2018.8319185.

[66] E. H. K. Wu, Y. U. C. Huang, and G. K. Chang, "EJTCP: Enhanced Jitter-based TCP for Wireless Broadband Networks," *Journal of Information Science and Engineering*, vol. 23, no. 6, pp. 1663–1679, 2007.

[67] J. M. Chen, C. H. Chu, E. H. K. Wu, M. F. Tsai, and J. R. Wang, "Improving SCTP performance by jitter-based congestion control over wired-wireless networks," *Eurasip Journal on Wireless Communications and Networking*, vol. 2011, pp. 1–13, 2011, doi: 10.1155/2011/103027.

[68] S. Burleigh *et al.*, "Delay-tolerant networking: An approach to interplanetary internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003, doi: 10.1109/MCOM.2003.1204759.

[69] J. J. P. C. Rodrigues, *Advances in delay-tolerant networks (DTNs): Architecture and enhanced performance*. Woodhead Publishing, 2020.

[70] S. Burleigh, "Interplanetary Overlay Network: An implementation of the DTN bundle protocol," 2007. https://trs.jpl.nasa.gov/bitstream/handle/2014/41732/07-0133.A.pdf?sequence=1 (accessed Oct. 10, 2022).

[71] I. Katz, "A Delay-Tolerant Networking Framework for Mobile Underwater Acoustic Networks," 2007.

[72] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 4, pp. 23–33, 2007, doi: 10.1145/1347364.1347372.

[73] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi, "Hardware Design Experiences in ZebraNet," in *Proceedings of the 2nd international conference on Embedded networked sensor systems - SenSys '04*, 2004, pp. 227–238. doi: 10.1145/1031495.

[74] T. S. Field and Z. J. Haas, "The shared wireless infostation model: a new ad hoc networking paradigm (or where there is a whale, there is a way)," in *Proceedings of the 4th*

ACM international symposium on Mobile ad hoc networking & computing - MobiHoc '03, 2003, pp. 233–244. doi: 10.1145/778415.778443.

[75] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting Mobility for Energy Efficient Data Collection in Wireless Sensor Networks," *Mobile Networks and Applications*, vol. 11, pp. 327–339, 2006, doi: 10.1007/S11036-006-5186-9.

[76] R. Sherwood and S. Chien, "Sensor webs for science: New directions for the future," in *2007 AIAA InfoTech at Aerospace Conference*, 2007, pp. 1360–1372. doi: 10.2514/6.2007-2842.

[77] V. N. G. J. Soares, F. Farahmand, and J. J. P. C. Rodrigues, "A layered architecture for vehicular delay-tolerant networks," in *2009 IEEE Symposium on Computers and Communications*, 2009, pp. 122–127. doi: 10.1109/ISCC.2009.5202332.

[78] P. R. Pereira, A. Casaca, J. J. P. C. Rodrigues, V. N. G. J. Soares, J. Triay, and C. Cervelló-Pastor, "From delay-tolerant networks to vehicular delay-tolerant networks," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 4, pp. 1166–1182, 2012, doi: 10.1109/SURV.2011.081611.00102.

[79] K. Scott, "Disruption tolerant networking proxies for on-the-move tactical networks," in *IEEE Military Communications Conference MILCOM*, 2005, pp. 3226–3231. doi: 10.1109/MILCOM.2005.1606153.

[80] J. Jormakka, H. Jormakka, and J. Väre, "A lightweight management system for a military ad hoc network," in *Information Networking. Towards Ubiquitous Networking and Services. ICOIN 2007. Lecture Notes in Computer Science*, vol. 5200, Springer, 2008, pp. 533–543. doi: 10.1007/978-3-540-89524-4_53.

[81] H. K. Kalitay and M. K. Nambiarz, "Designing WANem : A Wide Area Network emulator tool," in *2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011)*, 2011, pp. 1–4. doi: 10.1109/COMSNETS.2011.5716495.

[82] J. Sánchez, A. Mallorquí, A. Briones, A. Zaballos, and G. Corral, "An Integral Pedagogical Strategy for Teaching and Learning IoT Cybersecurity," *Sensors*, vol. 20, no. 14, p. 3970, 2020, doi: 10.3390/s20143970.

[83] A. Mallorquí and A. Zaballos, "A heterogeneous layer-based trustworthiness model for long backhaul nvis challenging networks and an iot telemetry service for antarctica," *Sensors*, vol. 21, no. 10, p. 3446, 2021, doi: 10.3390/s21103446.

[84] A. Mallorquí, A. Zaballos, A. Briones, and G. Corral, "Confiabilidad en la capa de transporte para la red de sensores antártica," in *Actas de las XV Jornadas de Ingeniería Telemática (JITEL 2021)*, 2021, pp. 63–70.

[85] A. Mallorquí, A. Zaballos, and A. Briones, "DTN Trustworthiness for Permafrost Telemetry IoT Network," *Remote Sensing*, vol. 13, no. 22, p. 4493, 2021, doi: 10.3390/rs13224493.

[86] A. Mallorquí, A. Zaballos, and D. Serra, "The Antarctic Delay Tolerant Network," in *Proceedings of the 27th IEEE Symposium on Computers and Communications (ISCC 2022)*, 2022, pp. 1–7.

[87] A. Mallorqui, A. Zaballos, and D. Serra, "A Delay Tolerant Network for Antarctica," *IEEE Communications Magazine*, pp. 1–7, 2022, doi: 10.1109/MCOM.007.2200147.

[88] "Sprint 4.0 – Sprint 4.0 Project." https://www.sprint40.eu/ (accessed Oct. 10, 2022).

[89] "Home - ATHIKA." https://athika.eu/ (accessed Oct. 10, 2022).

[90] "REWIRE." https://rewireproject.eu/ (accessed Oct. 10, 2022).

[91] "Antarctic Stations - Scientific Research Bases and Facilities." https://www.coolantarctica.com/Community/antarctic_bases.php (accessed Oct. 10, 2022).

[92] S. Jerez, M. Motas, J. Benzal, J. Diaz, and A. Barbosa, "Monitoring trace elements in Antarctic penguin chicks from South Shetland Islands, Antarctica," *Marine Pollution Bulletin*, vol. 69, no. 1–2, pp. 67–75, 2013, doi: 10.1016/j.marpolbul.2013.01.004.

[93] L. G. Sancho, A. Pintado, and T. G. A. Green, "Antarctic Studies Show Lichens to be Excellent Biomonitors of Climate Change," *Diversity*, vol. 11, no. 3, p. 42, 2019, doi: 10.3390/D11030042.

[94] A. Galati, *Delay Tolerant Network*. LAP Lambert Academic Publishing, 2010.

[95] M. Regi, M. de Lauretis, G. Redaelli, and P. Francia, "ULF Geomagnetic Activity Signatures in the Atmospheric Parameters in Antarctica," in *Antarctica - A Key To Global Change*, IntechOpen, 2019, pp. 5–20. doi: 10.5772/intechopen.81106.

[96] S. P. Palm, Y. Yang, and V. Kayetha, "New Perspectives on Blowing Snow in Antarctica and Implications for Ice Sheet Mass Balance," *Antarctica - A Key To Global Change*, 2018, doi: 10.5772/intechopen.81319.

[97] E. R. Thomas and D. R. Tetzner, "The Climate of the Antarctic Peninsula during the Twentieth Century: Evidence from Ice Cores," *Antarctica - A Key To Global Change*, 2018, doi: 10.5772/intechopen.81507.

[98] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of Things (IoT) communication protocols: Review," in *2017 8th International Conference on Information Technology (ICIT)*, 2017, pp. 685–690. doi: 10.1109/ICITECH.2017.8079928.

[99] L. Oliveira, J. J. P. C. Rodrigues, S. A. Kozlov, R. A. L. Rabêlo, and V. H. C. de Albuquerque, "MAC layer protocols for internet of things: A survey," *Future Internet*, vol. 11, no. 1, pp. 1–42, 2019, doi: 10.3390/fi11010016.

[100] S. Zaidi, M. Atiquzzaman, and C. T. Calafate, "Internet of Flying Things (IoFT): A Survey," *Computer Communications*, vol. 165, pp. 53–74, 2021, doi: 10.1016/j.comcom.2020.10.023.

[101] D. Liu *et al.*, "Opportunistic UAV utilization in wireless networks: Motivations, applications, and challenges," *IEEE Communications Magazine*, vol. 58, no. 5, pp. 62–68, 2020, doi: 10.1109/MCOM.001.1900687.

[102] A. Guillen-Perez and M. D. Cano, "Flying ad hoc networks: A new domain for network communications," *Sensors*, vol. 18, no. 10, p. 3571, 2018, doi: 10.3390/s18103571.

[103] S. Lee, Y. Wu, and D. Mortari, "Satellite constellation design for telecommunication in Antarctica," *International Journal of Satellite Communications and Networking*, vol. 34, no. 6, pp. 725–737, 2016, doi: 10.1002/sat.1128.

[104] E. A. Lee, "Cyber physical systems: Design challenges," in *2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC).*, 2008, pp. 363–369. doi: 10.1109/ISORC.2008.25.

[105] N. Chalaemwongwan and W. Kurutach, "State of the art and challenges facing consensus protocols on blockchain," in *2018 International Conference on Information Networking (ICOIN)*, 2018, pp. 957–962. doi: 10.1109/ICOIN.2018.8343266.

[106] M. Nitti, R. Girau, L. Atzori, and V. Pilloni, "Trustworthiness management in the IoT: The importance of the feedback," in *2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN)*, 2017, pp. 325–327. doi: 10.1109/ICIN.2017.7899434.

[107] A. Penning, L. Baumgärtner, J. Höchst, A. Sterz, M. Mezini, and B. Freisleben, "DTN7: An Open-Source Disruption-Tolerant Networking Implementation of Bundle Protocol 7," in *International Conference on Ad-Hoc Networks and Wireless*, 2019, pp. 196–209. doi: 10.1007/978-3-030-31831-4_14.

[108] A. U. Rehman, A. Jiang, A. Rehman, and A. Paul, "Weighted Based Trustworthiness Ranking in Social Internet of Things by using Soft Set Theory," in *2019 IEEE 5th International Conference on Computer and Communications (ICCC)*, 2019, pp. 1644–1648. doi: 10.1109/ICCC47050.2019.9064242.

[109] J. Seo, D. Ko, S. Kim, and S. Park, "A coordination technique for improving scalability of Byzantine fault-tolerant consensus," *Applied Sciences*, vol. 10, no. 21, p. 7609, 2020, doi: 10.3390/app10217609.

[110] M. T. Refaei, L. A. Dasilva, M. Eltoweissy, and T. Nadeem, "Adaptation of reputation management systems to dynamic network conditions in Ad Hoc networks," *IEEE Transactions on Computers*, vol. 59, no. 5, pp. 707–719, 2010, doi: 10.1109/TC.2010.34.

[111] A. D. Jun *et al.*, "Modeling and Simulation of LoRa in OPNET," in *Advanced Multimedia and Ubiquitous Engineering*, 2017, pp. 551–559. doi: 10.1007/978-981-10-5041-1_88.

[112] S. Distefano, "Evaluating reliability of WSN with sleep/wake-up interfering nodes," *International Journal of Systems Science*, vol. 44, no. 10, pp. 1793–1806, 2013, doi: 10.1080/00207721.2012.670293.

[113] X. Pan, F. di Maio, and E. Zio, "A benchmark of dynamic reliability methods for probabilistic safety assessment," in *2017 2nd International Conference on System Reliability and Safety (ICSRS)*, 2017, pp. 82–90. doi: 10.1109/ICSRS.2017.8272801.

[114] M. Castro and B. Liskov, "Practical Byzantine Fault Tolerance and Proactive Recovery," *ACM Transactions on Computer Systems (TOCS)*, vol. 20, no. 4, pp. 398–461, 2002, doi: 10.1145/571637.571640.

[115] K. Lei, Q. Zhang, L. Xu, and Z. Qi, "Reputation-Based Byzantine Fault-Tolerance for Consortium Blockchain," in *2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS)*, 2018, pp. 604–611. doi: 10.1109/PADSW.2018.8644933.

[116] R. M. Alsina-Pagès, M. Hervás, F. Orga, J. L. Pijoan, D. Badia, and D. Altadill, "Physical layer definition for a long-haul HF antarctica to Spain radio link," *Remote Sensing*, vol. 8, no. 5, p. 380, 2016, doi: 10.3390/rs8050380.

[117] "Location map of Low Island in the South Shetland Islands." https://en.wikipedia.org/wiki/Low_Island_(South_Shetland_Islands)#/media/File:Low-Island-location-map.png (accessed Oct. 10, 2022).

[118] A. Tovar, T. Friesen, K. Ferens, and B. McLeod, "A DTN wireless sensor network for wildlife habitat monitoring," in *Canadian Conference on Electrical and Computer Engineering*, 2010, pp. 1–5. doi: 10.1109/CCECE.2010.5575142.

[119] R. Matsuzaki, H. Ebara, and N. Muranaka, "Rescue support system with DTN for earthquake disasters," *IEICE Transactions on Communications*, vol. 98, no. 9, pp. 1832–1847, 2015, doi: 10.1587/transcom.E98.B.1832.

[120] V. N. G. J. Soares, F. Farahmand, and J. J. P. C. Rodrigues, "A layered architecture for Vehicular Delay-Tolerant Networks," in *2009 IEEE Symposium on Computers and Communications*, 2009, pp. 122–127. doi: 10.1109/ISCC.2009.5202332.

[121] S. Schildt, J. Morgenroth, W. B. Pöttner, and L. Wolf, "IBR-DTN: A lightweight, modular and highly portable Bundle Protocol implementation," *Electronic Communications of the EASST*, vol. 37, pp. 1–10, 2011, doi: 10.14279/tuj.eceasst.37.512.544.

[122] G. von Zengen, F. Büsching, W. Pöttner, and L. Wolf, "An Overview of µDTN: Unifying DTNs and WSNs," in *Proceedings of the 11th GI/ITG KuVS Fachgespräch Drahtlose Sensornetze (FGSN)*, 2012, pp. 1–4.

[123] F. M. Al-Turjman, A. E. Al-Fagih, W. M. Alsalih, and H. S. Hassanein, “A delay-tolerant framework for integrated RSNs in IoT,” *Computer Communications*, vol. 36, no. 9, pp. 998–1010, 2013, doi: 10.1016/j.comcom.2012.07.001.

[124] Z. Guo, B. Wang, and J. H. Cui, “Generic prediction assisted single-copy routing in underwater delay tolerant sensor networks,” *Ad Hoc Networks*, vol. 11, no. 3, pp. 1136–1149, 2013, doi: 10.1016/j.adhoc.2012.11.012.

[125] K. S. Wong and T. C. Wan, “Reliable multicast disruption tolerant networking: Conceptual implementation using message ferry,” in *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, 2017, pp. 1817–1822. doi: 10.1109/TENCON.2017.8228153.

[126] Y. Mao, C. Zhou, Y. Ling, and J. Lloret, “An optimized probabilistic delay tolerant network (DTN) routing protocol based on scheduling mechanism for internet of things (IoT),” *Sensors*, vol. 19, no. 2, p. 243, 2019, doi: 10.3390/s19020243.

[127] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, “Opportunistic IoT: Exploring the harmonious interaction between human and the internet of things,” *Journal of Network and Computer Applications*, vol. 36, no. 6, pp. 1531–1539, 2013, doi: 10.1016/j.jnca.2012.12.028.

[128] Y. Xu, V. Mahendran, and S. Radhakrishnan, “Internet of Hybrid Opportunistic Things: A novel framework for interconnecting IoTs and DTNs,” in *2016 IEEE conference on computer communications workshops (INFOCOM WKSHPS)*, 2016, pp. 1067–1068. doi: 10.1109/INFCOMW.2016.7562256.

[129] A. Elmangoush, A. Corici, M. Catalan, R. Steinke, T. Magedanz, and J. Oller, “Interconnecting standard M2M platforms to delay tolerant networks,” in *Proceedings - 2014 International Conference on Future Internet of Things and Cloud, FiCloud 2014*, 2014, pp. 258–263. doi: 10.1109/FiCloud.2014.48.

[130] A. Sathiaseelan, D. Trossen, I. Komnios, J. Ott, and J. Crowcroft, “Information Centric Delay Tolerant Networking: An Internet Architecture for the Challenged,” University of Cambridge, 2013. doi: 10.48456/tr-841.

[131] P. Manzoni, E. Hernández-Orallo, C. T. Calafate, and J. C. Cano, “A proposal for a publish/subscribe, disruption tolerant content island for fog computing,” in

SMARTOBJECTS 2017 - Proceedings of the 3rd Workshop on Experiences with the Design and Implementation of Smart Objects, co-located with MobiCom 2017, 2017, pp. 47–52. doi: 10.1145/3127502.3127511.

[132] F. M. R. Junior and C. A. Kamienski, "A Survey on Trustworthiness for the Internet of Things," *IEEE Access*, vol. 9, pp. 42493–42514, 2021, doi: 10.1109/ACCESS.2021.3066457.

[133] N. S. Labib, M. R. Brust, G. Danoy, and P. Bouvry, "Trustworthiness in IoT - A Standards Gap Analysis on Security, Data Protection and Privacy," in *2019 IEEE Conference on Standards for Communications and Networking, CSCN 2019*, 2019, pp. 1–7. doi: 10.1109/CSCN.2019.8931393.

[134] V. Bioglio, C. Condo, and I. Land, "Design of Polar Codes in 5G New Radio," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 29–40, 2021, doi: 10.1109/COMST.2020.2967127.

[135] J. Li, X. Li, X. Cheng, J. Yuan, and R. Zhang, "A trustworthiness-enhanced reliable forwarding scheme in mobile Internet of Things," *Journal of Network and Computer Applications*, vol. 140, pp. 40–53, 2019, doi: 10.1016/j.jnca.2019.05.003.

[136] V. Caballero, D. Vernet, and A. Zaballos, "Social Internet of Energy - A New Paradigm for Demand Side Management," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9853–9867, 2019, doi: 10.1109/JIOT.2019.2932508.

[137] C. Marche and M. Nitti, "Trust-Related Attacks and Their Detection: A Trust Management Model for the Social IoT," *IEEE Transactions on Network and Service Management*, vol. 18, no. 3, pp. 3297–3308, 2021, doi: 10.1109/TNSM.2020.3046906.

[138] "All about Moteino | LowPowerLab." https://lowpowerlab.com/guide/moteino/ (accessed Oct. 10, 2022).

[139] Y. Fang, P. Chen, G. Cai, F. C. M. Lau, S. C. Liew, and G. Han, "Outage-limit-approaching channel coding for future wireless communications: Root-protograph low-density parity-check codes," *IEEE Vehicular Technology Magazine*, vol. 14, no. 2, pp. 85–93, 2019, doi: 10.1109/MVT.2019.2903343.

[140] L. Fernandez, J. A. Ruiz-De-Azua, A. Calveras, and A. Camps, "Assessing LoRa for satellite-to-earth communications considering the impact of ionospheric scintillation," *IEEE Access*, vol. 8, pp. 165570–165582, 2020, doi: 10.1109/ACCESS.2020.3022433.

[141] I. Zacarias, L. P. Gaspary, A. Kohl, R. Q. A. Fernandes, J. M. Stocchero, and E. P. de Freitas, "Combining Software-Defined and Delay-Tolerant Approaches in Last-Mile Tactical Edge Networking," *IEEE Communications Magazine*, vol. 55, no. 10, pp. 22–29, 2017, doi: 10.1109/MCOM.2017.1700239.

[142] Z. Lu and H. Yang, *Unlocking the power of OPNET modeler*. Cambridge University Press, 2012.

[143] "El buque Hespérides interrumpe su trayecto hacia la Antártida por casos COVID-19," 2022. https://www.ciencia.gob.es/Noticias/2022/Febrero/El-buque-Hesperides-interrumpe-su-trayecto-hacia-la-Antartida-por-casos-COVID-19.html (accessed Oct. 10, 2022).

[144] Y. Meshcheryakov, A. Melman, O. Evsutin, V. Morozov, and Y. Koucheryavy, "On Performance of PBFT Blockchain Consensus Algorithm for IoT-Applications with Constrained Devices," *IEEE Access*, vol. 9, pp. 80559–80570, 2021, doi: 10.1109/ACCESS.2021.3085405.

[145] M. Zamani, M. Movahedi, and M. Raykova, "Rapid-Chain: Scaling Blockchain via Full Sharding," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018, pp. 931–948. doi: 10.1145/3243734.

[146] F. Holik, U. Roedig, and N. Race, "LoRa-SDN: Providing Wireless IoT Edge Network Functions via SDN," in *2020 43rd International Convention on Information, Communication and Electronic Technology, MIPRO 2020 - Proceedings*, 2020, pp. 1795–1800. doi: 10.23919/MIPRO48935.2020.9245378.

[147] B. Sipos, M. Demmer, J. Ott, and S. Perreault, "Delay-Tolerant Networking TCP Convergence-Layer Protocol Version 4," *RFC 9174*, 2022. https://www.rfc-editor.org/rfc/rfc9174.html (accessed Oct. 10, 2022).

[148] B. Sipos, "Delay-Tolerant Networking UDP Convergence Layer Protocol," *IETF Draft*, 2021. https://www.ietf.org/archive/id/draft-sipos-dtn-udpcl-01.html (accessed Oct. 10, 2022).

[149] J. Morgenroth, T. Pögel, R. Heitz, and L. Wolf, "Delay-tolerant networking in restricted networks," in *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*, 2011, pp. 53–55. doi: 10.1145/2030652.2030668.

[150] B. Rosado *et al.*, "Volcano-tectonic dynamics of Deception Island (Antarctica): 27 years of GPS observations (1991–2018)," *Journal of Volcanology and Geothermal Research*, vol. 381, pp. 57–82, 2019, doi: 10.1016/J.JVOLGEORES.2019.05.009.

[151] G. Shi, D. Dong, I. R. Petersen, and K. H. Johansson, "Reaching a Quantum Consensus: Master Equations That Generate Symmetrization and Synchronization," *IEEE Transactions on Automatic Control*, vol. 61, no. 2, pp. 374–387, 2016, doi: 10.1109/TAC.2015.2434034.

Appendix

PUBLISHED ARTICLES

The following pages present the articles encompassed in this dissertation as published or accepted. Article references are: [83], [85], [87].

TABLE XIII
LIST OF ARTICLES OF THE COMPENDIUM

Authors	**Title**	**Journal**	**Quartile**	**Reference**
A. Mallorquí, A. Zaballos	A Heterogeneous Layer-Based Trustworthiness Model for Long Backhaul NVIS Challenging Networks and an IoT Telemetry Service for Antarctica	Sensors (MDPI)	Q2	[83]
A. Mallorquí, A. Zaballos, A. Briones	DTN Trustworthiness for Permafrost Telemetry IoT Network	Remote Sensing (MDPI)	Q1	[85]
A. Mallorquí, A. Zaballos, D. Serra	A Delay Tolerant Network for Antarctica	IEEE Communications Magazine (IEEE)	Q1	[87]

sensors

MDPI

Article

A Heterogeneous Layer-Based Trustworthiness Model for Long Backhaul NVIS Challenging Networks and an IoT Telemetry Service for Antarctica

Adrià Mallorquí * and Agustín Zaballos

GRITS, Engineering Department, La Salle, Universitat Ramon Llull (URL), 08022 Barcelona, Spain; agustin.zaballos@salle.url.edu
* Correspondence: adria.mallorqui@salle.url.edu; Tel.: +34-932-902-436

Abstract: Antarctica is a key location for many research fields. The lack of telecommunication systems that interconnect remote base camps hardens the possibility of building synergies among different polar research studies. This paper defines a network architecture to deploy a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service. Long backhaul NVIS links were used to interconnect remote networks. This architecture presents some properties from challenging networks that require evaluating the viability of the solution. A heterogeneous layer-based model to measure and improve the trustworthiness of the service was defined and presented. The model was validated and the trustworthiness of the system was measured using the Riverbed Model simulator.

Keywords: trustworthiness; model; telemetry; IoT; NVIS; challenging network; Antarctica

check for updates

Citation: Mallorquí, A.; Zaballos, A. A Heterogeneous Layer-Based Trustworthiness Model for Long Backhaul NVIS Challenging Networks and an IoT Telemetry Service for Antarctica. *Sensors* **2021**, *21*, 3446. https://doi.org/10.3390/s21103446

Academic Editor: Lorenzo Vangelista

Received: 31 March 2021
Accepted: 13 May 2021
Published: 15 May 2021

1. Introduction

During the last half-century, Antarctica has been a key location for many research studies in several fields such as oceanography, bioscience, geoscience, physical sciences, and other environmental studies [1]. Although many bases have been settled in the peripheral areas of the Antarctic continent [2], the difficult environment and terrain provoke numerous challenges when it comes to implementing new operational services for modern studies. One of these challenges is the lack of telecommunication systems in Antarctica [3], especially wireless sensor networks (WSNs). Without WSNs, new research studies tend to use non-automatized ways of gathering data, which are more complex logistically, less scalable, and more error prone. Moreover, most Antarctic bases are not interconnected [3]. This fact lowers the possibilities for different research groups to collaborate on similar studies, and the advantages of providing synoptic region-wide observations and building synergies are lost [3].

The lack of conventional telecommunication services in Antarctica leverages the use of satellite communications or other systems such as high-frequency (HF) links to build a network of interconnected remote WSNs [4]. The first option is commonly discarded because of economic reasons, given the high costs of subscribing to this type of service. Furthermore, the degree of coverage offered by satellite constellations in Antarctic latitudes is not desirable [4]. To overcome these difficulties, the SHETLAND-NET [5] project aims to expand the use of communications in HF (3–30 MHz) by ionospheric reflection to the establishment of a low consumption communications system that allows the collection of sensor data distributed throughout the archipelago of the South Shetland Islands. This technology, called near vertical incidence skywave (NVIS), does not require direct vision and is totally independent of the satellite since the signal is transmitted upwards, allowing it to overcome any geographical feature [4,6,7]. The long backhaul NVIS link has a coverage range of up to 250 km, and its reliability is dependent on ionospheric conditions and solar activity. Researchers from our university have previously participated in research campaigns on Livingston Island, studying and verifying the NVIS communication system's viability [8–10]. A new campaign is planned to be carried out in the Antarctic field, with the goal to test the new improvements of the NVIS link [11] and deploy an IoT network for three different use cases: a telemetry service for light data (e.g., penguin tracking [12], a telemetry service for fat data (e.g., lichen observation [13]), and a distributed computing service to map the ionosphere along Antarctica).

However, a network deployed with long backhaul NVIS links may present some situations typical of challenging networks [14], such as intermittent connectivity, end-to-end disconnection, and variable error rates, making the implementation of the aforementioned services over a traditional TCP/IP architecture difficult. For the sake of the project, it is not feasible to wait until the Antarctic campaign starts to test the system in the field. Antarctic campaigns are usually very time restricted due to the meteorological conditions. Its remote location makes it challenging to overcome unanticipated difficulties that may arise (e.g., incorrect dimensioning of the needed equipment, poor performance of the proposed architecture). For this reason, it is necessary to study the viability and the expected trustworthiness of implementing this kind of network before its deployment in the field. The pre-deployment phase of the SHETLAND-NET project needs this previous research on the factors that affect the robustness of a communication network, which will help us build more reliable expectations for our proposed service's results and minimize the number of unexpected adversities (e.g., degraded service performance and reliability, loss of connectivity). In our case, this study was executed by simulating the conditions and the service that will be deployed in Antarctica.

This paper focuses on the use case of the telemetry service for light data. Many Antarctic studies could be helped by automating the data gathering of their research (e.g., geomagnetic studies [15], blowing snow monitoring [16], climate change [17], biological monitoring [12], or permafrost analysis [18]). Most of the data for these studies are currently gathered manually, and some zones might be challenging to reach, even with special vehicles such as snow motorbikes. For these reasons, the studies are focused on small areas of the Antarctic region. Thus, a WSN that provides a broader coverage area and the interconnection of remote areas could increase the results' relevance (e.g., more samples could be collected, broader synergies could be built). Moreover, the long backhaul links in charge of communicating remote WSNs could also be used to interconnect different Antarctic bases [4].

The paper has two main objectives. First, it was necessary to define which architecture, technologies, and protocols the telemetry service will use. As mentioned before, the drawbacks of challenging networks in addition to the extreme conditions sensors and other equipment need to work within is that it can provoke the service to reach low levels of performance and trustworthiness in the face of adversities. Thus, the paper's second objective was to propose and validate a model for visualizing, understanding, and measuring the trustworthiness of the overall service before its deployment in the field. With this model, the service's weaknesses could be detected, and countermeasures could be proposed to improve its trustworthiness and foresee their impact. We used the Riverbed Modeler simulator [19] to validate the model and measure the service's trustworthiness. To confirm the results, the tests were performed by modeling the permafrost use case of [18], where Ground Terrestrial Network-Permafrost (GTN-P) stations were used to measure 32 different parameters. These tests can be replicated to other concrete telemetry use cases by modeling them too.

The rest of the paper is organized as follows. In Sections 2 and 3, the background and related work are described, respectively. Section 4 defines the use case's service architecture. Section 5 presents the trustworthiness model. In Section 6, the performed simulations are described, and the extracted results are discussed in Section 7. Finally, the conclusions of the paper and future work are detailed in Section 8.

2. Background Work in IoT

This section presents mature IoT and WSN technologies that can help to define the network architecture of our telemetry use case for remote regions. In terms of network architecture for WSNs, it is necessary to differentiate between the access network and the backbone network. On the one hand, the access network provides connectivity to the IoT sensors in a variable coverage range, depending on the technology. On the other hand, the backbone is in charge of interconnecting the access networks to build a global WSN. The backbone network can use long backhaul links to reach remote areas and broader coverage than access network technologies.

