IARIA

Maria-Dolores Cano, Universidad Politécnica de Cartagena, Spain
Juan-Vicente Capella-Hernández, Universitat Politècnica de València, Spain
Eduardo Cerqueira, Federal University of Para, Brazil
Bruno Chatras, Orange Labs, France
Marc Cheboldaeff, Deloitte Consulting GmbH, Germany
Kong Cheng, Vencore Labs, USA
Dickson Chiu, Dickson Computer Systems, Hong Kong
Andrzej Chydzinski, Silesian University of Technology, Poland
Hugo Coll Ferri, Polytechnic University of Valencia, Spain
Noelia Correia, University of the Algarve, Portugal
Noël Crespi, Institut Telecom, Telecom SudParis, France
Paulo da Fonseca Pinto, Universidade Nova de Lisboa, Portugal
Orhan Dagdeviren, International Computer Institute/Ege University, Turkey
Philip Davies, Bournemouth and Poole College / Bournemouth University, UK
Carlton Davis, École Polytechnique de Montréal, Canada
Claudio de Castro Monteiro, Federal Institute of Education, Science and Technology of Tocantins, Brazil
João Henrique de Souza Pereira, University of São Paulo, Brazil
Javier Del Ser, Tecnalia Research & Innovation, Spain
Behnam Dezfouli, Universiti Teknologi Malaysia (UTM), Malaysia
Daniela Dragomirescu, LAAS-CNRS, University of Toulouse, France
Jean-Michel Dricot, Université Libre de Bruxelles, Belgium
Wan Du, Nanyang Technological University (NTU), Singapore
Matthias Ehmann, Universität Bayreuth, Germany
Wael M El-Medany, University Of Bahrain, Bahrain
Imad H. Elhajj, American University of Beirut, Lebanon
Gledson Elias, Federal University of Paraíba, Brazil
Joshua Ellul, University of Malta, Malta
Rainer Falk, Siemens AG - Corporate Technology, Germany
Károly Farkas, Budapest University of Technology and Economics, Hungary
Huei-Wen Ferng, National Taiwan University of Science and Technology - Taipei, Taiwan
Gianluigi Ferrari, University of Parma, Italy
Mário F. S. Ferreira, University of Aveiro, Portugal
Bruno Filipe Marques, Polytechnic Institute of Viseu, Portugal
Ulrich Flegel, HFT Stuttgart, Germany
Juan J. Flores, Universidad Michoacana, Mexico
Ingo Friese, Deutsche Telekom AG - Berlin, Germany
Sebastian Fudickar, University of Potsdam, Germany
Stefania Galizia, Innova S.p.A., Italy
Ivan Ganchev, University of Limerick, Ireland / University of Plovdiv "Paisii Hilendarski", Bulgaria
Miguel Garcia, Universitat Politecnica de Valencia, Spain
Emiliano Garcia-Palacios, Queens University Belfast, UK
Marc Gilg, University of Haute-Alsace, France
Debasis Giri, Haldia Institute of Technology, India
Markus Goldstein, Kyushu University, Japan
Luis Gomes, Universidade Nova Lisboa, Portugal
Anahita Gouya, Solution Architect, France
Mohamed Graiet, Institut Supérieur d'Informatique et de Mathématique de Monastir, Tunisie
Christos Grecos, University of West of Scotland, UK
Vic Grout, Glyndwr University, UK
Yi Gu, Middle Tennessee State University, USA
Angela Guercio, Kent State University, USA
Xiang Gui, Massey University, New Zealand
Mina S. Guirguis, Texas State University - San Marcos, USA

Tibor Gyires, School of Information Technology, Illinois State University, USA
Keijo Haataja, University of Eastern Finland, Finland
Gerhard Hancke, Royal Holloway / University of London, UK
R. Hariprakash, Arulmigu Meenakshi Amman College of Engineering, Chennai, India
Go Hasegawa, Osaka University, Japan
Eva Hladká, CESNET & Masaryk University, Czech Republic
Hans-Joachim Hof, Munich University of Applied Sciences, Germany
Razib Iqbal, Amdocs, Canada
Abhaya Induruwa, Canterbury Christ Church University, UK
Muhammad Ismail, University of Waterloo, Canada
Vasanth Iyer, Florida International University, Miami, USA
Imad Jawhar, United Arab Emirates University, UAE
Aravind Kailas, University of North Carolina at Charlotte, USA
Mohamed Abd rabou Ahmed Kalil, Ilmenau University of Technology, Germany
Kyoung-Don Kang, State University of New York at Binghamton, USA
Sarfraz Khokhar, Cisco Systems Inc., USA
Vitaly Klyuev, University of Aizu, Japan
Jarkko Kneckt, Nokia Research Center, Finland
Dan Komosny, Brno University of Technology, Czech Republic
Ilker Korkmaz, Izmir University of Economics, Turkey
Tomas Koutny, University of West Bohemia, Czech Republic
Evangelos Kranakis, Carleton University - Ottawa, Canada
Lars Krueger, T-Systems International GmbH, Germany
Kae Hsiang Kwong, MIMOS Berhad, Malaysia
KP Lam, University of Keele, UK
Birger Lantow, University of Rostock, Germany
Hadi Larijani, Glasgow Caledonian Univ., UK
Annett Laube-Rosenpflanzer, Bern University of Applied Sciences, Switzerland
Gyu Myoung Lee, Institut Telecom, Telecom SudParis, France
Shiguo Lian, Orange Labs Beijing, China
Chiu-Kuo Liang, Chung Hua University, Hsinchu, Taiwan
Wei-Ming Lin, University of Texas at San Antonio, USA
David Lizcano, Universidad a Distancia de Madrid, Spain
Chengnian Long, Shanghai Jiao Tong University, China
Jonathan Loo, Middlesex University, UK
Pascal Lorenz, University of Haute Alsace, France
Albert A. Lysko, Council for Scientific and Industrial Research (CSIR), South Africa
Pavel Mach, Czech Technical University in Prague, Czech Republic
Elsa María Macías López, University of Las Palmas de Gran Canaria, Spain
Damien Magoni, University of Bordeaux, France
Ahmed Mahdy, Texas A&M University-Corpus Christi, USA
Zoubir Mammeri, IRIT - Paul Sabatier University - Toulouse, France
Gianfranco Manes, University of Florence, Italy
Sathiamoorthy Manoharan, University of Auckland, New Zealand
Moshe Timothy Masonta, Council for Scientific and Industrial Research (CSIR), Pretoria, South Africa
Hamid Menouar, QU Wireless Innovations Center - Doha, Qatar
Guowang Miao, KTH, The Royal Institute of Technology, Sweden
Mohssen Mohammed, University of Cape Town, South Africa
Miklos Molnar, University Montpellier 2, France
Lorenzo Mossucca, Istituto Superiore Mario Boella, Italy
Jogesh K. Muppala, The Hong Kong University of Science and Technology, Hong Kong
Katsuhiro Naito, Mie University, Japan
Deok Hee Nam, Wilberforce University, USA

Sarmistha Neogy, Jadavpur University- Kolkata, India
Rui Neto Marinheiro, Instituto Universitário de Lisboa (ISCTE-IUL), Instituto de Telecomunicações, Portugal
David Newell, Bournemouth University - Bournemouth, UK
Ngoc Tu Nguyen, Missouri University of Science and Technology - Rolla, USA
Armando Nolasco Pinto, Universidade de Aveiro / Instituto de Telecomunicações, Portugal
Jason R.C. Nurse, University of Oxford, UK
Kazuya Odagiri, Sugiyama Jyogakuen University, Japan
Máirtín O'Droma, University of Limerick, Ireland
Jose Oscar Fajardo, University of the Basque Country, Spain
Constantin Paleologu, University Politehnica of Bucharest, Romania
Eleni Patouni, National & Kapodistrian University of Athens, Greece
Harry Perros, NC State University, USA
Miodrag Potkonjak, University of California - Los Angeles, USA
Yusnita Rahayu, Universiti Malaysia Pahang (UMP), Malaysia
Yenumula B. Reddy, Grambling State University, USA
Oliviero Riganelli, University of Milano Bicocca, Italy
Antonio Ruiz Martinez, University of Murcia, Spain
George S. Oreku, TIRDO / North West University, Tanzania/ South Africa
Sattar B. Sadkhan, Chairman of IEEE IRAQ Section, Iraq
Husnain Saeed, National University of Sciences & Technology (NUST), Pakistan
Addisson Salazar, Universidad Politecnica de Valencia, Spain
Sébastien Salva, University of Auvergne, France
Ioakeim Samaras, Aristotle University of Thessaloniki, Greece
Luz A. Sánchez-Gálvez, Benemérita Universidad Autónoma de Puebla, México
Teerapat Sanguankotchakorn, Asian Institute of Technology, Thailand
José Santa, University Centre of Defence at the Spanish Air Force Academy, Spain
Rajarshi Sanyal, Belgacom International Carrier Services, Belgium
Mohamad Sayed Hassan, Orange Labs, France
Thomas C. Schmidt, HAW Hamburg, Germany
Véronique Sebastien, University of Reunion Island, France
Jean-Pierre Seifert, Technische Universität Berlin & Telekom Innovation Laboratories, Germany
Dimitrios Serpanos, Univ. of Patras and ISI/RC ATHENA, Greece
Roman Y. Shtykh, Rakuten, Inc., Japan
Salman Ijaz Institute of Systems and Robotics, University of Algarve, Portugal
Adão Silva, University of Aveiro / Institute of Telecommunications, Portugal
Florian Skopik, AIT Austrian Institute of Technology, Austria
Karel Slavicek, Masaryk University, Czech Republic
Vahid Solouk, Urmia University of Technology, Iran
Peter Soreanu, ORT Braude College, Israel
Pedro Sousa, University of Minho, Portugal
Cristian Stanciu, University Politehnica of Bucharest, Romania
Vladimir Stantchev, SRH University Berlin, Germany
Radu Stoleru, Texas A&M University - College Station, USA
Lars Strand, Nofas, Norway
Stefan Strauβ, Austrian Academy of Sciences, Austria
Álvaro Suárez Sarmiento, University of Las Palmas de Gran Canaria, Spain
Masashi Sugano, School of Knowledge and Information Systems, Osaka Prefecture University, Japan
Young-Joo Suh, POSTECH (Pohang University of Science and Technology), Korea
Junzhao Sun, University of Oulu, Finland
David R. Surma, Indiana University South Bend, USA
Yongning Tang, School of Information Technology, Illinois State University, USA
Yoshiaki Taniguchi, Kindai University, Japan
Anel Tanovic, BH Telecom d.d. Sarajevo, Bosnia and Herzegovina

Rui Teng, Advanced Telecommunications Research Institute International, Japan
Olivier Terzo, Istituto Superiore Mario Boella - Torino, Italy
Tzu-Chieh Tsai, National Chengchi University, Taiwan
Samyr Vale, Federal University of Maranhão - UFMA, Brazil
Dario Vieira, EFREI, France
Lukas Vojtech, Czech Technical University in Prague, Czech Republic
Michael von Riegen, University of Hamburg, Germany
You-Chiun Wang, National Sun Yat-Sen University, Taiwan
Gary R. Weckman, Ohio University, USA
Chih-Yu Wen, National Chung Hsing University, Taichung, Taiwan
Michelle Wetterwald, HeNetBot, France
Feng Xia, Dalian University of Technology, China
Kaiping Xue, USTC - Hefei, China
Mark Yampolskiy, Vanderbilt University, USA
Dongfang Yang, National Research Council, Canada
Qimin Yang, Harvey Mudd College, USA
Beytullah Yildiz, TOBB Economics and Technology University, Turkey
Anastasiya Yurchyshyna, University of Geneva, Switzerland
Sergey Y. Yurish, IFSA, Spain
Jelena Zdravkovic, Stockholm University, Sweden
Yuanyuan Zeng, Wuhan University, China
Weiliang Zhao, Macquarie University, Australia
Wenbing Zhao, Cleveland State University, USA
Zibin Zheng, The Chinese University of Hong Kong, China
Yongxin Zhu, Shanghai Jiao Tong University, China
Zuqing Zhu, University of Science and Technology of China, China
Martin Zimmermann, University of Applied Sciences Offenburg, Germany

## CONTENTS

Simona Vasilache, University of Tsukuba, Japan
Shin Takahashi, University of Tsukuba, Japan

# Theoretical Models for Underwater RFID and the Impact of Water Salinity on the Design of Wireless Systems

Caroline Peres, Melusine Pigeon, Nadeem Rather, Dinesh Gawade, John Buckley, Hamed Jafarzadeh, Brendan O'Flynn

Tyndall National Institute
University College Cork
Cork, Ireland
caroline.peres@tyndall.ie, melusine.pidgeon@tyndall.ie, nadeem.rather@tyndall.ie, dinesh.gawade@tyndall.ie, john.buckley@tyndall.ie, hamed.jafarzadeh@tyndall.ie, brendan.oflynn@tyndall.ie

*Abstract*— Underwater wireless communications present challenges due to the characteristics of water as a propagation channel medium. Regardless, wireless communications are needed for a range of systems that operate underwater. Commonly used technologies for these use cases (radio-frequency, acoustic and optical communications) are lacking, as they generally suffer from strong attenuation, multipath effects and propagation delays. In this context, we explore the theoretical models for Path Loss of Radio Frequency Identification (RFID) systems underwater in regards to the salinity of the water. We also discuss RFID systems feasibility in such applications as aquaculture and fish stock management. This paper aims to discuss the theoretical transmission models for RFID systems underwater, separating them into near-field systems – which use Magnetic Induction (MI) to communicate – and far-field systems – that transfer data via Radio Frequency (RF). We determine the path loss for each case, the effect of the salinity in the model for the path loss, and present preliminary measurements of magnetic field strength underwater for different salinity values.

*Keywords- RFID; underwater wireless communications; underwater RFID; near-field communication; magnetic induction; salinity.*

## I. INTRODUCTION

This paper is an extension of a previous conference submission [1]. Underwater wireless communications present some challenges due to the characteristics of the channel medium. The underwater environment has different characteristics and phenomena compared to those typical for terrestrial radio propagation channel [2]. Despite these difficulties, underwater wireless communications are needed for a variety of underwater systems. Practical applications include seismic activity monitoring, equipment monitoring and control, underwater wireless sensor networks, underwater robots and Underwater Autonomous Vehicles (UAVs), aquaculture, fish stock management and underwater environmental monitoring [3][4].

There are three commonly used technologies for underwater communications [2][5][6]. Radio-frequency (RF) communication consists of propagating electromagnetic waves, and it has high data rates at short ranges but suffers from multipath propagation, strong attenuation and Doppler effect [2]. Due to the increasing attenuation for higher frequencies, it requires that systems operate at lower frequencies to achieve longer ranges of transmission, which in turn demands the use of large antennas making it unsuitable for some applications. Acoustic communication makes use of propagating sound waves, which have low attenuation underwater, achieving the longest range [2]. However, this type of communication exhibits a large propagation delay due to the speed of sound underwater, suffers from multipath propagation, and is affected by a large delay spread that leads to inter-symbol interference. Temperature gradients and ambient noise are also problems for acoustic communications. Another technology that can be used for underwater use is optical communication, which uses electromagnetic waves in the visible spectrum to transmit data. Such technologies have large data rates with low propagation delay. However, they suffer severe absorption in water and strong backscatter due to turbidity (e.g., suspended particles in the medium) [2].

Underwater Radio-Frequency Identification (RFID) is not an extensively explored topic due to the problems outlined above for RF communications, specifically strong attenuation and multipath propagation. However, some RFID systems communicate via Magnetic Induction (MI), which could provide an alternative for the existing technologies [4][7]. In this paper, we want to explore the potential for such technology to be used in the marine environment. We examine the different methods of communication that different RFID systems employ, separating them into two categories: near-field communication and far-field communication. Theoretical mathematical models exist for terrestrial RFID systems, from which the system functionality, communication properties and link budget can be derived. This paper aims to derive similar models for underwater RFID communications, by describing the underwater channel physical properties for near-field and far-field electromagnetic fields by presenting the path loss for each. This can then be used to predict communication range, link budget and channel capacity.

This paper is organised as follows. Section II discusses the related work. In Section III, we give a brief overview of an RFID system and its components. Section IV then presents the model for underwater RFID for near-field and far-field communications. In Section V, we present preliminary results of measurements done of magnetic field strength in various water solutions of different salinity values. Section VI discusses underwater RFID in light of the theory presented and the measurement results. We conclude the paper in Section VII.

## II. RELATED WORK

Underwater RFID is not a common topic due to the challenges that the underwater environment poses to RF communications. However, some preliminary work has been done. For example, [8] explores the use of Near-Field Communication (NFC) underwater. Using smartphones and smart cards operating at 13.56 MHz, they tested the read range achieved and the influence of dissolved salts in water in the read range. Another group used Low-Frequency (LF) RFID to track the sediment movements in a beach [9]. Transponders were coupled to pebbles, creating "smart" pebbles that could be detected at up to 50 cm underwater. They were then released into the beach and tracked to map the sediment movement. Systems that use LF RFID underwater can be found in Passive Integrated Transponder (PIT) tags used to uniquely identify fish in fisheries and research [10][11].

The authors in [12] summarised the current understanding of underwater RFID, examining the penetration depth in freshwater and seawater. However, the model presented is simplified and accounts only for the far-field operation. They also showcase other uses of RFID underwater, such as underwater pipeline monitoring. Other authors have explored MI communications underwater, where the system communicates via induction coupling. [13] provides an overview of the current research findings and challenges for MI. Models for MI can be found in [14]–[18].

## III. RFID SYSTEM

A typical RFID system is comprised of a tag and a reader (initiator). The tag is used to identify or measure ambient parameters, typically temperature for instance [19]. Furthermore, a reader is used to read and write data from or into such tag located within its proximity or vicinity. The tag consists of a coupling or backscatter element such as a conductor loop or an antenna, and an RFID radio that stores the data or ultra-low-power embedded system to measure various ambient parameters and store relevant metadata. The reader also comprises similar antennas as a coupling element along with a control unit and an RFID radio. Generally, near-field RFID uses inductive coupling between the reader and tag loop antennas to communicate with each other when located within each other's proximity or vicinity. RFID operates at 120–135 kHz low frequency (LF) unregulated band and high frequency (HF) 13.56 MHz Industrial,

Scientific, and Medical (ISM) band. LF RFID follows the ISO/IEC 18000-2 standard and HF RFID follows ISO/IEC 18000-3 standard along with additional smart cards ISO/IEC 15693, ISO/IEC 14443A and 14443B standard [20]–[24][25, p. 1][26]. Additional NFC standards ensure the interoperability of NFC-enabled devices and enable communication between them. The NFC standard defines the data rate (26.48 to 424 kbit/s), data frame formats such as NFC Data Exchange Format (NDEF), modulation, initialization, and collision control during initialization [27].

In Figure 1 the block diagram of a typical RFID sensing system is shown. The conductor loop antenna of the reader generates the magnetic alternating field. The magnetic flux ($\Phi$) generated by the reader loop antenna is used as a power supply for the sensor tag. This is achieved by utilising the voltage which is induced in the tag antenna by mutual inductance ($M$) between the transmitter and receiver antenna. Due to this induced voltage, a current starts flowing in the tag antenna and its value can be theoretically calculated from the quotient of the voltage divided by the impedance of the tag antenna [28]. Furthermore, the NFC radio Analog Front End (AFE) consists of an RF interface and the energy harvesting circuitry, which will connect to the loop antenna. The harvested voltage will further be regulated using a low dropout regulator and will be used to power up an ultra-low-power Microcontroller Unit (MCU), as well as a sensor [28]. The NFC radio consist of an Amplitude Shift Keying (ASK) demodulator which will demodulate the messages from the reader and responds to the reader with the help of load modulation. The load modulation is achieved by varying the impedance of the tag antenna [28]. The NFC radio and sensor will be interfaced with MCU using the Inter-Integrated Circuit (I2C) Protocol. The sensor will start sensing a parameter and its raw value will be transmitted to the MCU through the I2C. The MCU will then generate an NDEF message with the sensed value and forward it to the NFC radio [19][29].
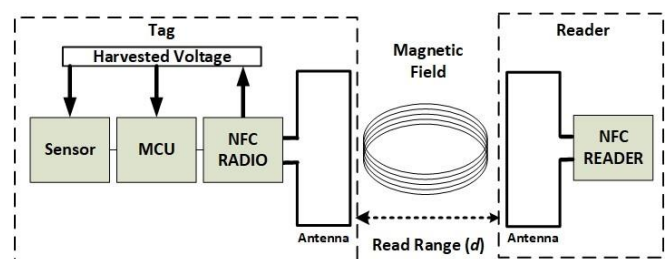


Figure 1. Block diagram of a RFID sensing system.

The tag receives the signal via the coupling element, and utilises the induced voltage to power up the tag's RFID radio and other electronics. RFID radio then sends data back to the reader via load modulation or backscatter. In general, such a tag is battery-less, and it is powered by the Magnetic flux ($\Phi$) generated by the reader. Other battery assisted power (BAP) system models exist incorporating an active RFID device that

consist of a battery as a power source. The added battery is used to power up additional tag electronics or sensors and to extend the communication distance range between the transmitter and receiver.

Equation (1) shows the relationship between the quality factor of the antenna $Q$, the bandwidth ($BW$) and the resonant frequency ($f$) for the system. For example, larger bandwidth is required to cover the sidebands of communication for the ISO/IEC 14443B standard and is particularly important to have if using higher data rates such as 424 or 848 kbps. In addition, for other applications based on the ISO/IEC15693 standard, the Q factor can be significantly higher, as the sidebands do not need such a wide bandwidth [30].

$$Q = f / BW \quad (1)$$

## IV. RFID CHANNEL PHYSICAL CHARACTERISTICS

The antenna or coil of the RFID reader generates an electromagnetic field. These fields can be described as time-harmonic fields in a lossy medium [31]:

$$\nabla^2 \mathbf{E} = \gamma \mathbf{E} \quad (2)$$

$$\nabla^2 \mathbf{H} = \gamma \mathbf{H} \quad (3)$$

where $\gamma$ is the propagation wave number, with $\alpha$ as the attenuation and $\beta$ as the phase variables. The wavelength $\lambda$ is $\lambda = 2\pi/\beta$.

$$\gamma = \alpha + j\beta = \sqrt{j\omega\mu(\sigma + j\omega\varepsilon)} \quad (4)$$

$$\alpha = \omega\sqrt{\mu\varepsilon}\left[\frac{1}{2}\left(\sqrt{1 + \left(\frac{\sigma}{\omega\varepsilon}\right)^2} - 1\right)\right]^{1/2} \quad \left(\frac{Np}{m}\right) \quad (5)$$

$$\beta = \omega\sqrt{\mu\varepsilon}\left[\frac{1}{2}\left(\sqrt{1 + \left(\frac{\sigma}{\omega\varepsilon}\right)^2} + 1\right)\right]^{1/2} \quad \left(\frac{rad}{m}\right) \quad (6)$$

The magnetic permeability $\mu = \mu_0 = 4\pi \cdot 10^{-7}$ H/m of the medium does not change for non-magnetic media. $\sigma$ is the conductivity of the medium, which in this case is dependent on the salinity of the water, its temperature and pressure. The salinity of the water is proportional to the concentration of dissolved salts (chloride, sodium, sulphate, etc.). In marine water, the conductivity ranges from 2 S/m to 6 S/m for frequencies lower than 10 GHz, being considered constant 4 S/m in most cases [12]. In freshwater, the considerations are the same. However, the salinity is lower, which means that the conductivity is lower (typically ranging from 30 to 2000 μS/cm) [12]. Due to this high conductivity, Eddy currents are induced within the water, caused by the propagating magnetic field [32]. These Eddy currents are a source of attenuation of the magnetic field.

The conversion between salinity and conductivity for seawater has been defined in the practical-salinity-scale PSS-78 [33]. This scale defines a standard ratio between any measured combination of salinity, conductivity, and temperature in relation to a standard value of conductivity and temperature for seawater of salinity 35 $g/Kg$.
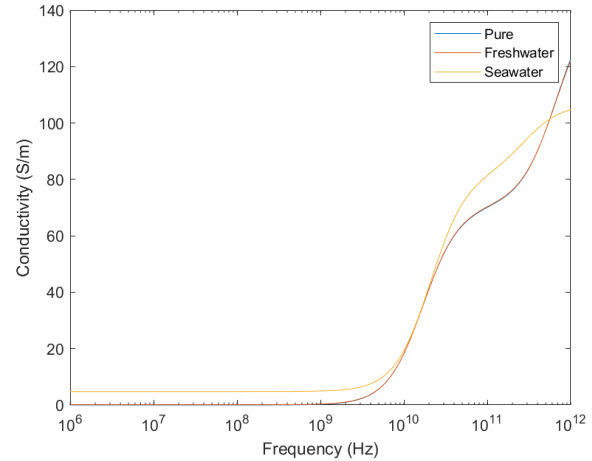


Figure 2: Conductivity of pure, freshwater and seawater for different values of propagating frequencies.

The dielectric permittivity of the medium $\varepsilon$ is defined by $\varepsilon = \varepsilon_r\varepsilon_0$, $\varepsilon_0 = 8.854 \times 10^{-12}$ F/m being the permittivity in free-space and $\varepsilon_r$ the relative permittivity of the medium. This relative permittivity is dependent on the composition of the medium that is polarised when placed under an electric field [34][35]. Equation (7) shows the relationship between the relative permittivity and the frequency of the propagating electromagnetic wave for pure water as modelled by Debye [35].

$$\varepsilon_r(\omega) = \varepsilon_\infty + \frac{\varepsilon_S - \varepsilon_\infty}{1 + j\omega\tau} \quad (7)$$

In this equation, $\tau$ is a time constant of the exponentially increasing orientation polarisation called relaxation time, and $\varepsilon_S$ and $\varepsilon_\infty$ are the static and infinite frequency relative permittivities of the medium. All these parameters are dependent on the temperature.

However, for freshwater and seawater, due to the interaction between molecules and the presence of ions that increase the conductivity of the medium, the simple model is not enough to accurately predict the permittivity [34].

For freshwater, there are extensive experimental studies and various models that predict the dielectric permittivity [36]–[40].

Work has been done by [38][41]–[43] to empirically determine a model for the relative permittivity of seawater, but some results disagree with each other. The International Telecommunication Union released a recommendation [44] that advises the model to use when calculating the dielectric permittivity and conductivity of seawater based on its

salinity, which will be used in this paper. Equation (8) shows how to calculate the relative dielectric permittivity based on this model.

$$\varepsilon_r = \frac{\varepsilon_{SS}-\varepsilon_{1S}}{1+(f_{GHz}/f_{1S})^2} + \frac{\varepsilon_{1S}-\varepsilon_{\infty S}}{1+(f_{GHz}/f_{2S})^2} + \varepsilon_{\infty S} \qquad (8)$$

The following equations show the calculations for each of the parameters from Equation (8).

$$\varepsilon_{SS} = \varepsilon_S exp(-3.33330 \times 10^{-3}S + 4.74868 \times 10^{-6}S^2) \quad (9)$$

$$f_{1S} = f_1\big(1 + S(2.3232 \times 10^{-3} - 7.9208 \times 10^{-5}T + 3.6764 \times 10^{-6}T^2 + 3.5594 \times 10^{-7}T^3 + 8.9795 \times 10^{-9}T^4)\big) \qquad (10)$$

$$\varepsilon_{1S} = \varepsilon_1 exp(-6.28908 \times 10^{-3}S + 1.76032 \times 10^{-4}S^2 - 9.22144 \times 10^{-5}TS) \qquad (11)$$

$$f_{2S} = f_2\big(1 + S(-1.99723 \times 10^{-2} + 1.81176 \times 10^{-4}T)\big) \qquad (12)$$

$$\varepsilon_{\infty S} = \varepsilon_\infty\big(1 + S(-2.04265 \times 10^{-3} + 1.57883 \times 10^{-4}T)\big) \qquad (13)$$

$$\varepsilon_S = 77.66 + 103.3 \qquad (14)$$

$$\varepsilon_1 = 0.0671\varepsilon_S \qquad (15)$$

$$\varepsilon_\infty = 3.52 - 7.52 \qquad (16)$$

$$\Theta = \frac{300}{T+273.15} - 1 \qquad (17)$$

$$f_1 = 20.20 - 146.4\Theta + 316\Theta^2 \qquad (18)$$

$$f_2 = 39.8f_1 \qquad (19)$$

in which $T$ is the temperature in degrees Celsius (°C), $f_{GHz}$ is the frequency of the signal in GHz, $S$ is the salinity in g/kg or ppt, and $f_1$ and $f_2$ are the Debye relaxation frequencies for pure water.

From the same model, the conductivity is:

$$\sigma = \sigma_{35}R_{35}R_{T15} \quad (S/m) \qquad (20)$$

The following equations show the calculations for each of the parameters from Equation (20).

$$\sigma_{35} = 2.903602 + 8.607 \times 10^{-2}T + 4.738817 \times 10^{-4}T^2 - 2.991 \times 10^{-6}T^3 + 4.3047 \times 10^{-9}T^4 \qquad (21)$$

$$R_{35} = S\frac{(37.5109+5.45216S+1.4409\times10^{-2}S^2)}{(1004.75+182.283S+S^2)} \qquad (22)$$

$$R_{T15} = 1 + \frac{a_0(T-15)}{a_1+T} \qquad (23)$$

$$a_0 = \frac{(6.9431+3.2841S-9.9486\times10^{-2}S^2)}{(84.850+69.024S+S^2)} \qquad (24)$$

$$a_1 = 49.843 - 0.2276S - 0.198 \times 10^{-2}S^2 \qquad (25)$$

Figure 2 shows the influence of the frequency of the signal in the conductivity of the medium. Figure 3 shows the complex permittivity for pure water, freshwater and seawater as a function of the frequency according to Equation (26).

$$\hat{\varepsilon} = \varepsilon_r - j\varepsilon_r'' = \frac{\varepsilon}{\varepsilon_0} - j\frac{\sigma}{\omega\varepsilon_0} \qquad (26)$$
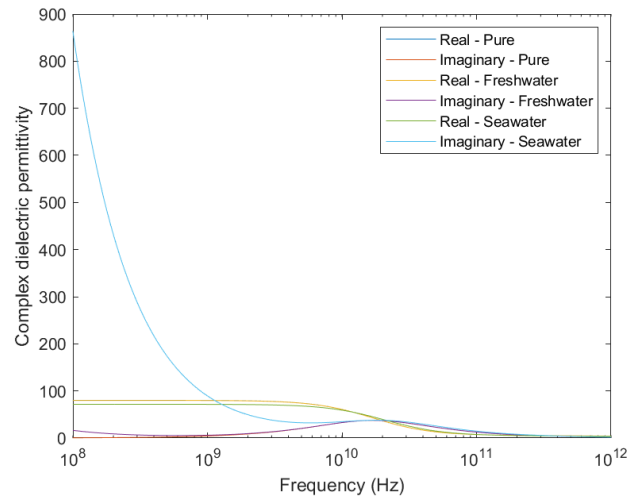


Figure 3: Real (relative permittivity) and imaginary parts of the complex dielectric permittivity for pure water, freshwater (Salinity = 0.5 g/kg) and seawater (Salinity = 35 g/kg) at temperature T = 20°C.

The dielectric permittivity and the conductivity are then used to determine the attenuation factor α. In [45], the authors propose a review of this model to account for the difference between the theoretical calculations and the empirical data of the attenuation of radio waves underwater. The experiments show that the signal attenuation at higher distances (≫10 m) is not as strong as predicted. Therefore, they redefine α as a corrected absorption factor α' that matches experimental results closely:

$$\alpha' = \alpha\left(\frac{\lambda}{\lambda+z}\right) \qquad (27)$$

For the Transverse Electromagnetic Mode to the positive $z$ direction in lossy medium (in this case, water), **E** and **H** can be derived as [31]:

$$\mathbf{E(z)} = \hat{a}_x E_0 e^{-\gamma z} \qquad (28)$$

$$\mathbf{H(z)} = \hat{a}_y \frac{\gamma}{j\omega\mu} E_0 e^{-\gamma z} \qquad (29)$$

For a given antenna, the space that surrounds it can be separated into three regions: (a) a reactive near-field, (b) a radiating near-field and (c) the far-field. There are no abrupt changes at their boundaries [46]. A representation of these regions can be seen in Figure 4.
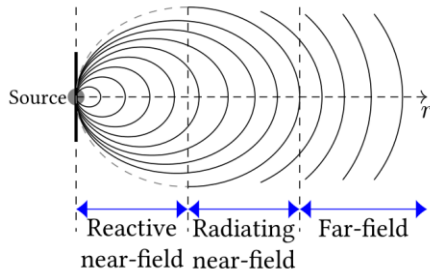


Figure 4: Field regions for a propagating electromagnetic wave leaving an antenna.

The *Reactive Near-Field* is the space immediately near the antenna where the reactive field predominates (magnetic field). For most antennas, the limit of this region is at $0.62\sqrt{D^3/\lambda}$ [46], where $D$ is the biggest dimension of the antenna. The *Radiating Near-Field* is located between the reactive near-field and the far-field and is the space wherein radiation fields are dominant. The angular field distribution is determined by the distance from the antenna. This field existence depends on the ratio between antenna size $D$ and the wavelength $\lambda$: if $D/\lambda \ll 1$ then this region does not exist. The *Far-Field* is the region wherein the electrical and magnetic components of the field become orthogonal to each other as they separate from the antenna and propagate as an electromagnetic wave. The lower boundary of this region is located at $2D^2/\lambda$ for any antenna [31], also considered to be $\lambda/2\pi$ for dipole antennas. According to [28], a good approximate rule for RFID systems is to place the beginning of the far-field at $\lambda/2\pi$.

The field boundary distance is different for each medium due to the difference in wavelength. Tables I and II show the values for the attenuation coefficient, wavelength, and far-field boundary for freshwater and seawater, respectively.

Current RFID systems can be separated into two categories: near-field systems that work with inductive coupling due to the dominance of the magnetic field in the near-region, and far-field systems that receive power from the propagating electromagnetic waves in the far-field [47]. The frequencies used in each region are different. Since the lower frequencies – such as Low Frequency (LF) at around 134.2kHz and High Frequency (HF) at 13.56MHz – have a far-field boundary that is further away, they are mainly used in inductive coupling systems. Higher frequencies are then used mostly in far-field systems.

TABLE I. VALUES OF ATTENUATION FACTOR A, WAVELENGTH Λ AND FAR-FIELD BOUNDARY $z_F = \lambda/2$ FOR FRESHWATER (S = 0.5 G/KG).

| Frequency | $\alpha$ (Np/m) | $\lambda$ (m) | $z_F$ (m) |
|---|---|---|---|
| 134.2 kHz | 2.16E-01 | 2.89E+01 | 4.60E+00 |
| 13.56 MHz | 1.58E+00 | 2.10E+00 | 3.34E-01 |
| 433.9 MHz | 2.83E+00 | 7.73E-02 | 1.23E-02 |
| 915 MHz | 6.16E+00 | 3.67E-02 | 5.84E-03 |
| 1.5 GHz | 1.34E+01 | 2.24E-02 | 3.57E-03 |
| 2.4 GHz | 3.11E+01 | 1.41E-02 | 2.24E-03 |
| 5 GHz | 1.24E+02 | 6.91E-03 | 1.10E-03 |

TABLE II. VALUES OF ATTENUATION FACTOR A, WAVELENGTH Λ AND FAR-FIELD BOUNDARY $z_F = \lambda/2$ FOR SEAWATER (S = 35 G/KG).

| Frequency | $\alpha$ (Np/m) | $\lambda$ (m) | $z_F$ (m) |
|---|---|---|---|
| 134.2 kHz | 1.59E+00 | 3.94E+00 | 6.27E-01 |
| 13.56 MHz | 1.59E+01 | 3.90E-01 | 6.20E-02 |
| 433.9 MHz | 7.63E+01 | 5.80E-02 | 9.22E-03 |
| 915 MHz | 9.51E+01 | 3.34E-02 | 5.32E-03 |
| 1.5 GHz | 1.07E+02 | 2.19E-02 | 3.49E-03 |
| 2.4 GHz | 1.25E+02 | 1.42E-02 | 2.27E-03 |
| 5 GHz | 2.01E+02 | 7.08E-03 | 1.13E-03 |

### A. Near-field

In the near-field, the magnetic field created by the reader's antenna induces a voltage in the transponder immersed in this field. This is called *inductive coupling* and the interaction between reader and transponder can be considered as coupled inductors. This method of communication can also be called *Magnetic Induction (MI)*.

Consider the equivalent circuit for the inductively coupled system shown in Figure 5. The transmitter antenna is fed by a source with internal impedance $Z_S$ and the receiver antenna is terminated by a load impedance $Z_L$. The transmitter coil antenna has a impedance of $Z_{TX} = R_{TX} + j\omega L_{TX} + 1/(j\omega C_{TX})$ and the receiver coil antenna is $Z_{RX} = R_{RX} + j\omega L_{RX} + 1/(j\omega C_{RX})$.
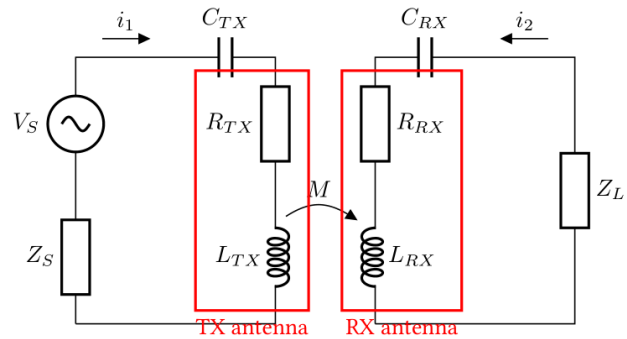


Figure 5: Inductive coupling between reader and transponder.
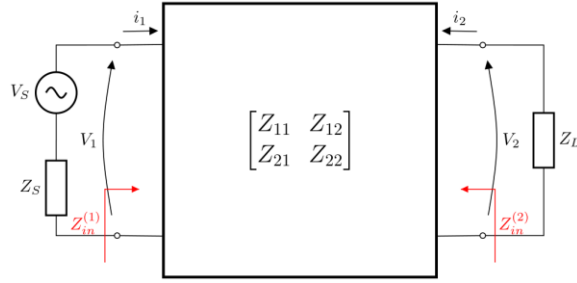
Figure 6: Two-port network equivalent of the system.

Using the two-port network equivalent (Figure 6) and considering an ideal source for $V_S$, $Z_{11} = Z_{TX}$ and $Z_{22} = Z_{RX}$ are the self-impedances of the coils and $Z_{12} = Z_{21} = j\omega M$ are the mutual impedances due to the coupling.

$$\begin{pmatrix} V_1 \\ -Z_L I_2 \end{pmatrix} = \begin{bmatrix} z_{11} & z_{12} \\ z_{21} & z_{22} \end{bmatrix} \cdot \begin{pmatrix} I_1 \\ I_2 \end{pmatrix} \quad (30)$$

The resistance of a coil is $R = N \cdot 2\pi a \cdot R_0$, where $N$ is the number of turns of the coil, $a$ is the diameter of the coil and $R_0$ is the resistance of a unit of length of the wire used to fabricate the coil. The self-inductance is

$$L = \frac{\mu \pi^2 N^2 a}{l} \quad (H) \quad (31)$$

where $l$ is the length of the coil. In the free space, the magnetic field strength generated by a coil antenna in the near field is [28]:

$$H_0 = \frac{Na^2 I}{2(a^2+z^2)^{3/2}} \quad (A/m) \quad (32)$$

The magnetic field magnitude for a lossy medium is then $H = H_0 exp(-\alpha z)$ according to (29). This magnetic field induces a voltage in the tag's coil antenna, given by:

$$U_2 = -N_2 \frac{d\Phi_{21}}{dt} = -M \frac{di_1}{dt} \quad (33)$$

where $\Phi_{21} = \int bB \cdot d\mathbf{S}$ is the magnetic flux through each turn, $\mathbf{B} = \mu\mathbf{H}$ the magnetic field and $\mathbf{S}$ the surface area of the coil. Considering that the reader's and tag's coils are aligned, and using (32):

$$B_z = \left( \frac{\mu Na_1^2 I}{2(a_1^2+z^2)^{2/3}} \right) e^{-\alpha z} \quad (34)$$

Therefore, the mutual inductance $M = k\sqrt{L_1 L_2}$ is:

$$M = \left( \frac{\mu \cdot \pi \cdot N_1 \cdot a_1^2 \cdot N_2 \cdot a_2^2}{2 \cdot (a_1^2+z^2)^{3/2}} \right) \cdot e^{-\alpha z} \quad (35)$$

where $\alpha$ is the attenuation constant of the medium.

The transmission power can be defined as the power consumed by the radiation resistance in the reader (transmitter) antenna:

$$P_{TX} = \frac{1}{2} Re(Z_{11}) \cdot |I_1|^2 \quad (36)$$

The received power is defined as the power consumed in the load:

$$P_{RX}(z) = \frac{1}{2} Re(Z_L) \cdot |I_2|^2 \quad (37)$$

Using (30) and considering $Z_S \approx 0$, the received power can be written as:

$$P_{RX}(z) = P_{TX} \frac{Re(Z_L)\omega^2 M^2}{Re(Z_{TX})|Z_L+Z_{RX}|^2} \quad (38)$$

Path loss in decibels (dB) can be defined as:

$$PL = -10 \log_{10} \left( \frac{P_{RX}}{P_{TX}} \right) (dB) \quad (39)$$

The path loss is a function of the number of turns and radius of both coils and the impedances of the system, as well as the frequency and the distance between reader and tag. The highest amount of power is transferred to the load when its impedance is matched with the impedance of the antenna.

The path loss for the MI system increases with the increasing distance between reader and tag. Also, the path loss is higher for seawater due to the higher conductivity of the medium. As the frequency increases, the distance from the reader where the border between the near and the far-field is located decreases. This implies that the maximum theoretical range decreases with frequency. We can then conclude that there is an optimal combination of frequency and distance for each application. In addition to that, the attenuation factor α is higher for higher frequencies.

The influence of the number of turns of the transmitter or the receiver coil in the mutual inductance $M$ is linear. Therefore, the power received would increase quadratically with the increase in the number of turns. However, it also increases the energy losses in the internal resistance of the coil as it increases. It is also worth noting that a bigger coil diameter allows for more magnetic flux to pass through, but it also has the effect of increasing the internal resistance of the coil.

A more in-depth model of underwater magnetic induction communication can be found in [15]. The model shown here assumes that the reader and tag coils are oriented in the same direction, with the field strength reaching zero if the angle between coils is 90°. To remove this limitation, the authors in [48] present a model of the Underwater MI channel for a tri-directional coil. To increase the achievable range of MI systems, waveguides can be used [14][49]. The authors in

[17] provide a different but similar model that is based on the quality factor of the coil inductor.

### 1) Data transmission from tag

When a transponder is located in the magnetic alternating field generated by the reader, the reader 'sees' the transponder as the secondary wing of the transformer. This means that the transponder's impedance is reflected back to the reader as the *transformer impedance $Z_T$*.

If the transponder antenna impedance changes, this is reflected back to the reader's coil via the reflected impedance $Z_T$. Therefore, a data stream can be transmitted via modulation of the voltage $Z_L$ in the reader's coil (called *Load Modulation*); this can be demodulated by the reader via rectification of the voltage [28]. This is only feasible in the near-field as if the transponder leaves the appropriate read range, the coupling is lost and the transmission link is not operational anymore.

For an amplitude modulating system, due to the weak coupling between reader and transponder antennas, the voltage fluctuation is orders of magnitude smaller than the voltage provided by the reader. As a direct result, the reader has to integrate a complex circuitry to separate noise from the signal and detect the data stream. On the other hand, if the transponder modulates the signal at a frequency $f_s$, smaller than the frequency of the magnetic field ($f_0$), two spectral lines $\pm f_s$ are created and they can be filtered with a band-pass filter and demodulated more easily [47].

### B. Far-field

In the far-field, the electromagnetic fields separate completely from the reader's antenna and become propagating waves, no longer retroacting upon the reader's antenna. These waves are captured by the antenna on the transponder. The energy on the antenna is rectified and used to power up the IC. The frequency range commonly used for this type of transmission is the Ultra-High Frequency (UHF) and Microwave.

A linearly polarized plane EM wave propagating in lossy media in the $z$-direction can be described by the electric field strength $E_x$:

$$E_x = E_0 e^{j\omega t - \gamma z} \qquad (40)$$

with $\gamma = \alpha + j\,\omega\beta$ as the propagating constant according to (5) and (6).

The radiation power density $S$ is the instantaneous value of the Poynting vector $\mathbf{S} = \mathbf{E} \times \mathbf{H}$. From [31] and considering (29):

$$\mathbf{S} = \tfrac{1}{2} Re(\mathbf{E} \times \mathbf{H}) = \hat{a}_z \frac{|E_0|}{2} e^{-2\alpha z} Re\left(\frac{1}{\eta_c^*}\right) \qquad (41)$$

where $\eta_C$ is the intrinsic impedance of the medium, given by (42).

$$\eta_C = \sqrt{\frac{j\omega\mu}{\sigma + j\omega\varepsilon}} \quad (42)$$

For the transmitting antenna in the free-space, $s_0$ is the power supplied to it over the area of the spread surface:

$$S_0 = \frac{P_{EIRP}}{4\pi z^2} = \frac{P_{TX} G_{TX}}{4\pi z^2} \quad (43)$$

Whereas the radiation power density in a lossy medium is then:

$$S = S_0 e^{-2\alpha z} \quad (44)$$

For the receiving antenna, the average power received is the radiation power density times its effective receiving area $A_e$ [45]:

$$P_{RX} = S \cdot A_e = S \cdot \frac{G_{RX} \lambda^2}{4\pi} \qquad (45)$$

The transmission equation then can be written as:

$$P_{RX} = P_{TX} \left(\frac{G_{TX} G_{RX} \lambda^2}{(4\pi z)^2}\right) e^{-2\alpha z} \qquad (46)$$

where $G_{TX}$ and $G_{RX}$ are the antenna gains for transmitter and receiver respectively, $\lambda = (2\pi)/\beta$ is the wavelength and $z$ is the distance between antennas. This equation assumes that the antennas are aligned and have the same polarization. The path loss $PL_{EM}$ in decibels is then defined as $PL_{EM} = -10 \log_{10}(P_{RX}/P_{TX})$.

### 1) Data transmission from tag

For passive RFID, the method of transmitting back to the reader is via *Backscatter*. Electromagnetic waves are reflected by objects that are larger than half the wavelength ($\lambda/2$) [28]. The efficiency of this reflection depends on the radar cross-section of the object: antennas that are resonant with the waves have a larger reflection cross-section. The reflection characteristics can be altered by changing the load that is connected to the antenna. For example, if a load $R_L$ is switched on and off while connected to the antenna, this changes the reflection characteristics of the antenna, generating a modulated backscatter signal [28]. The range is limited by the amount of energy that reaches the tag (path loss) and the sensitivity of the reader's receiver to the reflected signal (reflected signal strength $\propto 1/x^4$) [50]. The authors in [51] present a method for measuring the backscatter of an RFID tag and for calculating its radar cross-section. They utilise a network analyser connected to an anechoic chamber.
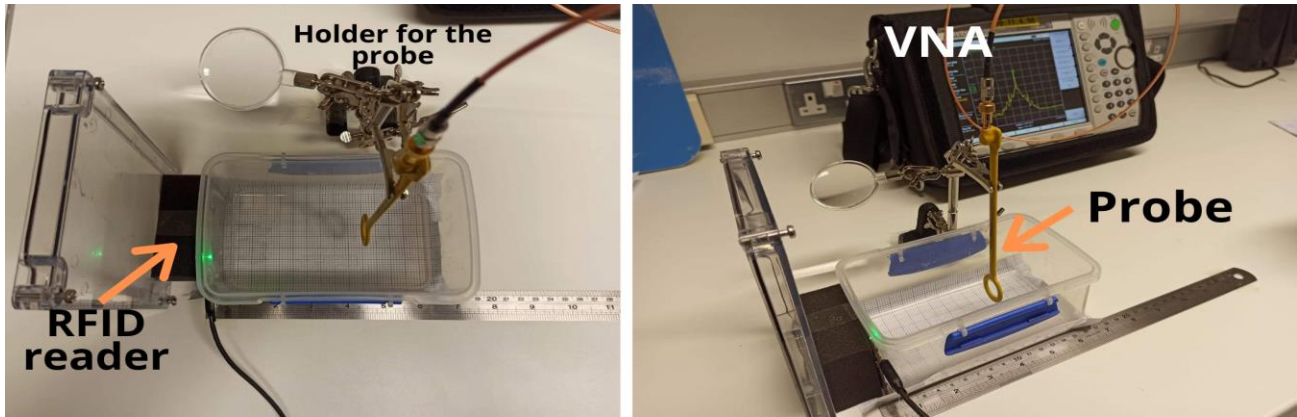
Figure 7: Experimental setup for measuring the magnetic field strength consisting of a plastic container, magnetic probe, holder and stand for the probe and VNA. Each RFID reader system was placed and secured on the side of the plastic container.

## V. MEASUREMENTS

To explore the difference in magnetic field strength between free-space and water in the near-field region, a preliminary experiment was designed. The Anritsu MS2038C VNA Master [52] vector network analyser and the probe 100C from Beehive Electronics, USA [53] were used to measure the magnetic field strength at the system's resonant frequency. Two Evaluation Kit RFID readers were used: *MRD2EVM* from Texas Instruments, USA that operates at 134.2 kHz [54] and *Pepper Wireless C1 USB* from Eccel Technology Ltd, UK that operates at 13.56 MHz [55]. Both have square loop antennae embedded on the printed circuit board, with sides of length 3.0 cm and 4.5 cm, and number of turns 14 and 3, respectively.

For a square loop antenna with *N*-turns, the magnetic field strength in free-space can be written as [56]:

$$H_0 = \frac{NI}{2\pi\left(\frac{z^2}{l^2}+\frac{1}{4}\right)\sqrt{z^2+\frac{l^2}{2}}} \quad (47)$$

where $z$ is the distance from the centre of the antenna and $l$ is the length of the side of the antenna. Using (29), for a lossy medium (in this case saline water), the magnetic field strength is then

$$H = H_0 exp(-\alpha z) \quad (48)$$

Using α from (5) with the salinity and frequency, we can then calculate the theoretical values for the magnetic field strength for any distance and compare this with the measurements made with the probe.

For both systems, the setup for the experiments was the same, as seen in Figure 7. The readers were placed and secured on the side of a plastic transparent container. The probe was placed in different distances $z$ from the centre of the embedded antenna of the reader. A measurement ruler

and grid paper were used for the precise placement of the probe. The free-space tests were done without water inside the container. For the other measurements, the probe was submerged in the water solution to get the measured field strength.

For the experimental test, 4 water solutions were used, named as: distilled water, freshwater, brackish water, and saline water. The water solutions were prepared as follows. First the container was placed in a scale, and the mass of NaCl was added, according to the target value of salinity for each solution. Then, distilled water was added until the whole solution mass reached 700g (the volume of the plastic container used for the experiments). Table III shows the mass of salt for each solution and their salinity and calculated conductivity.

TABLE III. SOLUTIONS USED FOR THE EXPERIMENT, TEMPERATURE AT WHICH THEY WHERE MIXED, THEIR SALINITY AND CALCULATED CONDUCTIVITY FOR EACH SOLUTION ACCORDING TO [44].

| Solution | NaCl mass | Total mass | Salinity | Tempe-rature | Calculated Conductivity (at 1 MHz) |
|---|---|---|---|---|---|
| Freshwater | 0.35 g | 0.7002 kg | 0.499 g/kg | 19 C | 0.093 S/m |
| Brackish Water | 10.5 g | 0.7003 kg | 14.99 g/kg | 18 C | 2.32 S/m |
| Saline Water | 24.5 g | 0.7003 kg | 34.98 g/kg | 19 C | 5.04 S/m |

For each solution, the probe was placed and held at different distances from the centre of the antenna and the peak value of the magnetic field was measured using the VNA. After this was done, the probe was wiped so there was no contamination between solutions.

The output power values measured by the probe $P_{out}$ were then converted from *dBm* to magnetic field strength in *A/m* using the probe manufacturer's guidelines [53], using Equations (49) and (50), where $f_{MHz}$ is the frequency of the system in *MHz*, $B_{out}$ is the magnetic flux density, and $H_{out}$

is the magnetic field strength. The raw data collected from the probe and the VNA in *dBm* can be found in Tables IV and V.

$$20 \times log_{10}(B_{out}) = P_{out} - 42.2 - 20 \times log_{10}(f_{MHz}) \quad (49)$$

$$H_{out} = \frac{10^B}{\mu} \quad (A/m) \quad (50)$$

Figure 8 shows the measured values of magnetic field strength for the 134.2 kHz system, while Figure 9 shows the values for the 13.56MHz system.

Figures 10 and 11 show the comparison between measured values of field strength and the theoretical values expected using a calculated α from the water salinity. For this, the peak current (A) flowing through the antennas was measured under the same experimental conditions, and Equation (47) was used.