2.1. IoT Access Network

The access network technologies for WSNs are commonly known as the IoT communication protocols [20] or IoT MAC layer protocols [21]. These protocols are commonly classified, depending on the size of the coverage area, as short-range coverage protocols and long-range protocols. Networks built on the latter kind of protocols are commonly known as low-power wide area networks (LPWANs). The authors of [20,21] classified the most used technologies in IoT. For short-range networks, the most common technologies are RFID, NFC, Bluetooth Low Energy (BLE), Zigbee, 6LoWPAN, and Z-Wave. For LPWANs, the most used communication protocols are narrow-band IoT (NB-IoT), long-term evolution-enhanced machine-type communication (LTE eMTC), Sigfox, and LoRa. To the best of our knowledge, in the specific case of Antarctica, the deployment of WSNs are scarce and limited to temporary testing deployments but not persistent. One example of short-range communications is the SNOWWEB project [22], where a network of weather stations was built using Zigbee transceivers. LPWANs seem to be more suitable options since the coverage area for deploying the WSN is more extended. For that reason, the authors of [23] studied the applicability of LoRa in Antarctic regions by characterizing its channel in the field, achieving a coverage area of up to a 30 km radius. Despite this, it seems feasible that some sensors of the WSN can be located out of range of the gateway due to the geographic conditions. In this case, there is the need to use mobile gateways and deploy mobile ad hoc networks (MANETs) [4,24–26].

2.2. IoT Backbone Network

On the other hand, the backbone network is in charge of interconnecting remote WSNs to build a single major network. For this purpose, LPWAN communications are not valid because the links that need to be established must have a broader range (several tenths of kilometers). Moreover, since the Antarctic region has many terrain variations, it is expected that two nodes separated by several kilometers do not have line of sight (LOS) [7]. Satellite communications are a solution to overcome these problems. However, geostationary Earth orbit (GEO) satellites do not cover Antarctica's latitudes adequately, and current low-altitude Earth orbit (LEO) satellites provide partial or no coverage in deep polar regions [27]. The authors of [27] studied the possibility of covering the whole Antarctic continent with a three-satellite constellation in elliptical orbits, but it has not been implemented. A significantly lower cost solution suitable for WSNs in remote areas is the use of HF communications. Specifically, the NVIS technique has already been tested in Antarctica [4,6,7]. Results show that this kind of long backhaul link can reach a throughput of up to 20 Kbps and a coverage radius of up to 250 Km without the need for LoS [28]. The main drawback of NVIS is the considerable variation of the transmitting channel's characteristics, the ionosphere, which can lead to periods of non-connectivity, becoming a challenging network [14]. Thus, it is necessary to test and measure NVIS networks' trustworthiness when used to transport data from actual use cases.

3. Related Work on Cyber Physical Systems' Trustworthiness

This section describes the related work by other authors to define and measure the trustworthiness of cyber physical systems (CPS). A CPS is defined as a system with integrated computational and physical capabilities. Common examples of CPSs include industrial control systems, automated vehicles and aircraft controls, wireless sensor networks, smart grids, and almost all devices typically encompassed by the Internet of Things [29,30]. The trustworthiness of CPS is defined in the literature, in general terms, as the property of behaving as expected under adversarial conditions [31]. However, these adversarial conditions can come from different reasons, e.g., faulty nodes, byzantine errors, malicious behaviors, and network malfunction [32]. For this reason, in the literature, there can be found many different approaches to measuring or providing trustworthiness that refer to disparate elements. We propose to classify them into the following four categories that will be used later to define our trustworthiness model:

1. Data Trustworthiness: It is defined as the possibility of ascertaining the correctness of the data provided by the source [33]. Many methods try to detect faulty nodes, false alarms, and sensor misreading using different approaches [32]. For instance, the authors of [34] presented a distributed Bayesian algorithm to detect faulty nodes, while the authors of [35] used a fog computing architecture to detect, filter, and correct abnormal sensed data. In addition, the authors of [36] presented a data intrusion detection system to trigger false data from malicious attacks;
2. Network Trustworthiness: It can be defined as the probability that a packet will reach its destination unaltered despite the adversities (e.g., link failure, link saturation, malicious attacks), and it is a crucial factor of low-power and lossy networks (LLNs) [37]. Improving network trustworthiness and performance is a challenge that has been addressed from different perspectives such as transmission coding [38–41], load balancing and redundancy protocols [42], transport protocols [43], dynamic routing and topology control protocols [44,45], cybersecurity mechanisms [46], and delay tolerant network (DTN) architectures and protocols [47]. In the case of routing, both proactive routing protocols (e.g., the IPv6 Routing Protocol for low-power and lossy networks (RPL) and optimized link state routing (OLSR)) and reactive routing protocols (e.g., ad hoc on-demand distance vector (AODV) and link-quality source routing (LQSR)) have been proposed in the literature to solve the drawbacks of LLNs and MANETs [44,45];
3. Social Trustworthiness: This trend has gained more attention since the irruption of the Social Internet of Things (SIoT) concept [48,49]. In SIoT trustworthiness, the capability of the objects to establish social relationships autonomously between them is leveraged to define more complex trust and reputation models that take into account several input parameters. The authors of [50] define a subjective model that considers factors as the computational capabilities of the nodes, the type of relationship between them, the total number of transactions, the credibility of a node, and the feedback provided by other nodes, among others. In [51], the authors evolved their previous model and based it on more parameters, such as the neighborhood of nodes, and presented a new objective model with a faster transitory response. The authors of [52] proposed another model that defines the input parameters as the expected gain on success, the expected damage on a failure, the expected cost, the expected result, and the goal. The authors of [53] define a decentralized, self-enforcing trust management system based on a feedback system and reputationally secure multiparty calculations to ensure the privacy of each party's provided data;
4. Consensus: This represents a state where all participants of the same distributed system agree on the same data values [54]. Consensus protocols can be divided into two general blocks: proof-based consensus and byzantine consensus. The first group is oriented to blockchain technology, where all participants compete with each other to mine a block, and the most used protocols are proof-of-work, proof-of-stake, and its variants [55–59]. The main drawback of these protocols for IoT is that most devices have simple hardware specifications and low processing power, being incapable of performing the mining tasks of blockchain [60]. The second major group of consensus protocols is the more classical byzantine based. These kinds of protocols

implement voting-based mechanisms to reach an agreement rather than competing among them, which generally results in less resource consumption. The drawback of these mechanisms is the number of messages that need to be delivered through the network to reach an agreement. The most well-known protocols in this category are Practical Byzantine Fault Tolerance, RAFT, PaXoS, and Ripple, although several variants have emerged year-by-year [55].

To the best of our knowledge, all the approaches that can be found in the literature focus on specific areas of trustworthiness, but none of them include all of the four trustworthiness topics. This fact can lead to misinterpreting the reasons for an inferior service's trustworthiness level, and wrong countermeasures to improve it could be applied if the interdependencies between different trustworthiness categories are not considered (as will be seen in Section 5.4). For this reason, we found the need to design our own model to measure a system's trustworthiness level that included the four categories mentioned above, which could help us anticipate and identify the possible weaknesses in our IoT telemetry system. Table 1 summarizes the characteristics of the analyzed trustworthiness approaches.

Table 1. Qualitative benchmark of the studied trustworthiness approaches.

Trustworthiness Use	**[34–36]**	**[38–41]**	**[42]**	**[43]**	**[44,45]**	**[47]**	**[50–53]**	**[54–60]**	**Own Model**
Data Trustworthiness	High	None	None	None	None	None	Medium	Medium	High
Network Trustworthiness	Low	Medium	High	Medium	High	High	Low	Low	High
Social Trustworthiness	None	None	None	None	None	None	High	None	High
Consensus	None	None	None	None	None	None	None	High	High
Metrics used	Faulty Sensed Data	Bit Error Rate	Packet Delivery Ratio (PDR)	PDR, Throughput, Delay	PDR, Delay	PDR, Delay	Successful Transactions	Successful Transaction, Byzantine Node Tolerance, Throughput	Faulty Sensing Ratio, PDR, Successful Transaction Rate, Byzantine Node Tolerance

4. Network and Service Architecture

Prior to applying the model of trustworthiness, our first goal was to define the architecture of the telemetry use case that was to be deployed in the Antarctic campaign of the SHETLAND-NET project. As mentioned before, the concrete case was the improvement of permafrost studies by automating data gathering from the GTN-P stations (the sensors), which measure 32 different parameters. Currently, data are gathered only once a day, and the authors from [18] left the complete automation of the GTN-P stations as an open challenge, given that their approach suffers from a lack of connectivity. It is important to remark that the architecture described below applies to any telemetry use case. However, we will use the example of the GTN-P stations' permafrost research that will be carried out during our campaign in the field.

We propose to use the architecture defined for the deployment phase SHETLAND-NET project [5]. In our approach [28], we aim to interconnect all remote sensors to a control center, building a heterogeneous global wireless sensor network (GWSN) composed of several wide wireless sensor networks (WWSN), able to gather data more frequently. The first approach to designing a remote sensing system for the Antarctic region was described in [4] during the SHETLAND-NET project's early stages, describing how sensors could reach and use NVIS as a long backhaul link. However, it was mostly centered on designing the characteristics of the physical layer of the NVIS (backbone) network. In this paper, a more detailed description of the overall network architecture is presented. The network diagram is detailed in Figure 1.

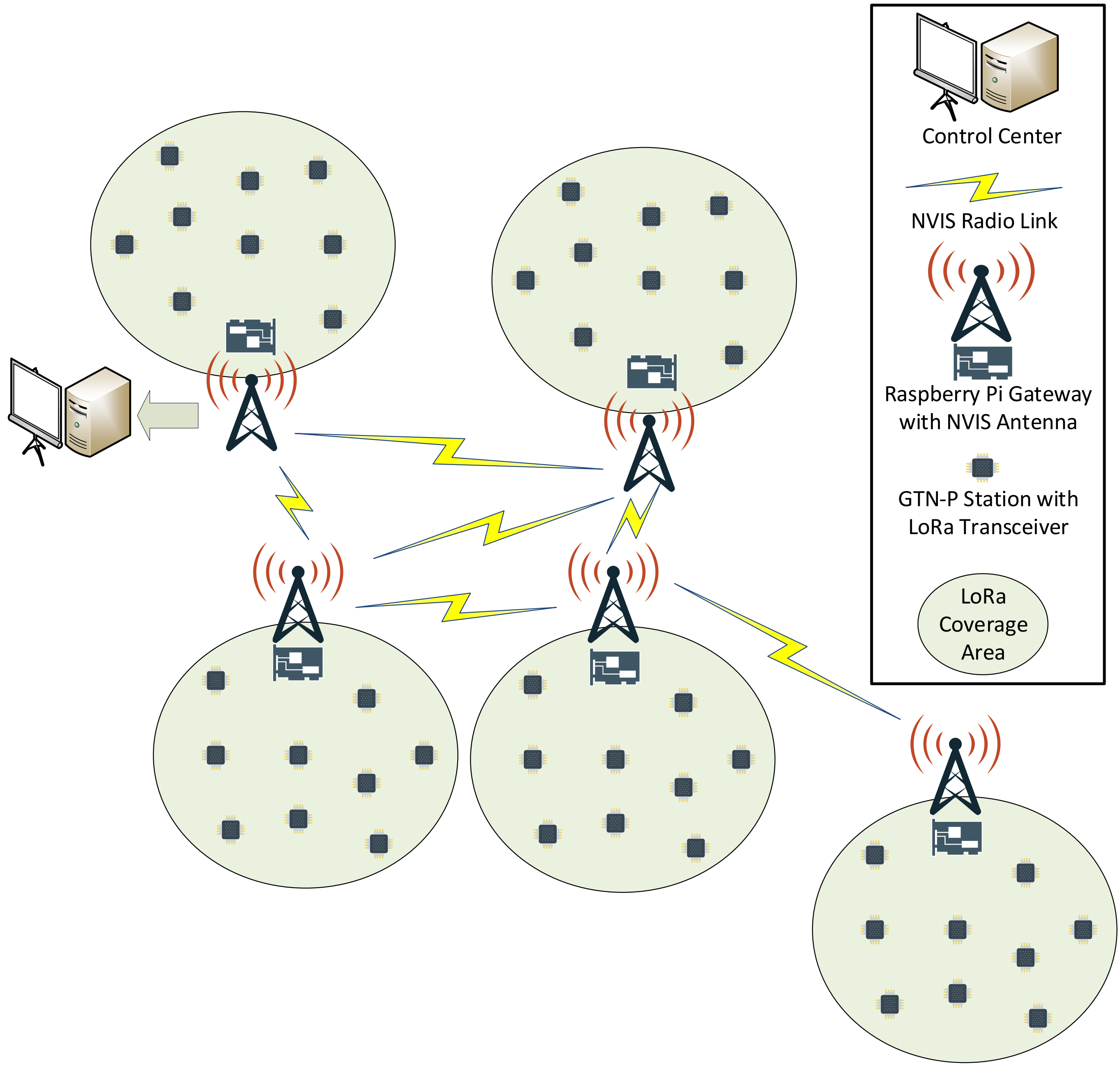

Control Center
NVIS Radio Link
Raspberry Pi Gateway with NVIS Antenna
GTN-P Station with LoRa Transceiver
LoRa Coverage Area

Figure 1. Network diagram of the SHETLAND-NET project telemetry service.

The access network (WWSN) will be in charge of providing connectivity to the remote sensors, transporting the gathered data from the sensors (GTN-P stations connected to a low-consumption board) to the gateways (e.g., a Raspberry Pi). The main gateway of each WWSN will be located near the research base, with the GTN-P stations located around it in a few-kilometer radius. For redundancy reasons, groups of GTN-P stations can be clustered and placed close enough to interpret that they measure the same permafrost values. These stations will sense the data and send it to the gateway once per hour. For this use case in Antarctica, it is logical to think that the wider the area that can be covered by the access communication technology the better, because more sensors will be able to be placed far from the research base so researchers will have access to sensors placed farther away while saving valuable time in collecting the data. For this reason, short-range communications are less suitable, and LPWAN communications are preferred. The lack of telecommunication operators providing service in Antarctica forces operator-dependent communication services, such as Sigfox, NB-IoT, or LTE eMTC, to be discarded. This leaves LoRa as the main candidate to deploy the access network. LoRa transceivers will be placed in each GTN-P station, responsible for sending the measured data to the LoRa gateway. As explained in [4], this gateway was implemented with a Raspberry Pi 3B+ in previous Antarctic campaigns of the SHETLAND-NET project, and it is responsible for storing the gathered data from all the sensors it is giving service to, ready to send all these data through the backbone network.

The backbone network will be composed of all NVIS nodes, which will interconnect remote WWSNs through the long backhaul links to form the GWSN. Each NVIS node mainly consists of a Red Pitaya, a Raspberry Pi 3B+, and an HF antenna [7]. The link that can be established between two NVIS nodes has a range of up to 250 km. In order to interconnect all the WWSNs and reach all remote areas, a multi-hop network will need to be deployed. Thus, some of the NVIS nodes will have to act as repeaters. At least one NVIS node will need to be connected to the control center in order to send all the data to it. To avoid a single point of failure (SPF), having more than one NVIS node connected to the control center is recommended. The possibility of having multiple paths to reach one destination demands the need for a routing protocol able to find the best possible loop-free path in the network [28].

The operation of the backbone network can be summarized as follows. Each NVIS node will act as a concentrator, gathering the data from every GTN-P station inside their LoRa coverage area. Once all possible data are collected, the NVIS node will forward it to the node connected to the control center, following the path determined by the routing protocol through the backbone network.

However, we can find three main issues that can provoke this architecture to become a challenging network [14]:

1. Due to the fact of Antarctica's extreme weather and environmental conditions, both sensors and gateways could experience temporary or persistent malfunctioning;
2. The irregular elevations of the Antarctic terrain might create situations where sensors do not have a LoS path through the gateway. This fact degrades the performance of LoRa communications considerably [23];
3. Depending on the ionosphere's state and the solar activity, NVIS links may become unavailable temporally or intermittently [4,6,7,11].

For this reason, our primary goal was to establish a model to measure the trustworthiness of a CPS, with which the performance of the proposed architecture can be evaluated and its weaknesses detected and improved. Our model will be used in the pre-deployment phase of the SHETLAND-NET project to foresee performance difficulties of the defined architecture that may arise during its deployment in the field and predict the effect of the proposed countermeasures.

5. Trustworthiness Model Definition

Our model proposal to measure the trustworthiness and evaluate a CPS's performance (in our case, a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service) is a layer-based model. This model is characterized by two base layers (Data Trustworthiness Layer and Network Trustworthiness Layer), two extension layers (Social Trustworthiness Layer and Consensus Layer), and the interaction among all of them. The Data Trustworthiness, Network Trustworthiness, Social Trustworthiness, and Consensus Layers can collectively define a system's trustworthiness. A graphic representation of our layered model is shown in Figure 2.

We postulated that each layer is characterized by its definition (scope), how the trustworthiness of that layer is measured (metric), and how the value of this metric can be improved (correction).

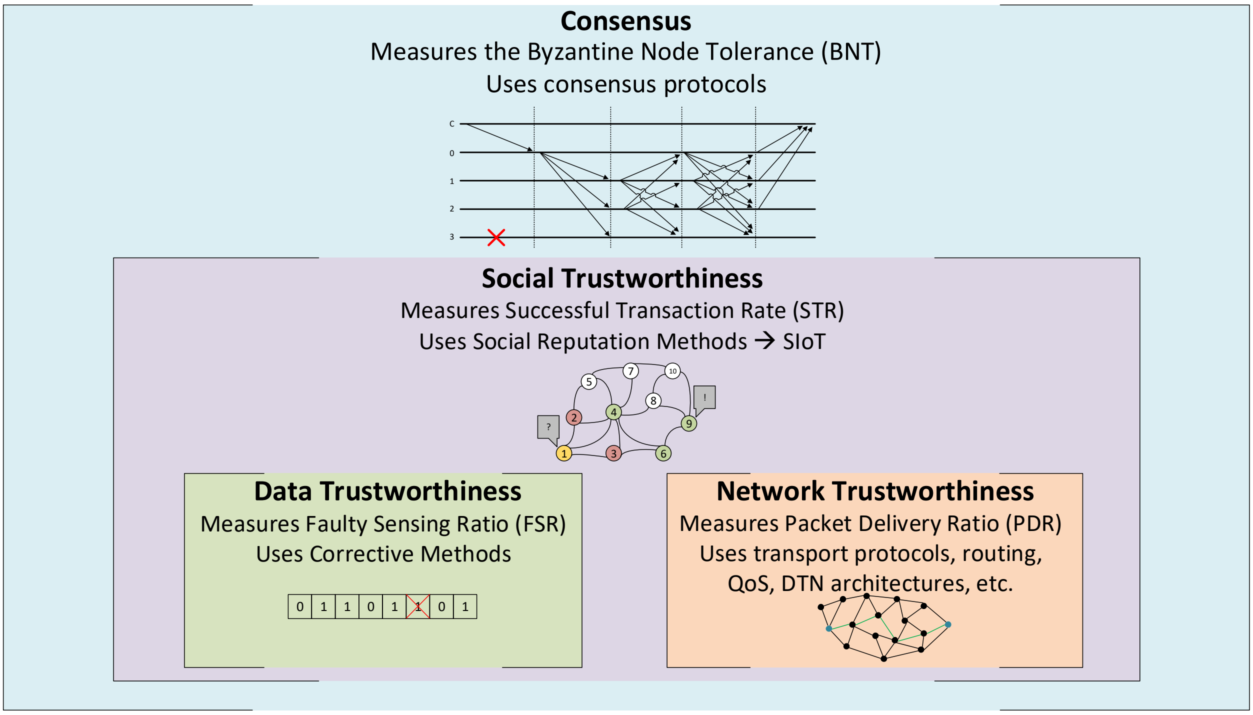

Consensus
Measures the Byzantine Node Tolerance (BNT)
Uses consensus protocols
C
0
1
2
3
Social Trustworthiness
Measures Successful Transaction Rate (STR)
Uses Social Reputation Methods → SIoT
?
!
1
2
3
4
5
6
7
8
9
10
Data Trustworthiness
Measures Faulty Sensing Ratio (FSR)
Uses Corrective Methods
0 1 1 0 1 1 0 1
Network Trustworthiness
Measures Packet Delivery Ratio (PDR)
Uses transport protocols, routing,
QoS, DTN architectures, etc.

Figure 2. Layers of the proposed trustworthiness model.

On the one hand, Data and Network Trustworthiness are the base layers of our model, because the system that we want to measure is meaningless if we do not have data to be exchanged between nodes through a network. On the other hand, Social Trustworthiness and Consensus are the extension layers because they include functionalities that are not needed in the service architecture but are optional to implement.

5.1. Trustworthiness Layers' Definitions

We propose the following definitions for each layer, based on the classification of trustworthiness approaches we defined in Section 3:

1. Data Trustworthiness Layer: Is the layer responsible for ascertaining the correctness of the data provided by the source;
2. Network Trustworthiness Layer: Is the layer responsible for assuring that a packet reaches its destination on time and unaltered despite the adversities (e.g., link failure, link saturation, or malicious attacks);
3. Social Trustworthiness Layer: Is the layer responsible for leveraging the capability of the objects to establish social relationships autonomously between them to improve the trust between them and the correctness of gathered data;
4. Consensus Layer: Is the layer responsible for reaching a state where all participants of a group agree on the same response or result.

5.2. Trustworthiness Layers' Metrics

Managing the trustworthiness of a system is possible when the different layers are separately understood. This way, objectives and metrics can be defined to measure the level of trustworthiness. In order to measure the four layers of trustworthiness, we have defined a quantitative metric for each layer. Once metrics are defined, a trustworthiness target can be determined, which is a quantitative objective given to a trustworthiness metric. If a trustworthiness characteristic does not meet its target, a change factor is needed to revert the situation. The combination of all change factors defined to meet the trustworthiness targets is called the trustworthiness implementation.

We propose that the trustworthiness model will use the normalized metrics defined in Table 2 to quantify and measure trustworthiness. We selected these well-known metrics as they are also used to measure the impact of the technologies and approaches described in Section 3. We normalized all of them for better cohesion with our layer-based approach.

Table 2. Trustworthiness metrics.

Layer	Metric	Range
Data	Faulty Sensing Ratio [35,36]	[0, 1]
Network	Packet Delivery Ratio [44,45]	[0, 1]
Social	Successful Transaction Rate [50,51]	[0, 1]
Consensus	Byzantine Node Tolerance [55,56]	[0, 1]

The faulty sensing ratio (FSR) is defined as the proportion of false sensed values (FSV) by all nodes and total sensed values (TSV) in a defined time period as stated in Equation (1).

$$\text{FSR} = \frac{FSV}{TSV} \ . \tag{1}$$

We considered that a sensed value is every independent and semantically significant measured data that a sensor stores in its memory (e.g., RAM, Flash, hard drive). Suppose no corrective methods are used in the system. In that case, sensed data (e.g., temperature, humidity, position, ice content) are considered to be faultily sensed if the value stored in the sensing (source) node's memory is different from the value that the sensor should have correctly read (within a tolerance percentage). In real implementations, the number of FSV can only be measured if the sensed data's value is known a priori (ground truth) [61]. Otherwise, only in simulations it is possible to quantify the number of FSV. FSV and TSV are parameters that must be gathered within the same time slot to calculate the ratio correctly. The lower the FSR, the better the data trustworthiness.

The packet delivery ratio (PDR) is calculated as the quotient between the total number of packets received (Pr) by all nodes and the total number of packets sent (Ps) by all nodes in the same time slot as stated in Equation (2). A packet is considered received if and only if the reception time (T_{rx}) is less or equal to the transmission time (T_{tx}) plus a defined threshold offset η ($\text{T}_{\text{rx}} \leq \text{T}_{\text{tx}} + \eta$), and the packet content is not altered. The higher the PDR is, the better the network trustworthiness. In our proposal, retransmitted packets (if any) and original packets are counted separately to compute the metric value.

$$\text{PDR} = \frac{Pr}{Ps} \ . \tag{2}$$

The successful transaction rate (STR) is the proportion between the number of successful transactions (STs) and the total number of transactions (TTs) in a defined time slot as stated in Equation (3). We defined a transaction, l, as a sensed value, v, that a node, j, expects to receive from a node or group of nodes, i. Retransmitted or duplicated packets for the same value, v, are considered part of a single transaction, l. A transaction, l, is considered successful when a node, j, expects to get some information or data (v) from node i before a defined maximum reception time (Trx_{max}) and receives it as expected, thus providing good feedback (${f_{ij}}^{l}$ = 1) for that transaction to node i. ST and TT are parameters that must be gathered within the same time slot to calculate the ratio correctly. The higher the STR, the better the social trustworthiness.

$$\text{STR} = \frac{ST}{TT} \ . \tag{3}$$

The byzantine node tolerance (BNT) is defined as the proportion of the supported byzantine nodes (Nb) that can participate in the consensus system without affecting the correctness of the general agreement and the total number of nodes (Nt) that participate in the consensus system as stated in Equation (3). A node is considered to be byzantine if it experiences a crash or soft fault that incapacitates it to behave as expected, or if it does not behave as expected on purpose (malicious node). The higher the BNT, the higher the probability to reach a correct general agreement. Although theoretically, the BNT value range is between 0 and 1, in practice, it is not possible to reach a correct consensus with a BNT ≥ 0.5.

$$\mathrm{BNT} = \frac{Nb}{Nt}\,. \tag{4}$$

5.3. Trustworthiness Improvement Examples

Now that we have defined the four trustworthiness layers and their associated metrics, we can give some examples of techniques and protocols that can be used as countermeasures at each layer to improve the metrics' values.

5.3.1. Data Trustworthiness Countermeasures

At the Data Trustworthiness Layer, corrective methods can be applied that try to detect abnormal data (false sensed values) stored in the source node due to the fact of a sensor malfunctioning, a misreading of the sensed data, or erratic writing to the node's memory. Corrective methods can be used to detect and correct these abnormal values by comparing them to the values sensed by the same node previously and other mechanisms such as hashes, checksums, and parity bits. If these corrections are performed at the post-processing stage by the receiving server or gateway, the nodes' malicious data manipulation can also be detected. However, our model assumes that corrections are only made by the own node (source node). Otherwise, errors that originated during the data transport through the network, which are out of our scope of definition for the Data Trustworthiness Layer, could be misinterpreted as source node errors. The drawback of this assumption is that only non-malicious errors are likely to be corrected at this layer because malicious nodes might not correct data on purpose. Our model specifies that other layers of the model are responsible for mitigating malicious behaviors (e.g., the Network Trustworthiness Layer).

The method presented in [35] is a suitable example of a corrective method for data trustworthiness. This value-level corrective method defines thresholds to detect potential abnormal data (e.g., a lower-value limit $\mathrm{t_{low}}$, an upper-value limit $\mathrm{t_{up}}$, and an abrupt change threshold $\mathrm{t_{ch}}$). When a potential abnormal value is detected, it is compared with the values sensed from the node's neighbors, computing the group value similarity (G). Since this breaks our model's assumption, this value similarity could be computed with the historical values from the sensor itself as in [36]. If the similarity is lower than a threshold $\mathrm{t_{sim}}$, then the abnormal data are confirmed and corrected (e.g., interpolation with previous and posterior correct values sensed by the own node). This method might experience false positives (by detecting a correct value as abnormal and modifying it) and false negatives (by not detecting an abnormal value), which can be grouped into faulty sensed values (FSV). If the thresholds are too strict, the number of false positives will increase, while the number of false negatives increases if the thresholds are too lax. The fewer the number of FSV, the better the data trustworthiness, so an optimal trade-off value for the thresholds must be found to minimize the overall number of FSV. This number is easy to gather in simulation scenarios, but in real implementations, it is only possible if the values are well-known a priori (ground truth values).

5.3.2. Network Trustworthiness Countermeasures

At the Network Trustworthiness Layer, transmission coding techniques such as FEC convolutional codes [38], LDPC codes [39,40], and polar codes [41] are used to increase the robustness of the transmitted signal. Routing protocols and quality of service (QoS) mechanisms are used to find the best path from a source to a destination by quantifying the quality or performance of each link in the network. For each destination, more than one path can be determined as feasible thus providing load balancing. Many metrics exist to calculate the best path such as the number of hops, the bandwidth of the link, the delay, and the expected number of retransmissions. These routing protocols can be classified under different categories such as proactive/reactive, link-state/distance-vector, or monometric/multimetric [45]. Selecting the best path for a traffic flow will eventually improve network statistics such as throughput, delay, jitter, or packet delivery ratio (PDR). In the case of challenging networks, DTN overlay architectures and protocols, such as the Bundle Protocol [62], is also a solution that can be used to improve the network trustworthiness.

Another relevant element to take into account in this layer is the data security through the network. While traveling from the source to the destination, data should remain private, available, and unaltered, preventing it from cyberattacks [63]. For this purpose, network elements such as next-generation firewalls or intrusion detection systems and security mechanisms, such as data encryption, authentication, anti-spoofing techniques, and network filters, are used in the network.

5.3.3. Social Trustworthiness Countermeasures

At the Social Trustworthiness Layer, most solutions tend to use reputational mechanisms to determine which nodes to trust when exchanging information. This reputation is commonly based on previous transactions' feedback to build an opinion for the node's trustworthiness [64]. More complex and robust mechanisms also incorporate parameters such as the indirect opinion of other nodes, the relevance (weight) of each transaction, the node's centrality, the node's computational capacity, and the type of relationships between the nodes [50].

The model in [51] provides two different ways for computing the reputation of a node. On the one hand, a subjective model of social trustworthiness is presented to compute the reputation of node *i* under the perspective of every other node (R_{ij}), these reputations being different from each other, because the experience of interaction with node *i* for two different nodes can be different. Moreover, reputations are asymmetric, meaning that the reputation that node *j* calculates from node *i* can be different from the reputation that node *i* calculates for node *j* ($R_{ij} \neq R_{ji}$). Thus, the system's overall trustworthiness can be represented as an N × N matrix for the reputation that each node calculates for all the other nodes, where N is the total number of nodes. On the other hand, objective models calculate one single reputation for each node (R_i), representing the trustworthiness that the system as a whole perceives from node *i*. This reputation takes into account the opinion and the feedback from all the other nodes. Thus, the system's overall trustworthiness is represented as an N-size vector with the reputation that the whole network perceives for each node.

Both the subjective and objective approaches aim to leverage the transactions between trustful nodes and isolate those with bad reputations, which are considered more faulty or malicious prone. Thus, their goal is to maximize the number of successful transactions (STs).

5.3.4. Consensus Countermeasures

At the Consensus Layer, several mechanisms can be used to reach a decentralized general agreement (GA) that all nodes in the group consider to be true. Theoretically, if the number of byzantine nodes is more than 50% of the total number of participating nodes, every consensus mechanism will fail to reach a benevolent agreement. Consensus mechanisms aim to reach the GA while tolerating a percentage of byzantine nodes. Consensus protocols are generally classified into competing mechanisms (proof-based) and voting-based mechanisms. The latter are more suitable for IoT devices because they consume fewer resources from the node. These protocols commonly consist of various voting phases to reach the GA, and their goal is to maximize the number of tolerated byzantine nodes (BNs). A drawback of these mechanisms is that they need participating nodes to exchange a large number of messages between them to reach a consensus which can be a problem in low-bandwidth networks, consuming most of this bandwidth. Some protocols look for a trade-off between the number of tolerated BNs, throughput, and scalability.

5.4. Trustworthiness Layers' Dependencies

Trustworthiness layers' dependencies must also be understood before deploying the system's architecture. In this way, we can build more accurate expectations of how the model's overall trustworthiness and concrete layers will be affected by applying a trustworthiness countermeasure in one layer. If the impact of applying specific countermeasures could not be foreseen, their implementation in the field could negatively affect the overall system's trustworthiness. For instance, how will the data trustworthiness affect the consensus? Can a robust consensus protocol lower the trustworthiness of the network because it is causing bottleneck congestion? In the SHETLAND-NET project, the trade-offs between these layers need to be carefully analyzed before deploying the system in the field to obtain the optimal overall trustworthiness level. If we were not considering these dependencies, it could be possible to experience a degraded performance of the deployed architecture without the necessary resources or response time to correct it during the campaign. Our model dependencies proposal is exhibited in Figure 3. These dependencies are qualitatively analyzed below, and the simulation tests performed in Section 6 were necessary to validate the model and quantify their actual impact on the overall system's trustworthiness.

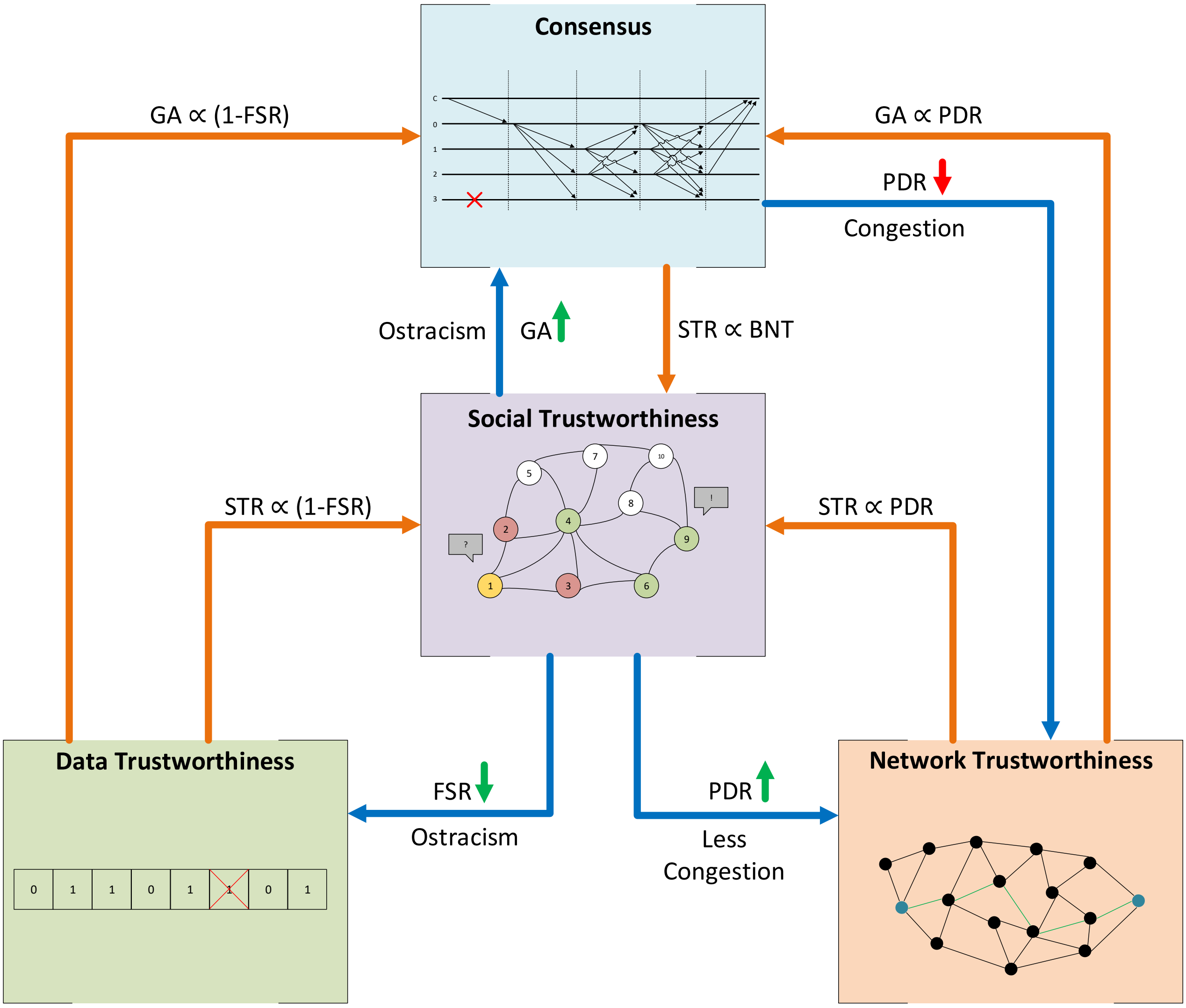


Figure 3. Dependency diagram between trustworthiness layers.

The Consensus Layer is directly affected by the other three layers. If FSV (Data Layer) is closer to 0, it means that nodes tend to measure the sensed values correctly, so they will be more prone to reach a correct general agreement. From the Social Layer, it is possible to ostracize those nodes with a lower reputation (which should be the ones with more false sensed values) if the application can afford to lose the data from them. In this case, if nodes with the worst reputation were omitted, it should be more probable to reach a correct general agreement for the rest of the nodes. Finally, suppose the PDR (network trustworthiness) is closer to 1. In that case, it means that the whole network delivers the most packets unaltered and on time, so fewer nodes will be considered byzantine due to the network issues and reaching correct general agreements will be more feasible. It is important to notice that all these dependencies do not affect the Consensus Layer metric, the byzantine node tolerance, which depends only on the consensus algorithm used and the total number of nodes participating in the consensus group.

We propose that the Social Layer can also be directly affected by the other layers. On the one hand, FSV and STR are inversely related. If the FSV is close to 0, a transaction coming from that node is less probable to have a false sensed value, meaning that it will become a successful transaction if the network delivers it properly to the destination. In addition, the source node will obtain good feedback from the receiving node, increasing its reputation. On the other hand, PDR and STR are directly related. As the PDR decreases, it is more feasible that packets targeted to a node are lost in the network, decreasing the STR. Thus, the receiver would evaluate the transaction as a failure, providing bad feedback and decreasing the sender's reputation. Finally, if the Consensus Layer is implemented, the negative effect of some false sensed values from byzantine nodes and lost packets can be masked thanks to the consensus algorithm. Nodes could still reach a correct general agreement, marking that transaction as successful and increasing the STR.

The network layer can be directly affected by the Social and Consensus Layers in terms of congestion [65,66]. Depending on the application, if nodes with the lowest reputation could be ostracized, their sensed data might not be sent through the network because they might not be requested. Thus, these nodes' links might be less congested and less prone to packet drops, increasing the PDR. Adversely, as mentioned before, using a consensus mechanism introduces a considerable amount of network traffic. In addition, the number of messages exchanged between a group of nodes is directly proportional to the number of nodes in the group. Thus, if the network bandwidth was not enough to support this extra traffic, the network could be more prone to be congested and drop packets, decreasing the PDR.

Finally, it is intuitive to think that the Data Layer should not be affected by the other layers. The variability of the FSV should depend on the error probability of the sensors and the node itself (e.g., equipment quality, battery degradation), which could also be affected by external factors (e.g., environmental characteristics). However, we propose that the Data Layer can be affected by the Social Layer. Suppose the Social Layer is implemented and is being used to ostracize the lowest reputation nodes. In that case, we considered that sensed values from omitted nodes must not be counted for the FSR computation. Thus, if the lowest reputation nodes were the ones with more false sensed values, the overall FSR should increase.

It is important to see that Data and Network Layers (the base layers, which are always present) are entirely independent, given that the correctness of data is always measured on the source node, never on the destination. This way, data loss or alteration caused by the network does not affect the data correctness measure.

Notice that Social and Consensus Layers (the extension layers, which are optional) are the ones affected by the rest of the layers. However, the way they are affected is different. On the one hand, the dependencies from other layers to the Social Layer directly affect the value of its trustworthiness metric, the STR. On the other hand, the Consensus Layer metric, the BNT, is not affected by other layers, but these dependencies can improve the probability of reaching a correct general agreement, which in final terms improves the Social Layer metric, the STR.

In that sense, we considered that the system's overall trustworthiness can be measured with the STR metric, which is the one affected by the four layers of our model, and intrinsically incorporates the effects of the other three metric values (i.e., FSR, PDR, and BNT). Moreover, notice that without implementing the extension layers, the STR can still be computed, which will combine the effects of the base layers (i.e., Data and Network Trustworthiness).

Although the dependencies between the layers and metrics of our model have been identified, it is still challenging to quantify the effect of looped dependencies on the system's trustworthiness. We identified two actions that can provoke a direct looped dependency. First, if Social Trustworthiness Layer is used to ostracize the lowest reputation nodes, their sensed values will be omitted, decreasing the FSR and eventually increasing the STR. However, suppose more traffic than supported by the network is concentrated on the links that lead to most reputation nodes. In that case, it is possible to create network congestion that will decrease the PDR and eventually decrease the STR. Second, implementing a consensus mechanism might help tolerate byzantine nodes and faulty network links, which eventually increases the STR. Nonetheless, suppose the network bandwidth is not large enough to allocate the extra traffic introduced by the consensus mechanism. In that case, the network may suffer from congestion, decreasing its PDR and eventually decreasing the STR.

To quantify the effects and trade-off points between these dependencies, it is essential to test the model's applicability with a use case and measure the trustworthiness metrics under different circumstances and several times. Given the complexity and cost of performing such a number of tests in the field during the Antarctic campaign, we opted to use simulation tests, which provides more flexibility. These pre-deployment simulations will help us decide which are the most suitable and trustworthy architectures for our system and anticipate the possible weaknesses and problems that may arise during the deployment in the field.

6. Simulation Tests

To validate the trustworthiness model, it was necessary to measure the metrics values for the use case scenario several times under different circumstances. For this purpose, the use case scenario was represented and evaluated in the Riverbed Modeler Simulator [19]. As stated before, our use case scenario was a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service. Concretely, the telemetry service will be used to automatize the data gathering of GTN-P stations to study the permafrost of the Antarctic region. The remote sensors of WSNs will be connected to a concentrator gateway through LoRa (access network), and these gateways will be interconnected between them and a control center through long backhaul NVIS links (backbone network). The extreme conditions GTN-P stations need to work with, added to the challenges of NVIS links and a LoRa network without LoS, might degrade the overall system's trustworthiness. In order to foresee which problems may occur during the Antarctic campaign and build more accurate expectations of the system's performance and outcomes, we applied our proposed trustworthiness model to measure and evaluate them.

The first step was the modeling of the network, the nodes, and the application. Once the model is designed and implemented in the simulator, the set of tests and the simulation parameters must be defined. After that, the simulations were run, and results were collected and evaluated.

6.1. Network Models

For the use case scenario, the backbone network (NVIS) and the access network (LoRa) were modeled separately. On the one hand, the NVIS channel was modeled following the characteristics described in [7]. The transmission frequency was 4.3 MHz with a channel bandwidth of 2.3 kHz and a bit rate of 4.6 kbps. An FEC convolutional coding with a $\frac{1}{2}$

rate code and interleaving were used to increase the reliability of the transmission. The range of the HF link was up to 250 km.

Moreover, given that the ionosphere characteristics vary considerably throughout a day, we also modeled the probability of a packet being correctly delivered through an NVIS link hour by hour, following the results in [11]. These results showed that the NVIS links are unlikely to be available from 17:00 until 6:00. In contrast, the channel availability from 6:00 to 17:00 varied from 70% to 100% when both the ordinary and extraordinary waves received were combined as shown in Figure 4.

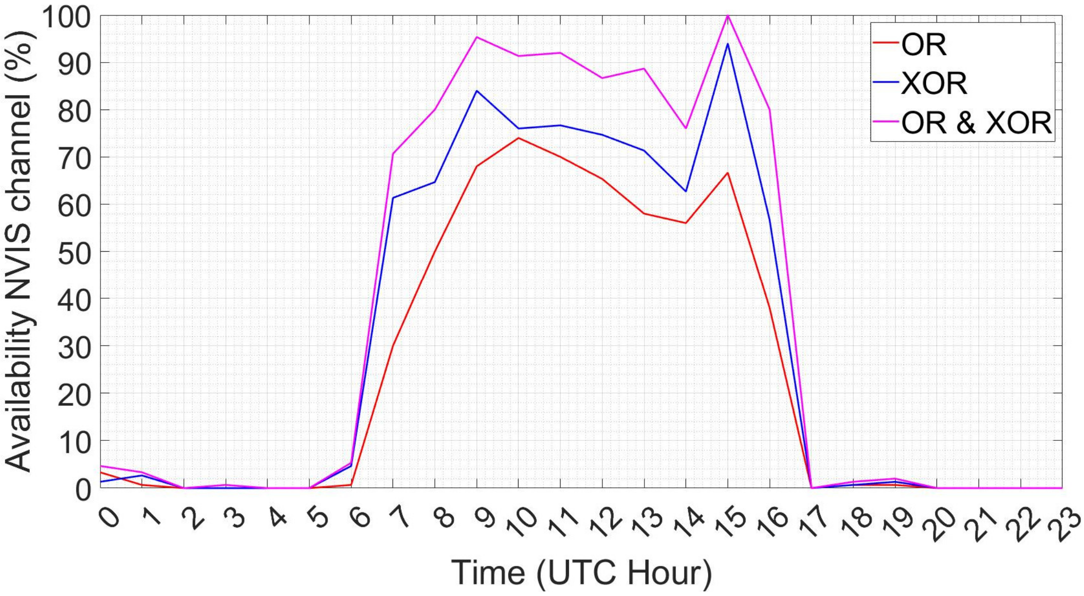


Figure 4. NVIS link availability depending on solar activity and the ionosphere's state [11]. The graph's legend is defined as follows: OR refers to the performance of the ordinary wave, XOR refers to the performance of the extraordinary wave received. OR and XOR refer to the total performance between both the ordinary and extraordinary modes. We have the copyright of [11], it belongs to the GRITS research group from La Salle.

On the other hand, the LoRa channel was modeled based on the results shown in [23,67]. The transmission frequency was 868 MHz with a channel bandwidth that varied depending on the chosen data rate (DR) and the rate spreading factor (SF). In our case, we chose CR3 (4/7) and SF7, resulting in a channel bandwidth of 125 kHz and a bit rate of 5.47 kbps. The range of the link was up to 30 km. In the line of sight (LoS) case, the channel was always available with a packet loss of 0%. Otherwise, with no LoS, the packet loss varied from 0% to 98% depending on the signal reflections, with an average value of 72%. Due to the Antarctic surface's irregularities, we cannot assume that the GTN-P stations will be located in LoS with the gateway. For this reason, we considered that 25% of the sensors will have a LoS to the LoRa gateway, while the remaining 75% will not have LoS. Table 3 summarizes the characteristics of our network models.

Table 3. Network parameters used to model the scenario.

Parameter	NVIS	LoRa
Transmission Band	4.3 MHz	868 MHz
Channel Bandwidth	2.3 kHz	125 kHz
Channel Bitrate	4.6 kbps	5.47 kbps
Coverage Range	Up to 250 km	Up to 30 km
Daytime Availability (6 a.m.–5 p.m.)	70–100%	100% (LoS), 2–100% (No LoS)
Night Availability (5 p.m.–6 a.m.)	0%	100% (LoS), 2–100% (No LoS)
Maximum Payload Size	242 bytes	140 bytes

6.2. Node Model

In the case of the node, both the GTN-P station and the gateway will use the same finite state machine model. The "INIT" state initializes the model and its attributes. The "IDLE" state is used when the node is waiting for a packet to arrive, transitioning to the "PROCESS" state, or a self-interruption to send the sensed data values, transitioning to the "SEND" state.

6.3. Application Model

Pseudocode algorithms for the application modeling are shown in Appendix A, Algorithms A1, A2, and A3. The application consists of the telemetry service to gather data from measured values by sensors and send them to the control center. To better understand the below explanations, we encourage revisiting Figure 1 to recall the proposed network architecture.

Each measured value, v, is considered a transaction, l, that must reach the control center. The application can be run without implementing any of the extension layers of the proposed trustworthiness model (standard mode) or can implement the Social or Consensus Layers of the model (redundancy mode), inclusively or exclusively. In standard mode, each value, v, is measured by a single GTN-P station, while in redundancy mode, the implementation of a reputational or consensus mechanism leverages the creation of clusters (groups of GTN-P stations) that measure the same value v.

The GTN-P stations will send data packets once per hour, simulating the moment when the 32 values are gathered from the GTN-P station sensors, stored in memory, and delivered to the gateway. We decided to sense these values hourly because it is the same sensing frequency that the members of the PERMASNOW project [18] used when they performed their automatization tests. In this process, if no consensus mechanism is performed, a hardcoded value, v, for each parameter will be inserted into a 132 byte payload (32 values and a timestamp, 4 bytes each). With a probability, Pb, the value, v, will be modified to another value out of an acceptable range (v_{min}, v_{max}), and the total number of FSV will be increased by one. This payload will then be inserted into the packet to be sent to the gateway. If an ACK packet is not received from the gateway before a timeout T_{out}, the data packet will be retransmitted up to a maximum of three times. In the case of implementing a consensus mechanism, all the GTN-P stations participating in the cluster (which are measuring the same v) will start the process to reach a general agreement. Once they have reached it, only the cluster leader will send the payload to the gateway with the agreed value v. During the consensus process, if the Social Layer is also implemented, each packet exchanged between the nodes participating in the consensus group will be used to compute the reputation Ri of nodes. The node with the highest reputation will be elected as the group leader. Moreover, a node i with a reputation R_i lower than R_{min} will not have the right to vote for the value election. However, it will be allowed to continue participating in the consensus group to increase its reputation until it can be granted the right to vote again. On the contrary, if no reputational mechanism is being used, all group members will always have the right to vote, and the leader will be chosen randomly.

On the other site, gateways will collect the data from the GTN-P stations inside its LoRa coverage area and then forward it through the NVIS backbone network until it reaches the control center. Given that gateways are also nodes, they may experience a byzantine failure with probability Pb. In that case, the gateway will modify the payload's content. In standard mode, each value v received from node i must be forwarded to the control center. In redundancy mode, if no consensus mechanism is being used (only the Social Layer is implemented), the gateway will receive several candidates for the value v from every node in the cluster. The gateway will inspect the values from it and check if they are in the acceptable range (v_{min}, v_{max}). In an affirmative case, gateway j will provide positive feedback for that transaction l from node i ($f_{ij}{}^{l}$ = 1). Otherwise, the feedback will be negative ($f_{ij}{}^{l}$ = 0). After providing feedback for every transaction, the reputation Ri of the nodes will be updated, and the value provided from the node with the greatest reputation will be chosen as the definitive value v. Alternatively, if a consensus mechanism is used in redundancy mode, the gateway will only receive a single value v for each cluster, which will have to be forwarded to the gateway.

Due to the NVIS backbone network's unavailability during night hours (from 17:00 until 6:00), values received by the gateway during this period will be stored in the gateway's memory and forwarded to the control center later (when the NVIS links start functioning at 6:00). On the contrary, values received during daytime (from 6:00 until 17:00) will be forwarded to the control center as soon as the gateway receives and processes them. As GTN-P stations do, the gateways also expect to receive an ACK packet for every payload packet they send to the control center. If an ACK packet is not received from the control center before a timeout, T_{out}, the data packet will be retransmitted up to a maximum of three times.

Finally, the control center will receive all the transactions that had not been lost through the network. Each value v from the received payload by the control center from node or node cluster i will be considered a transaction l. The control center will compute the STR by comparing the received values for each payload with the hardcoded values.

The probability, Pb, of a node having a byzantine fault is unlikely to be constant over time. As stated in [68], by associating the battery discharge to the WSN node aging process, the node reliability can be identified and associated with the battery charge level. Thus, following the model in [69], we can assume the impact of aging following a linear form as defined in Equation (5):

$$Pb(t) = Pb_0 + kt, \tag{5}$$

where Pb_0 is the probability of a node having a byzantine fault at time $t = 0$, and k is the aging factor. Thus, the probability of a node having a byzantine fault will increase hour by hour until its battery is completely drained at $t = t_d$, when it experiences a crash fault and $Pb(t_d) = 1$. However, this model will only be applied to GTN-P stations that will be powered by batteries. On the contrary, we assume that the gateways will always have a constant power supply in our use case because they will be placed in the research base. Thus, their probability of experiencing a byzantine fault will remain constant over time as defined in Equation (6):

$$Pb(t) = Pb_0. \tag{6}$$

As explained in Section 4, the use of corrective methods to improve the data trustworthiness provoke, in practice, that the probability, Pb_0, of a node experiencing a byzantine fault will decrease, thus reducing the number of FSV. For that reason, different values of Pb_0 will be used in our simulations to emulate the use of different corrective methods.

6.4. Social Trustworthiness Model

The reputational model for implementing social trustworthiness in our use case is a simplified version of the objective model defined in [51]. Our use case simplification assumes that all transactions will have the same weight, all nodes have the same computational capability, and the relationship factors between them are equal. Thus, the reputation R_i of node i can be measured as defined in Equation (7):

$$R_i = \alpha O_i^{short} + (1-\alpha) O_i^{short} \tag{7}$$

where O_i^{short} is the short-term opinion of node i, O_i^{short} is the long-term opinion of node i, and α is a design value between (0, 1) to ponder the importance of short-term and long-term opinions. The short-term opinion of node i is measured as stated in Equation (8):

$$O_i^{short} = \sum_{j=1}^{M} \sum_{l=1}^{L^{short}} C_{ij} f_{ij}^{l} / \sum_{j=1}^{M} \sum_{l=1}^{L^{short}} C_{ij}, \tag{8}$$

where M is the total number of nodes of the group, excluding node i, L^{short} is the number of last l transactions considered to be relevant for building the short-term opinion, f_{ij}^{l} is the feedback that node j gave to node i for transaction l, and C_{ij} is the credibility of node j to evaluate node i.

Analogously, the long-term opinion is calculated as defined in Equation (9):

$$O_i^{long} = \sum_{j=1}^{M} \sum_{l=1}^{L^{long}} C_{ij} f_{ij}^{l} / \sum_{j=1}^{M} \sum_{l=1}^{L^{long}} C_{ij} \,, \tag{9}$$

where L^{long} is the number of last l transactions considered to be relevant for building the long-term opinion, and $L^{long} > L^{short}$. The credibility of node j to evaluate node i is calculated as shown in Equation (10):

$$C_{ij} = \frac{R_j}{1 + \log(N_{ij} + 1)} \,, \tag{10}$$

where N_{ij} is the number of transactions between node j and node i.

6.5. Consensus Model

A consensus protocol can be modeled by knowing the background traffic (bps) that is introduced into the network and the number of byzantine nodes supported (Nb). In our use case, each group of redundant GTN-P stations will run the practical byzantine fault tolerance (PBFT) algorithm [70]. From [71], we can assume that the background traffic exponentially grows as the number of nodes participating in the consensus (Nt) group increases. Moreover, the number of tolerated byzantine nodes, Nb, is calculated as:

$$Nb = \frac{Nt - 1}{3} \,, \tag{11}$$

In the simulation, if more than Nb nodes experience a byzantine behavior, the agreement reached will have incorrect values. Otherwise, the resulting payload will contain the correct values.

6.6. Tests Definitions

A summary of the characteristics of the simulation tests is shown in Table 4.

Table 4. Simulation parameters.

Parameter	Value
Number of runs per test	30
Simulation duration	120 hours (5 days)
Simulation step	1 h
Pb_0	$[1 \times 10^{-3}, 2 \times 10^{-3}, 4 \times 10^{-3}, 8 \times 10^{-3}, 1 \times 10^{-2}, 2 \times 10^{-2}, 4 \times 10^{-2}, 8 \times 10^{-2}, 1 \times 10^{-1}]$
k	5.7×10^{-5}
Routing protocol	[AODV, OLSR]
Consensus mechanism	[None, PBFT]
Social Trustworthiness	[True, False]
Number of NVIS gateways	5
Number of GTN-P clusters per gateway	[8, 16, 32, 64, 128, 256, 512, 1024, 2048, 4096]
Number of GTN-P redundant stations per cluster	[1–10]

Each different test will be run 30 times, which gives us the total amount of 113,400 tests. Each test has a simulation duration of 5 days (120 hours), and the average value of the STR trustworthiness metric will be calculated. The different byzantine probabilities are proposed to simulate scenarios with different corrective methods that can reduce the byzantine probability of a node. On the other hand, two different routing protocols will be used to analyze which kind of method (reactive or proactive) has a better impact on the service's trustworthiness. Moreover, the Consensus and Social Trustworthiness Layers will be implemented or not to analyze their influence on the service's performance. Finally, the impact of the number of nodes connected to each gateway will also be studied by varying them. In standard mode, there was only one GTN-station per group (because there is no redundancy). This means that each of the 9 Pb_0 values must be tested against each number of GTN-P clusters per gateway, giving us a total of 90 possibilities. In redundancy mode, this number increases to 900 since the number of GTN-P stations per cluster varies from 1 to 10. If we sum the cases of standard mode, redundancy mode with consensus, and redundancy mode with social trustworthiness, we have a total of 1890 different cases, which are doubled to 3780 considering that we want to test the system with two different routing protocols. Considering that each test is repeated 30 times, a total of 113,400 simulations were run in the simulator.

7. Simulation Results

After performing all the simulations, the average value of the STR was calculated for every set of 30 runs per test. The obtained results had a maximum error deviation of 0.61% with a confidence interval of 99%. Three different operational modes for the telemetry service can be clearly identified: the standard mode, the redundancy mode with Social Trustworthiness Layer, and the redundancy mode with Consensus Layer. For every mode, an N x M-dimension grid with all the possible combinations of stimulation parameters was formed, where M is the number of different Pb_0 values (nine in our case as stated in Table 4, row 5), and N is the number of different GTN-P node combinations per gateway (10 in standard mode and 100 in redundancy mode). For every point in this grid, the average value of the trustworthiness STR metric was computed. If we link all the STR values for every neighboring point in the grid, a mesh with all the STR values is formed. We call this mesh the trustworthiness mesh. Figure 5 exhibits the trustworthiness mesh three-dimensional graph for all the operational modes. Given that the differences between the AODV and OLSR scenarios' obtained results are negligible, only the results for the AODV scenarios are shown. Figure 6 shows different two-dimensional perspectives of the trustworthiness mesh graph to understand and analyze the results better. For both figures, the "byzantine fault probability" axis has nine discrete points, which are (1×10^{-3}, 2×10^{-3}, 4×10^{-3}, 8×10^{-3}, 1×10^{-2}, 2×10^{-2}, 4×10^{-2}, 8×10^{-2}, 1×10^{-1}). The "redundant sensors x sensor clusters" axis has 100 discrete points, according to Table 4, rows 11 and 12, which are (1×2^{N}, 2×2^{N}, … , 10×2^{N}), where N = (3, 4, … , 12).

From Figures 5 and 6, we can analyze the behavior of the trustworthiness mesh. We can see how without redundancy, the STR is always lower than 0.8. Thus, we can conclude that due to the characteristics of the NVIS and LoRa networks, the threshold of maximum trustworthiness that can be achieved is approximately 80% of total transactions in our use case. This means that, on average, each monitored value arrives at the control center 19 times a day at most. The authors of [18] left the complete automatization of this use case as an open challenge, aiming to increase the monitoring frequency to visualize the daily variations of the monitored properties (air, snow, bottom snow, surface, and ground temperature, among others). Our project's objective was to receive 14 out of 24 (58.33%) values each day at least. This is the minimum acceptable threshold to achieve the goals in [18]. In the tests results, acceptable STR values (>0.58) are maintained if the number of sensor nodes is kept under 512, although it decreases below the desired trustworthiness threshold if the number of sensors per gateway is higher. Also, we can notice that the shape of the trustworthiness mesh is practically identical for all three cases in the "1 × N sensors" zone (no redundancy). This means that, as was expected, adding the Social or the Consensus Layers did not improve the level of trustworthiness if there was no redundancy.