TABLE IV.   MEASURED VALUES OF $P_{out}$ IN DBM FOR THE 134.4KHZ SYSTEM FOR FREE-SPACE (FREE), DISTILLED WATER (DW), FRESHWATER (FW), BRACKISH WATER (BW) AND SALINE WATER (SW) SOLUTIONS.

| z (cm) | F (dBm) | DW (dBm) | FW (dBm) | BW (dBm) | SW (dBm) |
|---|---|---|---|---|---|
| 0.2 | -2.2 | -2.18 | -2.63 | -3.62 | -1.96 |
| 1 | -9.66 | -7.68 | -6.84 | -8.34 | -7.35 |
| 2 | -13.82 | -14.24 | -13.36 | -15.42 | -14.13 |
| 3 | -20.2 | -21.13 | -19.6 | -21.06 | -20.22 |
| 4 | -26.1 | -25.95 | -25.41 | -25.81 | -25.72 |
| 5 | -30.15 | -30.69 | -30.27 | -30.04 | -30.43 |
| 6 | -34.02 | -34.44 | -33.41 | -34.06 | -34.16 |
| 7 | -37.7 | -37.36 | -36.75 | -37.24 | -37.3 |
| 8 | -40.9 | -40.76 | -40.42 | -40.56 | -40.79 |
| 9 | -43.34 | -43.66 | -42.79 | -43.07 | -43.26 |
| 10 | -45.65 | -45.66 | -45.31 | -45.84 | -45.9 |

TABLE V.   MEASURED VALUES OF $P_{out}$ IN DBM FOR THE 13.56MHZ SYSTEM FOR FREE-SPACE (F), DISTILLED WATER (DW), FRESHWATER (FW), BRACKISH WATER (BW) AND SALINE WATER (SW) SOLUTIONS.

| z (cm) | F (dBm) | DW (dBm) | FW (dBm) | BW (dBm) | SW (dBm) |
|---|---|---|---|---|---|
| 0.2 | 11.64 | 11.09 | 11.03 | 10.7 | 9.49 |
| 1 | 7.72 | 7.96 | 7.87 | 6.65 | 6 |
| 2 | 2.74 | 2.32 | 2.58 | 1.73 | 1.24 |
| 3 | -2.37 | -2.12 | -2.06 | -3.84 | -4.15 |
| 4 | -6.92 | -7.06 | -7.01 | -7.94 | -8.53 |
| 5 | -10.92 | -11.56 | -10.76 | -11.95 | -12.75 |
| 6 | -14.33 | -15.02 | -14.92 | -15.63 | -16.7 |
| 7 | -17.76 | -17.94 | -17.56 | -18.28 | -19.74 |
| 8 | -20.56 | -20.52 | -20.88 | -21.65 | -22.76 |
| 9 | -23.26 | -23.76 | -23.66 | -24.12 | -25.51 |
| 10 | -25.76 | -25.66 | -25.96 | -26.6 | -27.76 |
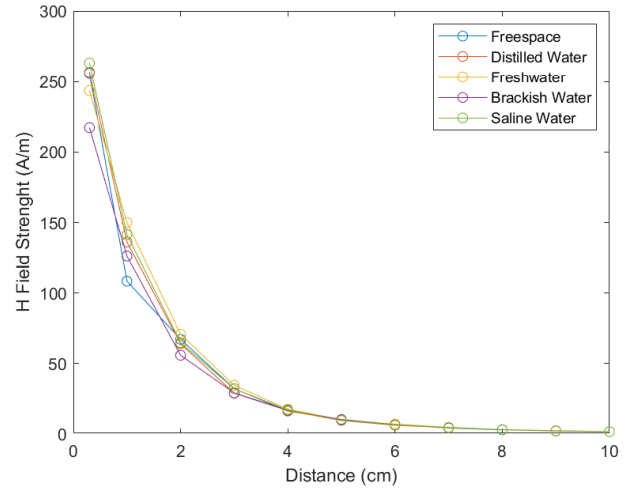


Figure 8: Measured magnetic field strength values for the Texas Instruments MRD2EVM evaluation kit (*f*=134.2 kHz).
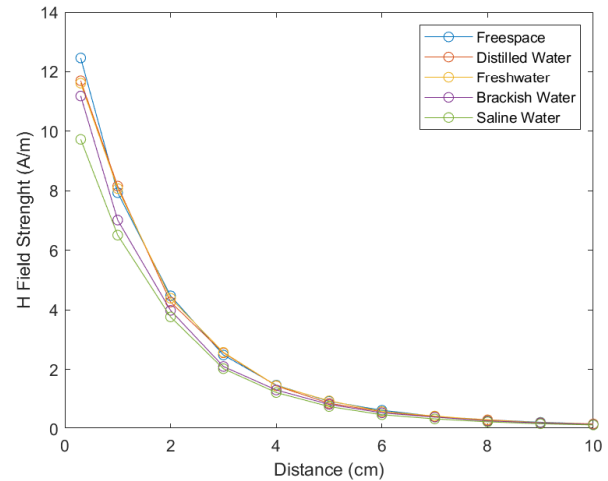


Figure 9: Measured magnetic field strength values for the Eccel Technology Pepper C1 USB evaluation kit (*f*=13.56 MHz).

VI.   DISCUSSIONS

The most common method of wireless transmission underwater is acoustic communication. This is due to the long range that can be accomplished with this technology. However, some applications do not need such long range and are deeply affected by acoustic noise and refractions, reflections and multipath due to the proximity to the water surface, such as coastal environments. In these cases, wireless communication can be better served by other methods that do not suffer from these problems. We explore the possibility of using RFID technology to better serve these environments.
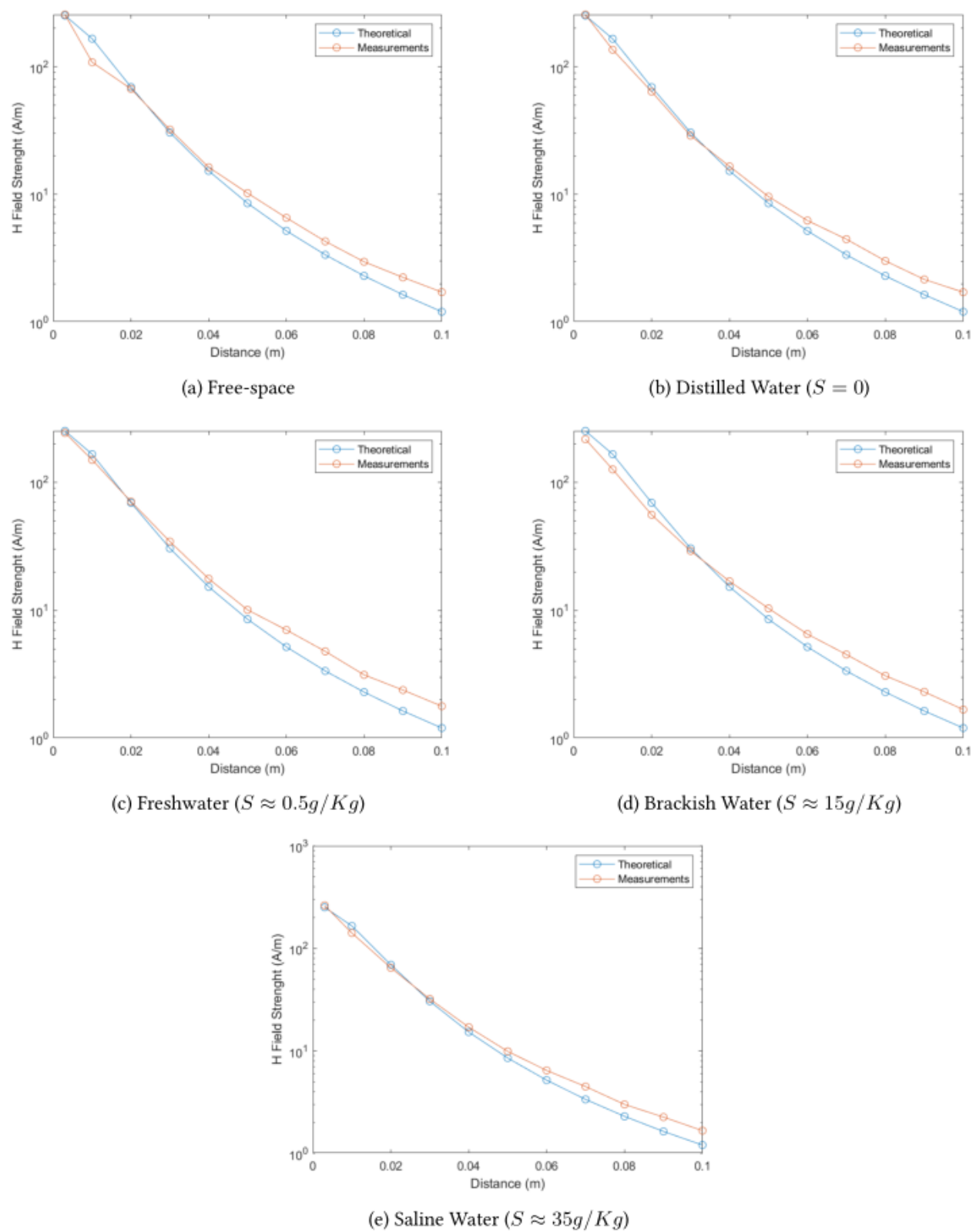
Figure 10: Comparison between measured values of magnetic field strength for different salinity values underwater for the 134.2kHz system.
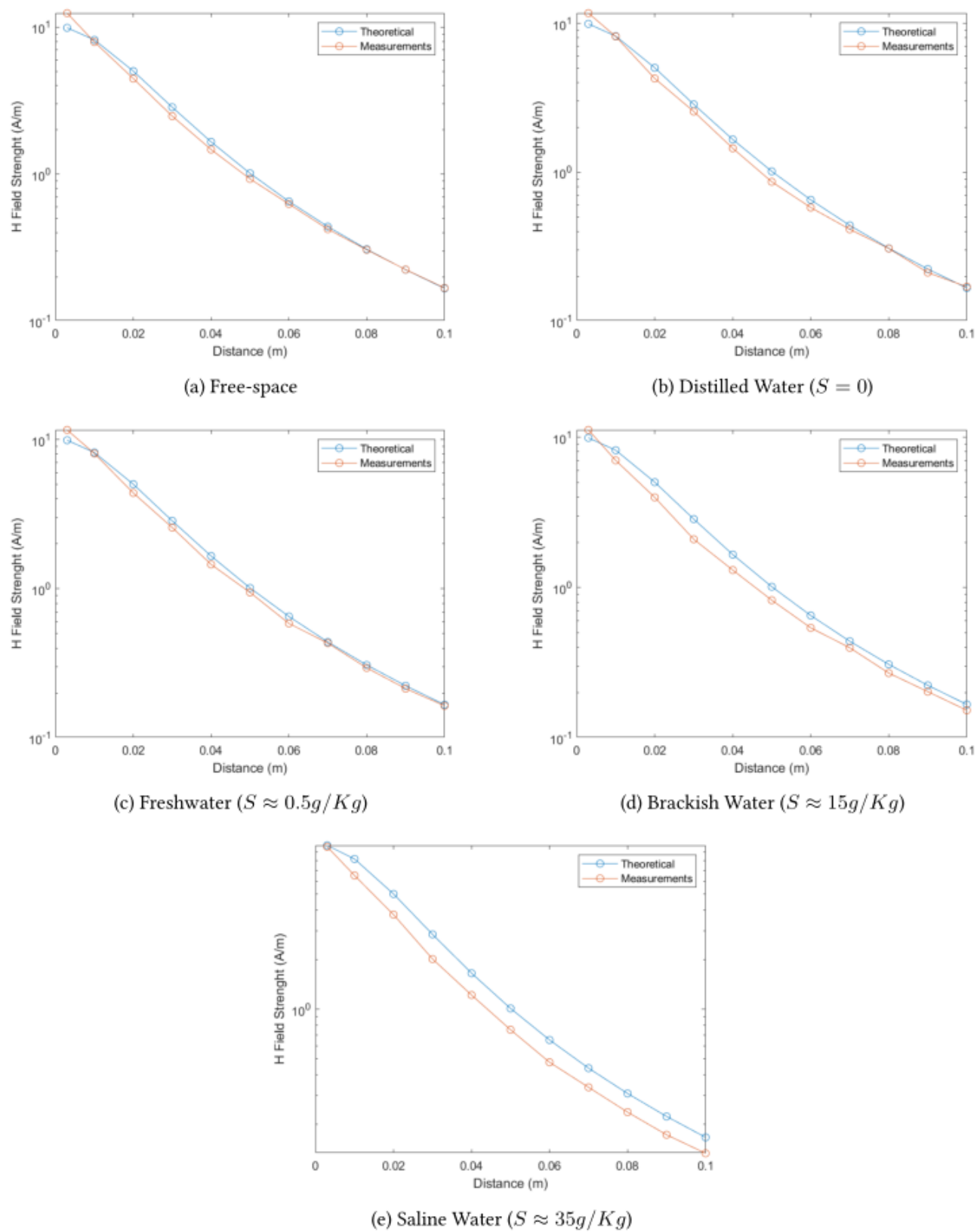
Figure 11: Comparison between measured values of magnetic field strength for different salinity values underwater for the 13.56MHz system.

In the near-field region, the magnetic component of the electromagnetic field dominates. The method of communication for RFID in this region is MI. Compared to other methods of underwater communications, MI has several advantages. It is not affected by multipath propagation or fading and the magnetic field can cross the water to air boundary with low attenuation [14]. The signal propagation delay is negligible if compared to acoustic waves. The channel response is predictable, and a sufficiently large range can be achieved with modest data rates [13].

For the far-field, the electromagnetic field propagates as a wave, and the communication is realized through radiofrequency. Due to the high attenuation, there is a severe constraint on data rates and propagation distances for this method [2]. Lower frequency signals have lower attenuation (due to conductivity of the water) but require larger antennas. This also limits the bandwidth of the system due to the lower frequency of operation. Higher frequency systems would then require more power to reach the same ranges. Shallow water environments, in particular, pose a problem to wave propagation due to the proximity to the water/air upper boundary and to the river/sea bed, which causes multipath propagation [5][6].

Both technologies do not require line-of-sight and are unaffected by light and acoustic ambient noise. Moreover, the channel response is independent of water quality conditions, such as turbidity. The literature generally agrees that the achievable range for a given transmission power is not great for both MI and RF. The reasoning is due to the high attenuation caused by the medium conductivity which increases with the salinity of the water. From this, it follows that long-range transmissions underwater, particularly in a marine environment, are best served by acoustic communication based systems [2].

However, there are some evidence that this attenuation could be lower than expected. The authors in [57] managed to transmit a RF signal at 90 m distance in seawater with a lower attenuation than expected. To reflect this results, [45] propose the change in the attenuation factor $\alpha$ to $\alpha'$ according to Equation (27). In addition to that, the dielectric permittivity for saline water is not completely understood [34]. There are models available extrapolated from measurement data, but they do not agree completely. Since the dielectric permittivity affects the attenuation, it follows that the attenuation itself could have a different value.

The results from our experiment in Figure 10 for the 134.2 kHz system show that there is a difference between the expected theoretical value and the measured value, especially as the distance increases. Yet, the same cannot be said for the results in Figure 11 for the 13.56 MHz system. In this latter case, the measurements seem to be lower than the expected value. It is not clear what is the cause, but this suggests that higher frequencies suffer higher attenuation in a manner not considered in the model.

As expected, a higher water salinity implies in a higher attenuation for the signal. Although the relative permittivity for seawater is slightly smaller than for freshwater, as can be observed in Figure 3, the conductivity for seawater is higher (see Figure 2) and dominates, increasing the attenuation factor. It is also worth noting that the 134.2 kHz system is not as affected by the increasing salinity of water as the 13.56 MHz system, as it can be seen looking at the values in Tables IV and V. This means that a MI communication system for a marine environment should be designed with a lower frequency, keeping in mind the needed bandwidth to transmit data and other requirements. This would ensure that the system is efficient, as there is lower energy being lost in the transmission.

Some authors argue that MI has a lower attenuation than RF for freshwater, and similar results for seawater [14]. This fact, combined with its immunity to multipath and fading, implies that MI could be a great alternative for wireless communications underwater. In addition, the achievable range of MI communications can be greatly extended by deploying waveguides that do not require power – simple passive relay coils that guide the magnetic field – such as demonstrated in [14][49]. For example, [14] uses an MI waveguide and achieves a range 26 times higher than a normal MI system. Another development that improves MI communications underwater is to use omnidirectional coils that remove the requirement of the transmitting and receiving coils being aligned [13][48].

However, to design an underwater RFID system it is required to balance a trade-off between range, transmission power and frequency (and therefore data rate and channel capacity). Nonetheless, the RFID system can always be engineered to achieve the best range given its power budget.

For an MI system, the size and number of turns of the transmitting and receiving coils also has an impact on the path loss. A bigger coil diameter increases the generated and captured magnetic flux for the transmitter/reader and receiver/tag, which increases the mutual induction and decreases the path loss. However, this is also a trade-off, as a bigger coil has a bigger internal resistance, requiring more power to transmit. It is also worth noting that some applications have size restrictions.

In the far-field category, the antenna can be carefully designed to provide the best radar cross-section, and therefore antenna gain, for the desired application. Again, the size of the antenna is important, as it is related to the wavelength. There is also a trade-off to be made for the frequency, attenuation, and antenna size: for a lower attenuation, the system would need a lower frequency, which requires a bigger antenna.

An example application that would benefit from MI communication over acoustic would be sensors deployed in coastal areas and fish farms [58]. In these environments, the acoustic noise – from waves, animal life and vessels – and the proximity with the water surface negatively impact acoustic underwater communications. In such scenarios, MI underwater communication would better fulfil the communication mechanisms for the of the system.

## VII. CONCLUSION AND FUTURE WORK

Given the existing challenges in wireless underwater communications, it is worth exploring alternatives, such as RFID. However, underwater RFID communication is not a well-explored topic in the scientific community. In this paper, we expanded on the existing theoretical model for RFID channel characteristics to account for the attenuation that the electromagnetic field suffers underwater. The water salinity is an important factor, which is used to calculate the dielectric permittivity and the conductivity of the water, and therefore, the attenuation. The RFID operation was separated into two categories: near-field and far-field. For both cases, the physical characteristics of the transmission were presented and from this, the equation for path loss was obtained.

In both technologies, the water salinity is a problem, as it increases the conductivity of the medium and, therefore, its attenuation of RFID signals. However, MI communication has advantages over RF in terms of immunity to multipath propagation and fading. In addition, a magnetic field can cross the air/water boundary, which is required for some applications. Therefore, near-field RFID communication is a promising alternative for underwater wireless communications.

The model presented in this paper considers that both the transmitting and receiving antennas are located underwater with no transition borders and other losses. This model could be expanded to account for transition borders such as the air-water interface located at the water surface or the interface with the waterproofing material of the reader and tag.

In this paper, we presented measurements for magnetic field strength for two near-field systems in different water salinity conditions. The results for the 134.2 kHz show that the attenuation may not be as strong as expected, especially for higher distances. But the results for the 13.56 MHz follow more closely the expected values, sometimes being even lower. This seems to imply that there may be a relationship between the attenuation factor and the frequency that is not currently expressed in the model. More experimental data is needed to draw any significant conclusions.

The results also suggest that a higher concentration of salt in the water increases the attenuation, which agrees with the model. However, the effect is more prominent the higher the frequency, which implies that the best communications solution for marine environments requires the use of lower frequencies to minimise attenuation.

## ACKNOWLEDGMENT

## REFERENCES

[1] C. Peres, J. Buckley, N. N. Rather, and B. O'Flynn, 'Theoretical models for underwater RFID', in *SENSORCOMM 2019*, Nice, France, Oct. 2019, pp. 80–88, Accessed: Sep. 01, 2020. [Online]. Available: https://cora.ucc.ie/handle/10468/9642.

[2] C. M. G. Gussen, P. S. R. Diniz, M. L. R. Campos, W. A. Martins, F. M. Costa, and J. N. Gois, 'A Survey of Underwater Wireless Communication Technologies', *JCIS*, vol. 31, no. 1, Art. no. 1, 2016, doi: 10.14209/jcis.2016.22.

[3] J. Heidemann, Wei Ye, J. Wills, A. Syed, and Yuan Li, 'Research challenges and applications for underwater sensor networking', in *IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006.*, Las Vegas, NV, USA, 2006, pp. 228–235, doi: 10.1109/WCNC.2006.1683469.

[4] M. C. Domingo, 'An overview of the internet of underwater things', *Journal of Network and Computer Applications*, vol. 35, no. 6, Art. no. 6, Nov. 2012, doi: 10.1016/j.jnca.2012.07.012.

[5] L. Lambo, Z. Shengli, and C. Jun-Hong, 'Prospects and problems of wireless communication for underwater sensor networks', *Wireless Communications and Mobile Computing*, vol. 8, no. 8, Art. no. 8, Oct. 2008, doi: 10.1002/wcm.654.

[6] X. Che, I. Wells, G. Dickers, P. Kear, and X. Gong, 'Re-evaluation of RF electromagnetic communication in underwater sensor networks', *IEEE Communications Magazine*, vol. 48, no. 12, Art. no. 12, Dec. 2010, doi: 10.1109/MCOM.2010.5673085.

[7] Y. Li, S. Wang, C. Jin, Y. Zhang, and T. Jiang, 'A Survey of Underwater Magnetic Induction Communications: Fundamental Issues, Recent Advances, and Challenges', *IEEE Communications Surveys Tutorials*, vol. 21, no. 3, Art. no. 3, thirdquarter 2019, doi: 10.1109/COMST.2019.2897610.

[8] A. Pozzebon, 'Bringing near field communication under water: short range data exchange in fresh and salt water', in *2015 International EURASIP Workshop on RFID Technology (EURFID)*, Oct. 2015, pp. 152–156, doi: 10.1109/EURFID.2015.7332401.

[9] D. Bertoni, G. Sarti, G. Benelli, A. Pozzebon, and G. Raguseo, 'Radio Frequency Identification (RFID) technology applied to the definition of underwater and subaerial coarse sediment movement', *Sedimentary Geology*, vol. 228, no. 3, Art. no. 3, Jul. 2010, doi: 10.1016/j.sedgeo.2010.04.007.

[10] E. B. Thorstad, A. H. Rikardsen, A. Alp, and F. Økland, 'The Use of Electronic Tags in Fish Research – An Overview of Fish Telemetry Methods', *Turkish Journal of Fisheries and Aquatic Sciences*, vol. 13, no. 5, Art. no. 5, 2013.

[11] S. J. Cooke, S. G. Hinch, M. C. Lucas, and M. Lutcavage, 'Biotelemetry and Biologging', in *Fisheries Techniques*, 3rd edition., American Fisheries Society, 2012, pp. 819–860.

[12] G. Benelli and A. Pozzebo, 'RFID Under Water: Technical Issues and Applications', in *Radio Frequency Identification from System to Applications*, M. I. B. Reaz, Ed. InTech, 2013.

[13]    I. F. Akyildiz, P. Wang, and Z. Sun, 'Realizing underwater communication through magnetic induction', *IEEE Communications Magazine*, vol. 53, no. 11, Art. no. 11, Nov. 2015, doi: 10.1109/MCOM.2015.7321970.

[14]    M. C. Domingo, 'Magnetic Induction for Underwater Wireless Communication Networks', *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 6, Art. no. 6, Jun. 2012, doi: 10.1109/TAP.2012.2194670.

[15]    B. Gulbahar and O. B. Akan, 'A Communication Theoretical Modeling and Analysis of Underwater Magneto-Inductive Wireless Channels', *IEEE Transactions on Wireless Communications*, vol. 11, no. 9, Art. no. 9, Sep. 2012, doi: 10.1109/TWC.2012.070912.111943.

[16]    Z. Sun and I. F. Akyildiz, 'Magnetic Induction Communications for Wireless Underground Sensor Networks', *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 7, Art. no. 7, Jul. 2010, doi: 10.1109/TAP.2010.2048858.

[17]    U. Azad, H. C. Jing, and Y. E. Wang, 'Link Budget and Capacity Performance of Inductively Coupled Resonant Loops', *IEEE Transactions on Antennas and Propagation*, vol. 60, no. 5, Art. no. 5, May 2012, doi: 10.1109/TAP.2012.2189696.

[18]    H. Nguyen, J. I. Agbinya, and J. Devlin, 'Channel Characterisation and Link Budget of MIMO Configuration in Near Field Magnetic Communication', *International Journal of Electronics and Telecommunications*, vol. 59, no. 3, Art. no. 3, Sep. 2013, doi: 10.2478/eletel-2013-0030.

[19]    D. Gawade *et al.*, 'A battery-less NFC sensor transponder for museum artefact monitoring - a review of NFC sensor technology and a proposed solution', in *SENSORCOMM 2019*, Nice, France, Oct. 2019, pp. 89–96, Accessed: Sep. 01, 2020. [Online]. Available: https://cora.ucc.ie/handle/10468/9634.

[20]    STMicroelectronics, 'ST25DV04K- Datasheet'. 2018, Accessed: Aug. 01, 2020. [Online]. Available: https://www.st.com/resource/en/datasheet/st25dv04k.pdf.

[21]    D. Sen, P. Sen, and A. M. Das, *RFID for energy & utility industries*. Pennwell Books, 2009.

[22]    S. A. Weis, 'RFID (radio frequency identification): Principles and applications', *System*, vol. 2, no. 3, pp. 1–23, 2007.

[23]    'ISO/IEC 18000-2:2009 Information technology — Radio frequency identification for item management — Part 2: Parameters for air interface communications below 135 kHz'. [Online]. Available: https://www.iso.org/standard/46146.html.

[24]    'ISO/IEC 18000-3:2010 Information technology — Radio frequency identification for item management — Part 3: Parameters for air interface communications at 13.56 MHz'. [Online]. Available: https://www.iso.org/standard/53424.html.

[25]    'ISO/IEC 15693-1:2010 Identification cards — Contactless integrated circuit cards — Vicinity cards — Part 1: Physical characteristics'. [Online]. Available: https://www.iso.org/standard/39694.html.

[26]    'ISO/IEC 14443-1:2018 Cards and security devices for personal identification — Contactless proximity objects — Part 1: Physical characteristics'. [Online]. Available: https://www.iso.org/standard/73596.html.

[27]    E. C. M. Association, *ECMA340–Near Field Communication Interface and Protocol (NFCIP-1)*. Genf, 2004.

[28]    K. Finkenzeller, *RFID Handbook: fundamentals and applications in contactless smart cards, radio frequency identification and near-field communication*. Hoboken, N.J.: Wiley, 2014.

[29]    P. Haigh, M. Hayes, D. Gawade, and B. O'Flynn, 'Towards autonomous smart sensing systems', in *2020 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2020, pp. 1–6.

[30]    Texas Instrument, 'Antenna Design Guide for the TRF79xxA'. Accessed: Aug. 01, 2020. [Online]. Available: http://www.ti.com/lit/an/sloa241b/sloa241b.pdf.

[31]    C. A. Balanis, *Advanced engineering electromagnetics*, 2nd ed. Hoboken, N.J: John Wiley & Sons, 2012.

[32]    E. E. Kriezis, T. D. Tsiboukis, S. M. Panas, and J. A. Tegopoulos, 'Eddy currents: theory and applications', *Proceedings of the IEEE*, vol. 80, no. 10, pp. 1559–1589, Oct. 1992, doi: 10.1109/5.168666.

[33]    P. Fofanoff and R. C. Millard Jr, 'UNESCO, Algorithms for computation of fundamental properties of seawater, UNSESCO Tech. Pap', *Marine Science*, no. 44, p. 55.

[34]    R. Somaraju and J. Trumpf, 'Frequency, Temperature and Salinity Variation of the Permittivity of Seawater', *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 11, Art. no. 11, Nov. 2006, doi: 10.1109/TAP.2006.884290.

[35]    P. J. W. Debye, *Polar molecules*. Chemical Catalog Company, Incorporated, 1929.