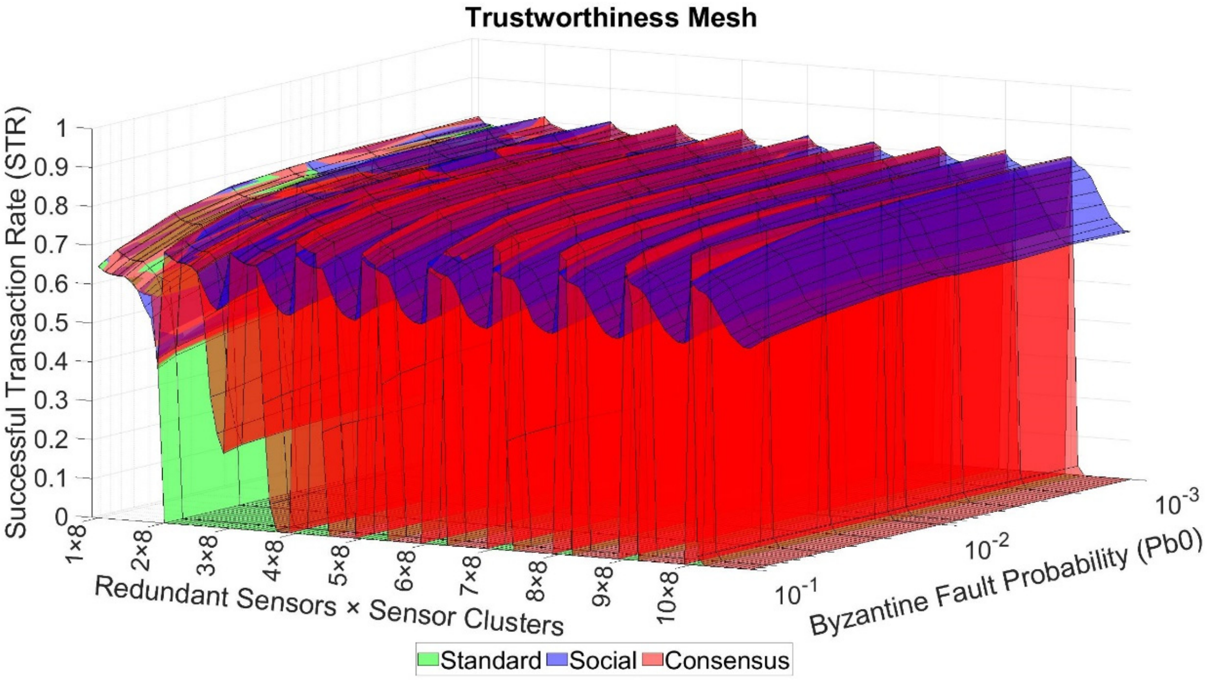
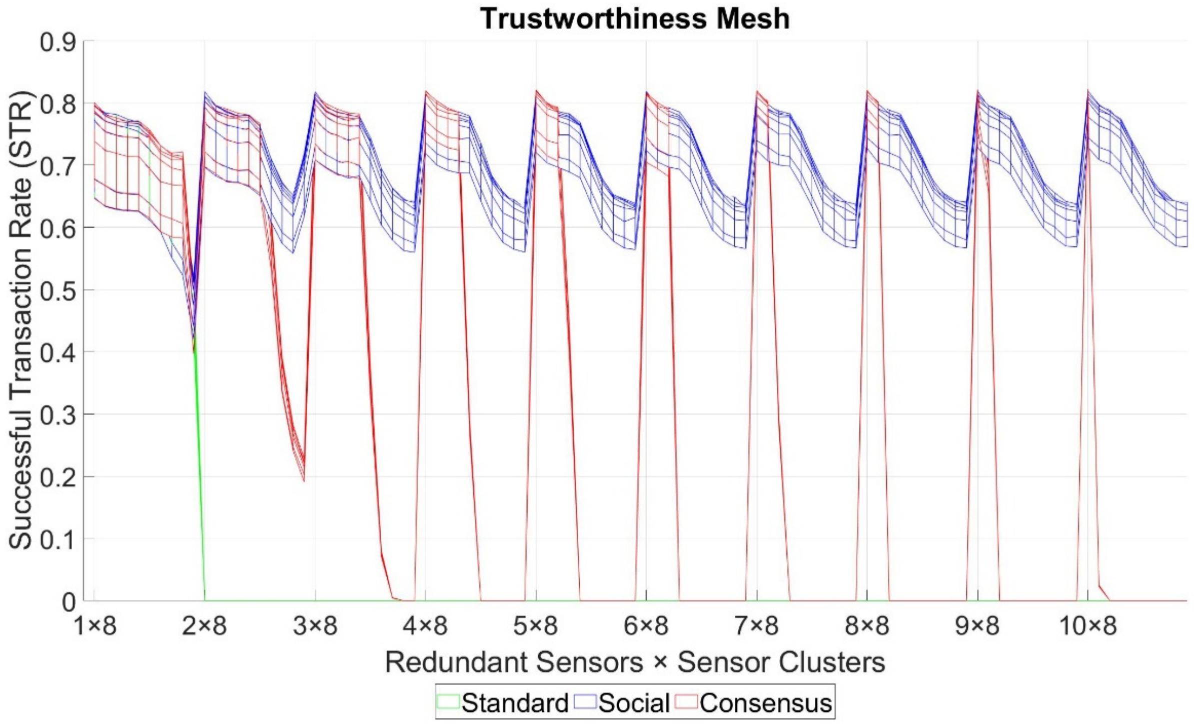


Figure 5. Trustworthiness mesh graph for the standard operational mode (green), the redundancy mode with Social Trustworthiness (blue), and the redundancy mode with Consensus mechanism (red).

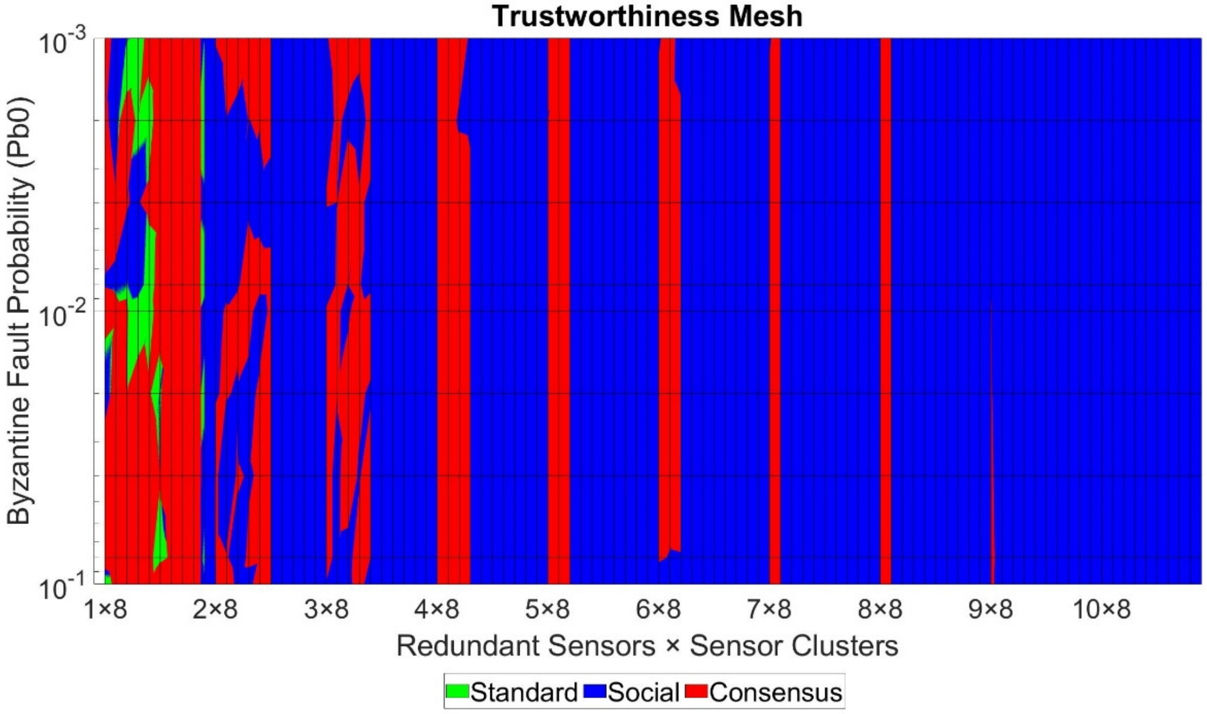
From Figures 5 and 6a, we can conclude that adding sensor redundancy and implementing the extension layers of our model improved the trustworthiness of the system, given that STR values greater than 0.8 were achieved. In cases of low redundancy ("2 × N sensors" and "3 × N sensors"), implementing the consensus mechanism did not improve the trustworthiness of the system when compared to the Social Trustworthiness case (the STR values are very similar). This is because, with two or three redundant nodes, the number of byzantine nodes tolerated by the consensus mechanism was still 0. Starting with four redundant nodes ("4 × N sensors"), the consensus mechanism's effects started to be noticed, achieving better STR values than the Social Trustworthiness case.

However, as the byzantine fault probability of the nodes decreases (meaning the FSR is lower), the difference between the STR values from the consensus mechanisms case and the Social Trustworthiness case becomes smaller. This means that implementing a consensus mechanism is more appropriate when the probability of the nodes experiencing byzantine behaviors is relatively high, and it is not necessary when this probability is low. In our cases, differences between STR values from both cases were not relevant as $Pb_0 \leq 0.01$.

Moreover, the quantity of network traffic that the consensus mechanism adds, combined with the LoRa and NVIS networks' low bandwidth, provokes low scalability for this solution. We can see that by looking at the evolution of the consensus trustworthiness mesh's STR values (red). We notice that as the number of sensor clusters increase, the STR values decreases until it drops to 0. This is because the nodes generate more traffic than the network supports. Thus, the network is congested, and the PDR rapidly decreases. Furthermore, the higher the number of redundant sensors per cluster, the sooner the STR dropping point (network saturation) happens. This resolves one of the looped dependencies postulated in Section 5.4.



(**a**)



(**b**)

Figure 6. Two-dimensional views of the trustworthiness mesh graph for the standard operational mode (green), the redundancy mode with Social Trustworthiness (blue), and the redundancy mode with consensus mechanism (red): (**a**) Frontal view of the trustworthiness mesh (STR vs. number of nodes); (**b**) top view of the trustworthiness mesh (byzantine fault probability vs. number of nodes).

On the contrary, it seems that implementing a social trustworthiness approach is more robust to these variations. Even if it did not achieve the same levels of STR as the consensus mechanism case when the number of sensor clusters was low, its STR values never dropped below 0.55 (very close to our desired minimum trustworthiness threshold), even in the scenario with more sensors and worse FSR. It is clear that the trustworthiness of the social case was also affected when the number of sensor nodes increased (which implies more network load and lower PDR), but its STR did not drop drastically and could maintain acceptable values. Due to the fact of our use case's modeling, the social trustworthiness implementation did not ostracize the nodes so that their sensed values were not collected, and the network load decreased. This is because in our simulations, each node had the same probability of experiencing a byzantine fault or sensing the value correctly, so ostracizing one of them could negatively affect the results. Thus, the behavior of the other looped dependency postulated in Section 5.4. remains uncertain.

From Figure 5, we can also conclude, as expected, that data trustworthiness had a direct affection on the overall system's trustworthiness. In all cases, as the byzantine fault probability, Pb_0, increased (meaning that more values are faultily sensed, increasing the FSR), the STR decreased.

Finally, Figure 6b shows for each of the 900 possible scenarios which is the most trustworthy option to implement the service. From this view, we can clearly see the robustness of the social trustworthiness case, showing how it gains ground as the number of sensors in the network increases.

8. Conclusions

This paper continues the SHETLAND-NET project's task to design a remote WSN for the Antarctic region using NVIS technology. The article focused on the use case of deploying a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service. A system and network architecture to implement the telemetry service was defined, using LoRa at the access network and NVIS long backhaul links at the backbone network. The extreme conditions remote sensors need to work with, added to the challenges of NVIS links and a LoRa network without LoS, can provoke a degradation of the overall system's trustworthiness. In order to study the viability of the service to be implemented before its deployment in the field during the Antarctic campaign, and aiming to anticipate the possible challenges that may arise, we proposed a model to measure and evaluate the trustworthiness of the system proposed. This trustworthiness model consists of four layers (two base layers and two extension layers) that can affect the successful transaction rate (STR) trustworthiness metric.

The trustworthiness model and the system architecture were validated using the Riverbed Modeler simulator. The obtained results have a maximum error deviation of 0.61% with 99% of confidence. The results show that the defined system architecture can reach acceptable levels of STR (>0.58) in case a relatively low number of sensors are deployed, although it drops too much with a large number of sensors. Adding redundancy to the measured values with multiple sensors and applying a social reputational mechanism improves the robustness of the system's trustworthiness, reaching higher STR values (>0.8) and never dropping below 0.55 even in high sensor-density scenarios. On the contrary, applying a consensus mechanism improves the system's trustworthiness when a low number of sensors are deployed. However, the STR values abruptly decrease as the number of deployed sensors increases.

Our model can also be used to visualize the work domain to implement our service, given a desired minimum trustworthiness level. For example, suppose our project requires a minimum STR of 0.7, so an average of 16 out of 24 sensed values per day reach the control center correctly. In this case, we could tolerate situations where the number of successful transactions that reached the control center was less than the average due to the fact of unexpected conditions but still achieving an STR higher than 58.33% to meet the objective of [18]. Figure 7 shows the work domain of our telemetry service for an STR higher than

0.7. For every point in the grid, if no solution provides an STR higher than the desired minimum value, the surface for that area is white-colored, meaning we cannot deploy the service with those conditions. On the contrary, if one or more solutions achieve an STR higher than the desired minimum value, the surface is painted with the color of the solution with the highest STR. However, it could be possible that we would prefer a solution that was not the one with the highest STR if all of the following criteria were met:

- The solution still has an STR greater than the desired minimum value;
- The solution has less cost than the solution with the highest trustworthiness in any sense (e.g., economic, computational resources, network load);
- The difference of the STR value achieved by the solution with less trustworthiness and the solution with the highest trustworthiness is less or equal to an established threshold value d_{max}.

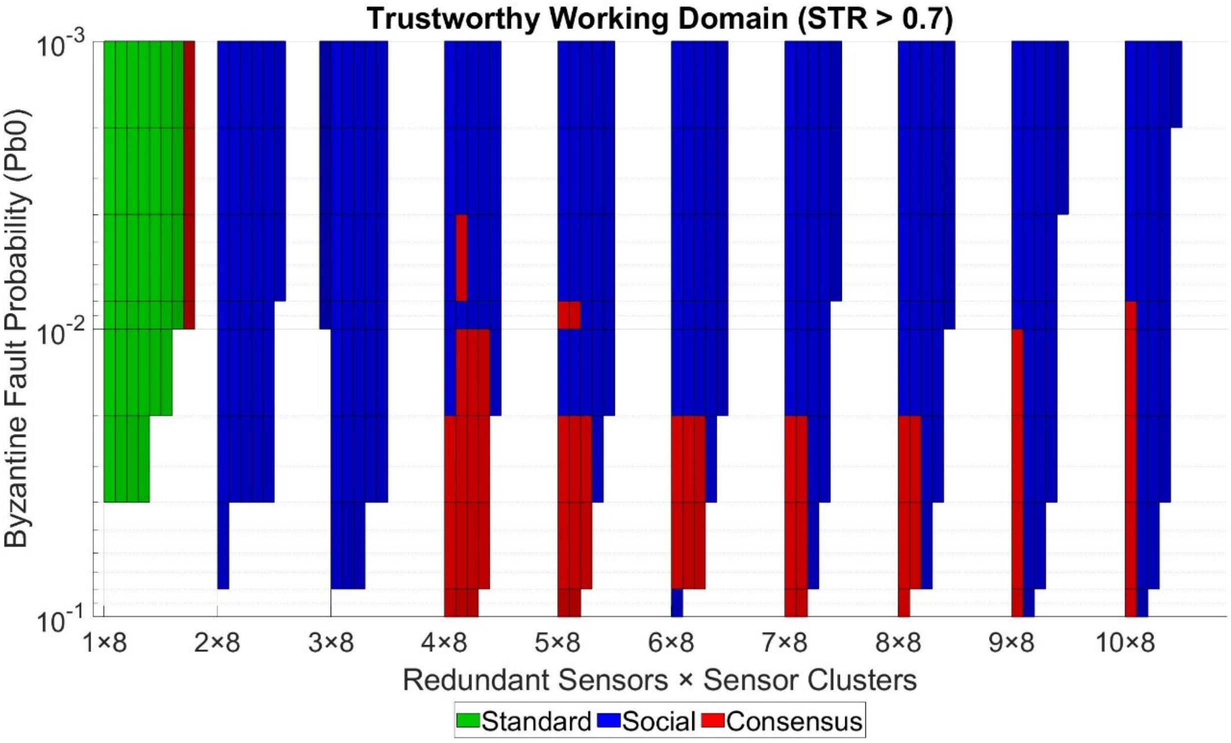


Figure 7. Trustworthiness mesh top view, coloring working domains with STR > 0.7.

We first preferred to deploy the standard mode in our use case, followed by the redundancy mode with the Social Trustworthiness Layer implementation and the redundancy mode with Consensus Layer implementation. This way, we prioritized the solution with lower resource consumptions (computational and network loads). To construct Figure 7, we set the value of d_{max} to 0.01. For our use case example, we chose this value arbitrarily. However, the value of this design parameter must be carefully analyzed for every particular use case to choose the actual optimal solution. The graph provides a clear vision of the work domains or areas that meet the necessary conditions to deploy the requested service guaranteeing the required minimum trustworthiness level. Furthermore, we can identify which solution to implement with the highest STR or the best trade-off between cost and STR for every grid point.

Performed simulations have also led us to understand better how all the proposed model actors work and relate to each other. Figure 8 synthesizes it. Blue-colored elements form part of our model base layers, and orange-colored elements form part of the extension layers. The final goal was to increase the STR to provide better trustworthiness. Three main factors directly help to increase the STR: (1) mitigating/tolerating byzantine errors;

(2) decreasing the FSR (2), and (3) increasing the PDR. These factors can be seen as subgoals that leverage the success of the final goal to provide trustworthiness. Each of these subgoals can be accomplished by implementing a set of actions or countermeasures. Each of these countermeasures affects only one of the subgoals.

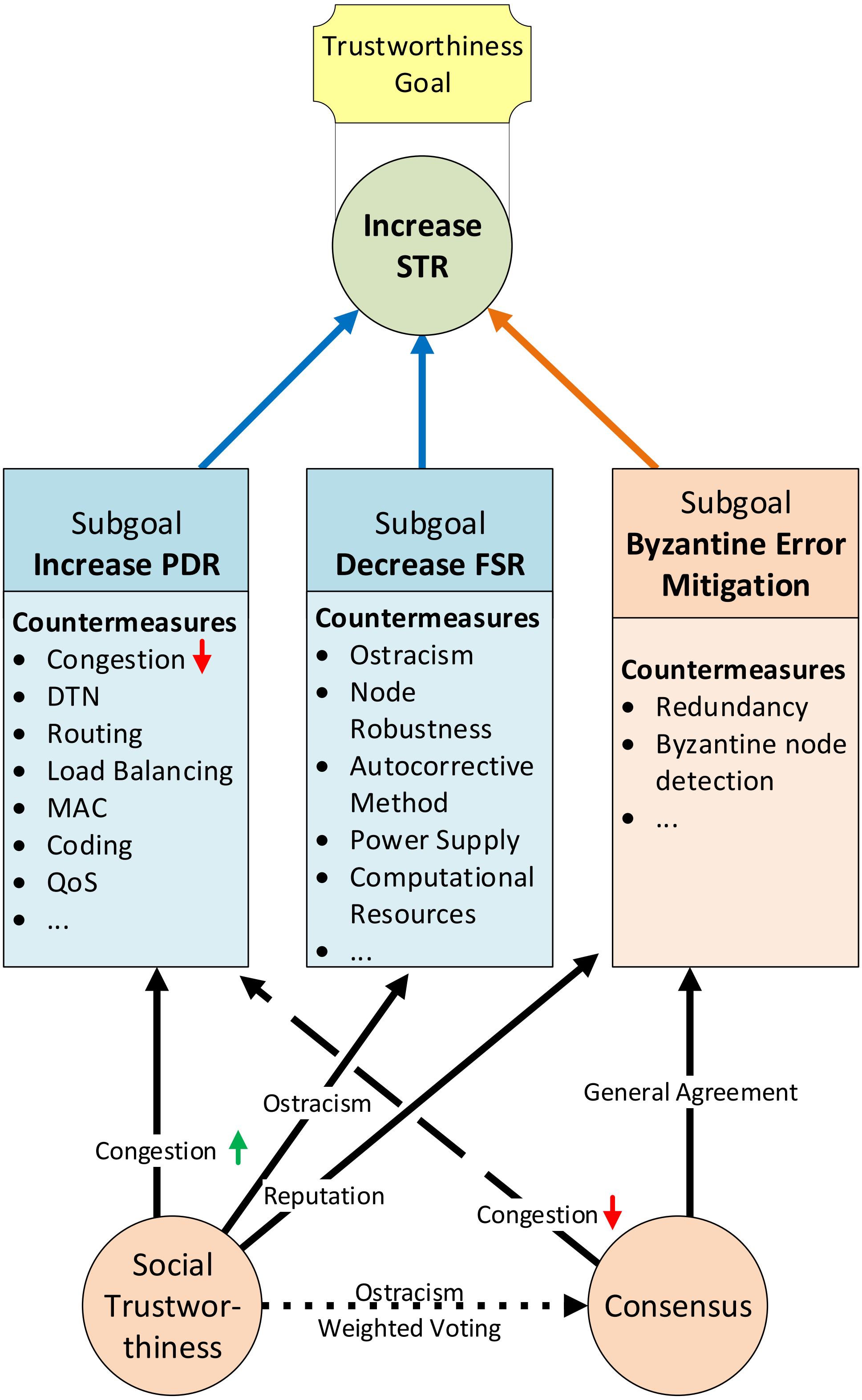


Figure 8. Trustworthiness model goals and countermeasures relationship.

Moreover, we have two transversal actions that affect more than one subgoal. These transversal actions implement the extension layers of our model: the Consensus Layer and the Social Trustworthiness Layer. Continuous-line arrows indicate a positive outcome, discontinuous-line arrows indicate a negative outcome, and dotted-line arrows indicate an uncertain outcome. Using social trustworthiness can reduce network congestion thanks to the ostracism of nodes with the worst reputation and send only the values from nodes with the highest reputation to the control center. Social trustworthiness also helps to reduce the FSR thanks to the ostracism of bad reputation nodes. It also leverages the mitigation of byzantine errors because only values from high reputation nodes (leaders) are trusted.

On the other hand, implementing a consensus mechanism mitigates byzantine errors thanks to the general agreements that are reached by all nodes from a consensus group. Contrarily, the Consensus Layer can negatively affect the PDR, given that it introduces a considerable amount of extra traffic to the network that could lead to link congestion. Finally, the Consensus Layer could also be affected by the Social Trustworthiness Layer if nodes' reputations were used to increase the reliability of general agreements (e.g., weighted voting based on node's reputation, ostracism of byzantine nodes), although its exact effect still remains unclear.

Future work aims to study the influence of implementing a DTN architecture at the NVIS backbone network, given that it has characteristics of challenging networks. The authors also plan to study the viability of deploying a FANET in the access network to provide connectivity to sensors placed outside the coverage area of the current LoRa network.

Author Contributions: Conceptualization, A.M. and A.Z.; methodology, A.M. and A.Z.; software, A.M.; validation, A.M. and A.Z.; formal analysis, A.M.; investigation, A.M.; resources, A.M. and A.Z.; data curation, A.M.; writing—original draft preparation, A.M.; writing—review and editing, A.M. and A.Z.; visualization, A.M.; supervision, A.Z.; project administration, A.Z.; funding acquisition, A.M. and A.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the "Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya", the European Union (EU), and the European Social Fund (ESF) (2020 FI_B 00448). The research was also funded by the "Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR)" of "Generalitat de Catalunya" (grant identification "2017 SGR 977"), and by the Spanish Ministry on Science, Innovation, and University, the Investigation State Agency and the European Regional Development Fund (ERDF) under the grant number RTI2018-097066-B-I00 (MCIU/AEI/FEDER, UE) for the project "NVIS SENSOR NETWORK FOR THE SOUTH SHETLAND ISLANDS ARCHIPELAGO" (SHETLAND-NET).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article, it is available based on reasonable request to the authors.

Acknowledgments: The authors would like to thank "La Salle-URL" (Universitat Ramon Llull) for their encouragement and assistance.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Algorithm A1: Sensor Node Application Pseudocode

```
int t, gateway_id, own_id, pk_id, tx_time, num_retries;
int data_values[3 2], node_reputations[N]; //N: number of redundnant nodes
float Pb0, Pb, k;
boolean consensus, social, is_leader, ack_received;
initializeVariables(Pb0,k,consensus,social, gateway_id, own_id, is_leader, ack_received);
for (t=0; t++; t<T_MAX){
    num_retries = 0;
    Pb = Pb0 + k * t;
    data_values = gatherData(Pb);
    if (consensus==TRUE){
        data_values = reachGeneralAgreement(data_values);
        if (social==TRUE) node_reputations = computeReputations();

        if (checkLeader(node_reputations)==TRUE)
            [tx_time, pk_id] = sendPacket(data_values, gateway_id, own_id);
        }else{
            pk_id = NULL;
        }
    }else{
        [tx_time, pk_id] = sendPacket(data_values, gateway_id, own_id);
    }
    if (pk_id != NULL){
        ack_received = checkAck(pk_id);
        while(ack_received==FALSE && num_retries<MAX_RETRIES){
            if (currentTime() >= tx_time + MAX_TIMEOUT){
                [tx_time, pk_id] = sendPacket(data_values, gateway_id, own_id);
                num_retries++;
            }
            ack_received = checkAck(pk_id);
        }
    }
    pk_id = NULL;
}
```

Algorithm A2: Gateway Node Application Pseudocode

```
int own_id, sensor_id; control_ctr_id, pk_id, tx_time, num_retries;
int data_values[32], stored_values[N][32], node_reputations[N]; //N: number of redundnant
nodes
float Pb;
boolean social, ack_received, data_pk_received;
initializeVariables(Pb, social, sensor_id, pk_id, own_id, control_ctr_id, ack_received,
data_pk_received);
while(TRUE){
num_retries = 0;
if (dataPkReceived()==TRUE){
[sensor_id, pk_id, data_values] = retrievePkData();
        if (social==FALSE){
            sendAck(pk_id, own_id, sensor_id);
            [tx_time, pk_id] = forwardDataPk(data_values, gateway_id, sensor_id,
control_ctr_id);
            ack_received = checkAck(pk_id);
while(ack_received==FALSE && num_retries<MAX_RETRIES){
                if (currentTime() >= tx_time + MAX_TIMEOUT){
                    [tx_time, pk_id] = forwardDataPk(data_values, gateway_id, sensor_id,
control_ctr_id);
                    num_retries++;
                }
                ack_received = checkAck(pk_id);
}
            ack_received=FALSE;
            num_retries=0;
        }else{
            stored_values[sensor_id] = data_values;
node_reputations[sensor_id] = computeReputation(data_values);
if (roundIsFinished()==TRUE){
                [data_values, sensor_id] = chooseData(node_reputations, stored_values);
                [tx_time, pk_id] = forwardDataPk(data_values, gateway_id, sensor_id,
control_ctr_id);
                ack_received = checkAck(pk_id);
while(ack_received==FALSE && num_retries<MAX_RETRIES){
if (currentTime() >= tx_time + MAX_TIMEOUT){
[tx_time, pk_id] = forwardDataPk(data_values, gateway_id, sensor_id, control_ctr_id);
                        num_retries++;
                    }
                    ack_received = checkAck(pk_id);
}
                ack_received=FALSE;
                num_retries=0;
            }
}
}
}
```

Algorithm A3: Control Center Application Pseudocode

```
int own_id, sensor_id; gateway_id, pk_id;
int data_values[32];
boolean data_pk_received;
initializeVariables(sensor_id, pk_id, own_id, gateway_id, data_pk_received);
while(TRUE){
    if (dataPkReceived()==TRUE){
        [sensor_id, gateway_id, pk_id, data_values] = retrievePkData();
        storeData(data_values, sensor_id);
        computeSTR(data_values);
        sendAck(pk_id, gateway_id);
    }
}
```

References

1. Kanao, M.; Genti, T.; Yamamoto, M.-Y. *Antarctica—A Key to Global Change*; IntechOpen: London, UK, 2019.
2. Antarctic Stations—Scientific Research Bases and Facilities. Available online: https://www.coolantarctica.com/Community/antarctic_bases.php (accessed on 29 March 2021).
3. Kennicutt, M.C.; Kim, Y.D.; Rogan-Finnemore, M.; Anandakrishnan, S.; Chown, S.L.; Colwell, S.; Cowan, D.; Escutia, C.; Frenot, Y.; Hall, J.; et al. Delivering 21st century Antarctic and Southern Ocean science. *Antarct. Sci.* **2016**, *28*, 407–423. [CrossRef]
4. Porté, J.; Lluis Pijoan, J.; Masó, J.; Badia, D.; Zaballos, A.; Maria Alsina-Pagès, R. Advanced HF Communications for Remote Sensors in Antarctica. In *Antarctica—A Key to Global Change*; IntechOpen: London, UK, 2019; pp. 21–39.
5. NVIS Sensors Network for the South Shetland Islands. Available online: https://www.salleurl.edu/en/research/research-lines-and-institutes/antarctica-project/summary (accessed on 29 March 2021).
6. Maso, J.; Porte, J.; Pijoan, J.L.; Badia, D. Internet of things communications for remote sensors in Antarctica using NVIS. In Proceedings of the Nordic HF, Fårö, Sweden, 12–14 August 2019; Volume 3.
7. Porte, J.; Maso, J.M.; Pijoan, J.L.; Badia, D. Sensing System for Remote Areas in Antarctica. *Radio Sci.* **2020**, *55*, 1–12. [CrossRef]
8. Vilella, C.; Miralles, D.; Pijoan, J.L. An Antarctica-to-Spain HF ionospheric radio link: Sounding results. *Radio Sci.* **2008**, *43*, 1–17. [CrossRef]
9. Ads, A.G.; Bergadà, P.; Vilella, C.; Regué, J.R.; Pijoan, J.L.; Bardají, R.; Mauricio, J. A comprehensive sounding of the ionospheric HF radio link from Antarctica to Spain. *Radio Sci.* **2013**, *48*, 1–12. [CrossRef]
10. Hervás, M.; Ma Alsina-Pagès, R.; Orga, F.; Altadill, D.; Pijoan, J.L.; Badia, D. Narrowband and wideband channel sounding of an antarctica to spain ionospheric radio link. *Remote Sens.* **2015**, *7*, 11712–11730. [CrossRef]
11. Male, J.; Porte, J.; Gonzalez, T.; Maso, J.M.; Pijoan, J.L.; Badia, D. Analysis of the Ordinary and Extraordinary Ionospheric Modes for NVIS Digital Communications Channels. *Sensors* **2021**, *21*, 2210. [CrossRef]
12. Jerez, S.; Motas, M.; Benzal, J.; Diaz, J.; Barbosa, A. Monitoring trace elements in Antarctic penguin chicks from South Shetland Islands, Antarctica. *Mar. Pollut. Bull.* **2013**, *69*, 67–75. [CrossRef]
13. Sancho, L.G.; Pintado, A.; Green, T.G.A. Antarctic Studies Show Lichens to be Excellent Biomonitors of Climate Change. *Diversity* **2019**, *11*, 42. [CrossRef]
14. Aidi, L.; Changsu, J. Delay Tolerant Network. 2012, pp. 1–19. Available online: https://www.academia.edu/download/33087272/delaytolerantnetwork.pdf (accessed on 29 March 2021).
15. Regi, M.; De Lauretis, M.; Redaelli, G.; Francia, P. ULF Geomagnetic Activity Signatures in the Atmospheric Parameters in Antarctica. In *Antarctica—A Key to Global Change*; IntechOpen: London, UK, 2019; pp. 5–20.
16. Palm, S.P.; Yang, Y.; Kayetha, V. New Perspectives on Blowing Snow in Antarctica and Implications for Ice Sheet Mass Balance. In *Antarctica—A Key to Global Change*; IntechOpen: London, UK, 2019; pp. 41–57.
17. Thomas, E.R.; Tetzner, D.R. The Climate of the Antarctic Peninsula during the Twentieth Century: Evidence from Ice Cores. In *Antarctica—A Key to Global Change*; IntechOpen: London, UK, 2019; pp. 75–92.
18. de Pablo Hernández, M.Á.; Jiménez, J.J.; Ramos, M.; Prieto, M.; Molina, A.; Vieira, G.; Hidalgo, M.A.; Fernández, S.; Recondo, C.; Calleja, J.F.; et al. Frozen ground and snow cover monitoring in livingston and deception islands, antarctica: Preliminary results of the 2015-2019 PERMASNOW project. *Geogr. Res. Lett.* **2020**, *46*, 187–222. [CrossRef]
19. Riverbed Modeler. Available online: https://www.riverbed.com/gb/products/npm/riverbed-modeler.html (accessed on 29 March 2021).
20. Al-Sarawi, S.; Anbar, M.; Alieyan, K.; Alzubaidi, M. Internet of Things (IoT) communication protocols: Review. In Proceedings of the IEEE 2017 8th International Conference on Information Technology (ICIT), Amman, Jordan, 17–18 May 2017; pp. 685–690.
21. Oliveira, L.; Rodrigues, J.J.P.C.; Kozlov, S.A.; Rabêlo, R.A.L.; de Albuquerque, V.H.C. MAC layer protocols for Internet of Things: A survey. *Futur. Internet* **2019**, *11*, 16. [CrossRef]

22. Jolly, B.; Willig, A.; McDonald, A.; Pannell, M.; Plank, G. SNOWWEB—Wirelessly connected weather stations in Antarctica. In Proceedings of the 38th Annual IEEE Conference on Local Computer Networks—Workshops, Sydney, Australia, 21–24 October 2013; pp. 194–202.
23. Gaelens, J.; Van Torre, P.; Verhaevert, J.; Rogier, H. Lora mobile-to-base-station channel characterization in the Antarctic. *Sensors* **2017**, *17*, 1903. [CrossRef] [PubMed]
24. Zaidi, S.; Atiquzzaman, M.; Calafate, C.T. Internet of Flying Things (IoFT): A survey. *Comput. Commun.* **2020**, *165*, 53–74. [CrossRef]
25. Liu, D.; Xu, Y.; Wang, J.; Chen, J.; Yao, K.; Wu, Q.; Anpalagan, A. Opportunistic UAV utilization in wireless networks: Motivations, applications, and challenges. *IEEE Commun. Mag.* **2020**, *58*, 62–68. [CrossRef]
26. Guillen-Perez, A.; Cano, M.D. Flying ad hoc networks: A new domain for network communications. *Sensors* **2018**, *18*, 3571. [CrossRef]
27. Lee, S.; Wu, Y.; Mortari, D. Satellite constellation design for telecommunication in Antarctica. *Int. J. Satell. Commun. Netw.* **2016**, *34*, 725–737. [CrossRef]
28. Porte, J.; Briones, A.; Maso, J.M.; Pares, C.; Zaballos, A.; Pijoan, J.L. Heterogeneous wireless IoT architecture for natural disaster monitorization. *EURASIP J. Wirel. Commun. Netw.* **2020**, *2020*, 184. [CrossRef]
29. Lee, E.A. Cyber physical systems: Design challenges. In Proceedings of the 2008 11th IEEE international symposium on object and component-oriented real-time distributed computing (ISORC), Orlando, FL, USA, 5–7 May 2008; pp. 363–369.
30. Zanero, S. Cyber-Physical Systems. *Computer* **2017**, *50*, 14–16. [CrossRef]
31. Crawford, M.; Liongosary, E. *IIC Journal of Innovation*; The Industrial Internet of Things Consortium: Milford, CT, USA, 2018; Volume 9.
32. Tang, L.A.; Yu, X.; Kim, S.; Gu, Q.; Han, J.; Leung, A.; La Porta, T. Trustworthiness analysis of sensor data in cyber-physical systems. *J. Comput. Syst. Sci.* **2013**, *79*, 383–401. [CrossRef]
33. Haron, N.; Jaafar, J.; Aziz, I.A.; Hassan, M.H.; Shapiai, M.I. Data trustworthiness in Internet of Things: A taxonomy and future directions. In Proceedings of the 2017 IEEE Conference on Big Data and Analytics (ICBDA), Kuching, Malaysia, 16–17 November 2017; pp. 25–30.
34. Yuan, H.; Zhao, X.; Yu, L. A Distributed Bayesian Algorithm for data fault detection in wireless sensor networks. In Proceedings of the IEEE 2015 International Conference on Information Networking (ICOIN), Siem Reap, Cambodia, 12–14 January 2015; pp. 63–68.
35. Zhang, G.; Li, R. Fog computing architecture-based data acquisition for WSN applications. *China Commun.* **2017**, *14*, 69–81. [CrossRef]
36. Fantacci, R.; Nizzi, F.; Pecorella, T.; Pierucci, L.; Roveri, M. False Data Detection for Fog and Internet of Things Networks. *Sensors* **2019**, *19*, 4235. [CrossRef]
37. Hassan, M.M.; Gumaei, A.; Huda, S.; Almogren, A. Increasing the Trustworthiness in the Industrial IoT Networks through a Reliable Cyberattack Detection Model. *IEEE Trans. Ind. Inform.* **2020**, *16*, 6154–6162. [CrossRef]
38. Dhaliwal, S.; Singh, N.; Kaur, G. Performance Analysis of Convolutional code over different Code rates and Constraint length in Wireless Communication. In Proceedings of the IEEE 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC), Palladam, India, 10–11 February 2017; pp. 464–468.
39. Feng, D.; Xu, H.; Zheng, J.; Bai, B. Nonbinary LDPC-Coded Spatial Modulation. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 2786–2799. [CrossRef]
40. Fang, Y.; Chen, P.; Cai, G.; Lau, F.C.M.; Liew, S.C.; Han, G. Outage-limit-approaching channel coding for future wireless communications: Root-protograph low-density parity-check codes. *IEEE Veh. Technol. Mag.* **2019**, *14*, 85–93. [CrossRef]
41. Bioglio, V.; Condo, C.; Land, I. Design of Polar Codes in 5G New Radio. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 29–40. [CrossRef]
42. Semong, T.; Maupong, T.; Anokye, S.; Kehulakae, K.; Dimakatso, S.; Boipelo, G.; Sarefo, S. Intelligent load balancing techniques in software defined networks: A survey. *Electronics* **2020**, *9*, 1091. [CrossRef]
43. Polese, M.; Chiariotti, F.; Bonetto, E.; Rigotto, F.; Zanella, A.; Zorzi, M. A survey on recent advances in transport layer protocols. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3584–3608. [CrossRef]
44. Alahari, H.P.; Yalavarthi, S.B. A Survey on Network Routing Protocols in Internet of Things (IOT). *Int. J. Comput. Appl.* **2017**, *160*, 18–22. [CrossRef]
45. Ahmed, D.E.M.; Khalifa, O.O. A Comprehensive Classification of MANETs Routing Protocols. *Int. J. Comput. Appl. Technol. Res.* **2017**, *6*, 141–158. [CrossRef]
46. Mrabet, H.; Belguith, S.; Alhomoud, A.; Jemai, A. A survey of IoT security based on a layered architecture of sensing and data analysis. *Sensors* **2020**, *20*, 3625. [CrossRef]
47. Bounsiar, S.; Benhamida, F.Z.; Henni, A.; Ipiña, D.L.d.; Mansilla, D.C. How to Enable Delay Tolerant Network Solutions for Internet of Things: From Taxonomy to Open Challenges. *Proceedings* **2019**, *31*, 24. [CrossRef]
48. Atzori, L.; Iera, A.; Morabito, G. SIoT: Giving a Social Structure to the Internet of Things. *IEEE Commun. Lett.* **2011**, *15*, 1193–1195. [CrossRef]
49. Caballero, V.; Vernet, D.; Zaballos, A. Social Internet of Energy—A New Paradigm for Demand Side Management. *IEEE Internet Things J.* **2019**, *6*, 9853–9867. [CrossRef]

50. Nitti, M.; Girau, R.; Atzori, L.; Iera, A.; Morabito, G. A subjective model for trustworthiness evaluation in the social Internet of Things. In Proceedings of the 2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications—(PIMRC), Sydney, Australia, 9–12 September 2012; pp. 18–23.
51. Nitti, M.; Girau, R.; Atzori, L. Trustworthiness Management in the Social Internet of Things. *IEEE Trans. Knowl. Data Eng.* **2013**, *26*, 1253–1266. [CrossRef]
52. Lin, Z.; Dong, L. Clarifying Trust in Social Internet of Things. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 234–248. [CrossRef]
53. Azad, M.A.; Bag, S.; Hao, F.; Shalaginov, A. Decentralized Self-Enforcing Trust Management System for Social Internet of Things. *IEEE Internet Things J.* **2020**, *7*, 2690–2703. [CrossRef]
54. Xiao, Y.; Zhang, N.; Lou, W.; Hou, Y.T. A Survey of Distributed Consensus Protocols for Blockchain Networks. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 1432–1465. [CrossRef]
55. Bodkhe, U.; Mehta, D.; Tanwar, S.; Bhattacharya, P.; Singh, P.K.; Hong, W.C. A survey on decentralized consensus mechanisms for cyber physical systems. *IEEE Access* **2020**, *8*, 54371–54401. [CrossRef]
56. Zoican, S.; Vochin, M.; Zoican, R.; Galatchi, D. Blockchain and Consensus Algorithms in Internet of Things. In Proceedings of the IEEE 2018 International Symposium on Electronics and Telecommunications (ISETC), Timisoara, Romania, 8–9 November 2018; pp. 1–4.
57. Sankar, L.S.; Sindhu, M.; Sethumadhavan, M. Survey of consensus protocols on blockchain applications. In Proceedings of the IEEE 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 6–7 January 2017; pp. 1–5.
58. Bano, S.; Sonnino, A.; Al-Bassam, M.; Azouvi, S.; McCorry, P.; Meiklejohn, S.; Danezis, G. SoK: Consensus in the Age of Blockchains. In Proceedings of the 1st ACM Conference on Advances in Financial Technologies, Zurich, Switzerland, 21–23 October 2019; ACM: New York, NY, USA, 2019; pp. 183–198.
59. Zheng, Z.; Xie, S.; Dai, H.; Chen, X.; Wang, H. An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends. In Proceedings of the 2017 IEEE International Congress on Big Data (BigData Congress), Honolulu, HI, USA, 25–30 June 2017; pp. 557–564.
60. Lao, L.; Li, Z.; Hou, S.; Xiao, B.; Guo, S.; Yang, Y. A survey of IoT applications in blockchain systems: Architecture, consensus, and traffic modeling. *ACM Comput. Surv.* **2020**, *53*, 1–32. [CrossRef]
61. Nitti, M.; Girau, R.; Atzori, L.; Pilloni, V. Trustworthiness management in the IoT: The importance of the feedback. In Proceedings of the IEEE 2017 20th Conference on Innovations in Clouds, Internet and Networks (ICIN), Paris, France, 7–9 March 2017; pp. 325–327.
62. Penning, A.; Baumgärtner, L.; Höchst, J.; Sterz, A.; Mezini, M.; Freisleben, B. DTN7: An Open-Source Disruption-Tolerant Networking Implementation of Bundle Protocol 7. In Proceedings of the International Conference on Ad-Hoc Networks and Wireless, Luxembourg, 1–3 October 2019; Springer: Cham, Switzerland, 2019; Volume 11803 LNCS, pp. 196–209.
63. Katsikas, S.; Gkioulos, V. Security, privacy, and trustworthiness of sensor networks and internet of things. *Sensors* **2020**, *20*, 3846. [CrossRef] [PubMed]
64. Rehman, A.U.; Jiang, A.; Rehman, A.; Paul, A. Weighted Based Trustworthiness Ranking in Social Internet of Things by using Soft Set Theory. In Proceedings of the 2019 IEEE 5th International Conference on Computer and Communications (ICCC), Chengdu, China, 6–9 December 2019; pp. 1644–1648.
65. Seo, J.; Ko, D.; Kim, S.; Park, S. A Coordination Technique for Improving Scalability of Byzantine Fault-Tolerant Consensus. *Appl. Sci.* **2020**, *10*, 7609. [CrossRef]
66. Refaei, M.T.; Dasilva, L.A.; Eltoweissy, M.; Nadeem, T. Adaptation of reputation management systems to dynamic network conditions in Ad Hoc networks. *IEEE Trans. Comput.* **2010**, *59*, 707–719. [CrossRef]
67. Jun, A.D.; Hong, S.; Lee, W.; Lee, K.; Joe, I.; Lee, K.; Park, T.J. Modeling and Simulation of LoRa in OPNET. In *Advanced Multimedia and Ubiquitous Engineering*; Springer: Singapore, 2017; pp. 551–559.
68. Distefano, S. Evaluating reliability of WSN with sleep/wake-up interfering nodes. *Int. J. Syst. Sci.* **2013**, *44*, 1793–1806. [CrossRef]
69. Pan, X.; Di Maio, F.; Zio, E. A benchmark of dynamic reliability methods for probabilistic safety assessment. In Proceedings of the IEEE 2017 2nd International Conference on System Reliability and Safety (ICSRS), Milan, Italy, 20–22 December 2017; pp. 82–90.
70. Castro, M.; Liskov, B. Practical Byzantine Fault Tolerance and Proactive Recovery. *ACM Trans. Comput. Syst.* **2002**, *20*, 398–461. [CrossRef]
71. Lei, K.; Zhang, Q.; Xu, L.; Qi, Z. Reputation-Based Byzantine Fault-Tolerance for Consortium Blockchain. In Proceedings of the 2018 IEEE 24th International Conference on Parallel and Distributed Systems (ICPADS), Singapore, 11–13 December 2018; pp. 604–611.

remote sensing

MDPI

Article

DTN Trustworthiness for Permafrost Telemetry IoT Network

Adrià Mallorquí *, Agustín Zaballos and Alan Briones

GRITS, Engineering Department, La Salle, Universitat Ramon Llull (URL), 08022 Barcelona, Spain; agustin.zaballos@salle.url.edu (A.Z.); alan.briones@salle.url.edu (A.B.)
* Correspondence: adria.mallorqui@salle.url.edu; Tel.: +34-932-902-436

Abstract: The SHETLAND-NET research project aims to build an Internet of Things (IoT) telemetry service in Antarctica to automatize the data collection of permafrost research studies on interconnecting remote wireless sensor networks (WSNs) through near vertical incidence skywave (NVIS) long fat networks (LFN). The proposed architecture presents some properties from challenging networks that require the use of delay tolerant networking (DTN) opportunistic techniques that send the collected data during the night as a bulk data transfer whenever a link comes available. This process might result in network congestion and packet loss. This is a complex architecture that demands a thorough assessment of the solution's viability and an analysis of the transport protocols in order to find the option which best suits the use case to achieve superior trustworthiness in network congestion situations. A heterogeneous layer-based model is used to measure and improve the trustworthiness of the service. The scenario and different transport protocols are modeled to be compared, and the system's trustworthiness is assessed through simulations.

Keywords: transport protocols; trustworthiness; Antarctica; IoT; NVIS; remote WSN; LFN

check for updates

Citation: Mallorquí, A.; Zaballos, A.; Briones, A. DTN Trustworthiness for Permafrost Telemetry IoT Network. *Remote Sens.* **2021**, *13*, 4493. https://doi.org/10.3390/rs13224493

Academic Editors: Andreas J. Dietz, Sebastian Roessler and Celia Amélie Baumhoer

Received: 24 September 2021
Accepted: 6 November 2021
Published: 9 November 2021

1. Introduction

Research studies from multiple disciplines are carried out every year in Antarctica [1]. Researchers are temporarily placed in Antarctic base stations, normally located in the peripheral areas of the continent. One of the main challenges in Antarctica is its lack of conventional telecommunication systems [1], which hinders the deployment of wireless sensor networks (WSNs). This fact reduces the possibilities of carrying out research studies (e.g., automation of data collection and remote bases interconnection).

To overcome these difficulties, our research project, the SHETLAND-NET, proposes the use of near vertical incidence skywave (NVIS) high-frequency (HF) radio links to provide low-consumption Antarctic communications, continuing previous research on ionospheric communications [2]. The ionosphere reflects this signal, providing a long backhaul link of a 250 km radius coverage area [3,4]. Networks using this type of links can be classified as long fat networks (LFNs), which are characterized by having long links with a bandwidth delay product (*BDP*) greater than 1×10^5 bits (12,500 bytes) [5], following Equation (1), where the link bandwidth (*BW*) is expressed in bits per second (bps) and the round-trip time (*RTT*) in seconds (s).

$$BDP = BW \times RTT. \quad (1)$$

The NVIS technology can be used to interconnect remote base stations [6]. Our final goal is to deploy a telemetry service by interconnecting remote WSNs [7], which will help in the automatization of data gathering for Antarctic research studies. This deployment will be carried out during the next Antarctic campaign in the field. However, this communication technique can be error-prone due to the variant properties of the ionosphere. It may present typical challenging network issues [8], such as intermittent connectivity, end-to-end disconnection, and variable error rates, which could degrade the performance of the overall offered IoT service.

Therefore, before the deployment phase of our project, we had to study and try to anticipate the expected trustworthiness of the IoT telemetry service we want to deploy. For this reason, we defined a model to assess the trustworthiness of our proposed system [7]. This enabled us to foresee the possible trustworthiness issues that might arise during the campaign in the field and decide on the respective countermeasures.

For our work, we focus on the use case of automating the monitoring of Ground Terrestrial Network-Permafrost (GTN-P) stations [9], which are used in permafrost research studies. Each of these GTN-P stations senses 32 different values hourly, which need to be remotely monitored from a control center. During the Antarctic campaign, we will deploy a test scenario. WSNs will be placed in two locations: the Spanish Juan Carlos I Base in Livingston Island, and the Uruguayan Artigas Base in King George Island, both part of the South Shetland Islands. The Artigas Base will provide Internet connectivity, so data gathered from the WSNs can be reached remotely. However, sensors in the Juan Carlos I Base will not have direct Internet connectivity, and the data from these sensors will need to be sent through an NVIS link to the Artigas base in order to reach the Internet. Figure 1 shows the test scenario in Antarctica.

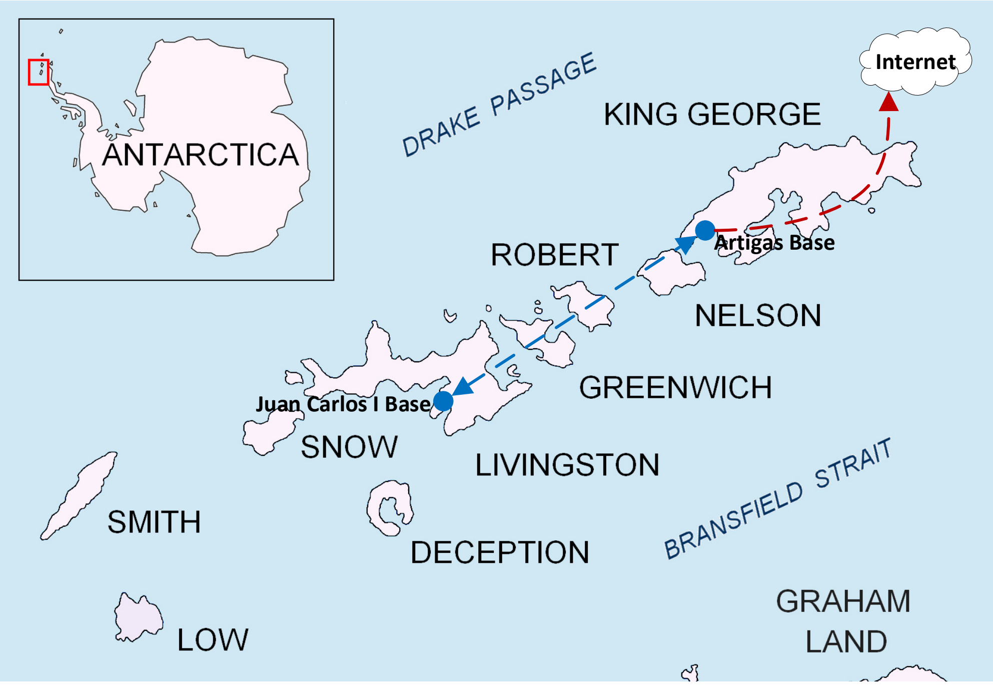


Figure 1. Map of the South Shetland Islands in Antarctica [10], showing the position of the WSNs (blue circles) during the test scenario of the campaign. The NVIS link is represented with the discontinuous blue line, and the Internet connectivity is represented with the discontinuous red line. The reproduction of the image was slightly modified under a Creative Commons License (CC BY-SA 3.0).

As seen in previous research [4], the main drawback of the NVIS link is its unavailability during the night, given that the ionosphere's characteristics vary drastically due to solar activity. For this reason, we decided to adopt a delay tolerant network (DTN) technique to opportunistically send all the data collected during the night as a bulk data transfer when the NVIS link becomes available in the morning. This complex scenario required a trustworthiness assessment to analyze its feasibility to be deployed in Antarctica before the campaign [7]. As shown in our first round of simulations, performing this opportunistic bulk data transfer in an LFN that presents network challenges could degrade the system's performance (packet losses) due to network congestion caused by the large quantity of data sent. On the other hand, in prior work, we also analyzed the suitability of different transport protocols for LFNs and designed a new one, the Enhanced Adaptive and Aggressive Transport Protocol [5,11]. Given that the NVIS links can also be considered as LFNs and given the strong performance that some modern transport protocols showed in our tests, we believed that it was crucial to assess how the use of modern transport protocols could improve or affect the performance and trustworthiness of the service, especially in this congestion situation provoked by the DTN technique. Having collected the initial results and analyzed the system's trustworthiness in previous work with the standard transport protocols of the devices' operative systems, this paper studies the trustworthiness and compares the usage of different transport protocols by modeling the scenario in the Riverbed Modeler. The paper contributions are as follows:

1. The definition and concretion of the remote sensor network architecture that will be deployed in Antarctica, detailing the type of nodes, protocol stack, and communication techniques that will be used.
2. The modeling of the Antarctic scenario in the simulator. To perform the simulation tests, we modeled the communication media (LoRa and NVIS), the telemetry application, the faulty behavior of Byzantine nodes, the social trust management and the consensus algorithms, the DTN technique, and the tested transport protocols.
3. The assessment and analysis of the results using our proposed trustworthiness model. From this analysis, we conclude which transport protocol best suits our use case and propose a modification of the scenario to be deployed in Antarctica.

The rest of this paper is structured as follows. Section 2 describes the related work in DTNs, transport protocols, and a system's trustworthiness. Section 3 defines our use case's network architecture. Section 4 reminds our proposed model to measure and evaluate a system's trustworthiness. Section 5 describes the simulation tests. Sections 6 and 7 present and discuss the obtained results, respectively. Finally, Section 8 concludes the paper.

2. Related Work

2.1. Delay Tolerant Networks

The DTN was first presented as an alternative network architecture designed for challenging networks [8] which suffer from high bit error rates, lack of end-to-end connectivity, and long delays [12]. It was initially designed for interplanetary communications in space [13], given the number of disconnections that this network suffers. However, over the years, many other types of terrestrial networks have emerged in response to similar problems (e.g., underwater networks [14], wildlife tracking networks [15], sparse wireless sensor networks [16], and vehicular networks [17]).

Conventional TCP/IP protocols are not suitable for these kinds of environments. In contrast, the RFC 5050 presented a DTN protocol, the Bundle Protocol (BP) [18], which enabled message delivery to cope with all the issues of challenging networks, even if the source and the destination were never connected to the network simultaneously. The BP is based on a store–carry–forward overlay network, where "bundles" are transported through endpoints on top of the transport layer of the OSI model when a connection opportunity is present between two endpoints. The BP version 7 draft was recently released [19], which introduces new features, such as optional CRCs for nonprimary blocks, and proposes other changes to make it simpler, more capable, and easier to use. Many implementations of the Bundle Protocol adapted to the constraints of IoT and WSNs exist nowadays, such as IBR-DTN [20], μDTN [21], and DTN7 [19], among others.

However, other DTN approaches are not based on the BP but use their own routing protocol designed to be disruption- and delay-tolerant [8]. DISRN [22], PASR [23], RMDTN [24], and PROPHET [25] are some examples of this kind of approach. Moreover, we can find other schemes that mix DTN with other kinds of technologies, such as opportunistic networking [26,27], machine to machine (M2M) communications [28], information-centric networking (ICN) [29], and fog computing [30].

As stated before, in our use case, we will use an opportunistic networking technique to send all the data collected during the night in the morning, when the NVIS link comes available, as a bulk data transfer. This kind of approach is possible because our research group has studied the behavior of the ionosphere and NVIS links in prior research [4], and were aware that the link is down at nighttime and becomes available at sunrise. However, we also know this bulk data transfer provokes network congestion, degrading the system's performance with packet losses. For this reason, it is crucial to study how modern transport protocols can help improve this performance, especially in LFNs such as the NVIS links.

2.2. Transport Protocols

The performance of transport protocols for network communications has been a topic under discussion and development since the Internet was conceived [5]. The first extensions of the original Transmission Control Protocol (TCP) were [31] TCP Tahoe, TCP Reno, TCP New-Reno, TCP SACK, and TCP-Vegas, which included new mechanisms such as the fast retransmit, the fast recovery, the packet pair link estimation, the duplicated acknowledgment (DUACK), and the selective acknowledgment (SACK).

However, these legacy transport protocols suffered performance degradation over some types of networks, including LFNs. The LFN concept and its effects on TCP performance were firstly defined and detailed in the Request For Comments (RFC) 1072, which was obsoleted by the RFC 1323 to finally become the standard RFC 7323. Some TCP variants and other transport protocols developed during the last decade have improved their performance over LFNs [5]. Some of these are Scalable TCP (S-TCP) [32], FAST TCP [33], High-Speed TCP (H-TCP) [34], Binary Increase Control TCP (BIC-TCP) [35], and its evolution: TCP CUBIC [36]. TCP CUBIC (RFC 8312) is the most commonly used transport protocol nowadays, given that it is the TCP variant used by default on most operating systems. However, most of these protocols consider that packet loss always occurs due to network congestion, reducing the congestion window. This assumption is false for wireless links, where packets can also be dropped for other reasons (e.g., fading, channel interference) [11]. Under these circumstances, reducing the congestion windows might also degrade the transmission performance, achieving lower throughput [11].