[36]    D. R. Lide, 'Handbook of Chemistry and Physics, CRC Press', *Boca Raton, FL*, vol. 1, no. 993, p. 1, 1993.

[37]    J. B. Hasted, S. K. Husain, F. A. M. Frescura, and J. R. Birch, 'The temperature variation of the near millimetre wavelength optical constants of water', *Infrared physics*, vol. 27, no. 1, pp. 11–15, 1987.

[38]    A. P. Stogryn, H. T. Bull, K. Rubayi, and S. Iravanchy, 'The microwave permittivity of sea and fresh water', *GenCorp Aerojet, Azusa, CA*, 1995.

[39]    D. G. Archer and P. Wang, 'The dielectric constant of water and Debye-Hückel limiting law slopes', *Journal of physical and chemical reference data*, vol. 19, no. 2, pp. 371–411, 1990.

[40]    H. J. Liebe, G. A. Hufford, and T. Manabe, 'A model for the complex permittivity of water at frequencies below 1 THz', *International Journal of Infrared and Millimeter Waves*, vol. 12, no. 7, pp. 659–675, 1991.

[41]    T. Meissner and F. J. Wentz, 'The complex dielectric constant of pure and sea water from microwave satellite observations', *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 9, pp. 1836–1849, 2004.

[42]    L. Klein and C. Swift, 'An improved model for the dielectric constant of sea water at microwave frequencies', *IEEE Transactions on Antennas and Propagation*, vol. 25, no. 1, pp. 104–111, 1977.

[43]    A. Stogryn, 'Equations for calculating the dielectric constant of saline water (correspondence)', *IEEE Transactions on Microwave Theory and Techniques*, vol. 19, no. 8, pp. 733–736, 1971.

[44]    'Recommendation ITU-R P.527-5'. International Telecommunication Union, 2019, Accessed: Aug. 01, 2020. [Online]. Available: https://www.itu.int/rec/R-REC-P.527-5-201908-I/en.

[45]  C. Uribe and W. Grote, 'Radio Communication Model for Underwater WSN', in *2009 3rd International Conference on New Technologies, Mobility and Security*, Dec. 2009, pp. 1–5, doi: 10.1109/NTMS.2009.5384789.

[46]  C. A. Balanis, *Antenna theory: analysis and design*, 3rd ed. Hoboken, NJ: John Wiley, 2005.

[47]  H. Lehpamer, *RFID design principles*, 2nd ed. Boston: Artech House, 2012.

[48]  H. Guo, Z. Sun, and P. Wang, 'Channel Modeling of MI Underwater Communication Using Tri-Directional Coil Antenna', in *2015 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2015, pp. 1–6, doi: 10.1109/GLOCOM.2015.7417399.

[49]  Z. Sun, I. F. Akyildiz, S. Kisseleff, and W. Gerstacker, 'Increasing the Capacity of Magnetic Induction Communications in RF-Challenged Environments', *IEEE Transactions on Communications*, vol. 61, no. 9, Art. no. 9, Sep. 2013, doi: 10.1109/TCOMM.2013.071813.120600.

[50]  R. Want, 'RFID Explained: A Primer on Radio Frequency Identification Technologies', *Synthesis Lectures on Mobile and Pervasive Computing*, vol. 1, no. 1, Art. no. 1, Jan. 2006, doi: 10.2200/S00040ED1V01200607MPC001.

[51]  K. V. S. Rao and P. V. Nikitin, 'Theory and measurement of backscattering from RFID tags', *IEEE Antennas and Propagation Magazine*, vol. 48, no. 6, Art. no. 6, Dec. 2006, doi: 10.1109/MAP.2006.323323.

[52]  'VNA master MS20xxC technical data sheet', manual, Aug. 2019.

[53]  '100 Series EMC Probes - Datasheet', Beehive Electronics, 100C, 2005.

[54]  'HDX RFID Reader System – Microreader RI-STU-MRD2', manual SCBU049, Aug. 2012.

[55]  'Pepper C1 user manual', manual SKU000395, Sep. 2019.

[56]  G. D. Durgin, 'ECE 3065 notes', TESSAL - Georgia Institute of Technology, manual.

[57]  A. Shaw, A. I. Al-Shamma'a, S. R. Wylie, and D. Toal, 'Experimental investigations of electromagnetic wave propagation in seawater', in *2006 European Microwave Conference*, 2006, pp. 572–575.

[58]  'IMPAQT Project'. https://impaqtproject.eu/about-impaqt/ (accessed Sep. 01, 2020).

# Suppressing Interference Between Periodic and Non-Periodic Traffic in Wireless Coexistence Scenarios

Ryota Ikeuchi and Hiroyuki Yomo

Graduate School of Engineering Science, Kansai University

3-3-35 Yamate-cho, Suita, Osaka, 564-8680 Japan

e-mail: {k097114, yomo}@kansai-u.ac.jp

*Abstract*—In this paper, we consider a wireless coexistence scenario where multi–radio platforms are employed to simultaneously support periodic and non–periodic traffic. Considering a scenario where wireless terminals generating periodic traffic over one frequency band change their operating band to the other band after detecting long–term communication failures, we consider how to suppress mutual interference between periodic and non–periodic traffic over the shared channel. In this paper, we propose a transmission control alleviating negative impact of mutual interference by exploiting interface heterogeneity, traffic periodicity, and queue management. As a means to suspend packet transmissions, we propose two types of queue management: transmission control in application layer and medium access control layer. The proposed schemes realize high packet delivery ratio of periodic traffic by suppressing transmissions of terminals with non–periodic traffic at the timing when periodic traffic is expected to be transmitted by their hidden terminals. We also investigate the feasibility of the proposed schemes with experiments. With computer simulations and experiments, we investigate the practicality and effectiveness of the proposed schemes.

*Keywords—Wireless Coexistence; Factory Automation; IEEE 802.11; IEEE 802.15.4; Internet of Things; Experimental Studies*

## I. Introduction

This paper is an extended version of [1], which investigates transmission control to suppress interference between heterogeneous traffic generated by wireless devices employing different access technologies. The proliferation of diverse wireless access technologies, such as LTE, WiFi, ZigBee, Bluetooth, etc., has been accelerated during the last decade to support heterogenous traffic with different requirements. Today, we have an option to simultaneously exploit these technologies with multi–radio platforms [2][3]: for instance, small, low–price IoT devices, which are equipped with multiple interfaces operating over different frequency bands, such as 2.4/5GHz and 920MHz, are commercially available [4].

In this paper, we exploit multi–radio platforms to enhance robustness of wireless networks in a highly noisy environment. A typical use–case is factory [5], where there are many metal objects blocking communication links between transmitters and receivers [6]. Furthermore, there can be noise emitted from industrial machines, as well as interference from many radio equipment around a factory. The resulting instability of communication channels causes temporal communication failure, which can last for a long period of time. If we employ wireless devices with a single interface operating over a specific frequency band in such an unstable environment,

we cannot offer reliable transmissions of data: once blocking, noise or interference is generated over an operating frequency band, each device cannot avoid them. The lack of reliability for data transmissions in a factory can result in serious incidents that could even cause human life to be in danger. Therefore, in our work, we focus on the usage of wireless devices equipped with multiple radio interfaces operating at different frequency bands, called Flexible Terminal (FT). With FT, even if noise or interference is generated over one frequency band, its operating band can be changed to the other frequency band, which enables us to avoid communication failures due to noise and interference. More specifically, we employ radio standards operating at unlicensed frequency bands: IEEE 802.11 at 2.4 GHz and IEEE 802.15.4g at 920MHz since these standards are widely employed in many industrial fields [7].

Besides the heterogeneity of radio interface, the heterogeneity of communication traffic has become a common trend in current wireless networks. In addition to non–periodic (bursty) traffic generated by classical applications, such as Internet access and video/image transfer, more deterministic and periodic traffic has become a dominant pattern especially in a scenario with sensor devices deployed for monitoring purpose [8][9]. In general, small amount of data is generated by sensor devices, for which 920MHz radio supporting low data rate with large coverage is a favorable option. On the other hand, 2.4GHz commonly used by WiFi offers higher data rate with smaller coverage than 920MHz, which makes it suited for supporting Internet access and transfer of large–size image/video files. In this work, we employ FTs to simultaneously support periodic and non–periodic traffic. In a normal operation mode without any noise or interference, FTs with non–periodic traffic employ an interface operating at 2.4 GHz while FTs with periodic traffic use an interface operating at 920MHz. Then, we consider a scenario where noise or interference is generated by surrounding devices/machines over 920MHz, and each FT with periodic traffic changes its operating interface to that at 2.4GHz. In this case, there is mutual interference between FTs with periodic traffic and FTs with non–periodic traffic. In this work, we propose a transmission control, which suppresses mutual interference by exploiting interface heterogeneity, traffic periodicity, and queue management. In the proposed scheme, FTs with non–periodic traffic detect possible hidden FTs with periodic traffic by using difference of propagation characteristics of different frequency bands. Then, FTs with non–periodic traffic predict the transmission timing of FTs with periodic traffic, and

suppress their packet transmissions at the predicted timing with adaptive queue management. With computer simulations and experiments, we investigate the practicality and possible gain of the proposed scheme. The new contributions of this paper in comparison to [1] are as follows.

- Besides the application–level queue management proposed in [1], in this paper, we propose a medium access control (MAC)–level queue management for our transmission control. With computer simulations, we show that the proposed MAC–level queue management achieves the best performance in terms of packet delivery ratio and throughput.
- Although the proposed MAC–level queue management achieves the best performance, it requires the queue management at physical layer (PHY)/MAC layer. In this paper, we investigate the feasibility of queue management at PHY/MAC layer by using a WiFi/Bluetooth coexistence function prepared in a off–the–shelf WiFi module. With the additional experiments, we show that the queue management at PHY/MAC layer is feasible, which confirms that the MAC–level queue management with WiFi/Bluetooth coexistence function is a promising means of interference management.
- More detailed explanations on the proposed schemes and experimental setting with additional figures are presented.

The rest of the paper is organized as follows. We review related work in Section II. After describing the system model in Section III, we present our proposed transmission control in Section IV. After showing and discussing some simulation results in Section V, Section VI presents the feasibility study of the proposed scheme with experiments. Finally, Section VII concludes the paper with several future work.

## II. STATE OF THE ART

The periodicity of traffic has been exploited to avoid packet collisions among wireless devices in several existing studies. Most of them propose to schedule/shift the timing of packet generations of periodic flow so that they are not overlapped over time. These approaches can be categorized into the application–level and MAC–level. The MAC–level approaches are difficult to implement into the current wireless standards as it requires the modification of MAC protocols. On the other hand, the application–level approaches are easy to implement since it can be implemented over the existing MAC protocols. For instance, a Time Division Multiple Access (TDMA)–based MAC protocol with scheduling of periodic flows to overcome packet contentions has been proposed in [10]. A self–organizing TDMA protocol supporting periodic message exchange in vehicular networks is analyzed in [11]. As an application–level approach, a scheduling method is introduced in [12], where packet creation timing of periodic flow is adjusted in order to reduce contentions and packet collisions. However, these works only consider avoiding collisions among periodic flow and the coexistence and packet collisions with non–periodic traffic are not investigated.

Another related work to our study is the investigation on hidden terminal problem in carrier sense multiple access (CSMA) networks. The most well–known solution to hidden terminal problem is RTS/CTS handshake defined in IEEE 802.11 [13]. However, it has been reported that the efficiency of RTS/CTS handshake is low when short packets, such as small amount of data generated by sensor nodes in our scenario, are involved in data transmissions [14]. Furthermore, RTS/CTS mechanism does not fundamentally solve problems on collisions among hidden terminals: RTS frames transmitted by hidden terminals can collide with high probability [15]. Another requirement specific to industrial applications is more strict and deterministic protection for sensing data in comparison to Internet access/file transfer [16], which is difficult to achieve with RTS/CTS handshake even with quality of service (QoS) differentiation defined in IEEE 802.11e [17]. In contrast, our work proposes a mechanism to deterministically avoid interference between hidden terminals without resorting to RTS/CTS mechanisms.

## III. SYSTEM MODEL

In this section, we describe the system model considered in this paper.

### A. System Model

In this work, we employ FTs with interfaces operating at 2.4GHz and 920MHz. In general, 920MHz signals have larger propagation distance than 2.4 GHz while the former achieves lower data rate than the latter. We consider a factory–like indoor area where FTs and a single Flexible Gateway (FG), which is in charge of aggregating data generated by FTs, are deployed as shown in Figure 1. The FG is also equipped with 2.4GHz and 920MHz interfaces to receive data from FTs. Some FTs are supposed to generate non–periodic, bursty, and heavy–load traffic, which are called NP–FTs. Since this type of traffic is in general supported by higher PHY rate at 2.4GHz that has limited communication range, we assume that NP–FTs are deployed near FG. On the other hand, FTs except for NP–FTs are assumed to generate periodic, light–load traffic, which are called P–FTs. A typical example of P–FT is a sensor device generating monitoring data of industrial machines and/or a given environment, which are deployed at various places within an area. This requires P–FTs to employ an interface and/or parameters realizing a larger communication range, for which 920MHz is more favorable option. We assume that the information on period of P–FT's traffic is known and shared by all FTs/FG. This is a reasonable assumption since these terminals and gateway are considered to be deployed by a single administrator of a factory. Furthermore, the timing of packet–generations of P–FTs are controlled to be equally separated over time so that they are not overlapped. This enables us to avoid contention among P–FTs. In a normal operation mode, NP–FTs employ 2.4GHz interface while P–FTs utilize 920MHz interface. Here, 2.4 GHz interface is supposed to follow IEEE 802.11 PHY/MAC protocol while 920MHz interface is in accordance with IEEE 802.15.4g/e PHY/MAC protocol. Note that both of these standards employ CSMA with collision avoidance (CSMA/CA) protocol. The FG receives data from both NP–FTs and P–FTs by using its
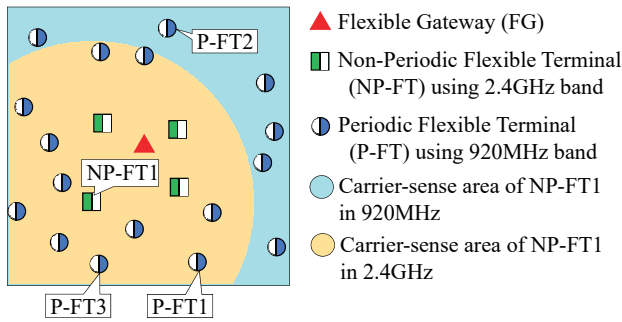
Figure 1. The considered, factory–like system model.



Figure 2. An example of hidden terminal detection.

two interfaces. It is assumed that the carrier–sense range of 2.4GHz interface is smaller than that of 920MHz as shown in Figure 1: for example, carrier–sense range of NP–FT1 in Figure 1 over 920MHz is sufficiently large to detect signals transmitted by all terminals while it can only sense signals transmitted by a part of terminals over 2.4GHz.

In this work, we consider a scenario where severe noise/interference is caused over 920MHz, which can be emitted from industrial machines and/or radio devices deployed inside/outside a factory area, and 920MHz interface suffers from continuous communication failures for a long period of time. As mentioned in Section I, FTs are able to switch their operating interface. Therefore, P–FTs, which operate with 920MHz interface in a normal operation mode, can switch their operating interface to 2.4GHz, e.g., after detecting continuous packet errors or after receiving some instruction if there is a central entity to monitor the radio environment. Here, each P–FT is assumed to employ low PHY rate (e.g., 1Mbps) at 2.4GHz, which enables each P–FT to achieve sufficiently large communication range to transmit data to FG. However, when P–FTs and NP–FTs share the same 2.4GHz frequency band, another problem can occur, which is a hidden terminal problem. For example, as shown in Figure 1, NP–FT1 and P-FT2 cannot sense their signals with each other at 2.4GHz. Therefore, CSMA/CA mechanisms do not work properly among these nodes after P–FT2 changes its operating band to 2.4GHz, which can cause packet losses at FG, thereby degrading packet delivery ratio and throughput. In this work, we propose a mechanism to suppress transmissions of NP–FTs to avoid interference with hidden P–FTs by exploiting interface heterogeneity, traffic periodicity, and adaptive queue management.

## IV. PROPOSED TRANSMISSION CONTROL

The proposed scheme controls packet transmissions of NP–FTs in order to suppress interference with their hidden P–FTs.

### A. Mechanism to Detect Hidden Terminals

The NP–FTs first need to identify possible hidden terminals in order to suppress their mutual interference. This is achieved by exploiting the heterogeneity of interface. Each NP–FT observes traffic over 920MHz and 2.4GHz while they are not transmitting their own data. In the normal operation mode, P–FTs transmit data at 920MHz. In this case, each NP–FT
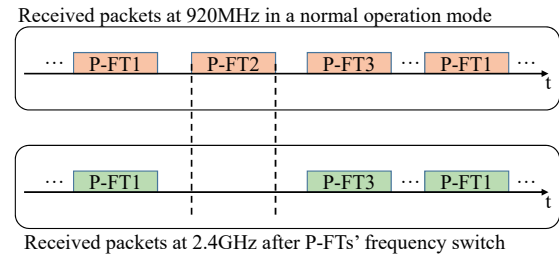
finds packets of all P–FTs over 920MHz since they can easily reach each NP–FT thanks to a large communication range of 920MHz. An example of packet receptions at NP–FT1 shown in Figure 1 in a normal operation mode is depicted in the upper part of Figure 2. Here, P–FT1, P–FT2, and P–FT3 are the terminals whose locations are specified in Figure 1. As shown in Figure 2, NP–FT1 observes periodic receptions of all P–FTs at 920MHz interface. After P–FTs detect noise/interference at 920MHz, they switch their interfaces to 2.4GHz, where NP–FT1 receives packets only from P–FTs located within its communication range at 2.4GHz. The lower part of Figure 2 shows an example of packet receptions at NP–FT1 over 2.4GHz. As shown in the figure, NP–FT1 cannot receive packets transmitted by P–FT2 since P–FT2 is out of communication range of NP–FT1. Then, NP–FT1 finds that it has a hidden terminal of P–FT2 over 2.4GHz. At this timing, NP–FTs can also find that P–FTs have changed their operating band to 2.4GHz. Thus, by comparing packet receptions at 920MHz and 2.4GHz, each NP–FT can identify its hidden terminals over 2.4GHz, whose packets can cause collisions against itself.

### B. Basic Idea of Proposed Transmission Control

While receiving packets from P–FTs in the normal operation mode, each NP–FT records the reception timing of each P–FT. Based on this information and pre–knowledge of the period of packet transmissions of each P–FT, each NP–FT predicts the timing of periodic packet transmissions. Then, each NP–FT suppresses its packet transmissions when the transmissions of its hidden P–FTs are expected. This is achieved by our proposed Transmission Control (TC), which executes queue management to control timing of packet transmissions at different layers.

The basic idea of the proposed TC is shown in Figure 3. Here, the blue solid arrow shows the predicted transmission timing of a hidden P–FT. With the proposed TC, a duration called Suspending Duration (SD), which consists of Pre–SD (before the predicted timing) and Post–SD (after the predicted timing) is prepared. A NP–FT attempts to suspend its packet transmission over SD with queue management described later. In Figure 3, the dashed green arrow represents the timing when packets are generated at upper layer of NP–FT. Once SD is over, NP–FT starts transmitting packets. Note that packets generated at non–SD duration can be immediately transmitted as in the packet P4 in Figure 3. The flowchart of these operations of the proposed transmission control is

Figure 3. Basic idea of the proposed transmission control.



Figure 4. A flowchart of basic operations of the proposed transmission control.



(a) Application-level Transmission Control (ATC)

(b) MAC-level Transmission Control (MTC)

Figure 5. Proposed queue management for transmission control: (a) application–level transmission control (ATC) and (b) MAC–level transmission control (MTC).
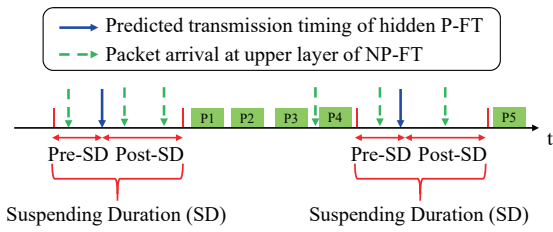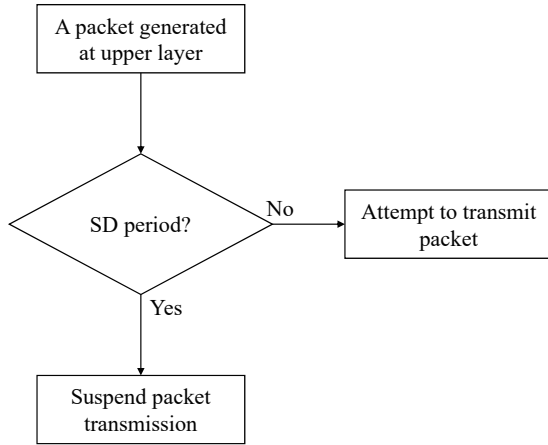


Figure 6. An example of problem on controlling packet transmissions with ATC.

shown in Figure 4. The duration of Pre–SD and Post–SD are decided considering trade–off between achievable Packet Delivery Ratio (PDR) of P–FTs and throughput of NP–FTs as discussed in Section V-B in more detail.

As a means to suspend packet transmissions during SD, we propose two types of queue management: application–level TC (ATC) and MAC–level TC (MTC), which are respectively depicted in Figure 5 (a) and (b). With ATC, even if packets are generated at upper layer, they are stored into upper layer queue without passing them into PHY/MAC layer. On the other hand, with MTC, packets are passed from upper layer to PHY/MAC layer even during SD. However, each NP–FT suspends packet transmissions at PHY/MAC level, i.e., it does not transmit any packet over the air (i.e., wireless channel) during SD. Note that MTC requires us to modify firmware installed into WiFi module/chip so that we can arbitrarily control transmission timing of packets at PHY/MAC level. Thus, MTC has higher complexity of implementation than ATC.

### C. Drawback of ATC

Although ATC has lower complexity than MTC as described above, it has difficulty to precisely control the timing when signals are actually transmitted at PHY/MAC level. This problem is explained in an example shown in Figure 6. Here, a NP–FT suspends passing packets to PHY/MAC layer during the first SD, and three packets are stored in the upper–layer queue. These packets are passed to PHY/MAC layer after the first SD is over, which are then stored in PHY/MAC queue. The transmissions of packets in PHY/MAC queue are managed by PHY/MAC module, therefore, these packets are transmitted

if they win contentions against the other terminals. In the example of Figure 6, it is supposed that NP–FT succeeds in transmitting a packet P1 by winning the contention. However, it fails to transmit packets P2 and P3 due to the lost contentions with BackGround (BG) traffic. Then, these 2 packets remain in PHY/MAC queue in the beginning of the next SD. As mentioned above, packet transmissions of these lower–layer packets are controlled by PHY/MAC module, therefore, they can be transmitted even during SD, which can cause a collision with packets transmitted by hidden P–FTs.

A possible solution to the above–mentioned problem is to control the number of packets to be passed to PHY/MAC layer module based on the congestion level over the channel, i.e., each NP–FT controls the number of packets passed to PHY/MAC layer module in the end of SD in such a way that these packets can be transmitted in the following non–SD period at the PHY/MAC level. This requires each NP–FT to continuously monitor the congestion level over the operating channel. Note that background traffic at 2.4GHz are not necessarily generated by WiFi terminals, whose packets can be decoded by NP–FT, but generated by the other radio equipment, e.g., Bluetooth or Microwave oven. In this case, each NP–FT needs to monitor the congestion level without decoding each background signal. Therefore, in the following subsection, we first investigate whether it is practically possible for a WiFi terminal to conduct real–time monitoring of busy rate (i.e., fraction of time, during which the channel is occupied by radio signals) of a channel.
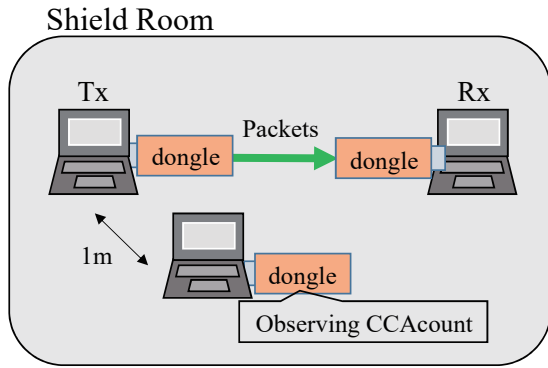
Figure 7. Experimental setting to investigate the feasibility to use CCAcount as a measure of busy rate.



Figure 8. Experimental results on CCAcount against traffic load.

## D. Feasibility to Monitor Congestion Level

We found a parameter called CCAcount in a device driver of an off–the–shelf WiFi module (Buffalo WL–U3–866DS [18]). The parameter seems to be related to busy rate of a channel, however, there was no evidence that this parameter represents our desired information on busy rate. Therefore, we conducted experiments to check the relationship between CCAcount and busy rate of a channel. The experimental setting is shown in Figure 7. In the experiments, we prepared 3 laptop PCs with USB dongles of WL–US–866DS. A laptop PC (Tx PC) was configured to be a transmitter of packets, which are directed to Rx laptop PC. A laptop PC to observe CCAcount was located at a sufficiently close position to Tx PC. The busy rate was varied by changing the number of packets transmitted per a unit time, for which the output of CCAcount was monitored at the observing PC. The PHY rate, packet size, and ACK size of packet transmissions were respectively set to be 54Mbps, 1496Bytes, and 46Bytes. The busy rate for each traffic load can be calculated based on these parameters. The measurements were conducted inside a shield room.

Figure 8 shows the output of CCAcount against traffic load (packets/s). From this figure, we can see that CCAcount increases as traffic load increases, which saturates over the range of high traffic load. There is a maximum traffic load that can be generated by a single WiFi terminal, which depends on back–off parameters and Inter–Frame Space (IFS) of IEEE 802.11, where the saturation is observed. From this figure, we can confirm that there is a direct relationship between CCAcount and traffic load, i.e., busy rate of the channel, which enables us to employ CCAcount as a measure of busy rate of the channel.