For this reason, other transport protocols, such as Performance-oriented Congestion Control (PCC) [37], TCP Veno [38], TCP Westwood+ [39], Dynamic TCP [40], Jitter TCP [41], and Jitter Stream Control Transmission Protocol (JSCTP) [42] are focused on implementing mechanisms to detect if lost packets occur due to network congestion or random channel loss. They only reduce the congestion window in the first case, achieving better performance [11].

In addition, other modern transport protocols, such as TCP BBR [43], Copa [44], Indigo [45], and Verus [46], can achieve high performance, as proven in several physical tests carried out by Stanford University's platform Pantheon [45]. TCP BBR is one of the top-performance protocols, managing the maximum bandwidth with the minimum RTT. Copa is a practical delay-based protocol that fixes an RTT target and adjusts its congestion windows based on the minimum RTT and the standing RTT measured during data transfers. Indigo is a data-driven protocol that uses a machine-learning congestion control scheme that learns from previous performance data. Verus is a transport protocol oriented to cellular networks that relates the congestion windows with delay variations through short-term RTT measurement.

Moreover, given that the aforementioned protocols did not meet the performance requirements of our cloud data-sharing use case from previous work [11], we presented the Adaptive and Aggressive Transport Protocol (AATP) [5] and its evolution, the Enhanced AATP (EAATP) [11], which incorporates mechanisms to differentiate the packet losses' cause, fairly adapting its sending rate accordingly to the network circumstances. The performance in these tests was solid, both in simulations and in a physical testbed with an LFN emulator, showing better results than other protocols, maximizing throughput and minimizing packet losses [5,11]. Figure 2 shows a summary of the tests' results. However, we did not know how these protocols (including ours) could affect the trustworthiness of a system, especially in the use case of this paper. For this reason, we thought that we needed to assess whether using the EAATP in the remote Antarctic WSN use case could improve the system's performance and trustworthiness, especially in congestion situations.

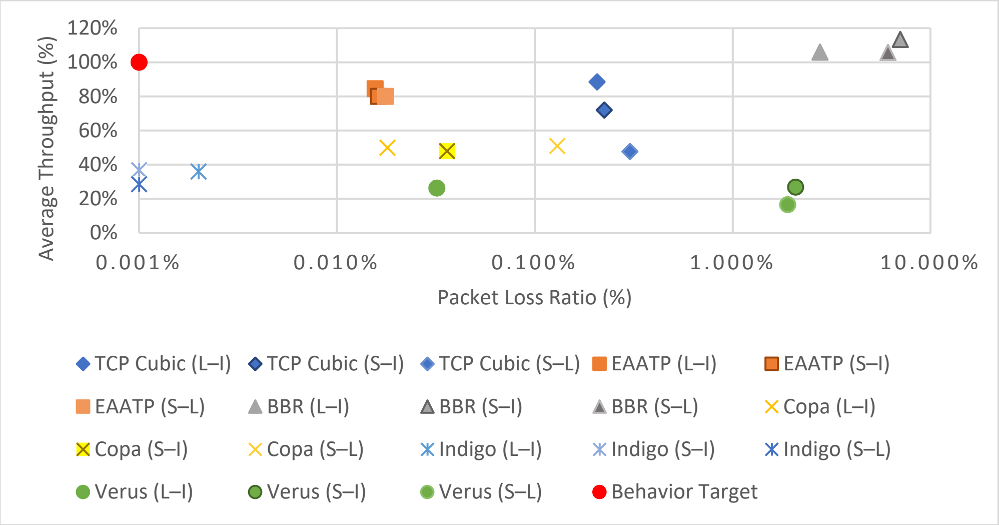


Figure 2. Results of average throughput (%) vs. packet loss ratio (%) of the transport protocols tested in previous work [11]. To represent the graph in semilogarithmic scale, the packet loss ratio values of 0% are represented as 0.001% in the graph. Each transport protocol was tested in three LFN scenarios: London to Iowa (L–I), Sidney to Iowa (S–I), and Sidney to London (S–L).

2.3. Trustworthiness in Cyber Physical Systems

A cyber physical system (CPS) is defined as a system with integrated computational and physical capabilities. Wireless sensor networks, smart grids, and some IoT devices are examples of CPSs [47]. Even though there is no consensus in the literature to define the trustworthiness property and its scope [48], we can define a CPS's trustworthiness, in general terms, as the property of behaving as expected under adversarial conditions [47]. Network malfunction, Byzantine errors, and faulty nodes are examples of adverse conditions that can affect a system's trustworthiness. Some authors limit this definition to system security issues only [49], while others propose a broader scope and relate trustworthiness with other terms such as resilience, availability, reliability, scalability, maintainability, heterogeneity, data quality, hardware resources, and fault management policies [48]. We can find many approaches to measuring or providing trustworthiness in literature, referring to different elements. We classify them into four main categories [7]:

1. Data trustworthiness: It is defined as the possibility to ascertain the correctness of the data provided by the source [50]. Many methods use different approaches that try to detect faulty nodes, false alarms, and sensor misreading using. For instance, authors in [51] use a fog computing architecture to detect, filter, and correct abnormal sensed data. In addition, authors in [52] present a data intrusion detection system to trigger false data from malicious attacks.
2. Network trustworthiness: Defined as the likelihood of a packet to reach its destination unaltered despite the adversities (e.g., link failure, link saturation, or malicious attacks, among others), it is a relevant aspect to consider in challenging networks [53], such as the use case we propose. The network's performance and trustworthiness have been addressed from several perspectives, such as channel coding [54], transport protocols [11], dynamic routing and topology control protocols [55,56], and DTN architectures and protocols [8].
3. Social trustworthiness: This field has become more popular since the appearance of the Social Internet of Things (SIoT) [57,58]. In SIoT trustworthiness, objects or network nodes interact and establish social relationships, which are used to define trust and reputation models that take into account several input parameters. Authors in [59] present a model that considers factors as the computational capabilities of the nodes, the type of relationship between them, the total number of transactions, the credibility of a node, and the feedback provided by other nodes, among others.

Authors in [60] present an evolution of the aforementioned trust management model, which applies a machine learning algorithm to calculate novel parameters such as the goodness, usefulness, and perseverance of a node. Thanks to these parameters, this upgraded trust model is resilient to more types of malicious node attacks. Authors in [61] propose another model that defines the input parameters as the expected gain on success, the expected damage on a failure, the expected cost, the expected result, and the goal. Authors in [62] define a decentralized self-enforcing trust management system which is based on a feedback system and reputational secure multiparty calculations to ensure the privacy of each party's provided data.

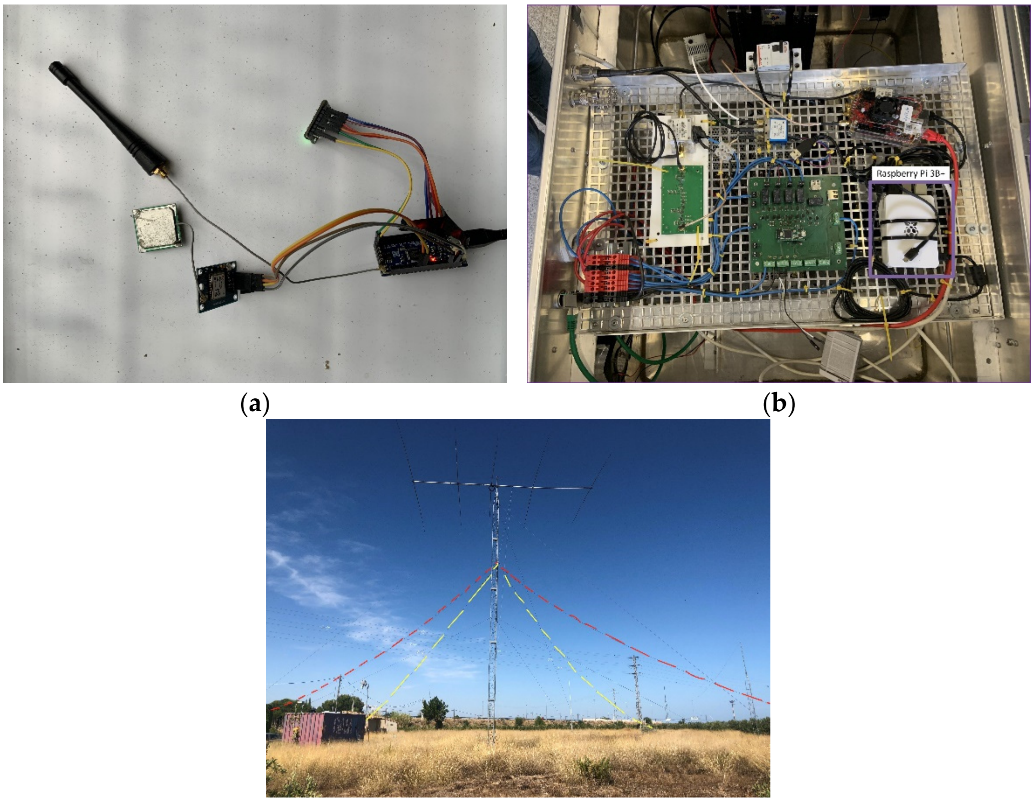
4. Consensus: It represents a state where all the participants of the same distributed system agree on the same data values [63]. Consensus protocols can be classified into two major groups: proof-based consensus and Byzantine consensus. The first group is related to blockchain technology, where all participants compete against each other to mine a block, and the most commonly used protocols are proof-of-work, proof-of-stake, and their variants [63]. The main drawback of these protocols for the IoT is that devices usually have lesser hardware resources and low processing power, which make the mining tasks of blockchain extremely difficult [63]. On the other hand, Byzantine-based protocols implement voting-based mechanisms to reach an agreement rather than competing among them, generating less resource consumption in general. Their main drawback is the number of messages that need to be delivered through the network to reach an agreement. Some well-known protocols from this category are Practical Byzantine Fault Tolerance (PBFT), RAFT, PaXoS, and Ripple, among others [63].

3. Remote Sensor Network Architecture

As stated before, the use case of this article is an IoT telemetry service to monitor remote WSNs in Antarctica interconnected through NVIS LFNs. The monitored data are used for permafrost studies and are gathered by GTN-P stations [9], which are the sensors of our network. Each of these GTN-P stations senses 32 different values hourly, and these values must reach the remote control center in Europe.

The GTN-P stations are equipped with a Moteino [64], an Arduino-based board designed for low-power consumption applications. The Moteino will send, through LoRa, its sensed values to a Raspberry Pi 3B+ gateway acting as a concentrator (access network). LoRa was preferred over other alternatives (e.g., Sigfox, NB-IoT) as the access network protocol because of its teleoperator independence. The LoRa network will be configured with a transmission frequency of 868 MHz, a code rate CR3 (4/7), and a spreading factor SF7, obtaining a 125 kHz channel bandwidth with a bit rate of 5.47 kbps. As proved in [65], this configuration can offer a coverage range of up to 30 km in Antarctica. Figure 3a shows the Moteino board with the LoRa transceiver that will be used during the campaign to collect and forward the data from the GTN-P stations.

The Raspberry Pi 3B+ gateway will forward these data through NVIS links (backbone network) to the Internet edge router in the Uruguayan Artigas Base in Antarctica. NVIS was preferred over satellite communication because the latter presents coverage issues in polar zones and has a higher economic cost [3]. The NVIS nodes will be configured to transmit at the 4.3 MHz transmission band, with a channel bandwidth of 2.3 kHz and a bit rate of 4.6 kbps. As in [3], we will increase the NVIS transmission reliability with an FEC convolutional code (1/2 rate code) and interleaving. With this configuration, an NVIS link range is up to 250 km. Figure 3b shows the NVIS node with the Raspberry Pi 3B+ gateway, and Figure 3c shows the NVIS antenna (inverted vee antenna).



(**a**) (**b**)

(**c**)

Figure 3. Antarctic WSN Hardware. (**a**) Moteino node with LoRa transceiver. (**b**) NVIS node with Raspberry Pi 3B+ gateway. (**c**) NVIS inverted vee antenna.

From the closest NVIS node to the Internet edge router (the one with Internet connectivity), data will be pushed to the Internet. From this moment, data monitoring and gathering will be available remotely from the control center. Figure 4 shows the network architecture diagram of the remote WSN.

The Artigas Base's Internet connectivity is supposed to have high reliability, so our trustworthiness assessment is focused on the access network (LoRa) and the backbone network (NVIS). As mentioned before, the reliability of NVIS links is very dependent on the ionosphere state, so it is not possible to send data during the night as all of it would be lost. For this reason, we believed it was necessary to apply a DTN technique to prevent the loss of data gathered during the night. In our case, we apply the DTN in the backbone network, as it is more likely to suffer from a lack of end-to-end connectivity, long delays, and network disruption.

Given that, in our case, we can predict a specific time slot when the NVIS links do not work (nighttime), we opted to implement a lightweight DTN approach, opportunistically sending the data collected during the whole night as a bulk transfer when the NVIS channel becomes available in the morning. Each concentrator should have collected 13 different sets of sensed values from each GTN-P station during the night. Our project requires that, on average, at least 9 out of the 13 datasets gathered from each station (around 70%) reach the control center correctly [7].

The DTN is usually implemented as an overlay network below the application layer of the Open Systems Interconnection model (OSI model) and needs a convergence layer as an interface to connect to the lower layers of the protocol stack. Figure 5 shows the protocol stack from our use case.

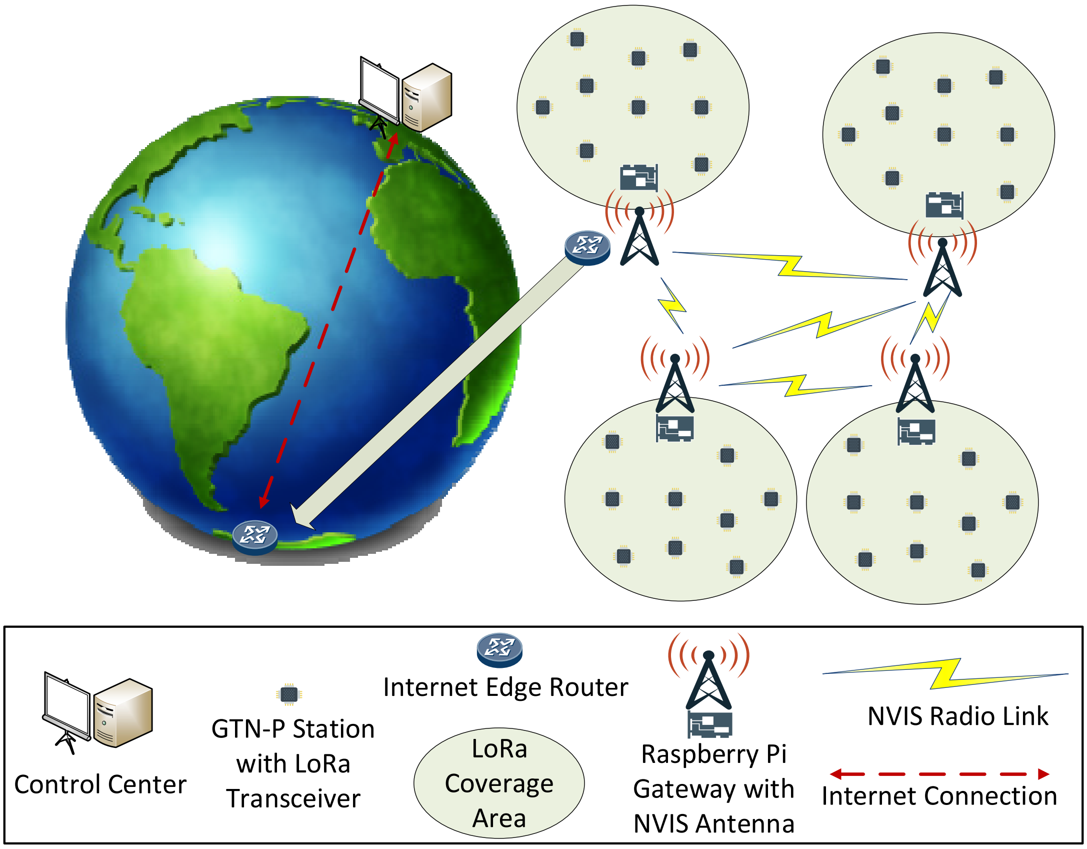


Figure 4. Network architecture of the remote WSN providing the IoT telemetry service.

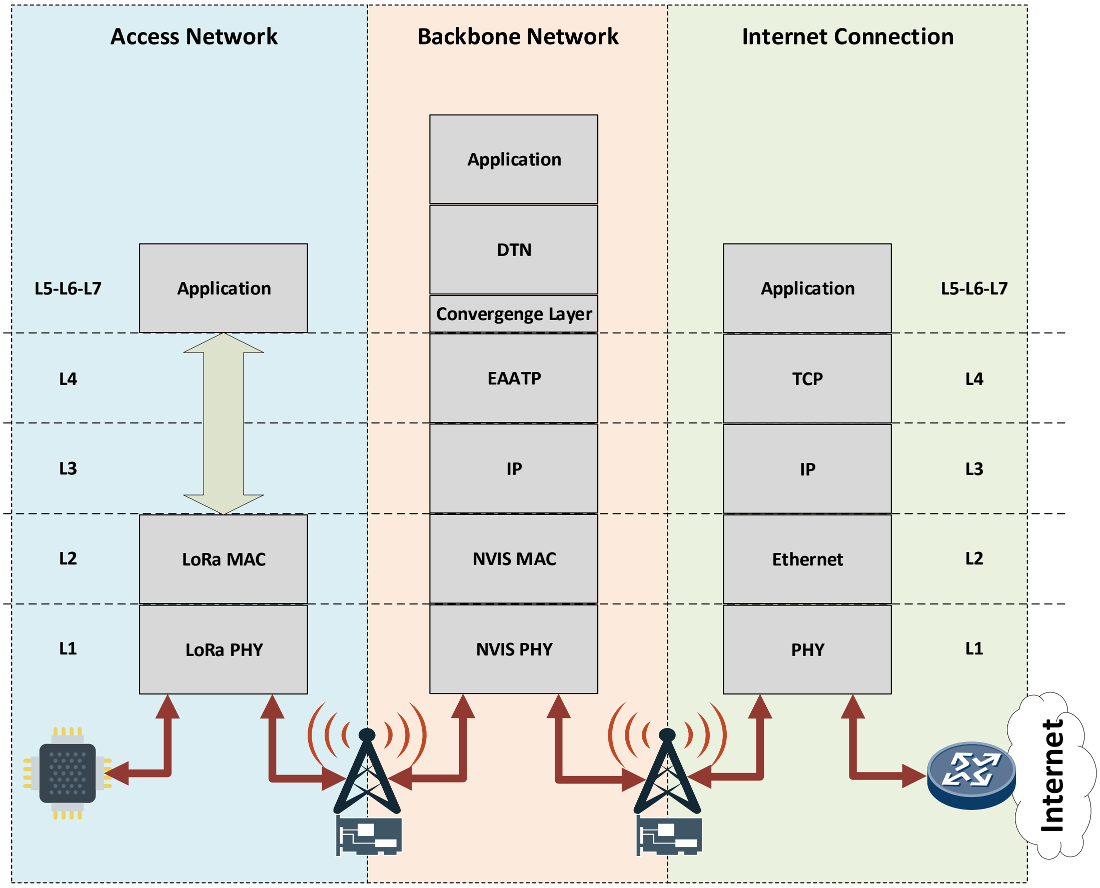


Figure 5. Antarctic IoT network protocol stack.

In the access network, LoRa uses a reduced protocol stack, thus avoiding layers 3 to 6 of the OSI model. The application data is directly encapsulated into the LoRa data link layer. Once data arrives at the NVIS node, the protocol stack introduces all the OSI model layers and adds the DTN layer below the application layer. The DTN layer needs a convergence layer to adapt to the transport protocol below. Figure 4 shows the EAATP as the transport protocol in the backbone network, although we test diverse transport protocols in our simulations, as discussed in Section 5. Finally, when the data arrives at the last NVIS node and must be forwarded through the Internet, the DTN and convergence layers are removed. The common, well-known TCP/IP model is used, given that end-to-end connectivity at this zone is assumed.

4. Trustworthiness Model Specification

In this section, we summarize our trustworthiness model. Further details of the model can be found in [7]. To the best of our knowledge, none of the prior analyzed trustworthiness approaches have tried to include all of the four trustworthiness areas but have instead focused on one or some of them without considering the interdependencies between all the four categories. This could lead to assuming incorrect reasons for a lower trustworthiness level and implementing the wrong countermeasures to improve it. For this reason, we believed it necessary to design our model to measure a system's trustworthiness level, which includes the four categories mentioned above and helps us to anticipate and identify the possible weaknesses of our IoT telemetry system.

We propose a layer-based model to measure the trustworthiness and evaluate a system's performance (in our case, a group of interconnected remote Antarctic wireless sensor networks providing an IoT telemetry service). This model is characterized by (1) two baseline layers (data trustworthiness layer and network trustworthiness layer), (2) two extension layers (social trustworthiness layer and consensus layer) that include optional functionalities, and (3) the interaction between all of them. The data trustworthiness, network trustworthiness, social trustworthiness, and consensus layers can collectively define a system's trustworthiness.

We postulate that each layer is characterized by its definition (scope), how the trustworthiness of that layer is measured (metric), and how the value of this metric can be improved (countermeasures).

4.1. Data Trustworthiness Layer

This layer aims to ascertain the correctness of the source's collected data. We propose the measurement of this layer's trustworthiness with the metric faulty sensing ratio (FSR), defined in Equation (2) as the proportion of false sensed values (FSV) by all nodes and total sensed values (TSV) in a defined period. The lower the FSR, the better the data trustworthiness.

$$FSR = \frac{FSV}{TSV}. \tag{2}$$

Corrective methods (e.g., [51,52]) which try to detect abnormal data (FSV) stored in the source node due to a sensor malfunctioning, a misreading of the sensed data, or erratic writing in the node's memory, can be applied. Additional examples of corrective methods are hashes, checksums, and parity bits, among others (see Figure 6).

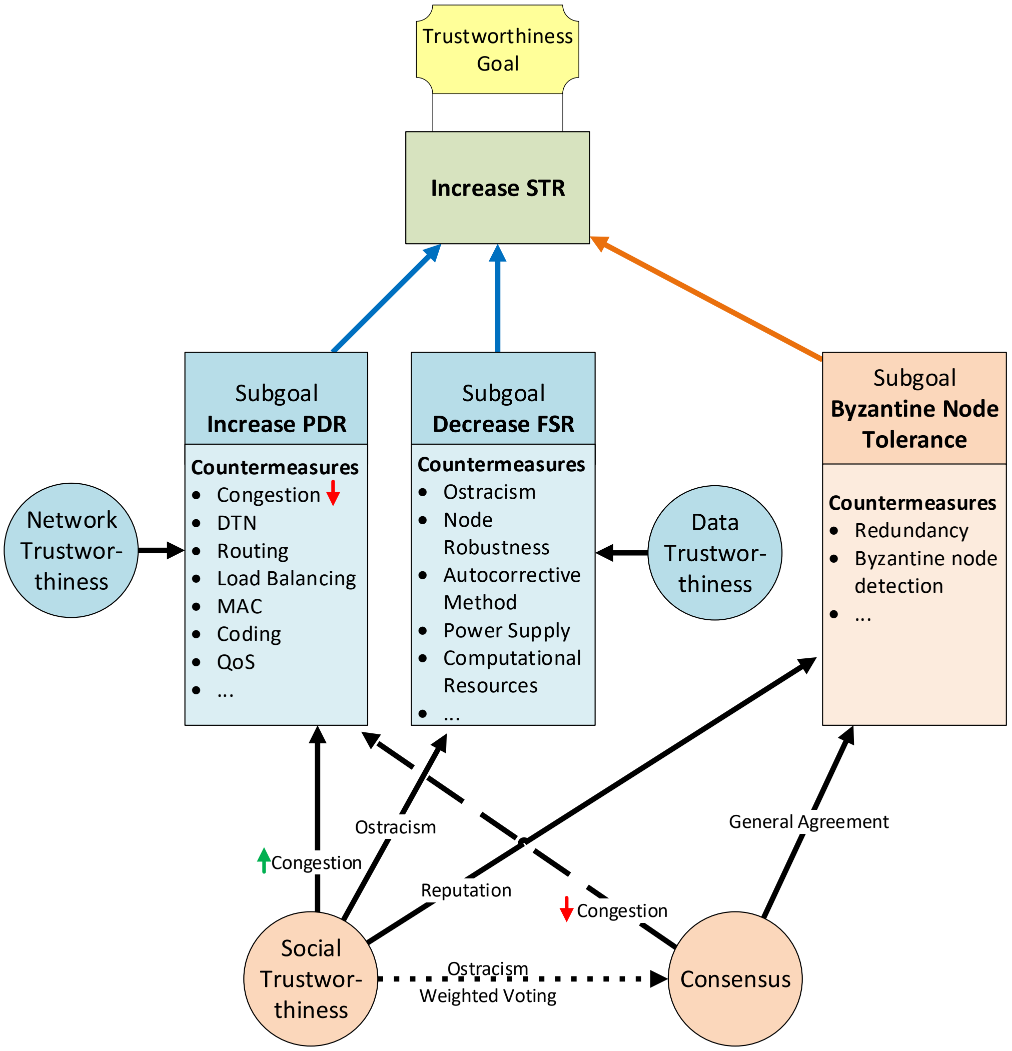


Figure 6. Trustworthiness model goals and countermeasures relationship [7].

4.2. *Network Trustworthiness Layer*

This layer is responsible for assuring that a packet reaches its destination on time and unaltered despite the adversities (e.g., link failure, network congestion). We measure this layer's trustworthiness with the packet delivery ratio (*PDR*), defined in Equation (3) as the quotient between the total number of packets correctly received (*Pr*) by all nodes and the total number of packets sent (*Ps*) by all nodes in the same time slot. The higher the *PDR* is, the better the network's trustworthiness.

$$PDR = \frac{Pr}{Ps}. \tag{3}$$

At the network trustworthiness layer, transmission coding techniques [66] are used to increase the robustness of the transmitted signal. Routing protocols and quality of service (QoS) mechanisms are used to find the best path from a source to a destination by quantifying the quality or performance of each link in the network [55,56]. Congestion control algorithms and other mechanisms of transport protocols [11] can also improve network trustworthiness. In the case of challenge networks, DTN overlay architectures and protocols, such as the Bundle Protocol [8], can also improve network trustworthiness (see Figure 6).

4.3. Social Trustworthiness Layer

This layer is responsible for leveraging the capability to autonomously establish social inter-object relationships to improve the trust between them and the correctness of the collected data. We measure this layer's trustworthiness with the successful transaction rate (*STR*), calculated as the proportion between the number of successful transactions (*ST*) and the total number of transactions (*TT*) in a defined time slot, as stated in Equation (4). A transaction *l* is considered successful when a node *j* expects to obtain some information or data (v) from node *i* before a defined maximum reception time (Trx_{max}) and receives it as expected, thus providing good feedback (f_{ij}^{l} = 1) for that transaction to node *i*. The higher the *STR* is, the better the social trustworthiness.

$$STR = \frac{ST}{TT}. \tag{4}$$

Most solutions tend to use reputational mechanisms to determine which nodes to trust when exchanging information. This reputation is commonly based on the feedback of previous transactions to build an opinion of the node's trustworthiness [59,60,62].

4.4. Consensus Layer

This layer is responsible for reaching a state where all group participants agree on the same response or result. We measure this layer's trustworthiness with the Byzantine node tolerance (*BNT*), defined as the proportion of supported Byzantine nodes (*Nb*) that can participate in the consensus system without affecting the correctness of the general agreement and the total number of nodes (*Nt*) that participate in the consensus system, as defined in Equation (5). A node is considered Byzantine if it experiences a crash or soft fault that incapacitates it to behave as expected or if it does not behave as expected on purpose (malicious node). The higher the *BNT* is, the higher the probability of reaching a correct general agreement (GA).

$$BNT = \frac{Nb}{Nt}. \tag{5}$$

Several mechanisms can be used to reach a decentralized GA that all group nodes consider to be true. Theoretically, if the number of Byzantine nodes is higher than 50% of the total number of participating nodes, none of the consensus mechanism will reach a benevolent agreement [63]. A drawback of these mechanisms is that participating nodes need to exchange a large quantity of messages between them to reach a consensus, which can degrade the performance of low-bandwidth networks.

4.5. Trustworthiness Layers Relationships

Figure 6 synthesizes our trustworthiness model actors. Blue-colored elements form part of our model baseline layers, and orange-colored elements form part of the extension layers. The primary goal is to increase the *STR* to provide better trustworthiness. Three main factors directly help increase the *STR*: (1) Mitigate/tolerate Byzantine errors; (2) decrease the *FSR*; and (3) increase the *PDR*. These factors can be seen as secondary goals that leverage the success of the final goal to provide trustworthiness. Each of these secondary goals can be accomplished by implementing a set of actions or countermeasures. Each of these countermeasures commonly affects only one of the goals. Moreover, two transversal actions impact more than one secondary goal. These transversal actions implement the extension layers of our model: the social trustworthiness layer and the consensus layer.

In Figure 6, continuous-line arrows indicate a positive outcome, discontinuous-line arrows indicate a negative outcome, and dotted-line arrows indicate an uncertain outcome. On the one hand, the use of social trustworthiness can reduce network congestion thanks to the ostracism of nodes with the worst reputation by only sending the values from nodes with the highest reputation to the control center. In addition, social trustworthiness also helps to reduce the *FSR* thanks to the ostracism of bad reputation nodes. It also leverages the mitigation of Byzantine errors because only values from high reputation nodes (leaders) are trusted. On the other hand, implementing a consensus mechanism mitigates Byzantine errors thanks to the general agreements reached by all nodes from a consensus group. Contrarily, the consensus layer can negatively affect the *PDR*, given that it introduces a considerable amount of extra traffic to the network, which could lead to link congestion.

5. Simulation Tests

As mentioned before, the first tests we performed to assess the system's trustworthiness in this use case [7] showed that it was possible to have an STR greater than 0.7 in some circumstances. However, we noticed that the DTN approach of using opportunistic bulk data transfers when the NVIS link becomes available produced network congestion in these periods. On the other hand, we also compared, evaluated, and designed modern transport protocols for heterogeneous LFNs to improve the performance of data transfers over this type of network. Our tests showed that our protocol, the EAATP, maximized throughput and minimized packet losses in LFNs. However, we did not evaluate how the use of these protocols could affect the trustworthiness of a system. Given that the NVIS links in the remote Antarctic WSN use case can be considered an LFN (with a *BDP* greater than 12,500 bytes, from Equation (1)), we thought that using a particular transport protocol might affect the system's trustworthiness. For this reason, we decided to run a second round of tests and check if the hypothesis was correct.

In order to (1) foresee which problems may occur during the Antarctic campaign, (2) decide which transport protocol to use, and (3) build more accurate expectations of the system's performance and outcomes, we applied our trustworthiness model to measure and evaluate them in this use case. For this purpose, the use case scenario was represented and evaluated in the Riverbed Modeler simulator. The first step is the modeling of the different elements that characterize our use case. More details about the modeling of this scenario and its technologies and protocols can be found in [7,11].

Firstly, the backbone network (NVIS) and the access network (LoRa) were modeled separately, characterized as stated in Table 1 following the aforementioned description of the network architecture (please revisit Section 3) and the link availability results from [4] and [65]. On the one hand, LoRa does not experience any availability variation between daytime and nighttime, being fully available if there is LoS between the sensor and the gateway, and with partial availability in the case of no LoS. On the other hand, NVIS is not affected by not having LoS. However, its availability varies hour by hour, depending on the ionosphere state, which is highly correlated to solar activity. During nighttime (5 p.m. to 6 a.m.), the NVIS links are not available, while during daytime (6 a.m. to 5 p.m.), their availability varies between 70% and 100%.

Table 1. Network parameters used to model the scenario.

Parameter	NVIS	LoRa
Transmission Band	4.3 MHz	868 MHz
Channel Bandwidth	2.3 kHz	125 kHz
Channel Bitrate	4.6 kbps	5.47 kbps
Coverage Range	Up to 250 km	Up to 30 km
Daytime Availability (6 a.m.–5 p.m.)	70–100%	100% (LoS), 2–100% (No LoS)
Night Availability (5 p.m.–6 a.m.)	0%	100% (LoS), 2–100% (No LoS)
Maximum Payload Size	242 bytes	140 bytes

Secondly, we modeled the following transport protocols as in our previous work [11]: BBR, Copa, CUBIC, EAATP, Indigo, and Verus. We focused on modern transport protocols that have been proven to perform well [45] and TCP CUBIC, which is the standard transport protocol in most operating systems nowadays. These protocols were modeled according to the results from our previous work in physical testbeds and simulations [5,11] and the Pantheon tests [45].

Thirdly, we needed to model the Byzantine behavior of nodes. As stated in [67], the probability *Pb* of a node having a Byzantine fault is unlikely to be constant over time. The node reliability can be related to the battery charge level by associating the battery discharge with the WSN node aging process. Following the model in [67], we can assume the impact of aging as following a linear form, as defined in Equation (6):

$$Pb(t) = Pb_0 + kt, \tag{6}$$

where Pb_0 is the probability of a node having a Byzantine fault at time $t = 0$, and k is the aging factor. This probability *Pb* increases hour by hour until its battery has practically run out at $t = t_d$, when it experiences a crash fault and $Pb(t_d) = 1$. In the simulations, we tested nine different values of Pb_0 to emulate the use of different corrective methods (see Table 2).

Table 2. Simulation parameters.

Parameter	Value
Number of runs per test	30
Simulation duration	120 h (5 days)
Pb_0	$[1\times10^{-3}, 2\times10^{-3}, 4\times10^{-3}, 8\times10^{-3}, 1\times10^{-2}, 2\times10^{-2}, 4\times10^{-2}, 8\times10^{-2}, 1\times10^{-1}]$
k	5.7×10^{-5}
Transport protocol	[BBR, Copa, CUBIC, EAATP, Indigo, Verus]
Redundancy Mode	[None, Social, Consensus (PBFT)]
Number of NVIS gateways	5
GTN-P clusters per gateway	[8,16,32,64,128,256,512,1024,2048,4096]
GTN-P redundant stations per cluster	[1,2,3,4,5,6,7,8,9,10]

As we are in a simulation environment and we can keep track of all collected, sent, and received values by all nodes, we can compute *FSV* and *ST* by comparing the values that the sensor should have collected with the values that the sensor actually sends and the values that the control center receives, respectively. In a testbed environment with real devices, this would only be possible if previously known ground truth values were sent, in order to compare them with the values received by other nodes.

To model the implementation of the social trustworthiness layer, we used a simplified version of the objective reputational model from [59]. Our use case simplification assumes that all transactions will have the same weight, all nodes have the same computational capability, and the relationship factors between them are equal. Finally, a consensus protocol can be modeled by knowing the background traffic (bps) introduced to the network and the number of Byzantine nodes supported (*Nb*). In our use case, each group of redundant GTN-P stations will run the PBFT algorithm [68]. The background traffic grows exponentially as the number of nodes participating in the consensus (*Nt*) group increases. Moreover, the number of tolerated Byzantine nodes *Nb* is calculated as in Equation (7):

$$Nb = \left\lfloor \frac{Nt-1}{3} \right\rfloor. \tag{7}$$

Our scenario has five NVIS gateways, each providing an independent LoRa coverage area (access network) with its own sensors. For each gateway, there are clusters of sensors measuring the same data. In our test on the field during the campaign, we will deploy eight clusters per gateway. However, in the simulations, we also tested larger numbers of clusters (as seen in Table 2) to assess the goodness of our model and the system's scalability. Each cluster will have a specific number of redundant sensors measuring the same data. From our previous tests, we defined that we would set seven redundant sensors (GTN-P stations) in each cluster in the field deployment, so two Byzantine nodes could be tolerated. Despite this, in the simulation tests, we varied this number from 1 to 10 in order to compare the results with different Byzantine node tolerances (from 0 to 4, following Equation (7)) and assess the system's scalability.

The simulations consider three different operational modes: the standard mode (no redundancy), the redundancy mode with social trustworthiness, and the redundancy mode with consensus. In the standard mode, all the values gathered by every sensor are pushed through the backbone network to the remote control center. On the contrary, in redundancy modes, only one value is forwarded to the control center by each cluster. This value is agreed by cluster members with the social or the consensus mechanism. This fact reduces the amount of traffic that has to pass through the NVIS backbone LFN, although, contrarily, it introduces more overload to the LoRa access network due to the messages that need to be exchanged between cluster members.

All these possibilities add up a total amount of 16,200 different scenarios. Each scenario was simulated for 120 h (5 days) to experience diverse nighttime and daytime cycles, and each test was repeated 30 times to assure results confidence. A summary of the simulation parameters to run our tests is shown in Table 2.

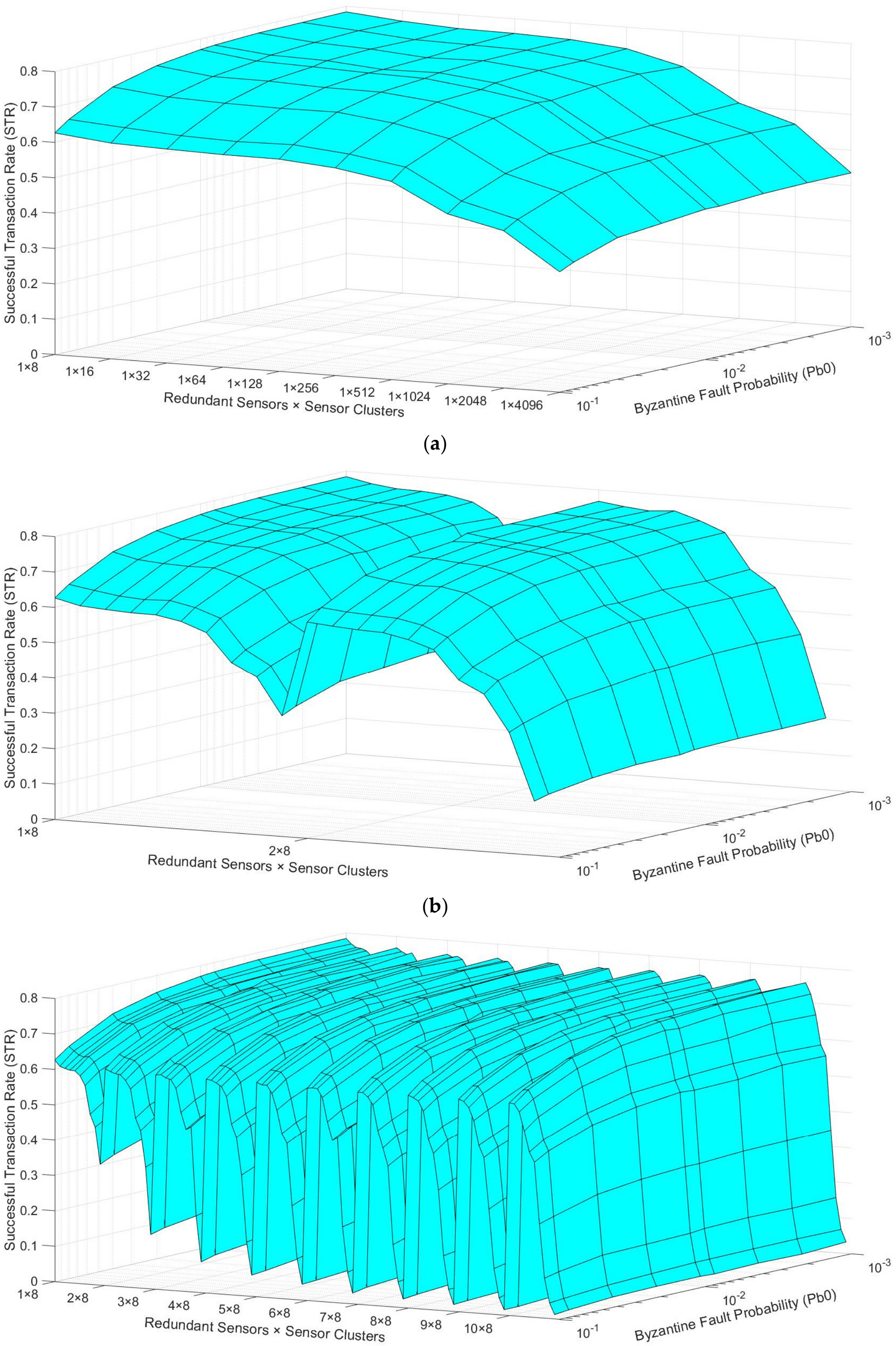
6. Results

After performing all the simulations, the average value of the *STR* was calculated for every set of 30 runs per test. The results obtained have a maximum error deviation of 0.68% with a confidence interval of 99%. Three different operational modes for the telemetry service can be identified: the standard mode, the redundancy mode with social trustworthiness layer, and the redundancy mode with consensus layer. For every mode, an N × M-dimension grid with all the possible combinations of stimulation parameters is formed, where M is the number of different Pb_0 values, and N is the number of different GTN-P node combinations per gateway. For every point in this grid and for every transport protocol, the average value of the trustworthiness *STR* metric is computed. If we link all the *STR* values for every neighboring point in the grid, a mesh with all the *STR* values for each transport protocol is formed. We call this mesh the trustworthiness mesh.

Given that it is complex to understand the trustworthiness mesh results, we first use an example to describe how the results are visualized. If we wanted to represent the results for only one transport protocol, when the number of redundant sensors per cluster is 1, and the number of clusters varies from 8 to 4096 (Table 2, row 9) we could obtain a mesh similar to Figure 7a. The "Byzantine Fault Probability" axis has nine discrete points, corresponding to the nine different Pb_0 values shown in Table 2, row 4. The "Redundant Sensors × Sensor Clusters" axis has 10 discrete points, which are 1×2^N, where N = [3, 4, . . . , 12], according to the values shown in Table 2, row 9. Figure 7a shows the general behavior that *STR* values will follow in the actual results. On the one hand, across the "Byzantine Fault Probability" axis, the *STR* decreases as the Pb_0 increases, given that more values are faulty sensed when the Pb_0 is higher. On the other hand, across the "Redundant Sensors × Sensor Clusters" axis, the *STR* decreases as the number of clusters increases, given that more devices are introduced to the network, provoking more packet losses caused by network congestion.

Similarly, suppose we wanted to show, in a single mesh, the results from the same scenario, but the number of redundant sensors per cluster varied between 1 and 2. In that case, we could obtain a mesh similar to Figure 7b. In this case, the "Byzantine Fault Probability" axis remains the same. In contrast, now the "Redundant Sensors × Sensor Clusters" axis has 20 discrete points, which are $[1 \times 2^N, 2 \times 2^N]$ where N = [3, 4, . . . , 12]. If all the discrete points of this axis were labeled, it could be too congested. For this reason, we only label the beginning of each "redundant sensors" series, i.e., the "1 × 8" and the "2 × 8" discrete points. The same behavior as before is observed, but now the *STR* values recover when we jump from the "1 × 4096" to the "2 × 8" discrete point, given that much fewer nodes are introduced to the network, i.e., fewer packets are dropped due to network congestion.

Analogously, Figure 7c shows the trustworthiness mesh if we wanted to visualize all the results simultaneously, varying the number of redundant sensors from 1 to 10 (Table 2, row 10). In this case, the "Redundant Sensors × Sensor Clusters" axis has 100 discrete points, which are [1 × 2^N, 2 × 2^N, . . . , 10 × 2^N] where N = [3, 4, . . . , 12]. In this case, we observe the same general behavior again. However, now we can also detect that, if we compare the discrete points with the same number of clusters, the *STR* also decreases as the number of redundant sensors per each cluster increases, i.e., more packet losses are caused by network congestion as more nodes are introduced to the network.



(**a**)

(**b**)

(**c**)

Figure 7. Trustworthiness mesh examples: (**a**) only one redundant sensor per cluster; (**b**) one or two redundant sensors per cluster; (**c**) one to ten redundant sensors per cluster.

Figure 8 shows the frontal view of the trustworthiness mesh from Figure 7c. From this view, we can observe how the *STR* varies across the “Redundant Sensors × Sensor Clusters” axis without showing the variance, depending on the Pb_0 of the nodes.

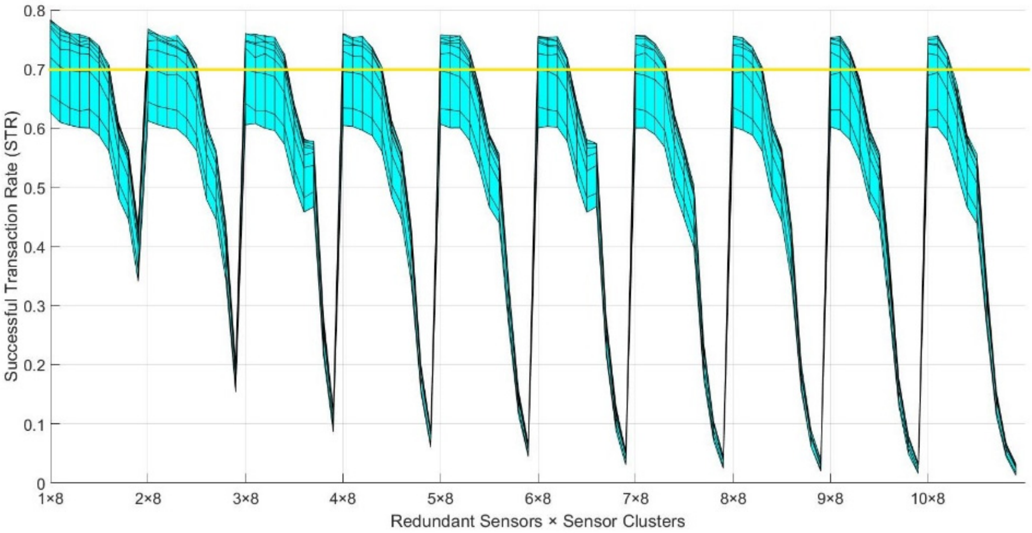


Figure 8. Example of frontal view of the trustworthiness mesh, corresponding to Figure 7c. The yellow line is used to construct the trustworthiness working domain shown in Figure 9.

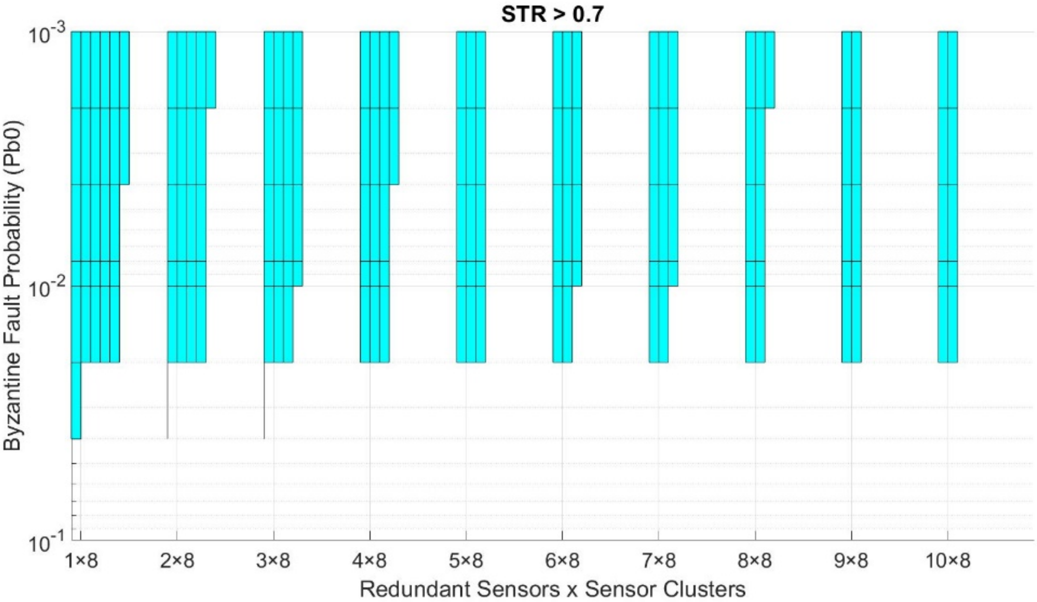


Figure 9. Example of the trustworthiness working domain corresponding to Figures 7c and 8 with a minimum *STR* required of 0.7.

Our model can also be used to visualize the working domain in which to implement our service, given a desired minimum trustworthiness level. As stated before, our use case requires a minimum *STR* of 0.7, so an average of 9 out of 13 sensed values per night reach the control center correctly to meet the objective of [9]. Figure 9 shows the working domain of the example trustworthiness mesh presented in Figures 7c and 8, requiring an *STR* higher than 0.7. For every point in the grid, if no solution provides an *STR* higher than the desired minimum value, the surface for that area is white-colored, meaning we cannot deploy the service with those conditions. On the contrary, if one or more solutions achieve an *STR* higher than the desired minimum value, the surface is painted with the color of the solution with the highest *STR*. This representation is achieved by “cutting” Figure 8 along the yellow line, which represents the minimum STR level that must be achieved. The part of the trustworthiness mesh above the yellow line meets the criteria and is part of the working domain, while the part below does not.

After clarifying how to visualize the data shown in these graphs, we present the tests' results in the following graphs. Figures 10–12 show the trustworthiness mesh for the standard mode, the redundancy mode with social trustworthiness, and the redundancy mode with consensus, respectively. In each graph, the trustworthiness mesh of each transport protocol is superposed with the others in order to visualize which one achieves the highest *STR*. Moreover, Figure 13 shows the trustworthiness working domain of our telemetry service for an *STR* higher than 0.7

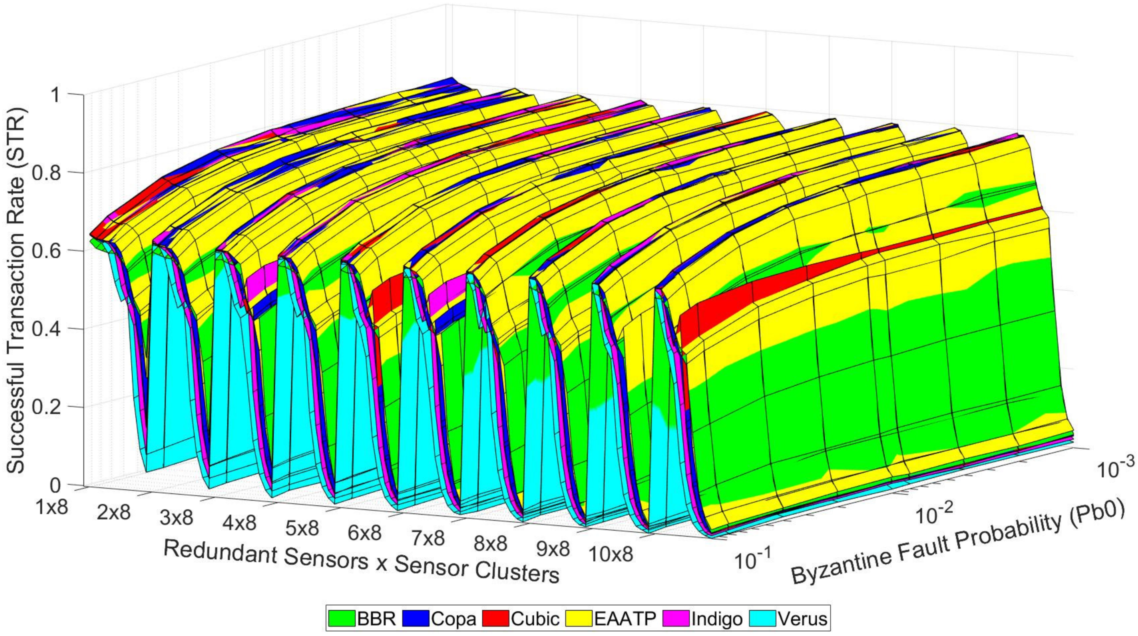


Figure 10. Trustworthiness mesh (standard mode).

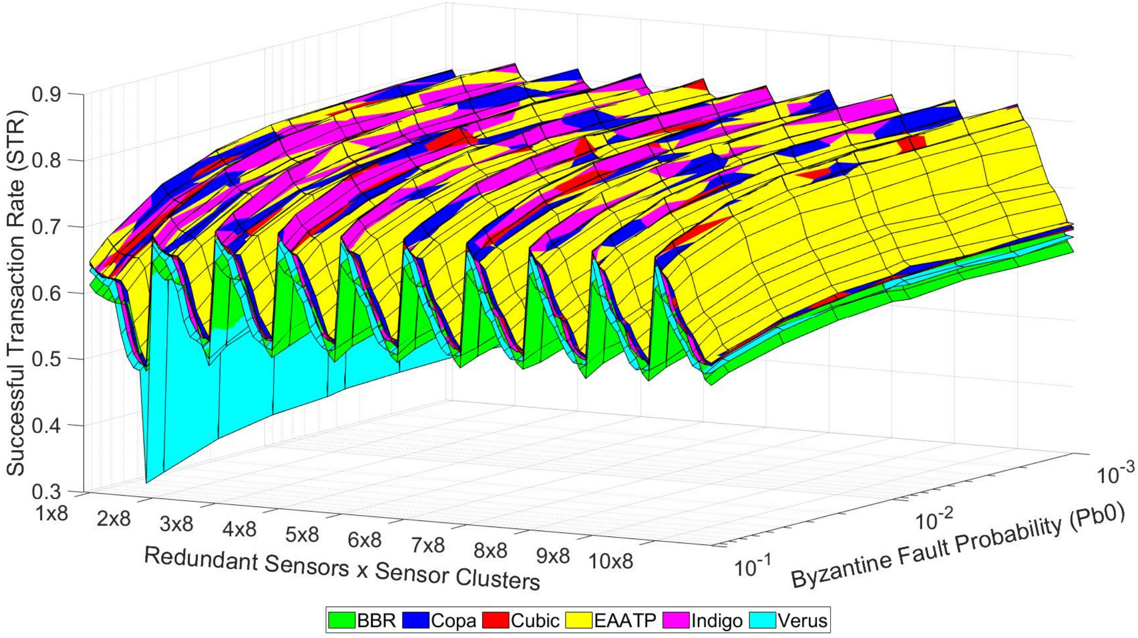


Figure 11. Trustworthiness mesh (social trustworthiness).

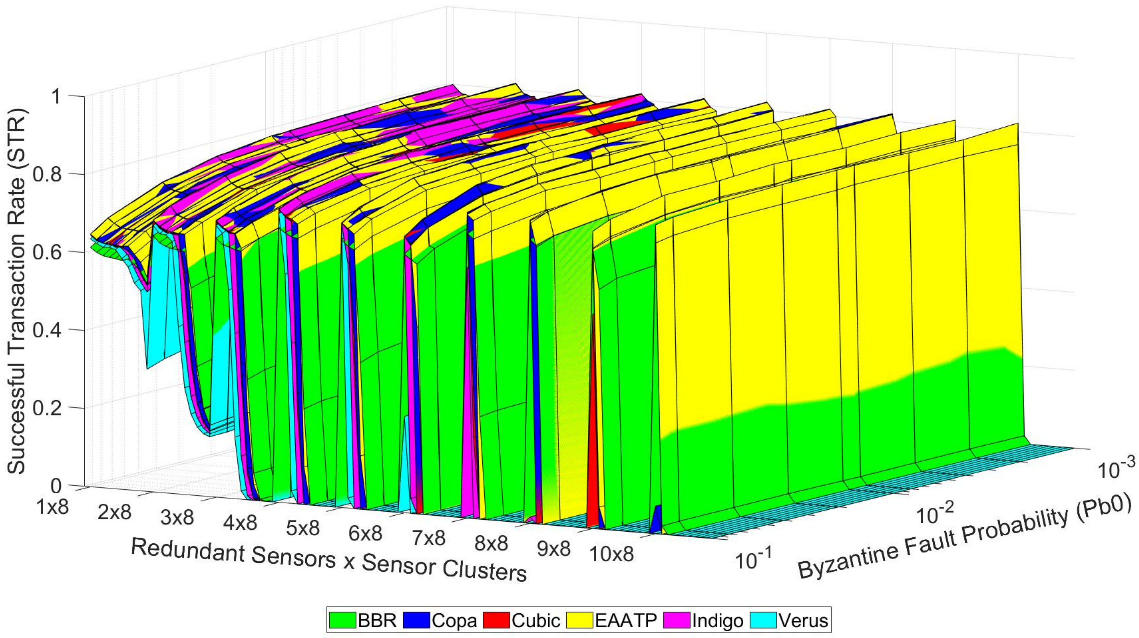


Figure 12. Trustworthiness mesh (consensus).

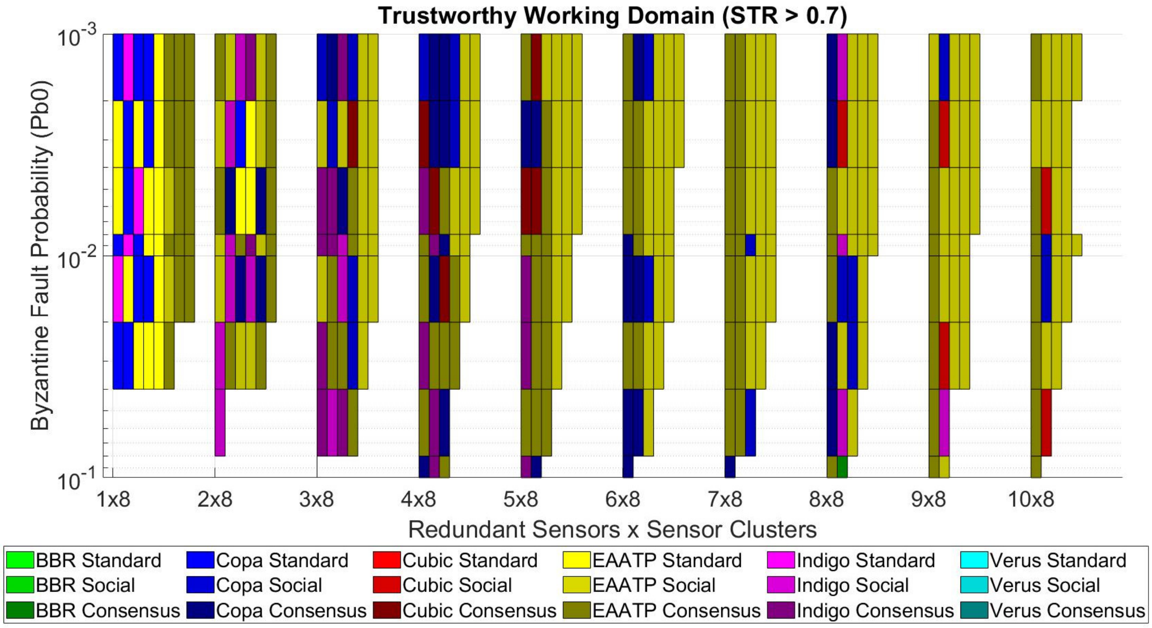


Figure 13. Trustworthiness working domain requiring *STR* > 0.7.