## E. ATC with traffic adaptation

In this work, we introduce traffic adaptation into ATC, which controls the number of packets to be passed to PHY/MAC layer based on the observed CCAcount. In ATC, each NP–FT observes CCAcount during each non–SD period. This can be realized only by obtaining the corresponding information from WiFI device driver. The output of CCAcount is converted to the traffic load by using a linear equation approximating the relationship between CCAcount and traffic
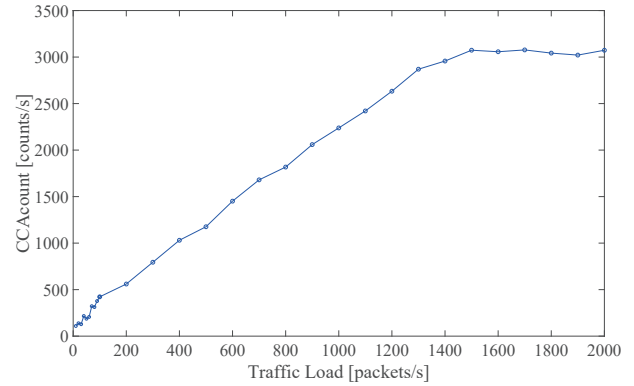
load over the load range of [0:1500] packets/s in Figure 8, which is used to calculate the busy rate. Based on the derived busy rate, the maximum number of packets permitted to be passed to PHY/MAC layer at the next non–SD period, $N_{max}$, is decided. $N_{max}$ is calculated as follows:

$$N_{max} = \frac{(1 - B_{ave})T_{NSD}}{T_D \cdot \alpha}. \qquad (1)$$

Here, $T_{NSD}$ is the duration of next non–SD period, $T_D$ is the duration required to transmit a single data frame including SIFS and ACK duration, and $\alpha$ is a parameter to vary effective number of $N_{max}$, and $B_{ave}$ is average busy rate calculated as

$$B_{ave} = \frac{\sum_{i=1}^{W} B_i}{W}, \qquad (2)$$

where $B_i$ is busy rate calculated for the $i$-th last non–SD period, and $W$ is the window size (number of non–SDs) used for calculating average busy rate. $N_{max}$ calculated with (1) represents the estimated (effective) number of packets that can be transmitted by a single NP–FT during free period in the following non–SD period. Note that $\alpha$ is introduced in order to take the impact of back–off duration and number of contending FTs into account. With smaller (larger) $\alpha$, the estimation of $N_{max}$ becomes more optimistic (pessimistic). The range of $\alpha$ considered in this paper is set to [0.4, 6.0].

The proposed ATC is executed in the end of every SD period. For instance, in the end of SD1 in Figure 6, $N_{max}$ is calculated by using busy rate over the last $W$ non–SD periods. Then, if the number of packets stored in the upper–layer queue is equal to or more than $N_{max}$, only $N_{max}$ packets out of stored packets are passed to PHY/MAC layer, and no more packets are passed to PHY/MAC layer during the following non–SD period. Otherwise if the number of packets stored in the upper–layer queue is less than $N_{max}$, all stored packets are passed to PHY/MAC layer. Then, newly arriving packets in the following non–SD period can be passed to PHY/MAC layer as long as the total number of packets passed to PHY/MAC layer does not exceed $N_{max}$. With these operations, we can reduce the probability that packets remain in PHY/MAC queue in the end of each non–SD period.
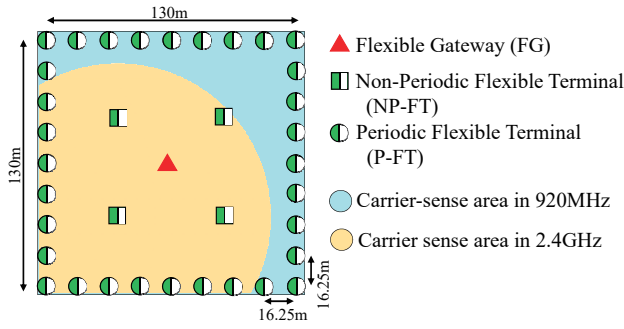
Figure 9. Simulation Model.

TABLE I: Simulation Parameters

|  | NP–FT | P–FT |
|---|---|---|
| PHY rate | 54Mbps | 1Mbps |
| Communication range | 75m | 100m |
| Carrier-Sense Range | 100m | 100m |
| Packet generation | Poisson (mean $\lambda$) | period = 1s |
| Data size | 2000Bytes | 200Bytes |
| ACK size | 30Bytes | |
| DIFS | $28\mu s$ | |
| SIFS | $10\mu s$ | |
| Slot time | $20\mu s$ | |
| Max. Num. of Retransmissions | 3 | |
| Min. Contention Window | 31 | |
| Simulation Duration | 20s | |

## V. SIMULATION MODEL AND RESULTS

In this section, we provide numerical results obtained by our computer simulations, and discuss the benefit brought by the proposed transmission control in detail.

### A. Simulation Model

The simulation model is shown in Figure 9. The layout given in Figure 9 is selected since it can increase the number of hidden terminals, which allows us to consider a worst–case scenario. In the simulations, communication performance after P–FTs change their operating frequency band to 2.4GHz is evaluated. The main parameters used in simulations are shown in Table I. Most of the parameters are taken from the IEEE 802.11g standard [19]. The P–FTs generate packets with period of 1s, and their generation timing are scheduled so that they do not overlap with each other. In the evaluation, since there are 32 P–FTs, a period of 1s is divided into 32 sections, and the beginning of each section is randomly assigned to each P–FT as its generation timing. Each NP–FT applies the proposed ATC/MTC to its hidden P–FTs. We use the application–level PDR of P–FT and throughput of NP–FT as performance measures. A packet is decided to be lost and discarded once the number of retransmissions reaches the maximum value. For simplicity, packet errors are assumed to occur only due to collisions. The throughput is defined as the amount of data successfully delivered by NP–FTs to FG. The simulation is conducted by a custom–made simulator developed with Matlab software.
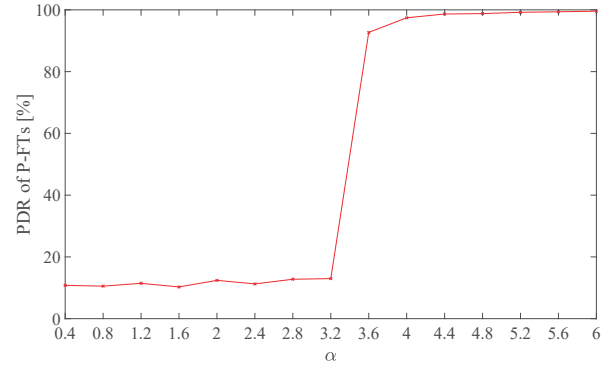


Figure 10. PDR of NP-FTs against $\alpha$ for ATC.
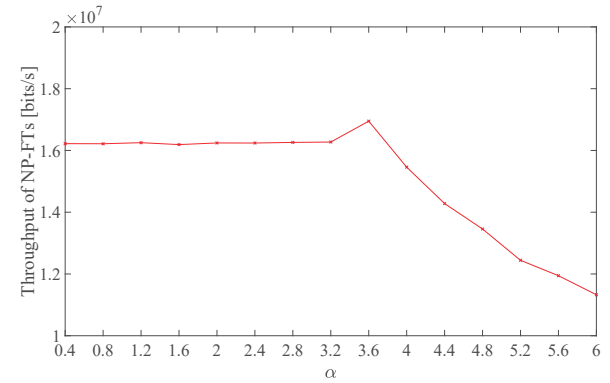


Figure 11. Throughput of NP-FTs against $\alpha$ for ATC.

### B. Simulation Results

Below, we show simulation results averaged over 5 simulation trials. Figure 10 shows PDR of P–FTs against the parameter of $\alpha$ in (1) when the proposed ATC is employed with Pre–SD = 2ms, Post–SD = 6ms, $W = 10$, and $\lambda = 400$ [packets/s]. From Figure 10, we can see that PDR of P–FT is degraded with smaller $\alpha$. With smaller $\alpha$, each NP–FT passes a larger number of packets to PHY/MAC layer in the end of SD as calculated by (1), which exceeds the number of packets that can be transmitted at PHY/MAC layer during the next non–SD period. In this case, packets remained in PHY/MAC queue can be transmitted simultaneously with hidden P–FTs, which causes collisions with high probability. This problem is alleviated by increasing the value of $\alpha$ where the number of packets passed to PHY/MAC layer is reduced. Therefore, PDR of P–FT is improved with larger value of $\alpha$. However, larger values of $\alpha$ force each NP–FT to keep more packets in its upper–layer queue, and degrade its throughput performance. This is confirmed in Figure 11, where throughput of NP–FTs against $\alpha$ is shown. The throughput of NP–FTs is largely degraded with too large $\alpha$, i.e., the range of $\alpha$ exceeding 3.6. From these results, we can see that there is an appropriate value of $\alpha$ to be employed to achieve both high PDR of P–FTs and high throughput of NP–FTs. In the following evaluations, we employ $\alpha = 3.6$ based on the above results.

Next, we investigate the impact of SD length on the achievable performance of the proposed ATC. We have observed the same tendency for ATC and MTC, therefore, we only
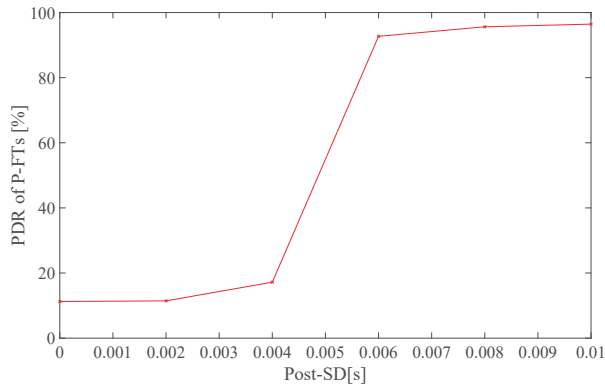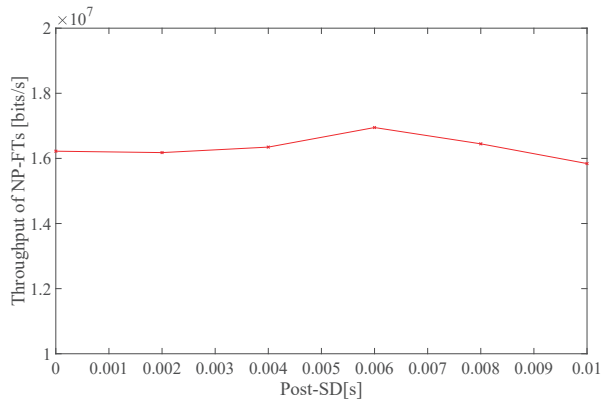
Figure 12. PDR of P-FTs against Post-SD for ATC.



Figure 14. PDR of P-FTs against traffic load of NP-FTs.



Figure 13. Throughput of NP-FTs against Post-SD for ATC.

show results of ATC here. Figures 12 and 13 respectively show PDR of P–FTs and throughput of NP–FTs against the length of Post–SD, where Pre–SD is fixed to be 2ms, $W = 10$, $\alpha = 3.6$, and $\lambda = 400$ [packets/s]. First, from Figure 12, we can see that a sufficiently large value of Post–SD is required to achieve high PDR of P–FTs. Each packet generated at P–FT is transmitted with CSMA/CA protocol, where its actual transmission timing at PHY/MAC level can be delayed due to contentions with the other NP–FTs and P–FTs within its carrier–sense range. Therefore, if NP–FT employs too small Post–SD, it can transmit packets with hidden P–FTs whose transmissions are delayed due to CSMA/CA operations. The increase of Post–SD also offers the improvement on throughput as shown in Figure 13 thanks to higher probability to avoid mutual collisions, however, too large Post–SD leads to the reduction of throughput of NP–FTs since it can reduce the duration for NP–FTs to be able to transmit their packets. From these figures, we can see that Post–SD of 6ms is an appropriate choice in our considered settings.

Finally, we respectively show PDR of P–FTs and throughput of NP–FTs against traffic load of NP–FTs for different schemes in Figures 14 and 15. Here, we set Pre–SD = 2ms, Post–SD = 6ms, $W = 10$, and $\alpha = 3.6$. The results of W/O TC in these figures represent achievable performance of an existing scheme, which follows conventional IEEE 802.11 MAC protocol without employing our proposed TC. We also show results of ATC without traffic adaptation in these figures.

From Figure 14, we can first see that PDR of P–FTs is largely degraded if we do not employ TC. This is due to packet collisions between NP–FTs and their hidden P–FTs. By introducing ATC, PDR of P–FTs can be improved, however, we can obtain gain only over the range of small traffic load of NP–FTs if we do not introduce traffic adaptation into ATC. As the traffic load of NP–FTs increases, more packets are stored in the upper–layer queue in the end of each SD, which can exceed the number of packets that can be handled at PHY/MAC level during the following non–SD period. Therefore, more collisions occur for larger traffic of NP–FTs, which degrades PDR of P–FTs. On the other hand, it can be seen that the proposed ATC with traffic adaptation achieves high PDR of P–FTs even for larger traffic load of NP–FTs thanks to the adjustment of number of packets passed to PHY/MAC queue, which is adapted to the observed traffic load. The proposed MTC achieves the highest PDR since it can stop/start the transmissions of packets at PHY/MAC level according to the schedule of SD and non–SD. However, we can see that the proposed ATC also achieves PDR close to MTC. Next, from Figure 15, we can see that the proposed ATC does not degrade throughput of NP–FTs even with the introduction of SD. The avoidance of collisions eventually leads to throughput improvement. With the proposed ATC, packets are stored in the upper–layer queue according to the estimated traffic load. If the actual traffic load is smaller than the estimated value, all packets passed to PHY/MAC queue can be transmitted at early timing within a non–SD period, after which no packet is transmitted since there is no packet in PHY/MAC queue. This problem does not occur with the proposed MTC, therefore, throughput of the proposed ATC does not reach close to MTC. From these results, we can confirm that the proposed ATC, which has lower complexity than MTC, can significantly improve PDR of P–FTs while achieving slightly better throughput of NP–FTs in comparison to the the case without TC. Furthermore, the proposed MTC has the highest PDR and throughput at the cost of complexity of implementation.

## VI. FEASIBILITY STUDY OF MTC WITH EXPERIMENTS

In the previous section, we have shown that the proposed MTC achieves the best PDR and throughput. However, the
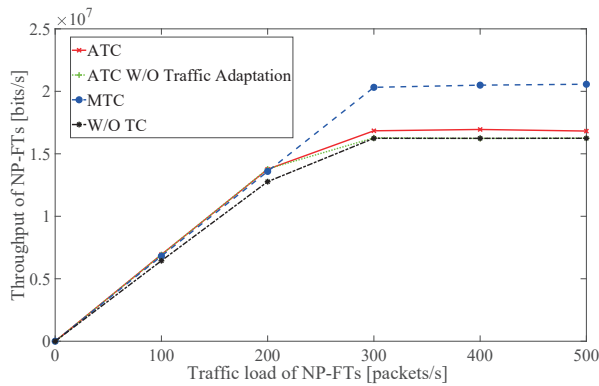
Figure 15. Throughput of NP-FTs against traffic load of NP-FTs.

proposed MTC requires us to control the transmissions of packets at the lowest level of protocol stack, i.e., to conduct queue management at PHY/MAC level. In this section, we investigate the feasibility of queue management at PHY/MAC layer with experiments.

### A. Experimental Setting

In order to realize the queue management at PHY/MAC layer, we attempt to utilize a function of WiFi/Bluetooth coexistence implemented in some WiFi/Bluetooth combined modules. This function is prepared for a module to stop the transmissions of WiFi packets while transmitting Bluetooth signals, thereby avoiding interference between them. The basic operation to realize queue management at PHY/MAC layer by exploiting WiFi/Bluetooth coexistence function is shown in Figure 16. In general, WiFi PHY/MAC module has layered structure of queue management. The lowest queue is called as CSMA queue, which stores a packet to be transmitted if the corresponding node wins the contention through CSMA operations. The other packets are stored in PHY/MAC queue, waiting for the process of CSMA. The WiFi/Bluetooth coexistence function is able to suspend passing packets from PHY/MAC queue to CSMA queue while it transmits Bluetooth signal. In this work, we configure the WiFi/Bluetooth coexistence function so that it can output the suspending command to WiFi module at arbitrary timing even if there is no actual transmission of Bluetooth signal. This way, we can control the timing for PHY/MAC queue to pass a packet to CSMA queue. Although we cannot control the transmission of CSMA queue once a packet is inserted into it, we can minimize the number of packets out of control by using this function. The function is implemented by using JWX6051 module with AR9380 chip onboard [20].

The experimental setting is shown in Figure 17. In the experiment, we use three JWX6051 WiFi modules. WiFi Module 1 is supposed to be a transmitter of periodic traffic (i.e., P–FT) while WiFi Module 2 is assumed to be a transmitter of non–periodic traffic (i.e., NP–FT). As described in Section III, we consider a scenario where these two modules are hidden with each other. In order to construct a situation where these two modules are hidden with each other, they are respectively put
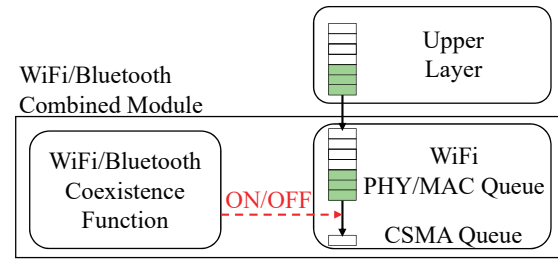


Figure 16. The queue management at PHY/MAC layer exploiting WiFi/Bluetooth coexistence function.
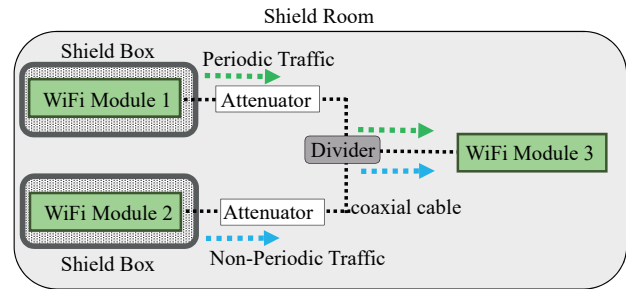


Figure 17. The experimental setting of feasibility study of MTC.

into shield boxes. This way, one module cannot sense radio signals transmitted by the other module. In order to extract radio signals transmitted by each module, they are connected to coaxial cable. These cables are combined by a divider after attenuation. Finally, the combined signals are received by WiFi Module 3 through a coaxial cable. With this setting, we can imitate a situation where P–FT and NP–FT are hidden with each other. All the experiments are conducted inside a shield room. The parameters employed in the experiments are shown in Table II.

### B. Experimental Results

In the experiments, the periodic traffic is generated by WiFi Module 1 with the interval of 1.024s. After 100 packets are transmitted by WiFi Module 1, its transmission timing is shifted by 0.1s. This is repeated 10 times. This realizes 10 different gaps between the generation timing of periodic traffic and SD set by WiFi Module 2 for the transmissions of non–periodic traffic.

Figure 18 shows PDR of periodic traffic over time when MTC exploiting WiFi/Bluetooth coexistence function is em-

TABLE II: Parameters for experiments

| | |
|---|---|
| Frequency band | 2.4 GHz |
| Data Size | 1554 Bytes |
| Data Rate of Non–periodic Traffic | 18 Mbps |
| Data Rate of Periodic Traffic | 1 Mbps |
| ACK Size | 46 Bytes |
| ACK Rate of Non–Periodic Traffic | 12 Mbps |
| ACK Rate of Periodic Traffic | 1 Mbps |
| Generation Rate of Non-Periodic Traffic | 200 packets/s |
| Period of SD | 1 s |
| SD | 0.2 s |
| Non–SD | 0.8 s |
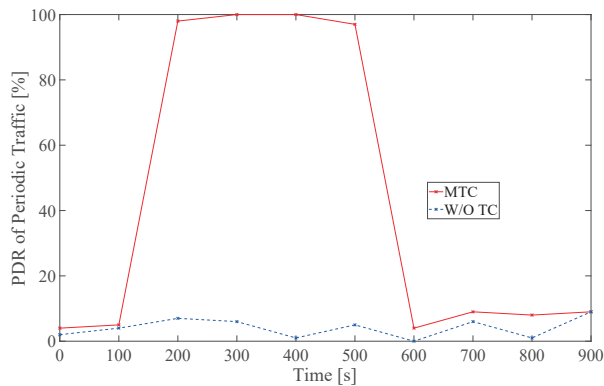
Figure 18. PDR of periodic traffic against time shift.

ployed. We also plot experimental results without employing TC in the same figure. Each value in the horizontal axis corresponds to the gap between the transmission timing of periodic traffic and SD set for MTC. From this figure, we can first see that PDR of periodic traffic is largely degraded when TC is not employed. This is because WiFi Module 1 and Module 2 are hidden with each other, and their transmitted packets collide at WiFi Module 3 with high probability if we do not employ any transmission control. Some periodic packets can be received successfully when they are transmitted during back–off periods of non–periodic traffic, therefore, PDR is larger than 0%. On the other hand, Figure 18 shows that PDR of MTC is 100% for the time of 300s and 400s. Over this range of time, periodic traffic is transmitted during SD period of MTC. This means that queue management exploiting WiFi/Bluetooth coexistence function properly works for avoiding collisions with periodic traffic. For 200s and 500s, some periodic packets are lost due to collisions. This is probably caused by the difficulty to control the transmission of packet stored into CSMA queue. For the other region of horizontal axis, PDR of MTC is degraded because the transmission timing of periodic traffic is outside of SD of MTC. We also confirmed that throughput of non–periodic traffic is not degraded even if we employ MTC using WiFi/Bluetooth coexistence function. From these results, we can conclude that WiFi/Bluetooth coexistence function is a promising means to realize MTC.

## VII. Conclusions

In this paper, focusing on a wireless coexistence scenario where multi–radio platforms are employed to support heterogenous traffic, we proposed transmission control, which suppresses mutual interference between hidden terminals generating periodic and non–periodic traffic. The proposed transmission control exploits interface heterogeneity, traffic periodicity, and queue management in order to suppress interference. As a means to suspend packet transmissions, we proposed two types of queue management: application–level transmission control and MAC–level transmission control. Furthermore, we proposed traffic adaptation for application–level transmission control, which adapts the amount of packets passed from the upper layer according to the observed congestion level. We first confirmed with experiments the practicality for WiFi

device to monitor congestion level in a real–time manner, which is required for traffic adaptation. Then, we evaluated the gain of the proposed application–level and MAC–level transmission control in terms of packet delivery ratio and throughput by computer simulations. Our numerical results showed that the proposed application–level and MAC–level transmission control significantly improve packet delivery ratio of periodic traffic while slightly improving throughput of non–periodic traffic in comparison to reference schemes. Finally, we investigated the feasibility of MAC–level transmission control with experiments. We realized the queue management at PHY/MAC level by exploiting WiFi/Bluetooth coexistence function. Our experimental results showed that WiFi/Bluetooth coexistence function is a promising means to realize MAC–level transmission control.

Our future work includes experimental evaluations of the proposed application–level and MAC–level transmission control with actual multi–radio platforms in a practical environment. More extensive verifications of simulation results, e.g., with a larger number of simulation trials and comparison with theoretical results, are also our future work. Furthermore, in this paper, it is assumed that the transmission timing of periodic traffic can be ideally estimated by the other terminals. However, in practice, this estimation can be incomplete, which can shift suspending duration from the desired duration. This causes degradation of packet delivery ratio of periodic traffic and throughput of non–periodic traffic. Therefore, we need to design a practical mechanism of synchronization, and to evaluate the impact of estimation error on the achievable performance of the proposed transmission control.

### References

[1] R. Ikeuchi and H. Yomo, "Transmission Control to Suppress Interference Between Periodic and Non-Periodic Traffic in Wireless Coexistence Scenarios," *Proc. of The Fifteenth International Conference on Digital Telecommunications (ICDT 2020)*, Feb. 2020.

[2] J. Gummeson, D. Ganesan, M. D. Corner, and P. Shenoy, "An Adaptive Link Layer for Heterogenous Multi-Radio Mobile Sensor Networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 1094–1104, Sept. 2010.

[3] D. K. Tosh and S. Sengupta, "Heterogenous Access Network(s) Selection in Multi-Interface Radio Devices," *IEEE International Workshop on Managing Ubiquitous COmmunications and Services*, pp. 117–122, 2015.

[4] "Smart Energy Gateway CUBE, NextDrive Co." accessed: 2020-012-04. [Online]. Available: https://www.nextdrive.io/en/cube-smart-energy-gateway/

[5] S. Dietrich, G. May, O. Wetter, H. Heeren, and G. Fohler, "Performance Indicators and Use Case Analysis for Wireless Networks in Factory Automation," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sep. 2017, pp. 1–8.

[6] M. Dungen *et al.*, "Channel Measurement Campaigns for Wireless Industrial Automation," *Automatisierungstechnik*, vol. 67, no. 1, pp. 7–28, 2019.

[7] A. Varghese and D. Tandur, "Wireless Requirements and Challenges in Industry 4.0," *Proc. of 2014 International Conference on Contemporary Computing and Informatics (IC3I)*, pp. 634–638, 2014.

[8] M. Sansoni *et al.*, "Comparison of M2M Traffic Models Against Real World Data Sets," *Proc. of IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, pp. 1–6, 2018.

[9] X. Cao, J. Chen, Y. Cheng, X. X. Shen, and Y. Sun, "An Analytical MAC Model for IEEE 802.15.4 Enabled Wireless Networks With Periodic Traffic," *IEEE Transactions on Wireless Communications*, vol. 14, no. 10, pp. 5261–5273, Oct. 2015.

[10] T. W. Carley, M. A. Ba, R. Barua, and D. B. Stewart, "Contention-free Periodic Message Scheduler Medium Access Control in Wireless Sensor/Actuator Networks," in *RTSS 2003. 24th IEEE Real-Time Systems Symposium, 2003*, 2003, pp. 298–307.

[11] T. Gaugel, J. Mittag, H. Hartenstein, S. Papanastasiou, and E. G. Strom, "In-depth Analysis and Evaluation of Self-organizing TDMA," in *2013 IEEE Vehicular Networking Conference*, 2013, pp. 79–86.

[12] A. H. Nguyen, Y. Tanigawa, and H. Tode, "Scheduling Method for Solving Successive Contentions of Heterogeneous Periodic Flows Based on Mathematical Formulation in Multi-Hop WSNs," *IEEE Sensors Journal*, vol. 18, no. 21, pp. 9021–9033, 2018.

[13] Kaixin Xu, M. Gerla, and Sang Bae, "How Effective is the IEEE 802.11 RTS/CTS Handshake in Ad Hoc Networks," in *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, vol. 1, 2002, pp. 72–76 vol.1.

[14] L. Zhang *et al.*, "Signal Strength Assistant Grouping for Lower Hidden Node Collision Probability in 802.11ah," *Proc. of 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 1–6, Oct. 2017.

[15] K. Shih, W. Liao, H. Chen, and C. Chou, "On Avoiding RTS Collisions for IEEE 802.11-based Wireless Ad Hoc Networks," *Elsevier Computer Communications*, vol. 32, no. 1, pp. 69–77, Jan. 2009.

[16] A. A. K. S., K. Ovsthus, and L. M. Kristensen, "An Industrial Perspective on Wireless Sensor Networks: A Survey of Requirements, Protocols, and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1391–1412, Third Quarter 2014.

[17] "IEEE Standard for Information technology–Local and metropolitan area networks–Specific requirements–Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," *IEEE Std 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003)*, pp. 1–212, 2005.

[18] "BUFFALO, WI-U3-866DS," accessed: 2020-12-04. [Online]. Available: https://www.buffalo.jp/product/detail/wi-u3-866ds.html

[19] D. Vassis, G. Kormentzas, A. Rouskas, and I. Maglogiannis, "The IEEE 802.11g Standard for High Data Rate WLANs," *IEEE Network*, vol. 19, no. 3, pp. 21–26, May 2005.

[20] "JWX6051, Qualcom Atheros Inc." accessed: 2020-12-04. [Online]. Available: minipci.biz/data/datasheets/jjplus/SpecSheet_JWX6051-0_20161006.pdf

# Teaching Machines to Understand Urban Networks: A Graph Autoencoder Approach

Maria Coelho*, Mark A. Austin[†], Shivam Mishra[‡], and Mark Blackburn [§]

*[†][‡] University of Maryland, College Park, MD 20742

*[†] Department of Civil and Environmental Engineering

[†][‡] Institute for Systems Research

E-mail: mecoelho@terpmail.umd.edu; austin@umd.edu; smishra8@umd.edu

[§] Stevens Institute of Technology, Hoboken, NJ, USA

E-mail: mblackbu@stevens.edu

*Abstract*—Due to remarkable advances in computer, communications and sensing technologies over the past three decades, large-scale urban systems are now far more heterogeneous and automated than their predecessors. They may, in fact, be connected to other types of systems in completely new ways. These characteristics make the tasks of system design, analysis and integration of multi-disciplinary concerns much more difficult than in the past. We believe these challenges can be addressed by teaching machines to understand urban networks. This paper explores opportunities for using a recently developed graph autoencoding approach to encode the structure and associated network attributes as low-dimensional vectors. We exercise the proposed approach on a problem involving identification of leaks in urban water distribution systems.

*Keywords-Systems Engineering; Machine Learning; Graph Embeddings; Graph Autoencoders; Digital Twins; Water Distribution Systems.*

## I. INTRODUCTION

This paper is concerned with the integration of recently developed graph embedding procedures with machine learning tasks, which together can enhance digital twin design and decision making in urban settings. It builds upon our previous work [1] on teaching machines to understand urban networks with graph analytics techniques.

### A. Problem Statement

The concept of creating digital replicas to serve as tools to improve decision-making has long been used in engineering. During the past three decades, however, remarkable advances in technology – including Internet of Things, artificial intelligence, and augmented and virtual reality (AR/VR) – have created opportunities to develop "digital twins," a cyber representation of an object, process or place that mirrors its implementation in the physical world through real-time monitoring, and synchronization of data with events. To this end, sensor and actuator systems, and algorithms and software are provided for observation, reasoning and physical systems control. NASA initially proposed the digital twin concept in the late 90s as a way to support the design and operation of air vehicles [2]. More recently, cities around the world have entertained use of the digital twin concept as a way to transform processes for day-to-day management and long-term urban planning and design.

The design of strategies to achieve superior levels of urban operation, even when available resources are limited, is complicated by the distributed, concurrent, and multi-disciplinary nature of large-scale urban systems. It is well known that when a disruption to urban operations requires the instantiation of recovery procedures, it is essential that the participating domains share information at key points in the system operation to enhance common knowledge, and their individual ability to make decisions appropriate to their understanding of the system state, its goals and objectives.

Despite the abundance of available urban data, current urban systems' potential for reaching enhanced capabilities in the decision-making and management of city infrastructure is hampered by lack of systematic knowledge exchange. The digital twins concept can overcome this barrier through its ability to integrate domains into the real-time knowledge discovery process from heterogeneous urban data. At this time, however, many challenges remain in digital twin design and implementation. First, there is a lack of unified models or a generic digital twin architecture in the literature, with no consensus on how to build a digital twin system [3]. Second, urban environments create further design difficulties, mainly related to the complexity of city-wide modeling and the lack of standards supporting cross-disciplinary data exchange. Research is needed to understand how to design the digital twin elements and their interactions so that collectively they can overcome these challenges.

### B. Architecting Urban Digital Twins

Requirements for architecting urban digital twins include the need for systems that can identify anomalies (faults) in system performance, and model the behavior of processes and interactions among the different domains within a city. Since a generic (or unifying) digital twin architecture does not exist at this time, we believe that the best pathway forward for digital twin design is with architectures that combine Machine Learning (ML) formalisms and Semantic Model representations that work side-by-side as a team, providing supportive roles for the collection and processing of data, identification of events,
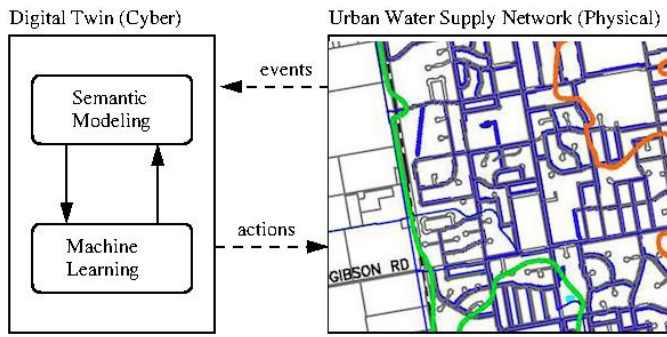
Figure 1. High-level representation for an urban water supply network digital twin (cyber) working alongside a physical urban water supply network.

and real-time management of city operations. Figure 1 shows the application of this concept to a small urban water supply system.

Figure 2 shows the proposed architecture, building upon our recent work in semantic modeling for (multi-domain) system of systems [4]–[6] and exploration of a combined semantic and ML framework exercised in energy-efficient buildings and brain cancer profiles [7], [8]. The proposed architecture is comprised of three sections: (1) multi-domain semantic modeling, (2) data mining, and (3) machine learning of graphs.

**Box 1: Multi-domain Semantic Modeling.** By their very nature, urban systems are a composition of many different types of domains and their interactions. These domains are geographically dispersed and intertwined, and have behaviors that are distributed and concurrent. Box 1 of Figure 2 shows that instead of modeling the dynamic behavior of systems with a centralized control and one large catch-all network, we explore opportunities for modeling systems as collections of domain-specific networks that dynamically evolve in response to events. Individual urban domains will operate as concurrent processes, each having their own thread of execution, and will respond to incoming data from external domains. Each domain will have a graph that evolves according to a set of domain-specific rules, and subject to satisfaction of constraints. Domains will interact when they need to in order to achieve collective objectives. If goals are in conflict, or resources are insufficient, then negotiation will need to take place. The model uses Semantic Web technologies for the implementation of ontologies, rules checking, and message passing mechanisms. The distinguishing feature of this framework is the concurrent development of ontologies, rules and data models, which are placed on an equal footing. When used in an urban context as part of the digital twin architecture, this approach to semantic modeling forces cross-domain data exchange that is more homogeneous than it would otherwise be, and establishes common knowledge among domains. Rule-based reasoning procedures can also be developed to impose fairness in domain operations and prevent deadlocks.

**Box 2: Data Mining.** Recent developments in the field of computational intelligence are sometimes termed machine learning (ML). Machine learning techniques deviate from traditional

models of computation in their ability to perform data analytics by learning patterns and hidden insights in data. Box 2 of Figure 2 shows ML for three classes – classification, clustering and association – of data mining. For the most part, these three data mining techniques were developed in the 1980s and 90s, and so the associated algorithms and software [9] are now quite mature. Data mining techniques also include use of recurrent neural network architectures to represent temporal sequences, and algorithms to detect anomalies in expected temporal behavior. For our purposes, these anomalies are events that can trigger the activation of urban recovery procedures modeled in Box 1.

**Box 3: Machine Learning of Graphs.** Remarkable advances in ML algorithms (2016-2019) include the ability of a machine to learn the structure of a graph and its attributes. The so called graph embedding methods learn a continuous vector space for the graph, assigning each node (and/or edge) in the graph to a specific position in the vector space. These embeddings can be later used to advance various learning tasks, such as node classification, node clustering, node recommendation, link prediction, and so forth [10].

*C. Scope and Objectives*

During 2018, our studies [7], [8], [11] focused on semantic foundations and data mining techniques working together as a team. Data mining/ML techniques for the classification and clustering of data provided useful feedback on the structuring of ontologies. In 2019, our studies started to explore graph embedding techniques for learning urban networks (i.e., see Box 3 of Figure 2) [1]. The Dynamic Attributed Network Embedding (DANE) [12] was used to generate low-dimensional vectors for a water distribution network. The embeddings were then fed to a Random Forest (RF) algorithm trained to identify water leaks. Although these initial studies showed successful results in identifying water leaks, the DANE framework did not incorporate the capability to verify that the embedding input to the RF Classifier was an accurate representation of the physical network topology and node attribute information. This lack of insight suggests that a better approach would replace hand generation of features with learning models trained to identify features encoded within embedding vectors, and then validate that the trained machine models faithfully replicate the physical graph topology. Research is also needed to understand the effects of graph size on learning performance. The latter is key for scalability of the proposed approach.

In a step toward resolving these problems, and extending our previous work, the objectives of this work are three fold: (1) identify an embedding framework that better fits our need for minimizing the loss of information during the network embedding process; (2) exercise the new embedding framework on the problem of leaks identification in an urban water distribution system; (3) explore the effects of network size on the learning performance. The remainder of this paper proceeds as follows: Related work in traditional and ML approaches for graph modeling is covered in Section II. An overview of different graph embedding techniques is provided in Section III. We exercise graph embedding and ML classification procedures in a water distribution system
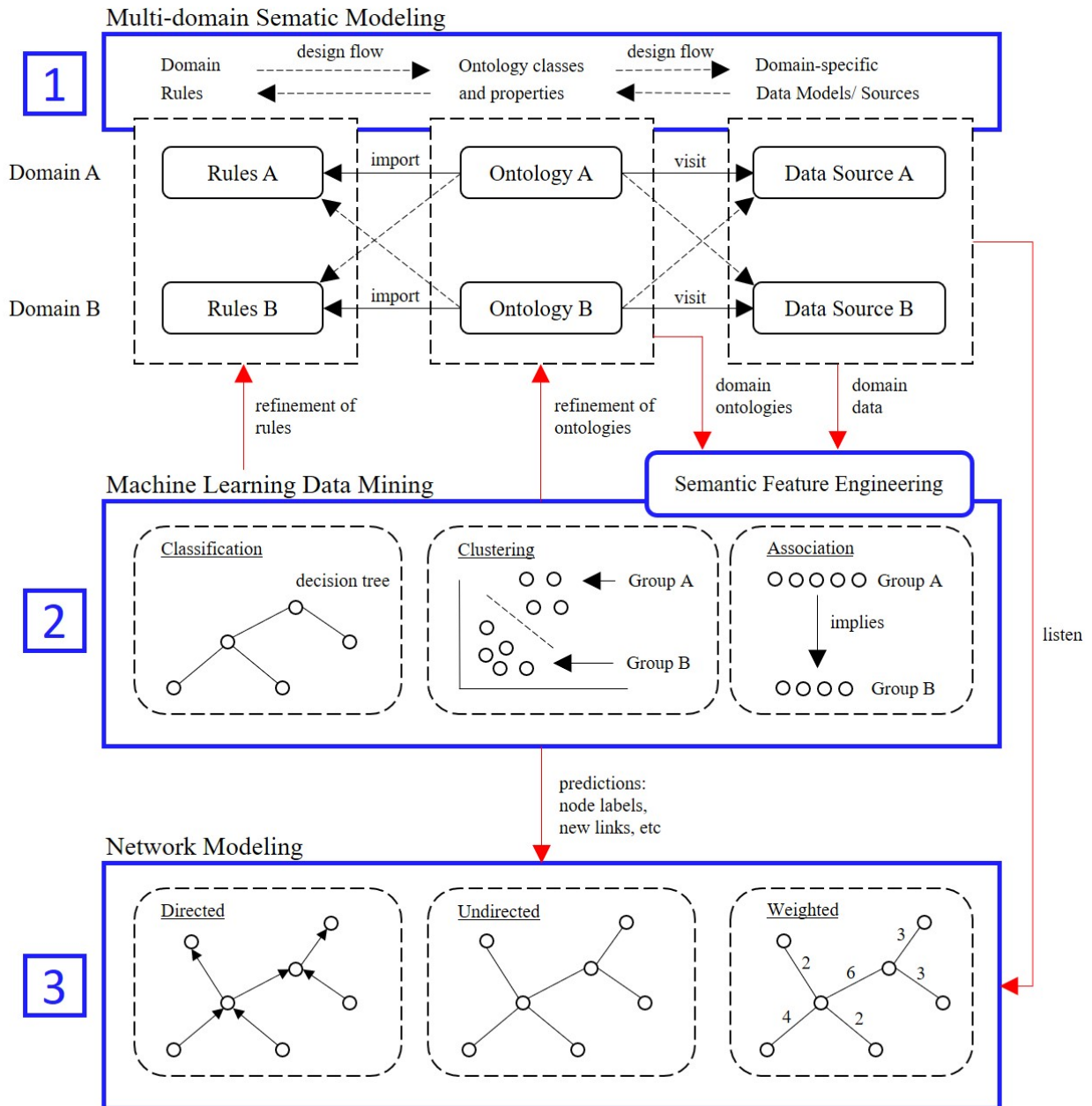
Figure 2. Digital twin architecture.

application involving identification of leaks in Section IV. We discuss ideas for scaling up the simulations in Section V. The conclusions and directions for future work are located in Section VI.

## II. RELATED WORK

This section covers related work in traditional and ML approaches to graph modeling.

### A. Graph Theory

In mathematical terms, a graph G = (V, E), where V is a set of vertices (also called nodes or points), E = set of edges (also called links or lines), and each edge is formed from pair of distinct vertices in V. V and E are usually taken to be finite. Graph theory is the study mathematical structures used to model pairwise relations between node and edge objects. It plays a central role in understanding how solutions to problems on graphs (e.g., traversal of nodes; reachability) are affected by the fundamental characteristics of the graph.

The study of urban systems as networks, and networks of interacting networks, draws upon many aspects of graph theory. Urban networks may be homogeneous, heterogeneous, and carry auxiliary information modeled as attributes. The information to be preserved in the network is strongly affected by the underlying characteristics of the system and, unfortu-
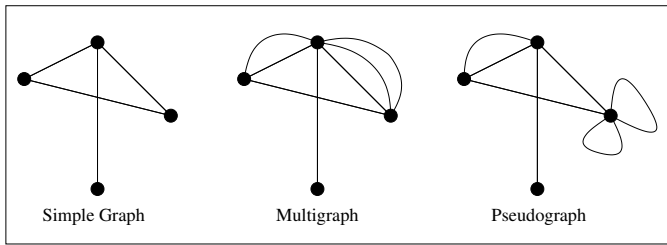
Figure 3. Structure of simple graphs, multigraphs and pseudographs.
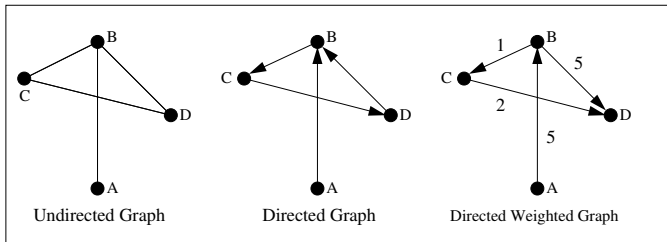


Figure 4. Graphs with undirected, directed, and directed weighted edges.

nately, in urban settings this is complicated by a wide range of possibilities. Figures 3 and 4 show, for example, that graph structures include simple, multigraphs and pseudographs. Their edges may be undirected, directed and/or weighted.

Kong and Simonovic [13] describe a few examples of how urban systems can be modeled as graphs. A street network can be represented as a graph composed of street junctions, end points and street segments. Generally, the edges are undirected, homogeneous, and the street network is fully connected. A water distribution network can be represented as graph where water works, storage facilities and pump stations are nodes with different attributes, and the water distribution pipes are directed edges, where the direction indicates the flow of water. A power grid can be represented by a graph where power plants, distribution and transmission substations are nodes with different attributes, and power lines are directed edges for the flow of electricity transmission. Information infrastructure can be represented as a graph where the Internet Service providers are nodes and cable connections are undirected edges, since these networks provide a bidirectional flow of information.

Regardless of the type of graph and its purpose in an urban setting, we observe that data mining techniques can be employed to access the information stored in graphs. This information can then be used to improve the design and operation of urban systems. Techniques for graph data mining can be divided into graph analysis and graph analytics.

### B. Graph Analysis

Graph analysis tasks are mainly focused on exploring the graph data to gain insights about its topological properties. Network topology (see Figure 3) nearly always affects function; therefore, it is important to characterize it. For instance, the topology of social networks affects the spread of information and disease, and the topology of the power grid

affects the robustness and stability of power transmission [14]. Traditional approaches to network/graph modeling employ adjacency matrices (or a simplified representation of network adjacency) to model the topology of graphs. The topological properties can then be explored through typical graph analysis tasks including connectivity analysis, traceability analysis, cycle detection, and shortest path identification.

### C. Graph Analytics

For high-dimensional problems that are data sparse, traditional approaches to graph representation and analysis can quickly become computationally prohibitive. A second problem is these traditional approaches do not capture the semantics of the network (i.e., node attributes). In recent years, there has been a surge in ML approaches that automatically learn to encode graph topology and attributes into low-dimensional embeddings.

Figure 5 shows simplified representations for traditional and machine learning approaches to graph representation. In a significant departure from traditional approaches to representing and studying the the properties of a graph, graph embedding methods learn a continuous vector space for the graph, assigning each node (and/or edge) in the graph to a specific position in the vector space, with the goal of preserving local linkage structure (not global structure) and/or network semantics. First the encoder maps the node to a low-dimensional vector embedding, based on the node's position in the graph and/or its attributes. Next, the decoder extracts information from the embedding vector (i.e., node's local graph neighborhood, or a classification label associated with the node). By jointly optimizing the encoder and decoder, graph embedding methods learn to compress information about graph structure and semantics into the low-dimensional embedding space. Because information can be lost in the embedding transformation process – it can be viewed as dimensionality reduction – the output embeddings are statistical in nature and, as such, should be interpreted as graph analytics (not graph analysis). Graph analytics can extract unseen or difficult to obtain properties of the graph, either directly or by feeding the learned representations to a downstream inference pipeline, such as node classification, node clustering, and link prediction.

### III. Teaching Machines to Understand Graphs

In recent years, many graph embedding approaches have been developed. This section introduces these different approaches, their advantages and limitations.

### A. Graph Embedding Methods

Goyal and Ferrara [15] have organized embedding methods into three broad categories: factorization based, random walk based, and deep learning based. Both the factorization and random walk based approaches train unique embedding vectors for each node independently, which results in several limitations. First, there is an absence of parameter sharing between the nodes, which leads to computational inefficiency
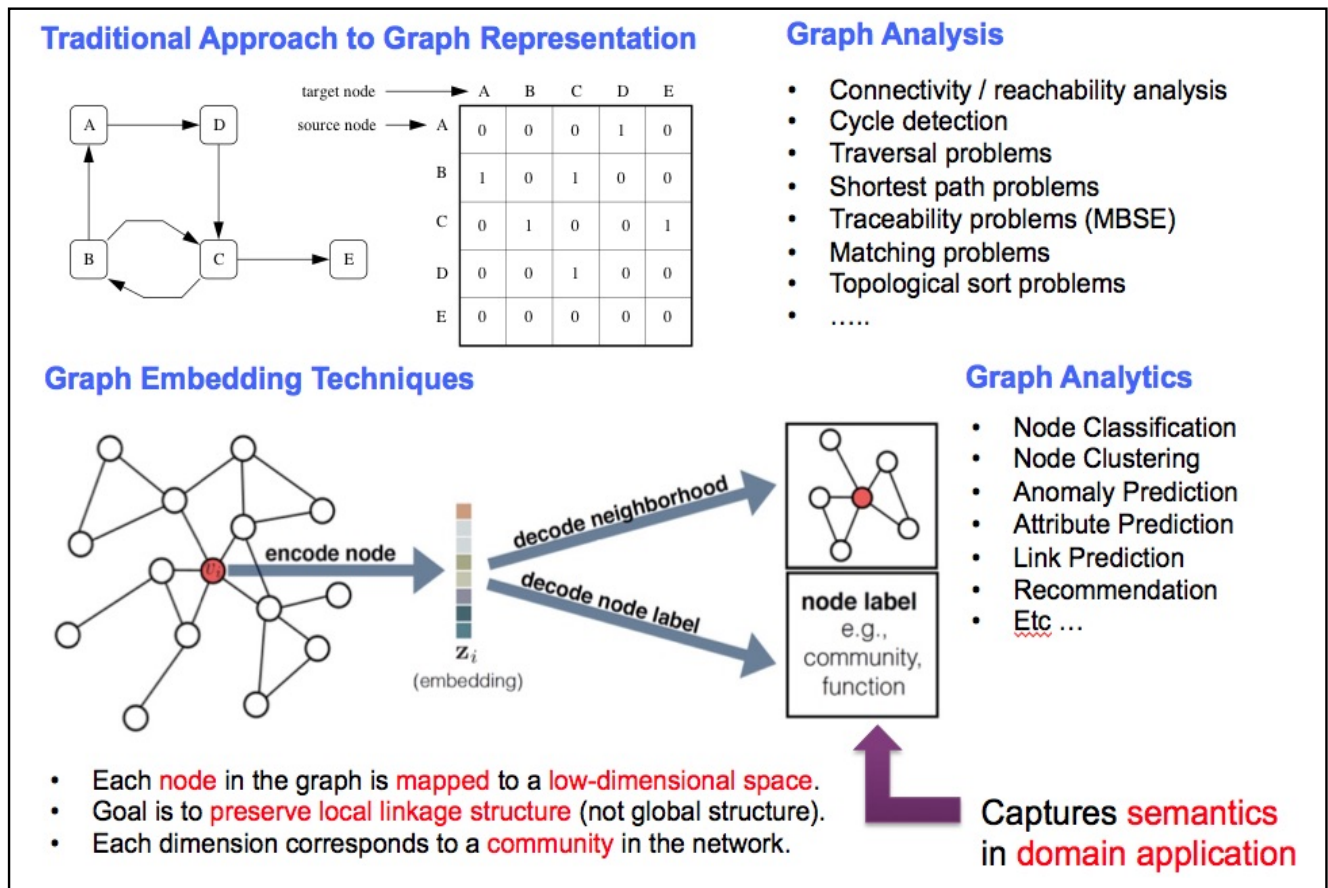
Figure 5. Graph analysis vs graph analytics.

since the number of trainable parameters grows linearly with the number of nodes in the graph. Second, these approaches are not extensible and are only able to generate embedding vectors for the nodes that were present during the training phase and not for any unseen nodes. A third problem is that these approaches lack an ability to incorporate node attributes during embedding generation, when node attributes can be highly informative about the node's position and role in the network.

In a concerted effort to mitigate these concerns, recent research efforts have led to the emergence of deep learning based methods. Deep learning based approaches use a deep neural network architecture, generally referred to as Graph Neural Networks (GNNs), to generate embeddings which depend both on the structure and the attributes of the graph. Wu et al. identified many types of GNNs among which are graph autoencoders (GAEs) [16].

*B. Graph Autoencoders*

GAEs are deep neural network architectures that are trained with the objective of reconstructing their original graph input. Figure 6 shows a high-level GAE architecture. First, an encoder takes a graph as its input and systematically compresses it into a low-dimensional vector. The decoder then takes the vector representation and attempts to generate a

reconstruction of some user-defined graph analysis tool (e.g., the adjacency matrix) of the original graph input. Encoder-decoder pairs are designed to minimize the loss of information between the input graph and the output (i.e., reconstructed) graph. These frameworks may be deterministic or probabilistic [17].

*C. Network Embedding in an Urban Context*

Figure 7 is a schematic for how the embedding process might be integrated in an urban network digital twin. We start by extracting a graph representation of the urban infrastructure and determining the initial parameters of the system. As time progresses, the digital twin will monitor changes in the system. Embeddings will be generated, and machines will be trained to understand a number of salient features of acceptable and unacceptable behavior. When an unacceptable behavior is identified, urban recovery procedures will be triggered. Since embeddings are a central and essential part of the learning process, there is a need to ensure the embedding input to the machine is an accurate representation of the information contained in the graph.

IV. CASE STUDY PROBLEMS

In this section, we exercise a graph autoencoding procedure that can encode the structure and network attributes

input: urban graph

minimize
loss

autoencoder

encoder

compress

$[\ -2.0,\ 0.5,\ 1.0\ .....\ -0.5\ ]$

lower–dimensional
representation vector

decoder

decompress

reconstruction of input: urban graph

Figure 6. Traditional encoder-decoder approach.

on problems involving the identification of leaks in an urban water distribution system.

*Topic 1. Case Study Applications*

The two case study applications presented in this work aim to explore the use of graph autoencoding (GAE) procedures and ML classification techniques for the detection and localization of leakages in water distribution systems (WDSs). We start by extracting a graph representation of the system at hand and determining the initial node attributes of the system. Topics 2 through 5 follow the analysis procedure and describe: (2) the encod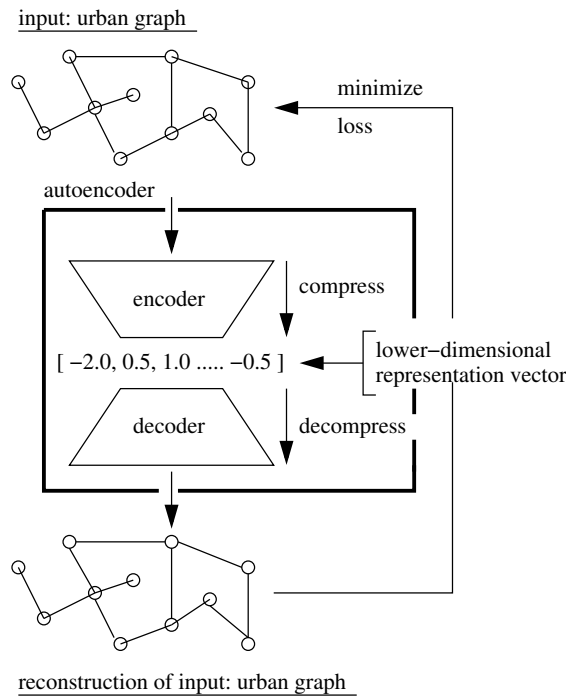ing of the network topology and node attribute information as node embeddings by a GAE framework; (3) synthetic generation of hydraulic (node pressure) data by the hydraulic simulation software EPANET [18]; (4) training and testing of a Random Forest (RF) algorithm [19] with the node embeddings to infer leak location; and (5) performance of the proposed framework.

*Topic 2. Node Embedding*

The selection of an appropriate GAE is not a trivial task. Water distribution networks are associated with a rich set of node attributes that are in constant flux due to day-to-day temporal variations in demand and, longer term, network expansion and urban growth. These variations cause the emergence of new content patterns and the fading of old content patterns. Despite these complications, it has been widely studied and reported that there exists a strong correlation among the attributes of linked nodes [20]. These node correlations and changing characteristics motivate us to seek an effective embedding representation to capture network

structure and attribute evolving patterns, which is of fundamental importance to learning in a dynamic environment. Early GAE algorithms such as Deep Neural Network for Graph Representations (DNGR) [21] and Structural Deep Network Embedding (SDNE) [22] only consider node structural information (i.e., connectivity between pairs of nodes), and ignore node attribute information. In 2016 Kipf et al. introduced Graph Autoencoder (GAE*) [23], an algorithm that leverages a ConvGNN [24] to encode node structural information and node feature information at the same time. We employ this algorithm in the two case studies of this work.