7. Discussion

On the one hand, Figures 10–12 show that the levels of trustworthiness achieved are similar for all the studied transport protocols with low network load (left side of the mesh and cases with fewer sensor clusters). This fact seems reasonable because we already selected the most suitable and top-performance transport protocols to perform our tests, discarding those that do not adapt well in LFNs. We believe that if other transport protocols less suitable for this kind of network had been tested, the difference in the results would be more evident. However, (1) the levels of BBR and Verus are slightly lower than their competitors, and (2) Copa, Indigo, and EAATP share the highest *STR* values in the case of low network load, although the predominance of EAATP grows as the network load increases (the yellow mesh is more visible than the others).

On the other hand, we can also see that the redundancy mode with social trustworthiness (Figure 11) is the most robust scenario, given that its *STR* decrease in high-load situations is less accentuated compared to the other cases (Figures 10 and 12), always maintaining *STR* values greater than 0.5. Furthermore, it is confirmed that, in general, as the probability of a node experiencing a Byzantine error decreases, the achieved *STR* values accordingly increase.

From the trustworthiness working domain (Figure 13), we can see the aforementioned predominance of the EAATP. As mentioned in Section 5, the scenario intended to deploy in the next Antarctic campaign was the "7 redundant sensors × 8 sensor clusters". Concretely, we can check that this case reaches the *STR* requirement of 0.7 for any Pb_0 value.

If we focus on this case, in Figure 13, we can see that the EAATP is the most trustworthy protocol except for the $Pb_0 = 1 \times 10^{-1}$ and $Pb_0 = 8 \times 10^{-2}$ cases, in which Copa performs better. Table 3 shows, in detail, the results for the "7 redundant sensors × 8 clusters" case. For each protocol and each Pb_0, we show the best *STR* achieved from the three possible operational modes (standard, social, and consensus). Although Copa, CUBIC, and EAATP have similar results, the latter can outperform Copa and CUBIC between 0.1% and 0.5% better in terms of *STR* in most cases, and also outperforms up to 7% more than its other competitors. These results confirm our hypothesis, i.e., using a particular transport protocol can directly affect the system's trustworthiness in our use case.

Table 3. Best *STR* achieved by each transport protocol in the "7 redundant sensors × 8 clusters" case. The best *STR* for each Pb_0 is highlighted in bold.

Pb_0	BBR	Copa	CUBIC	EAATP	Indigo	Verus
1×10^{-3}	0.767	0.818	0.817	**0.818**	0.814	0.801
2×10^{-3}	0.767	0.814	0.814	**0.819**	0.817	0.802
4×10^{-3}	0.772	0.819	0.819	**0.819**	0.811	0.795
8×10^{-3}	0.768	0.816	0.814	**0.817**	0.807	0.797
1×10^{-2}	0.767	0.818	0.817	**0.820**	0.805	0.794
2×10^{-2}	0.767	0.814	0.813	**0.815**	0.799	0.782
4×10^{-2}	0.762	0.811	0.809	**0.813**	0.777	0.765
8×10^{-2}	0.750	**0.796**	0.795	0.794	0.741	0.727
1×10^{-1}	0.731	**0.785**	0.781	0.779	0.724	0.710

We believe that the EAATP's superior trustworthiness is caused by the fact that it incorporates a fairness mechanism to share the network bandwidth, which reduces congestion and packet losses. Moreover, EAATP's congestion control tries to occupy the entire network bandwidth rapidly, and its mechanism to differentiate between random channel losses and congestion losses optimizes its achieved throughput in heavy congestion situations. These features give the EAATP a competitive advantage in terms of performance in our use case, where the DTN opportunistic scheme we use to send accumulated data during the night as a bulk data transfer congests the network.

For these reasons, we decided to use the EAATP as the backbone network transport protocol for our IoT telemetry service that will be deployed in the field during the next Antarctic campaign. Moreover, we can identify which of the three modes best suits the different scenarios which may arise. In general, the standard mode obtains the highest *STR* values when there is no redundancy (1 × N zone). If redundancy is applied, the consensus solution shows the highest levels of trustworthiness in most cases with a low network load. However, as mentioned before, when the network load increases, the social trustworthiness solution is more robust, achieving the highest *STR* values for those cases.

Finally, we also propose that the scenario to be deployed is reconsidered. In the "7 redundant sensors × 8 clusters" scenario, each gateway has 56 sensors connected, while only eight different values are sensed, which might be an excessive low efficiency. We propose to switch to the "5 redundant sensors × 16 clusters". In this case, increasing the number of sensors by 43% (80 sensors per gateway) results in increasing the number of different sensed values by 100% (16 values). Table 4 shows the detailed results for this use case. If we compare the results from Tables 3 and 4, the latter case achieves slightly worse *STR* values (which seems evident because we decrease the redundancy and increase the total number of sensors). However, Copa, CUBIC, EAATP, and Indigo still meet the required *STR* level of 0.7, providing trustworthiness to the service. In this case, we can also confirm the predominance of the EAATP, being the protocol with the highest *STR* in five of the nine Pb_0 cases, while Copa and CUBIC achieve the highest *STR* in two cases each. Moreover, EAATP outperforms its competitors by up to 5.1%, while in the cases where another protocol outperforms the EAATP, it is only by 0.3% at most. Thus, we believe that the EAATP would also be the most suitable transport protocol to be used in this case.

Table 4. Best *STR* achieved by each transport protocol in the "5 redundant sensors × 16 clusters" case. The best *STR* for each Pb_0 is highlighted in bold.

Pb_0	BBR	Copa	CUBIC	EAATP	Indigo	Verus
1×10^{-3}	0.757	0.797	**0.798**	0.797	0.795	0.783
2×10^{-3}	0.752	**0.799**	0.799	0.798	0.796	0.783
4×10^{-3}	0.748	0.796	**0.798**	0.797	0.792	0.777
8×10^{-3}	0.75	0.794	0.795	**0.801**	0.792	0.775
1×10^{-2}	0.749	0.793	0.795	**0.796**	0.786	0.775
2×10^{-2}	0.74	0.79	0.787	**0.792**	0.779	0.764
4×10^{-2}	0.73	0.776	0.781	**0.781**	0.757	0.747
8×10^{-2}	0.698	**0.74**	0.736	0.737	0.727	0.706
1×10^{-1}	0.672	0.717	0.714	**0.718**	0.704	0.692

8. Conclusions

This paper analyzes the applicability of the deployment of a remote WSN for the Antarctic region using NVIS technology and the provision of an IoT telemetry service for permafrost studies. This service will be deployed during the 2021–2022 Antarctic campaign of the SHETLAND-NET project. This work focuses on analyzing and comparing transport protocols' trustworthiness in our remote WSN with DTN use case, which uses LoRa at the access network and NVIS links at the backbone network. Due to certain ionospheric characteristics, NVIS links do not work correctly at night. For this reason, values sensed at night are sent opportunistically to the control center as bulk data when the NVIS channel becomes available, which might cause network congestion. In this situation, the choice to use a particular transport protocol might affect the overall system's trustworthiness. In order to study the viability of the service to be implemented before its deployment in the field during the Antarctic campaign and in an attempt to compare the performance of various transport protocols, we use our model to measure and evaluate the trustworthiness of the proposed system. This trustworthiness model consists of four layers that can affect the *STR* trustworthiness metric.

Three operational modes and six transport protocols were analyzed under different conditions using the Riverbed Modeler simulator. The results show a predominance of the EAATP as the most trustworthy transport protocol, while BBR and Verus have the worst trustworthiness. Adding redundancy to the measured values with multiple sensors and applying a social reputational mechanism improves the robustness of the system's trustworthiness, reaching higher *STR* values and never dropping below 0.5, even in high-load scenarios. On the contrary, a consensus mechanism improves the system's trustworthiness if the number of sensors is kept at a low value.

The research group decided to deploy eight clusters for each NVIS gateway and seven GTN-P redundant stations per cluster in the Antarctic campaign. The collected results confirm that this scenario achieves the minimum *STR* required of 0.7, resulting in a feasible deployment. In this case, the results show that the EAATP can outperform up to 7% of the other analyzed transport protocols in terms of trustworthiness (*STR*). However, we recommend sacrificing some redundancy (i.e., trustworthiness) and increasing the number of different sensed values, implementing the scenario with 16 clusters and five GTN-P redundant stations. In this case, although slightly worse *STR* values are achieved, the requirement of achieving at least an *STR* of 0.7 is met, while more data can be remotely monitored from the control center. The EAATP is also the most trustworthy transport protocol in this case, outperforming its competitors by up to 5.1%. Thus, the research group has decided to use the EAATP as the transport protocol for the offered telemetry service.

Future work aims to (1) study the viability of using the same network architecture to deploy an integrated sensing and communication system (ISAC) capable of using ionosondes as data transmission signals through NVIS; and (2) analyze the implementation of other DTN architectures and protocols to improve the trustworthiness of the entire system in situations when the availability of the NVIS link is not previously known (daytime).

Author Contributions: Conceptualization, A.B., A.M. and A.Z.; methodology, A.M. and A.Z.; software, A.M.; validation, A.B., A.M. and A.Z.; formal analysis, A.M.; investigation, A.M.; resources, A.B., A.M. and A.Z.; data curation, A.M.; writing—original draft preparation, A.M.; writing—review and editing, A.B., A.M. and A.Z.; visualization, A.M.; supervision, A.Z.; project administration, A.Z.; funding acquisition, A.B., A.M. and A.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the "Secretaria d'Universitats i Recerca del Departament d'Empresa i Coneixement de la Generalitat de Catalunya", the European Union (EU) and the European Social Fund (ESF) [2021 FI_B1 00175]. The research was also funded by the "Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR)" of "Generalitat de Catalunya" (grant identification "2017 SGR 977"). This work also received funding from the Spanish Ministry on Science, Innovation and University, the Investigation State Agency and the European Regional Development Fund (ERDF) under the grant number RTI2018-097066-B-I00 (MCIU/AEI/FEDER, UE) for the project "NVIS Sensor Network For The South Shetland Islands Archipelago" (SHETLAND-NET).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is contained within the article.

Acknowledgments: The authors would like to thank "La Salle-URL" (Universitat Ramon Llull) for their encouragement and assistance.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AATP	Adaptive and Aggressive Transport Protocol
BDP	Bandwidth Delay Product
BIC-TCP	Binary Increase Control TCP
BNT	Byzantine Node Tolerance
BP	Bundle Protocol
bps	Bits per second
BW	Bandwidth
CPS	Cyber Physical System
DTN	Delay Tolerant Network
DUACK	Duplicated Acknowledgment
EAATP	Enhanced AATP
FSR	Faulty Sensing Ratio
FSV	False Sensed Values
GA	General Agreement
GTN-P	Ground Terrestrial Network-Permafrost
HF	High Frequency

H-TCP	High-Speed TCP
ICN	Information-Centric Networking
IoT	Internet of Things
ISAC	Integrated Sensing and Communication System
JSCTP	Jitter Stream Control Transmission Protocol
LFN	Long Fat Network
M2M	Machine to Machine
NVIS	Near Vertical Incidence Skywave
OSI	Open Systems Interconnection
PBFT	Practical Byzantine Fault Tolerance
PCC	Performance-oriented Congestion Control
PDR	Packet Delivery Ratio
QoS	Quality of Service
RFC	Request For Comments
RTT	Round-Trip Time
SACK	Selective Acknowledgment
SIoT	Social Internet of Things
S-TCP	Scalable TCP
ST	Successful Transactions
STR	Successful Transaction Rate
TCP	Transmission Control Protocol
TSV	Total Sensed Values
TT	Total Transactions
WSN	Wireless Sensor Network

References

1. Kennicutt, M.C.; Kim, Y.D.; Rogan-Finnemore, M.; Anandakrishnan, S.; Chown, S.L.; Colwell, S.; Cowan, D.; Escutia, C.; Frenot, Y.; Hall, J.; et al. Delivering 21st century Antarctic and Southern Ocean science. *Antarct. Sci.* **2016**, *28*, 407–423. [CrossRef]
2. Alsina-Pagès, R.M.; Hervás, M.; Orga, F.; Pijoan, J.L.; Badia, D.; Altadill, D. Physical layer definition for a long-haul HF antarctica to Spain radio link. *Remote Sens.* **2016**, *8*, 380. [CrossRef]
3. Porte, J.; Maso, J.M.; Pijoan, J.L.; Badia, D. Sensing System for Remote Areas in Antarctica. *Radio Sci.* **2020**, *55*, 1–12. [CrossRef]
4. Male, J.; Porte, J.; Gonzalez, T.; Maso, J.M.; Pijoan, J.L.; Badia, D. Analysis of the Ordinary and Extraordinary Ionospheric Modes for NVIS Digital Communications Channels. *Sensors* **2021**, *21*, 2210. [CrossRef] [PubMed]
5. Briones, A.; Mallorquí, A.; Zaballos, A.; de Pozuelo, R.M. Adaptive and aggressive transport protocol to provide QoS in cloud data exchange over Long Fat Networks. *Futur. Gener. Comput. Syst.* **2021**, *115*, 34–44. [CrossRef]
6. Gonzalez, T.; Porte, J.; Pijoan, J.L.; Badia, D.; Male, J.; Navarro, J.; Maso, J.M. SC-FDE Layer for Sensor Networks in Remote Areas Using NVIS Communications. *Electronics* **2021**, *10*, 1636. [CrossRef]
7. Mallorquí, A.; Zaballos, A. A heterogeneous layer-based trustworthiness model for long backhaul nvis challenging networks and an iot telemetry service for antarctica. *Sensors* **2021**, *21*, 3446. [CrossRef]
8. Bounsiar, S.; Benhamida, F.Z.; Henni, A.; de Ipiña, D.L.; Mansilla, D.C. How to Enable Delay Tolerant Network Solutions for Internet of Things: From Taxonomy to Open Challenges. *Proceedings* **2019**, *31*, 24. [CrossRef]
9. de Pablo Hernández, M.Á.; Jiménez, J.J.; Ramos, M.; Prieto, M.; Molina, A.; Vieira, G.; Hidalgo, M.A.; Fernández, S.; Recondo, C.; Calleja, J.F.; et al. Frozen ground and snow cover monitoring in livingston and deception islands, antarctica: Preliminary results of the 2015–2019 PERMASNOW project. *Geogr. Res. Lett.* **2020**, *46*, 187–222. [CrossRef]
10. Location Map of Low Island in the South Shetland Islands. Available online: https://en.wikipedia.org/wiki/Low_Island_(South_Shetland_Islands)#/media/File:Low-Island-location-map.png (accessed on 2 September 2021).
11. Briones, A.; Mallorquí, A.; Zaballos, A.; de Pozuelo, R.M. Wireless loss detection over fairly shared heterogeneous long fat networks. *Electronics* **2021**, *10*, 987. [CrossRef]
12. Rodrigues, J.J.P.C. (Ed.) *Advances in Delay-Tolerant Networks (DTNs): Architecture and Enhanced Performance*, 2nd ed.; Woodhead Publishing: Sawston, UK, 2020.
13. Burleigh, S.; Hooke, A.; Torgerson, L.; Fall, K.; Cerf, V.; Durst, B.; Scott, K.; Weiss, H. Delay-tolerant networking: An approach to interplanetary internet. *IEEE Commun. Mag.* **2003**, *41*, 128–136. [CrossRef]
14. Partan, J.; Kurose, J.; Levine, B.N. A Survey of Practical Issues in Underwater Networks. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* **2007**, *11*, 23–33. [CrossRef]
15. Tovar, A.; Friesen, T.; Ferens, K.; McLeod, B. A DTN wireless sensor network for wildlife habitat monitoring. In Proceedings of the Canadian Conference on Electrical and Computer Engineering, Calgary, AB, Canada, 2–5 May 2010; pp. 1–5. [CrossRef]
16. Matsuzaki, R.; Ebara, H.; Muranaka, N. Rescue support system with DTN for earthquake disasters. *IEICE Trans. Commun.* **2015**, *E98B*, 1832–1847. [CrossRef]

17. Soares, V.N.G.J.; Farahmand, F.; Rodrigues, J.J.P.C. A layered architecture for Vehicular Delay-Tolerant Networks. In Proceedings of the 2009 IEEE Symposium on Computers and Communications, Sousse, Tunísia, 5–8 July 2009; pp. 122–127. [CrossRef]
18. Scott, K.L.; Burleigh, S. Bundle Protocol Specification. Available online: https://tools.ietf.org/html/rfc5050 (accessed on 20 September 2021).
19. Penning, A.; Baumgärtner, L.; Höchst, J.; Sterz, A.; Mezini, M.; Freisleben, B. DTN7: An Open-Source Disruption-Tolerant Networking Implementation of Bundle Protocol 7. In Proceedings of the International Conference on Ad-Hoc Networks and Wireless, LNCS, Luxembourg, 1–3 October 2019; Springer: Basel, Switzerland; Volume 11803, pp. 196–209.
20. Schildt, S.; Morgenroth, J.; Pöttner, W.B.; Wolf, L. IBR-DTN: A lightweight, modular and highly portable Bundle Protocol implementation. *Electron. Commun. EASST* **2011**, *37*. [CrossRef]
21. Von Zengen, G.; Büsching, F.; Pöttner, W.-B.; Wolf, L. An Overview of µDTN: Unifying DTNs and WSNs. In Proceedings of the 11th GI/ITG KuVS Fachgespräch Drahtlose Sensornetze (FGSN), Darmstadt, Germany, 13 September 2012; pp. 1–4.
22. Al-Turjman, F.M.; Al-Fagih, A.E.; Alsalih, W.M.; Hassanein, H.S. A delay-tolerant framework for integrated RSNs in IoT. *Comput. Commun.* **2013**, *36*, 998–1010. [CrossRef]
23. Guo, Z.; Wang, B.; Cui, J.H. Generic prediction assisted single-copy routing in underwater delay tolerant sensor networks. *Ad Hoc Netw.* **2013**, *11*, 1136–1149. [CrossRef]
24. Wong, K.S.; Wan, T.C. Reliable Multicast Disruption Tolerant Networking: Conceptual Implementation Using Message Ferry. In Proceedings of the IEEE Region 10 Annual International Conference, Proceedings/TENCON, Penang, Malaysia, 5–8 November 2017; pp. 1817–1822.
25. Mao, Y.; Zhou, C.; Ling, Y.; Lloret, J. An optimized probabilistic delay tolerant network (DTN) routing protocol based on scheduling mechanism for internet of things (IoT). *Sensors* **2019**, *19*, 243. [CrossRef]
26. Guo, B.; Zhang, D.; Wang, Z.; Yu, Z.; Zhou, X. Opportunistic IoT: Exploring the harmonious interaction between human and the internet of things. *J. Netw. Comput. Appl.* **2013**, *36*, 1531–1539. [CrossRef]
27. Xu, Y.; Mahendran, V.; Radhakrishnan, S. Internet of Hybrid Opportunistic Things: A Novel Framework for Interconnecting IoTs and DTNs. In Proceedings of the 2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), San Francisco, CA, USA, 10–14 April 2016; pp. 1067–1068. [CrossRef]
28. Elmangoush, A.; Corici, A.; Catalan, M.; Steinke, R.; Magedanz, T.; Oller, J. Interconnecting Standard M2M Platforms to Delay Tolerant Networks. In Proceedings of the Proceedings-2014 International Conference on Future Internet of Things and Cloud, FiCloud 2014, Barcelona, Spain, 27–29 August 2014; pp. 258–263.
29. Sathiaseelan, A.; Trossen, D.; Komnios, I.; Ott, J.; Crowcroft, J. *Information Centric Delay Tolerant Networking: An Internet Architecture for the Challenged*; University of Cambridge: Cambridge, UK, 2013.
30. Manzoni, P.; Hernández-Orallo, E.; Calafate, C.T.; Cano, J.C. A Proposal for a Publish/Subscribe, Disruption Tolerant Content Island for Fog Computing. In Proceedings of the SMARTOBJECTS 2017-Proceedings of the 3rd Workshop on Experiences with the Design and Implementation of Smart Objects, Co-Located with MobiCom 2017, Snowbird, UT, USA, 16 October 2017; pp. 47–52.
31. Kazmi, M.; Shamim, A.; Wahab, N.; Anwar, F. Comparison of TCP Tahoe, Reno, New Reno, Sack and Vegas in IP and MPLS Networks under Constant Bit Rate Traffic. In Proceedings of the International Conference on Advanced Computational Technology and Creative Media (ICACTCM), Pattaya, Thailand, 14–15 August 2014; pp. 33–38.
32. Kelly, T. Scalable TCP: Improving performance in highspeed wide area networks. *ACM SIGCOMM Comput. Commun. Rev.* **2003**, *33*, 83–91. [CrossRef]
33. Jin, C.; Wei, D.X.; Low, S.H. FAST TCP: Motivation, Architecture, Algorithms, Performance. *IEEE/ACM Trans. Netw.* **2006**, *14*, 1246–1259. [CrossRef]
34. Leith, D.; Shorten, R. H-TCP Protocol for High-Speed Long-Distance Networks. In Proceedings of the PFLDnet, Argonne, IL, USA, 16–17 February 2004.
35. Xu, L.; Harfoush, K.; Rhee, I. Binary Increase Congestion Control (BIC) for Fast Long-Distance Networks. In Proceedings of the IEEE INFOCOM 2004, Hong Kong, China, 7–11 March 2004; Volume 4, pp. 2514–2524.
36. Ha, S.; Rhee, I.; Xu, L. Cubic: A new TCP-friendly high-speed TCP variant. *ACM SIGOPS Oper. Syst. Rev.* **2008**, *42*, 64–74. [CrossRef]
37. Dong, M.; Li, Q.; Zarchy, D.; Godfrey, P.B.; Schapira, M. PCC: Re-Architecting Congestion Control for Consistent High Performance. In Proceedings of the 12th USENIX Symposium on Networked Systems Design and Implementation (NSDI'15), Oakland, CA, USA, 4–6 May 2015; pp. 395–408.
38. Fu, C.P.; Liew, S.C. TCP Veno: TCP Enhancement for Transmission Over Wireless Access Networks. *IEEE J. Sel. Areas Commun.* **2003**, *21*, 216–228.
39. Grieco, L.A.; Mascolo, S. Performance evaluation and comparison of Westwood+, New Reno, and Vegas TCP congestion control. *ACM SIGCOMM Comput. Commun. Rev.* **2004**, *34*, 25–38. [CrossRef]
40. Kanagarathinam, M.R.; Singh, S.; Sandeep, I.; Roy, A.; Saxena, N. D-TCP: Dynamic TCP Congestion Control Algorithm for next Generation Mobile Networks. In Proceedings of the 15th IEEE Annual Consumer Communications & Networking Conference (CCNC), Las Vegas, NV, USA, 12–15 January 2018; pp. 1–6. [CrossRef]
41. Wu, E.H.K.; Huang, Y.U.C.; Chang, G.K. EJTCP: Enhanced Jitter-based TCP for Wireless Broadband Networks. *J. Inf. Sci. Eng.* **2007**, *23*, 1663–1679.

42. Chen, J.M.; Chu, C.H.; Wu, E.H.K.; Tsai, M.F.; Wang, J.R. Improving SCTP performance by jitter-based congestion control over wired-wireless networks. *Eurasip J. Wirel. Commun. Netw.* **2011**, *2011*, 103027. [CrossRef]
43. Cardwell, N.; Cheng, Y.; Gunn, C.S.; Yeganeh, S.H.; Jacobson, V. BBR: Congestion-Based Congestion Control. *Queue* **2016**, *14*, 20–53. [CrossRef]
44. Arun, V.; Balakrishnan, H.; Csail, M.I.T.; Design, S.; Nsdi, I. Copa: Practical Delay-Based Congestion Control for the Internet. In Proceedings of the 15th USENIX Symposium on Networked Systems Design and Implementation (NSDI'18), Renton, WA, USA, 9–11 April 2018; pp. 329–342.
45. Yan, F.Y.; Ma, J.; Hill, G.D.; Raghavan, D.; Wahby, R.S.; Levis, P.; Winstein, K. Pantheon: The Training Ground for Internet Congestion-Control Research. In Proceedings of the 2018 USENIX Annual Technical Conference, USENIX ATC 2018, Boston, MA, USA, 11–13 July 2020; pp. 731–743.
46. Zaki, Y.; Pötsch, T.; Chen, J.; Subramanian, L.; Görg, C. Adaptive Congestion Control for Unpredictable Cellular Networks. *Comput. Commun. Rev.* **2015**, *45*, 509–522. [CrossRef]
47. Crawford, M.; Liongosary, E. The Industrial Internet of Things Consortium. *IIC J. Innov.* **2018**, *9*, 1–141.
48. Junior, F.M.R.; Kamienski, C.A. A Survey on Trustworthiness for the Internet of Things. *IEEE Access* **2021**, *9*, 42493–42514. [CrossRef]
49. Labib, N.S.; Brust, M.R.; Danoy, G.; Bouvry, P. Trustworthiness in IoT-A Standards Gap Analysis on Security, Data Protection and Privacy. In Proceedings of the 2019 IEEE Conference on Standards for Communications and Networking (CSCN), Granada, Spain, 28–30 October 2019; pp. 1–7. [CrossRef]
50. Haron, N.; Jaafar, J.; Aziz, I.A.; Hassan, M.H.; Shapiai, M.I. Data Trustworthiness in Internet of Things: A Taxonomy and Future Directions. In Proceedings of the 2017 IEEE Conference on Big Data and Analytics (ICBDA), Kuching, Malaysia, 16–17 November 2017; pp. 25–30.
51. Zhang, G.; Li, R. Fog computing architecture-based data acquisition for WSN applications. *China Commun.* **2017**, *14*, 69–81. [CrossRef]
52. Fantacci, R.; Nizzi, F.; Pecorella, T.; Pierucci, L.; Roveri, M. False Data Detection for Fog and Internet of Things Networks. *Sensors* **2019**, *19*, 4235. [CrossRef]
53. Hassan, M.M.; Gumaei, A.; Huda, S.; Almogren, A. Increasing the Trustworthiness in the Industrial IoT Networks through a Reliable Cyberattack Detection Model. *IEEE Trans. Ind. Inform.* **2020**, *16*, 6154–6162. [CrossRef]
54. Bioglio, V.; Condo, C.; Land, I. Design of Polar Codes in 5G New Radio. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 29–40. [CrossRef]
55. Alahari, H.P.; Yalavarthi, S.B. A Survey on Network Routing Protocols in Internet of Things (IOT). *Int. J. Comput. Appl.* **2017**, *160*, 18–22. [CrossRef]
56. Li, J.; Li, X.; Cheng, X.; Yuan, J.; Zhang, R. A trustworthiness-enhanced reliable forwarding scheme in mobile Internet of Things. *J. Netw. Comput. Appl.* **2019**, *140*, 40–53. [CrossRef]
57. Atzori, L.; Iera, A.; Morabito, G. SIoT: Giving a Social Structure to the Internet of Things. *IEEE Commun. Lett.* **2011**, *15*, 1193–1195. [CrossRef]
58. Caballero, V.; Vernet, D.; Zaballos, A. Social Internet of Energy-A New Paradigm for Demand Side Management. *IEEE Internet Things J.* **2019**, *6*, 9853–9867. [CrossRef]
59. Nitti, M.; Girau, R.; Atzori, L. Trustworthiness Management in the Social Internet of Things. *IEEE Trans. Knowl. Data Eng.* **2013**, *26*, 1253–1266. [CrossRef]
60. Marche, C.; Nitti, M. Trust-Related Attacks and Their Detection: A Trust Management Model for the Social IoT. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 3297–3308. [CrossRef]
61. Lin, Z.; Dong, L. Clarifying Trust in Social Internet of Things. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 234–248. [CrossRef]
62. Azad, M.A.; Bag, S.; Hao, F.; Shalaginov, A. Decentralized Self-Enforcing Trust Management System for Social Internet of Things. *IEEE Internet Things J.* **2020**, *7*, 2690–2703. [CrossRef]
63. Bodkhe, U.; Mehta, D.; Tanwar, S.; Bhattacharya, P.; Singh, P.K.; Hong, W.C. A survey on decentralized consensus mechanisms for cyber physical systems. *IEEE Access* **2020**, *8*, 54371–54401. [CrossRef]
64. All about Moteino | LowPowerLab. Available online: https://lowpowerlab.com/guide/moteino/ (accessed on 21 September 2021).
65. Gaelens, J.; Van Torre, P.; Verhaevert, J.; Rogier, H. Lora mobile-to-base-station channel characterization in the Antarctic. *Sensors* **2017**, *17*, 1903. [CrossRef]
66. Fang, Y.; Chen, P.; Cai, G.; Lau, F.C.M.; Liew, S.C.; Han, G. Outage-limit-approaching channel coding for future wireless communications: Root-protograph low-density parity-check codes. *IEEE Veh. Technol. Mag.* **2019**, *14*, 85–93. [CrossRef]
67. Pan, X.; Di Maio, F.; Zio, E. A Benchmark of Dynamic Reliability Methods for Probabilistic Safety Assessment. In Proceedings of the 2017 2nd International Conference on System Reliability and Safety (ICSRS), Milan, Italy, 20–22 December 2017; pp. 82–90.
68. Castro, M.; Liskov, B. Practical Byzantine Fault Tolerance and Proactive Recovery. *ACM Trans. Comput. Syst.* **2002**, *20*, 398–461. [CrossRef]

The next pages (omitted due to copyright reasons) correspond to the published version of the following article:
A. Mallorqui, A. Zaballos and D. Serra, "A Delay Tolerant Network for Antarctica," in *IEEE Communications Magazine*, doi: https://doi.org/10.1109/MCOM.007.2200147

Due to copyright reasons, content from pages 208-214 was omitted from this version.