Although GAE* is the graph embedding approach most suitable for our purposes, a key research question remains: Are the procedures for reconstruction of the graph topology guaranteed to give the correct answer only most of the time or all of the time? An essential prerequisite for understanding the robustness of GAE* reconstruction procedures is to first understand the procedure itself (i.e., graph encoding, embedding optimization and reconstruction). Hence, the objective of our first case study is to walk step by step through the application of the GAE* framework to urban graphs from input to output. We deliberately keep the network layout and node attributes very simple, and begin with encoding for a 4 node graph and its node attributes.

Figure 8 shows the architecture of the GAE* framework applied to this use case. The encoder consists of two graph convolutional layers and a simple inner product decoder, which aims to decode node relational information from generated embeddings by reconstructing the graph adjacency matrix. The first convolutional layer takes as input the graph's node feature matrix $X$ and the symmetrically normalized adjacency matrix $\tilde{A} = D^{-1/2}AD^{-1/2}$, where $A$ is the adjacency matrix with added self connections and $D$ is the diagonal node matrix of $A$. For the purposes of keeping the case study as simple as possible, all of the node features are simply assigned a value of 1. Note, however, in the later case study, different system nodes will have different feature values that are naturally changing.

The first convolutional layer generates a lower-dimensional feature matrix defined as:

$$\bar{X} = ReLu(\tilde{A}XW^0)$$

where $W^0$ is a trainable parameter matrix. The second convolutional layer takes as input the output of the previous layer and generates the node embeddings:

$$Z = \tilde{A}ReLu(\tilde{A}XW^0)W^1$$

where $W^1$ is also a trainable parameter matrix. The purpose of the decoder is to reconstruct the adjacency matrix $A$ (with added self-connections) from $Z$. By applying the inner product on the latent variable $Z$ and $Z^T$, the algorithm learns the similarity of each node inside $Z$. By applying the sigmoid function $\sigma(\cdot)$ the algorithm computes the probability of edges existing between the range of 0 and 1. Therefore, the reconstructed adjacency matrix is defined as:

$$A' = \sigma(ZZ^T)$$

In order to arrive at the optimal embedding matrix $Z$, the $W^0$ and $W^1$ parameters are systematically updated through
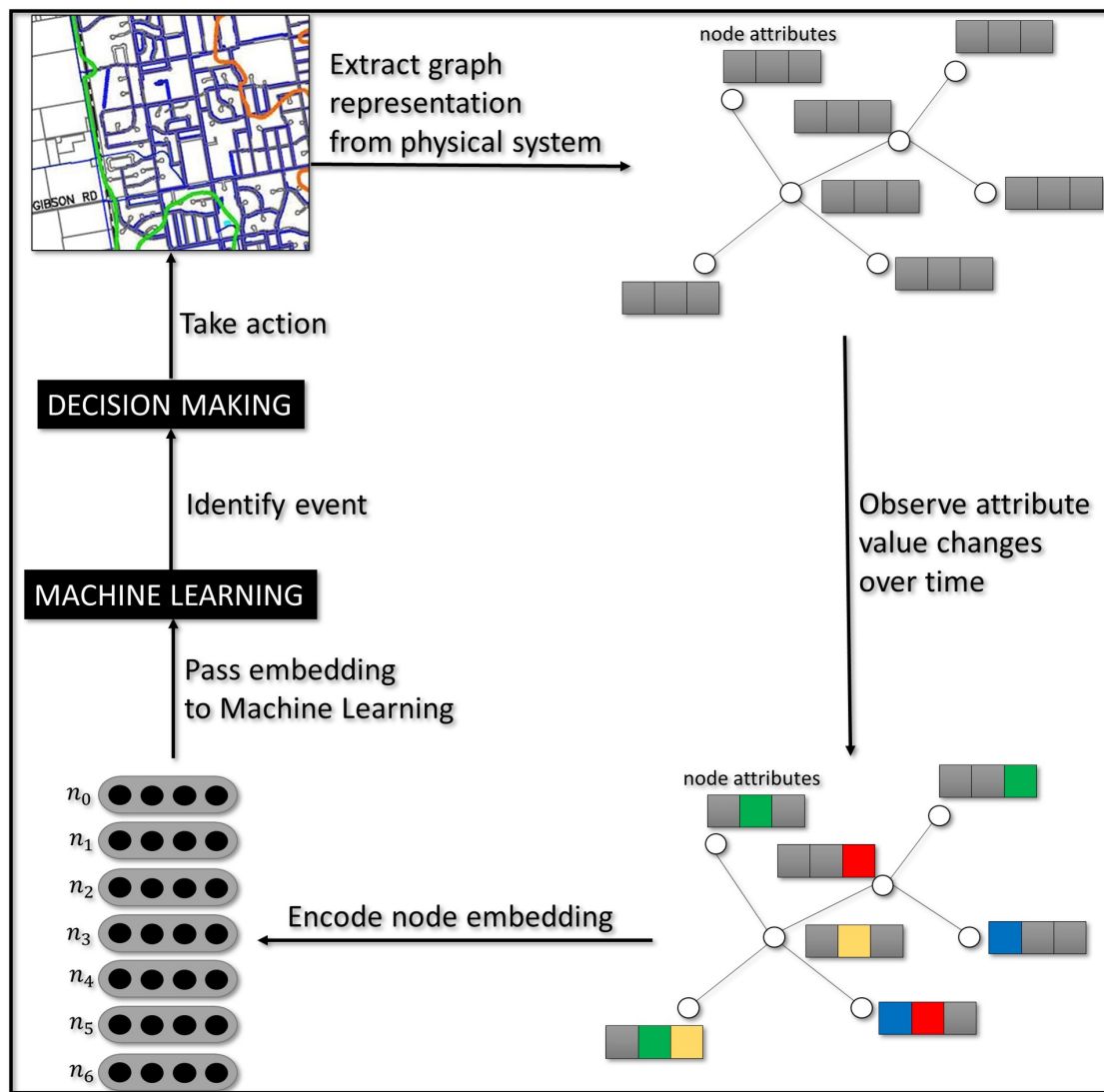
Figure 7. Process flowchart for training and executing machine.

an Adam optimization [25] of the weighted cross-entropy loss between the adjacency matrix $A$ and the soft reconstruction $A'$. Adam optimization uses a combination of extensions (i.e., adaptive gradient estimation and root mean square propagation) to stochastic gradient descent to estimate gradients and their moments as a moving average. These gradient estimates allow us to find the direction in which the weights should be adjusted in order to reduce the weighted cross-entropy loss.

By applying the GAE* architecture to the layout of Case Study 1, we obtain successful reconstruction of the test network topology. However, GAE* does not have the ability to reconstruct graph attribute information. Thus, an important research question that remains open is whether the embedding obtained using GAE* is an accurate representation of node attribute information needed for leak detection in WDS. In the following case study, we aim to answer this question by applying GAE* architecture to a leaking WDS, and use the generated embedding as input for a RF classifier trained to identify leaks.

*Topic 3. Data Generation*

Automatic water leakage detection can be performed with ML classification algorithms. These algorithms require training data involving normal operation conditions and abnormal operation conditions. In the case of a WDS, training data should involve hydraulic parameters at different locations, and pertain to normal operations and leaks from the past. Ideally, one would like to train a network using data measurements from a real-world network. For security reasons, however, WDS data (i.e., geographical layout of pipes, tanks, and demands) are kept confidential by the water utility companies and are not readily available to the public. A second option is to use simulators to synthetically generate data for machine learning. This study employs the simulation tool EPANET [18] to generate the training data for the WDS under consideration [26]. EPANET is a computerized model produced by the Environmental Protection Agency of the USA that simulates the dynamic hydraulic and water quality behavior within a WDS operating over an extended period of time. WDSs are
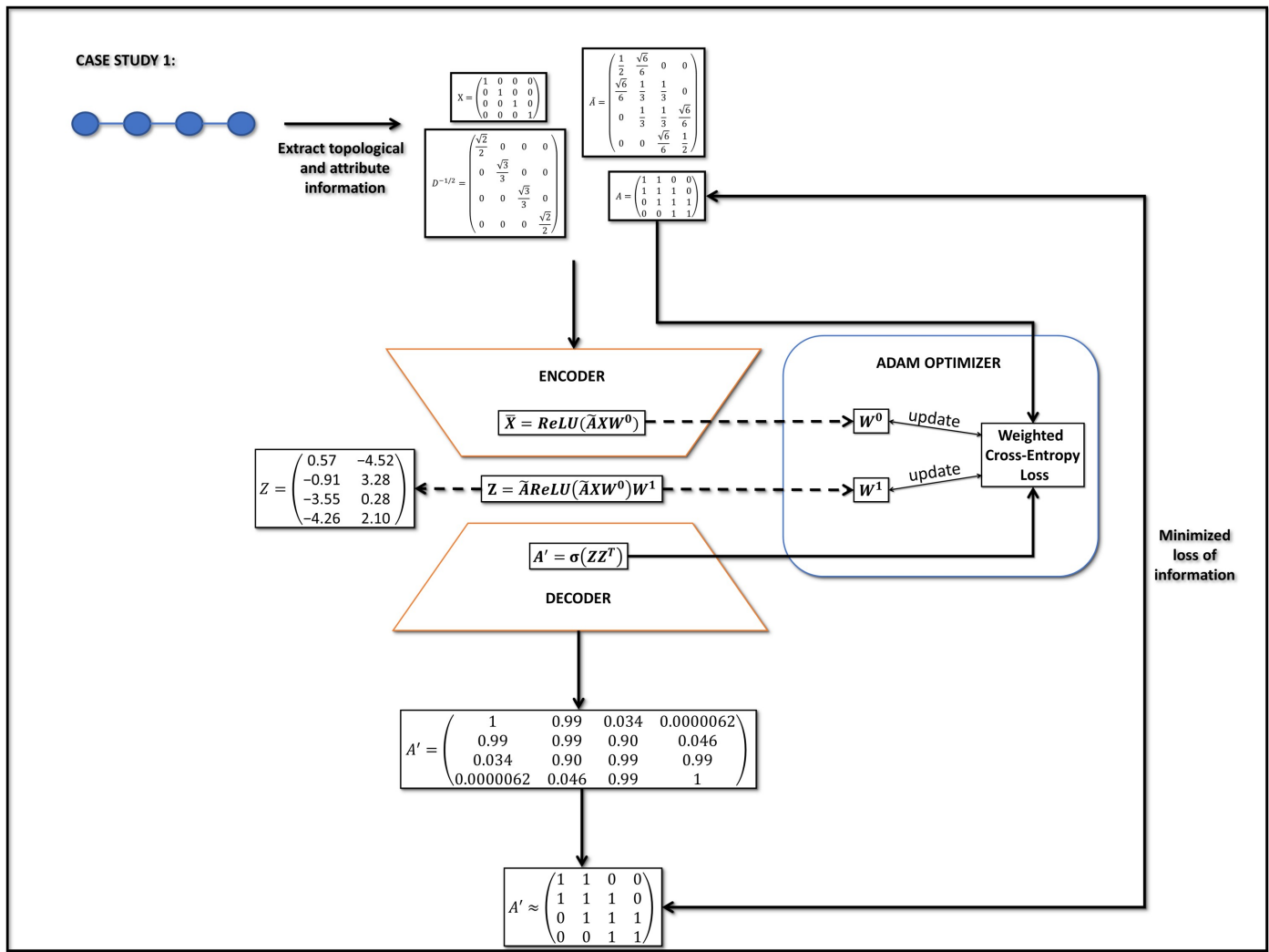
Figure 8. Flowchart of GAE* computations applied to Case Study 1.

composed by pipes (edges), nodes (junctions), pumps, valves, and storage tanks or reservoirs. During a simulation period, EPANET can track the flow of water in each pipe, the pressure at each node, the height of the water in each tank, the type of chemical concentration throughout the network, the age of the water, and source tracing. Various characteristics of a network element can be edited and a simulation can be performed to observe its effect on the overall system. For simplicity, we will limit the hydraulic parameter of interest in this study to node pressure. We obtain the pressure data by making two additional assumptions: (1) The data obtained through simulation does not involve any noise in it (i.e., the sensors are ideal), and (2) Water leakage is assumed to occur only at the junction nodes.

While EPANET does not explicitly support the modeling of leakages, its computational engine is demand-driven and, thus, leaks can be modeled as additional demand, and in a way that is independent of the pressure in a consumption node. The water output data at each node is defined as the base demand, and the demand can be increased at different times during the simulation. The network layout used to perform the hydraulic simulations in this work is shown in Figure 9, which from

here on forth we will refer to as Case Study 2. The network contains 4 junctions, 4 links, a pump station, a water source, and a tank. The same network configuration was also used in our previous work [1].

ML training sets require data for a large number of positive (i.e., leaking) and negative (i.e., non leaking) cases, meaning that EPANET simulations also need to be performed for a large number of cases. A practical way of automating this process is with the EPANET-Python Toolkit, an open-source scripting software that interfaces with the latest version of EPANET.

With data from the hydraulic network simulations in hand, the next step is to perform the graph embedding. For this case study we are interested in obtaining a low-dimensional node vector representation for each node in the network. The hope is that the learned embeddings will advance various learning tasks, particularly leak detection by node classification. Applying GAE* to the pressure data outputted from EPANET simulation, yields 2-dimensional node embedding vectors for each node of Case Study 2. The next challenge to
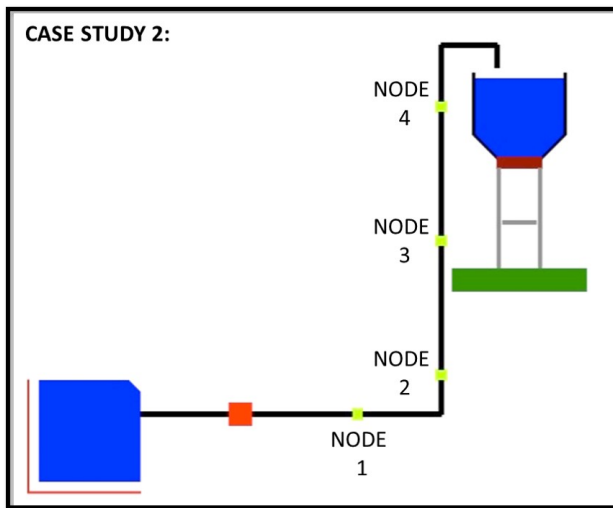
Figure 9. Elevation view of urban water distribution networks and junction (node) numbers used by EPANET simulation.

address is how to determine the optimal number of embedding dimensions? Since answers to this question are still an open research problem, we chose a set up for which the best topology reconstruction results were observed.

*Topic 4. Node Classification*

With the node embeddings obtained from GAE*, leakage detection can be performed. To be specific, we wish to identify the node or nodes where leakage has occurred during hydraulic simulation. To this end, the target function assigns a value of 1 to leaking nodes and a value of 0 to the non-leaking nodes. The embedding and target data are passed to a Random Forest classification algorithm. There are a number of good reasons to choose this pathway of analysis. First, the random forest classifier is composed of a number of decision trees. This problem solving strategy is well known to be accurate and robust. In addition, it does not suffer from the overfitting problem often encountered in other ML algorithms, since it takes the average of all the predictions of decision trees, and cancel out the biases. Random forests can also handle missing values, by using median values to replace continuous variables, or computing the proximity-weighted average of missing values. Finally, random forest provides a measurement of relative feature importance, which helps in the selection of contributing features for the classifier [19].

In this case study, we select a training data set that maximizes the expected variation in the target and environment, by building a hydraulic simulations where all of the nodes are leaking for half of the simulation duration, and for the other half of the duration none of the nodes are leaking. Since the simulation was set to last 24 hours, with pressure readings at every hour, the training set contains 24 cases for each node. The top portion of Figure 10 shows the plots for each node in Case Study 2, where the first of the embedding dimensions is plotted against time. Note that when a leak occurs at a specific time step, the embedding value for that dimension will also change; thus, this problem can be framed as anomaly detection.

In order to test the trained RF classifier, we generate a test set from a hydraulic simulation where none of the nodes are leaking for half of the simulation duration, and for the other half of the simulation duration only one of the nodes is leaking. In this use case we chose the leaking node in the testing set to be Node 3. Similar to the training set, the test set also contains 24 cases for each node. The bottom of Figure 10 shows the first of the embedding dimensions plotted against time. Notice that the embedding values change slightly compared to the previous scenario where all the nodes where leaking; therefore, the goal of the ML process is not only to detect the anomalies, but also identify which anomalies are actual leaks and which ones are just a propagation of the leak effects. Moreover, since the initial objective of this work is identification of leak location (and not timing of incipient leakage), leak durations are kept constant through all simulations. We do acknowledge, however, that the hour at which a leak occurs is relevant and future work will need to address variations in leakages in both space and time.

*Topic 5. Results*

As noted in the previous topic, the Random Forest classifier was trained on both leak and non leak data for each node. We were then able to test whether the classifier is able to detect a leak in the system. The test procedure involved feeding the node embeddings for a scenario where initially none of the nodes were leaking, and later only one node was leaking (node 3). As classification problems are perhaps the most common type of ML problem, there are a myriad of metrics that can be used to evaluate outputted classifications, the most common being classification accuracy (i.e., the ratio of number of correct predictions to the total number of input samples). After training and testing an RF classifier, we obtained a classification accuracy of 100 percent (i.e., the classifier correctly predicted the location of test leaks in the network). We recognize such a high performance may be due to several simplifications made in this experiment. First the network maintained a constant base demand for each node through the hydraulic simulation, when in reality WDSs experience changes in water demand depending on time of day. Second, the decoder component of GAE* framework successfully reconstructed network topologies in case study 1 and 2, and consequently generated accurate embeddings representations of the networks, in part because the topologies were relatively simple and static. An unresolved question from these initial experiments is if the same success can be obtained for larger or dynamic topologies. Section V discusses the effects of size on GAE* performance. Third, we assumed sensor nodes were deployed in each junction of the network; however, in our resource limited world, placing as many sensors as there are junction nodes to monitor all of the nodes in real time is extremely infeasible. In Section VI we discuss ideas for addressing these simplification issues in future work.

V. DISCUSSION

The automated identification of leaks in WDSs depends critically on accurate representations of network topology and
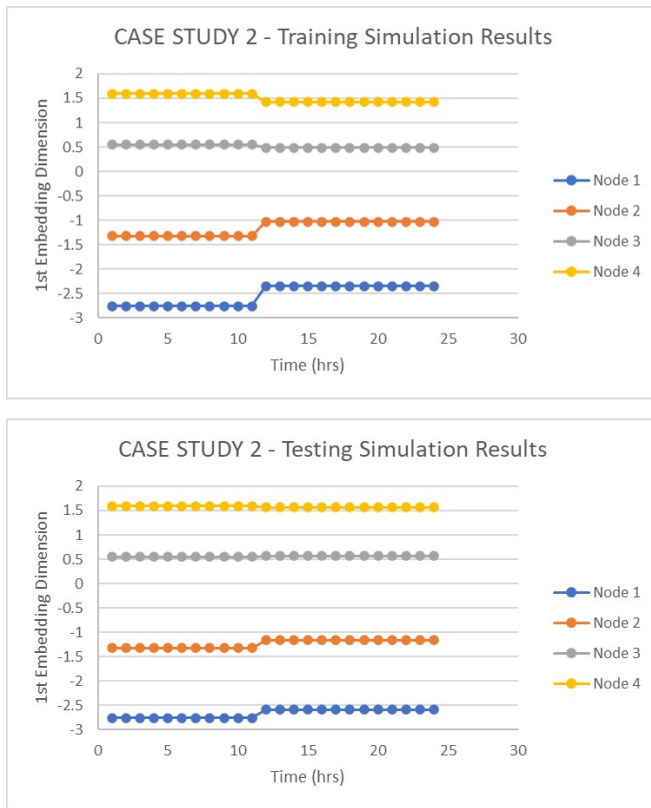
Figure 10. Case study 2 node embeddings (1st dimension) obtained for train and test sets plotted against time.

node attribute information. The embedding obtained by using GAE* framework is fed to downstream classifiers that will determine presence of leaks in the system. If the embedding is not accurate enough, errors could cascade across the fault detection mechanism and lead to false positive or false negative leakage identifications. Therefore, it is important to understand the limitations of GAE* in generating a representation of the network at hand. This section discusses these limitations and potential solutions.

### A. GAE* Limitations

Our results have indicated that when network topologies are simple, GAE* is capable of providing network embeddings that can be decoded to perfectly reconstruct the input network. One important research question that remains open is the effect of network size on GAE* learning performance. In order to address this concern, we have attempted to reconstruct three attributed WDSs of varying sizes using GAE*. The goal of this experimental study was to learn for what ranges of embedding dimensions the algorithm was able to reconstruct the input network accurately. In order to measure the accuracy of the reconstruction, the graph edit distance (GED) between the input graph and the reconstructed graph was used. GED is a graph similarity measure defined as the minimum cost of a sequence of node and edge edit operations transforming the reconstructed graph into the input graph. A GED value of zero shows that the reconstructed graph is isomorphic to the input graph. The higher the GED value, the higher the dissimilarity

between the reconstructed graph and the input graph. In order to maintain a sense of GED importance for different graph sizes, the GED values obtained in our experiments were then normalized by the size of the input graph.

Figure 11 summarizes the results of the numerical experiments, and shows that if the neural network used to train the GAE* has an insufficient number of neurons, then the reconstruction performance is poor. This makes perfect sense. But when the same network is modeled with too many neurons, the reconstruction performance is also poor. The results suggest that there is a window of convergence where good reconstruction can be achieved. For the networks with 6 and 11 nodes the algorithm is able to perfectly reconstruct (i.e., GED is zero) certain network configurations, but for the network with 36 nodes the algorithm is not able to reach perfect reconstruction. From Case Study 1, we now understand the inner architecture of GAE* and what limitations it may encounter, as well as the ways in which it might be changed to adapt to specific problems. To obtain a good reconstruction of the input graph, the underlying algorithm needs to work. In turn, the latter depends on: (1) convergence of the Adam optimization, (2) the encoder architecture design, and (3) the decoder architecture design. We suspect that the phenomena observed in Figure 11 may be related to limitations in these three GAE* components.

### A. Convergence of the Adam Optimizer

One of the key hyper-parameters to set in order to construct an Adam optimizer is the learning rate. This parameter scales the magnitude of our weight updates in order to minimize the network's loss function. If the learning rate is set too low, then training will progress very slowly as we are making very small updates to the weights values. However, if your learning rate is set too high, then it can cause undesirable divergent behavior in the loss function (i.e., the gradient of the weight oscillates back and forth, and it is difficult to make the loss reach the global minimum).

### B. Encoder Architecture Design

Although Kipf et al. used 2 hidden layers, 32 neurons in the first hidden layer, and 16 neurons in the second hidden layer when presenting the GAE* architecture [24], it is possible to modify these characteristics of the framework in order to adapt to specific needs. When considering the structure of GAE*, there are really two decisions that must be made: how many hidden layers to actually have in the neural network, and how many neurons will be in each of these layers? Prior to deep learning, problems that required more than two hidden layers were rare. Two or fewer layers will often suffice with simple data sets. However, with complex datasets additional layers can be helpful. According to Heaton a neural network with no hidden layers is only capable of representing linear separable functions or decisions, a network with one hidden layer can approximate any function that contains a continuous mapping from one finite space to another, and a network with two hidden layers can represent an arbitrary decision boundary to arbitrary accuracy with rational activation functions and can approximate any smooth mapping to any accuracy. And
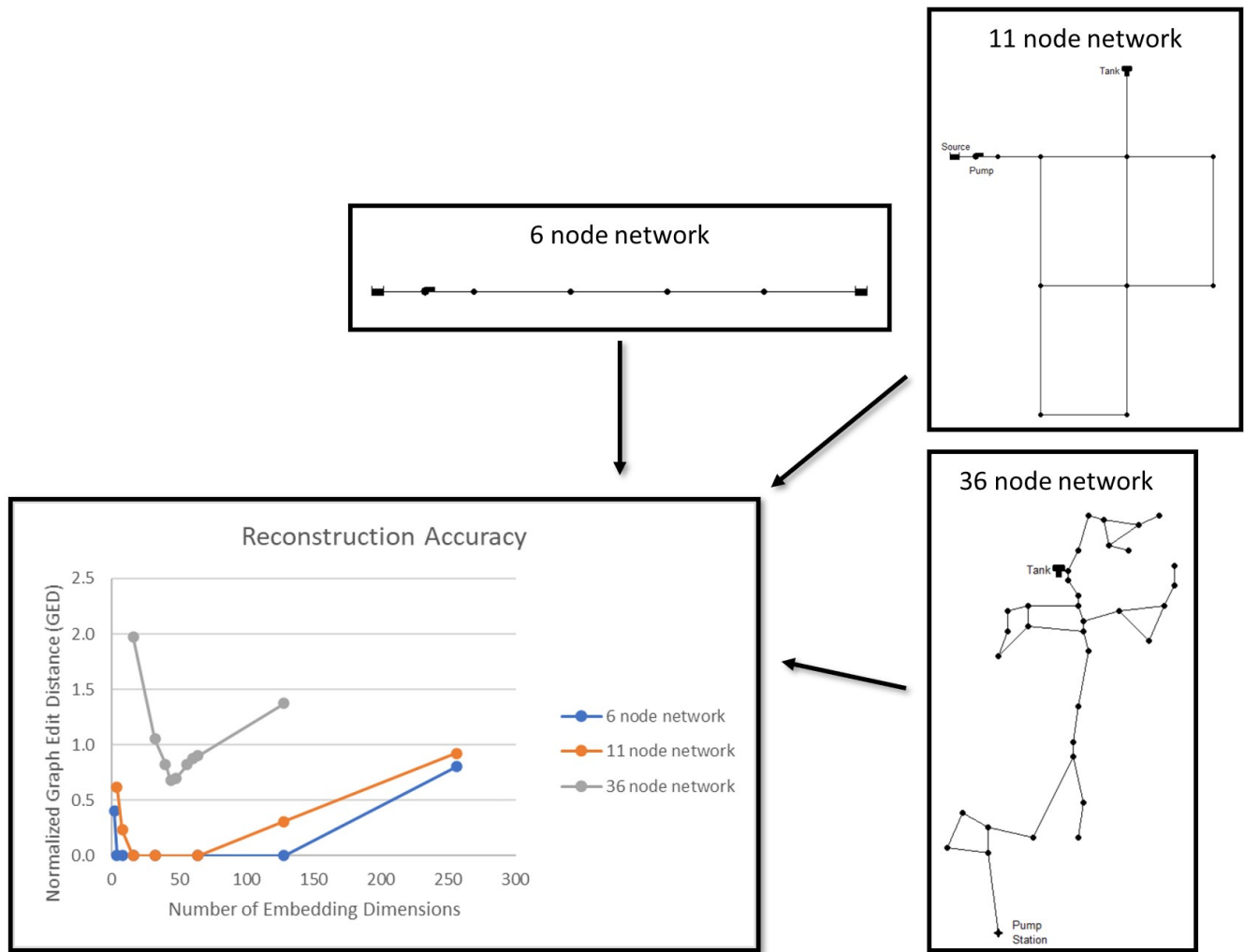
Figure 11. GAE* reconstruction accuracy for networks of varying sizes.

a network with more than two layers can learn complex representations (i.e., feature engineering) for layered layers [27]. For our purposes, we are interested in investigating whether adding more layers to GAE* algorithm can bring benefits (i.e., faster convergence, lower loss, fewer nodes, etc.). Deciding the number of hidden layers is only a small part of the problem, we must also determine how many neurons will be in each of these hidden layers. Using too few neurons in the hidden layers can result in underfitting. Underfitting occurs when there are too few neurons in the hidden layers to adequately process information in a complicated data set. Too many neurons on the other hand can result in overfitting. Overfitting occurs when the neural network has so much information processing capacity that the limited amount of information contained in the training set is not enough to train all of the neurons in the hidden layers [27]. Computational requirements also need to be taken into account. Generally speaking, a large number of neurons in the hidden layers will increase the time it takes to train the network, possibly to a point where it is computationally infeasible to adequately train the neural network [27]. Obviously, some compromise must be reached between too many and too few neurons in the hidden layers. Ultimately, the selection of an

architecture for our neural network will come down to avoiding underfit, overfit and convergence issues. Learning curves are widely used in machine learning for algorithms that learn (i.e., optimize their internal parameters) incrementally over time. The shape and dynamics of a learning curve can be used to diagnose the behavior of a machine learning model and, in turn, suggest changes that may be made to improve learning and/or performance. For our purposes, we are interested in investigating how learning curves can help us adapt the GAE* architecture to avoid the abovementioned issues.

### C. Decoder Architecture Design

Much of the node embedding research work performed in recent years has focused on refining and improving the encoder architecture. Most methods still rely on basic pairwise decoders (e.g., inner-product decoders), which predict pairwise relations between nodes and ignore higher-order graph structures involving more than two nodes [17]. Higher-order structural sub-graphs are essential to the structure and function of complex networks. Hence, testing the performance of decoding algorithms other than inner-product decoders will

be an important step in unveiling GAE* limitations.

## VI. CONCLUSIONS AND FUTURE WORK

The long term objective of this research is to understand how ML and semantic-modeling can work hand-in-hand to build digital twins architectures that enhance the collection of data, identify events, and manage operations of systems in an urban context. In this paper, we started by exploring the recently developed GAE* framework in Case Study 1, where a simple graph was encoded to a low-dimensional vector while minimizing the loss information in the encoding process. Then, we investigated the possibility of using a GAE* generated embedding as input for a RF classier trained to identify leaks in WDSs in Case Study 2. Finally, we examined the effects of network size on learning performance by generating embeddings for larger size networks. Although procedures for encoding network information (i.e., structure and attributes) with the GAE* framework, and using RF classifier to identify water leaks have shown to be successful, the work reported in this paper takes an initial step toward exploring potential applications of ML to the identification of leaks in urban networks. Many questions and issues remain either unanswered or unresolved. Opportunities for future work include varying base node demands according to time of day, experimenting with more complex topologies and topologies that are dynamic (i.e., edges are removed or created), investigating how many sensor nodes are needed and where to place them in the network, exploring the use of learning curves in finding the optimal number of neurons, layers, and learning rate values for optimizing learning performance, and exploring different decoder architectures with the aim of improving reconstruction results for more complex graphs.

## VII. ACKNOWLEDGMENT

## REFERENCES

[1] M. Coelho and M.A. Austin, "Teaching Machines to Understand Urban Networks," The Fifteenth International Conference on Systems (ICONS 2020), February 23-27 2020, pp. 37–42.

[2] E.H. Glaessgen and D.S. Stargel, "The Digital Twin Paradigm for Future NASA and U. S. Air Force Vehicles," in 53rd AIAA/ASME/ASCE/AHS/ASC Struct. Struct. Dyn. Mater. Conf., 2012.

[3] A. Fuller, Z. Fan, C. Day, and C. Barlow, "Digital Twin: Enabling Technologies, Challenges and Open Research," IEEE Access, vol. 8, June 2020, pp. 108 952–108 971.

[4] M. Coelho, M.A. Austin, and M.R. Blackburn, "Distributed System Behavior Modeling of Urban Systems with Ontologies, Rules and Many-to-Many Association Relationships," The Twelth International Conference on Systems (ICONS 2017), April 23-27 2017, pp. 10–15.

[5] ——, "Semantic Behavior Modeling and Event-Driven Reasoning for Urban System of Systems," International Journal on Advances in Intelligent Systems, vol. 10, no. 3 and 4, December 2017, pp. 365–382.

[6] ——, The Data-Ontology-Rule Footing: A Building Block for Knowledge-Based Development and Event-Driven Execution of Multi-Domain Systems. Proceedings of the 16th Annual Conference on Systems Engineering Research, Systems Engineering in Context, Chapter 21, Springer, 2019, pp. 255–266.

[7] P. Delgoshaei, M. Heidarinejad, and M.A. Austin, "Combined Ontology-Driven and Machine Learning Approach to Monitoring of Building Energy Consumption," in 2018 Building Performance Modeling Conference and SimBuild, Chicago, IL, September 26-28 2018.

[8] J. Abraham, "Semantic Foundations for Formalizing Brain Cancer Profiles," MS Thesis, Master of Science in Systems Engineering Program, University of Maryland, College Park, MD 20742, 2019.

[9] I.H. Witten, E. Frank, M.A. Hall, and J.P. Christopher, Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kauffmann, 2017.

[10] H. Cai, V. W. Zheng, and K. C. Chang, "A Comprehensive Survey of Graph Embedding: Problems, Techniques and Applications," IEEE Transactions on Knowledge and Data Processing, vol. 30, no. 9, 2018, pp. 1616–1637.

[11] M.A. Austin, P. Delgoshaei, M. Coelho, and M. Heidarinejad, "Architecting Smart City Digital Twins: A Combined Semantic Model and Machine Learning Approach," Journal of Management in Engineering, ASCE, vol. 36, no. 4, 2020.

[12] J. Li et al., "Attributed Network Embedding for Learning in a Dynamic Environment," in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, ser. CIKM '17. New York, NY, USA: ACM, 2017, pp. 387–396. [Online]. Available: http://doi.acm.org/10.1145/3132847.3132919

[13] J. Kong, S. Simonovic, and Z. Chao, "Resilience Assessment of Interdependent Infrastructure Systems: A Case Study Based on Different Response Strategies," Sustainability, vol. 11, November 2019, p. 6552.

[14] S. Strogatz, "Exploring Complex Networks," http://www.nature.com/nature/journal/v410/n6825/pdf/410268a0.pdf, vol. 410, March 2001.

[15] P. Goyal and E. Ferrara, "Graph Embedding Techniques, Applications, and Performance: A Survey," CoRR, vol. abs/1705.02801, 2017. [Online]. Available: http://arxiv.org/abs/1705.02801

[16] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P.S. Yu, "A Comprehensive Survey on Graph Neural Networks," CoRR, vol. abs/1901.00596, 2019.

[17] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation Learning on Graphs: Methods and Applications," CoRR, vol. abs/1709.05584, 2017. [Online]. Available: http://arxiv.org/abs/1709.05584

[18] L.A. Rossman L.A., "EPANET 2: Users Manual," Water Supply and Water Resources Division National Risk Management Research Laboratory Cincinnati, OH 45268, Tech. Rep. EPA/600/R-00/057, September 2000.

[19] L. Breiman, "Random Forests," in Machine Learning, vol. 45, no. 1. Norwell, MA, USA: Kluwer Academic Publishers, October 2001, pp. 5–32.

[20] J.J. Pfeiffer, S. Moreno, T. La Fond, J. Neville, and B. Gallagher, "Attributed Graph Models: Modeling Network Structure with Correlated Attributes," in Proceedings of the 23rd International Conference on World Wide Web, ser. WWW '14. New York, NY, USA: ACM, 2014, pp. 831–842. [Online]. Available: http://doi.acm.org/10.1145/2566486.2567993

[21] S. Cao, W. Lu, and Q. Xu, "Deep Neural Networks for Learning Graph Representations," 2016, In AAAI.

[22] D. Wang, P. Cui, and W. Zhu, "Structural Deep Network Embedding," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 1225–1234.

[23] T.N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," CoRR, vol. abs/1609.02907, 2016.

[24] ——, "Variational Graph Auto-Encoders," ArXiv, vol. abs/1611.07308.

[25] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," International Conference on Learning Representations, December 2014.

[26] J. Mashford, D. Silva, S. Burn, and D. Marney, "Leak Detection in Simulated Water Pipe Networks using SVM," Applied Artificial Intelligence, vol. 26, May 2012, pp. 429–444.

[27] J. Heaton, Artificial Intelligence for Humans, Volume 3: Deep Learning and Neural Networks, ser. Artificial Intelligence for Humans. Createspace Independent Publishing Platform, 2015. [Online]. Available: https://books.google.com/books?id=q9mijgEACAAJ

# Evaluation of Vehicle Position Estimation Method Combining Roadside Vehicle Detector and In-vehicle Sensors

1st Shunya Yamada
*Institutes of Innovation for Future Society*
*Nagoya University*
Nagoya, Aichi, Japan
s_yamada@ertl.jp

2nd Yousuke Watanabe
*Institutes of Innovation for Future Society*
*Nagoya University*
Nagoya, Aichi, Japan
watanabe@coi.nagoya-u.ac.jp

3rd Hiroaki Takada
*Institutes of Innovation for Future Society*
*Nagoya University*
Nagoya, Aichi, Japan
hiro@ertl.jp

*Abstract*—**Properly managing merging at junction is important in improving highway traffic safety and traffic flow. Accurate vehicle positions and velocities are necessary in realizing proper management, but existing sensors have both advantages and disadvantages. Roadside vehicle detectors are very accurate, but only available at fixed points. By contrast, in-vehicle Global Navigation Satellite System (GNSS) sensors can be used anywhere except in tunnels, but are less accurate. However, these sensors can compensate for each other's week points. In this paper, we propose a vehicle position estimation method that combines roadside vehicle detector and in-vehicle sensors. This method is used to gather data from the roadside vehicle detector and in-vehicle sensors via different wireless networks. It then applies Kalman filters to calculate the accurate position and velocity. When exchanging information over wireless networks, communication delays occur and the data arrival sequence is not guaranteed. Our method can retroactively calculate the vehicle position in the presence of delays below a maximum acceptable threshold. This study evaluates in several simulations using the data created from the GNSS position error and communication delay models. We also perform the evaluation using a dataset created based on the actual vehicle information and communication delays in the simulations. The simulation results show that our method can more accurately estimate the vehicle positions compared to that using data from either sensor alone. Moreover, our method is more suitable for managing traffic and controlling merging at junctions.**

*Index Terms*— *Sensor fusion; Position estimation; Communication delays; Intelligent transportation system.*

Fig. 1. Roadside vehicle detector.



Fig. 2. General-purpose GNSS sensor.

## I. INTRODUCTION

This paper is an enhanced version of [1]. Vehicles merging into the main lane at highway junctions are increasingly causing the traffic congestion in that lane [2]. Moreover, 20–30% of highway truck accidents occur at or near junctions [3]. Thus, appropriately managing traffic and controlling the merging at junctions are important in improving both highway safety and traffic flow.
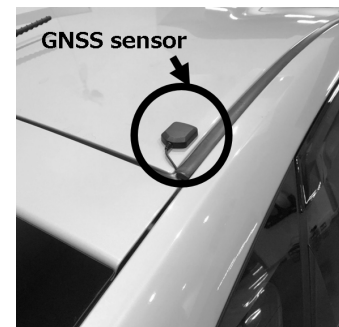
Several previous studies have investigated proper traffic management and merging control at junctions. Cui et al. [4] proposed a system for detecting collisions by estimating the vehicle arrival time at junctions. Their system obtained the vehicle positions and velocities from a monocular camera
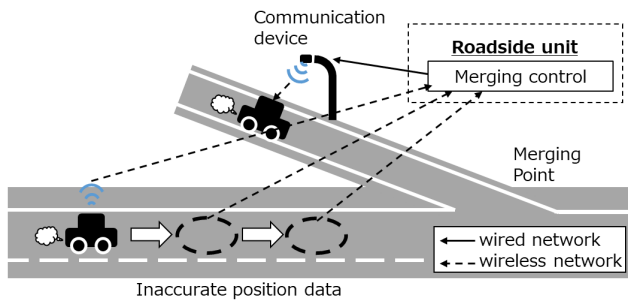
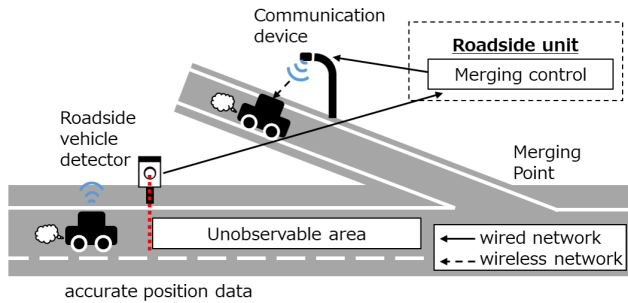Fig. 3. Existing approach1: using GNSS sensors.



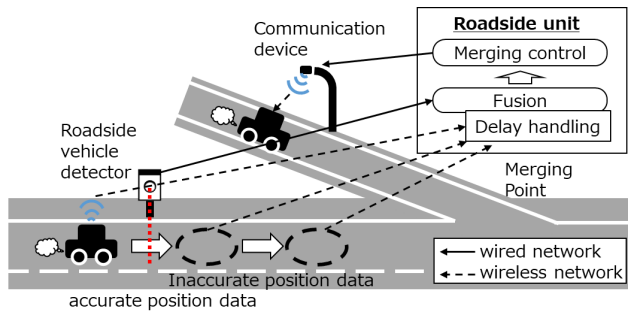Fig. 4. Existing approach2: using roadside vehicle detector.
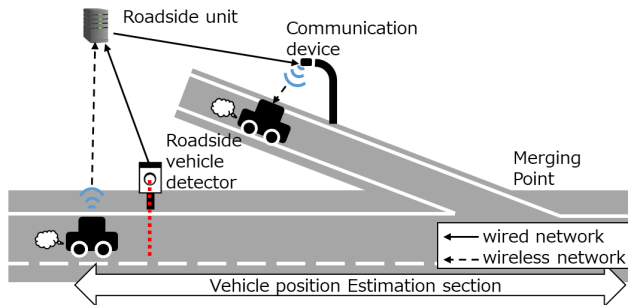


Fig. 5. Proposed approach: combining both sensors.



Fig. 6. Merging support system overview.

installed at the junction and used these to estimate the arrival times. Milanes et al. [5] installed a local control system near the junction, which receives the position and velocity information from approaching vehicles and sends them low-risk merging strategies. Chou et al. [6] proposed a merging method based on vehicle-to-vehicle (V2V) communication. The vehicles approaching the junction use this to exchange

their positions and velocities. The vehicles in the main lane then create gaps for entering vehicles before they have even reached the merging point. Hirai et al. [7] proposed a system that uses roadside vehicle detector installed before merging points. Roadside vehicle detector are often used to detect the presence of vehicles and the vehicle velocities on the lane to estimate traffic (Figure 1). Hirai's approach was estimated vehicles' arrival times at the merging points. When a vehicle on the on–ramp arrives at almost the same time as a vehicle in the main lane, the system alerts the latter, enabling it to prepare for merging, even if its driver has not seen the vehicle on the on–ramp. Giving drivers longer time to prepare makes the process safe. Japan's government has started field tests of autonomous driving in the Tokyo waterfront area [8]. Accordingly, information on roadside vehicle detector are available in these filed tests. Proper traffic management and merging control at junctions are considered by companies that participated in the field tests.

All the above-mentioned merging methods depend on the vehicle position and velocity information to properly manage traffic and control merging at junctions. The location estimation method using a camera must be applied such that occlusion does not occur, and the place where it can be used is limited. Although accurate vehicle velocities can be obtained from speed sensors, vehicle positions can be incorrect. These are often acquired from position estimation methods using light detection and ranging (LiDAR) or in-vehicle global navigation satellite system (GNSS) sensors. The position estimation method using LiDAR can more accurately estimate the position when compared to GNSS sensors; hence, autonomous vehicles often use it for position estimation. However, the cost of LiDAR usage is much higher than that of using general-purpose GNSS sensors and LiDAR cannot be immediately installed in conventional vehicles. GNSS sensors are obtained position by receiving the signals emitted from satellites and can function anywhere, except in tunnels. The signals received from the satellites contain noise. High-precision GNSS sensors can correct noise [10] and obtain accurate positions, albeit being expensive. The GNSS sensors mounted on vehicles are almost general-purpose products (Figure 2). They are inexpensive, and the positions measured using these sensors can differ from the true position by more than 10 m depending on the location. Properly managing traffic and controlling the merging at junctions cannot be performed with large vehicle position errors [6]. The system using roadside vehicle detector accurately estimates the vehicle arrival times when the sensor is close to the merging point. However, the error increases with the distance between the sensor and the merging point. The establishment cost is also high, and multiple installations are not reasonable because roadside vehicle detector are usually attached to poles installed at roadsides and gates across the road (Figure 1).

Exchanging information via such wireless networks leads to communication delays. Dedicated short-range communications (DSRC) or long-term evolution (LTE) is used for V2V and vehicle-to-infrastructure (V2I) communications. According to
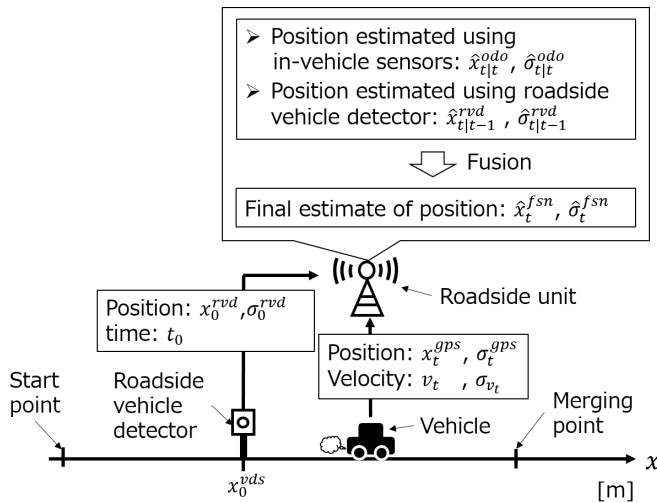
Fig. 7. Illustration of the position estimation model and variable definitions.

Dey et al. [11], these delays are approximately 100 ms for the DSRC (for the communication between a vehicle traveling at 80 km/h and a roadside unit). These delays also mean the data arrival times are not guaranteed. We believe that we can obtain more accurate vehicle positions by combining data from the GNSS and roadside vehicle detectors and compensating for the communication delays.

In this paper, we propose a vehicle position estimation method that combines data from roadside vehicle detector and in-vehicle sensors. This can retroactively calculate prior vehicle positions in the presence of delays below the maximum acceptable threshold. This study makes the following three main contributions.

The first contribution of this study is a vehicle position estimation method that combines data from roadside vehicle detector and in-vehicle sensors. In our system, a roadside vehicle detector is installed before the merging point and a roadside unit is installed near the junction. In-vehicle sensor information is used to estimate its position and velocity. A roadside vehicle detector is also used to estimate the vehicle position based on the sensor position and vehicle velocity. The two estimates are combined using a statistical approach proposed by Duffin [12]. Previous studies used only one of the position information obtained from the GNSS and roadside vehicle detector (Figures 3 and 4). Our method estimates positions using both position information (Figure 5).

The second contribution of this study is a communication delays compensation method. The vehicle positions at earlier times are retroactively calculated when older data arrive up to the predetermined maximum communication delay. When the roadside unit does not receive information from a vehicle, it estimates the vehicle's position based on the most recent information received from it. Modified Kalman filters that consider communication delays are proposed [14]. The contribution is applying the modified Kalman filters to the scene of merging support.

The third contribution of this study is the evaluation of our

system. We evaluated our proposed system by using only the data created from the GNSS position error and communication delay models in several simulations [1]. In this paper, our system was also evaluated by using a dataset created based on actual vehicle information and communication delays in simulations. The actual vehicle information were obtained when driving through a main lane of the airport west interchange in Tokyo, Japan. The dataset including communication delays was created by actually transmitting from a certain terminal to other terminal via LTE communication.

The remainder of this paper is organized as follows. Section II describes the assumptions made herein. Section III introduces our proposed system. Section IV evaluates the method using several simulations. Section V presents the results. Section VI discusses our proposed system, and Section VII concludes the paper.

## II. ASSUMPTION

In this study, roadside vehicle detector were installed before the merging point. A roadside unit was located near the junction (Figure 6). All vehicles are assumed to have GNSS devices, speed sensors, and V2I communication devices. Vehicles approaching the junction send their current position and velocity, as well as the time the data were acquired, to the roadside unit. This information is repeatedly sent at regular intervals in vehicle position estimation section and starts to send before the vehicle passes through the roadside vehicle detector. The system clocks in the vehicles, roadside vehicle detector, and roadside units are assumed to be synchronized. Some delay exists in the communications between the vehicles and the roadside units. Meanwhile, the communication delays between the roadside vehicle detector and the roadside unit are assumed to be negligible because the communication between them is via wire and dedicated connection.

Figure 7 shows the environmental model, where the vehicle drives from the starting point toward the merging point. A roadside vehicle detector is installed at $x = x_0^{rvd}$ m. The vehicle sends information about its position (namely the average $x_t^{gps}$ m and the standard deviation $\sigma_t^{gps}$ m) and velocity (average $v_t$ km/h and standard deviation $\sigma_{v_t}$ km/h) to the roadside unit. The roadside vehicle detector sends the position of the vehicle's center (average $x_0^{rvd}$ m and standard deviation $\sigma_0^{rvd}$ m) and the detection time $t_0$ s to the roadside unit when the vehicle passes through it.

The roadside unit estimates the vehicle's position using both the information received from the vehicle (the average $\hat{x}_{t|t}^{odo}$ m and standard deviation $\hat{\sigma}_{t|t}^{odo}$ m) and that from the roadside vehicular detector (average $\hat{x}_{t|t-1}^{rvd}$ ms and standard deviation $\hat{\sigma}_{t|t-1}^{rvd}$ m). It then combines these two estimates to obtain the final vehicle position (average $\hat{x}_t^{fsn}$ m and standard deviation $\hat{\sigma}_t^{fsn}$ m).

The assumptions in this study are according to the actual field test of Japan [8]. Support for autonomous driving by providing information for automatically adjusting the speed and timing of entering the main line at highway junctions has

been considered in the field tests [13]. The experiment has been conducted at an airport west interchange in Tokyo, Japan. Figure 8 shows the top view of the airport west interchange and the sensor location. Figures 9 and 10 illustrate the devices for merging control. Figure 11 depicts the merging point of the airport west interchange. The roadside vehicle detector detects the speed of the vehicles driving on the main lane and sends them to the roadside unit located near the merging point via wire and dedicated connection. The roadside unit estimates the speed and timing to safely join the main lane. These information are provided to the vehicles driving on the ramp via wireless at the place where the communication device is installed. Furthermore, previous studies have already discussed time synchronization [9]. Therefore, these assumptions are realistic.

## III. PROPOSED SYSTEM

In this section, we describe our proposed position estimation approach, followed by our method of compensating for the communication delays.

### A. Position Estimation

Figure 12 presents a flow diagram showing the steps performed to estimate the vehicle position when it passes through the gate at time $t_0$ s. Here, the vehicle position is estimated by applying Kalman filters to the position and velocity information received from it until it passes through the roadside vehicle detector. Kalman filters are often used to estimate the exact state based on inaccurate and noisy information; hence, we used them here to estimate the vehicle position from the noisy position and velocity information.

The Kalman filters usage is divided into the prediction and correction steps. During the prediction step, the vehicle position is estimated based on the estimate from the previous time step and the current vehicle velocity information. The correction step adjusts this estimated position using the current vehicle position information. The specific equations are presented below.

Prediction step:

$$\hat{x}_{t|t-1}^{odo} = \hat{x}_{t-1|t-1}^{odo} + \frac{5}{18} v_t dt, \qquad (1)$$

$$\left(\hat{\sigma}_{t|t-1}^{odo}\right)^2 = \left(\hat{\sigma}_{t-1|t-1}^{odo}\right)^2 + \left(\frac{5}{18}\sigma_{v_t} dt\right)^2. \qquad (2)$$

Correction step:

$$\hat{x}_{t|t}^{odo} = \hat{x}_{t|t-1}^{odo} + k_t \left(x_t^{gps} - \hat{x}_{t|t-1}^{odo}\right), \qquad (3)$$

$$\left(\hat{\sigma}_{t|t}^{odo}\right)^2 = (1 - k_t)\left(\hat{\sigma}_{t|t-1}^{odo}\right)^2, \qquad (4)$$

$$k_t = \frac{\left(\hat{\sigma}_{t|t-1}^{odo}\right)^2}{\left\{\left(\hat{\sigma}_{t|t-1}^{odo}\right)^2 + (\sigma_t^{gps})^2\right\}}. \qquad (5)$$

where $\hat{x}_{t|t-1}^{odo}$ m and $\hat{\sigma}_{t|t-1}^{odo}$ m are the average and standard deviation of the vehicle position, respectively, generated by
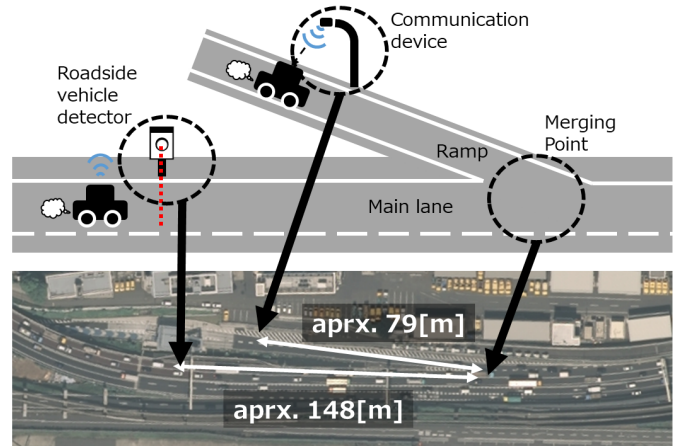


Fig. 8. Status of devices for merging support system at the airport west interchange in Tokyo.



Fig. 9. Roadside vehicle detector.



Fig. 10. Communication device.



Fig. 11. Merging point.

the prediction step for timestep $t$ s; $\hat{x}_{t|t}^{odo}$ m and $\hat{\sigma}_{t|t}^{odo}$ m are the average and standard deviation of the vehicle position, respectively, generated by the correction step for timestep $t$ s; $v_t$ km/h and $\sigma_{v_t}$ km/h are the average and standard deviation of the vehicle velocity, respectively, generated by the correction step for timestep $t$ s; $k_t$ is the Kalman gain at timestep $t$ s; and $\frac{5}{18}$ is a term used for converting the vehicle velocity from
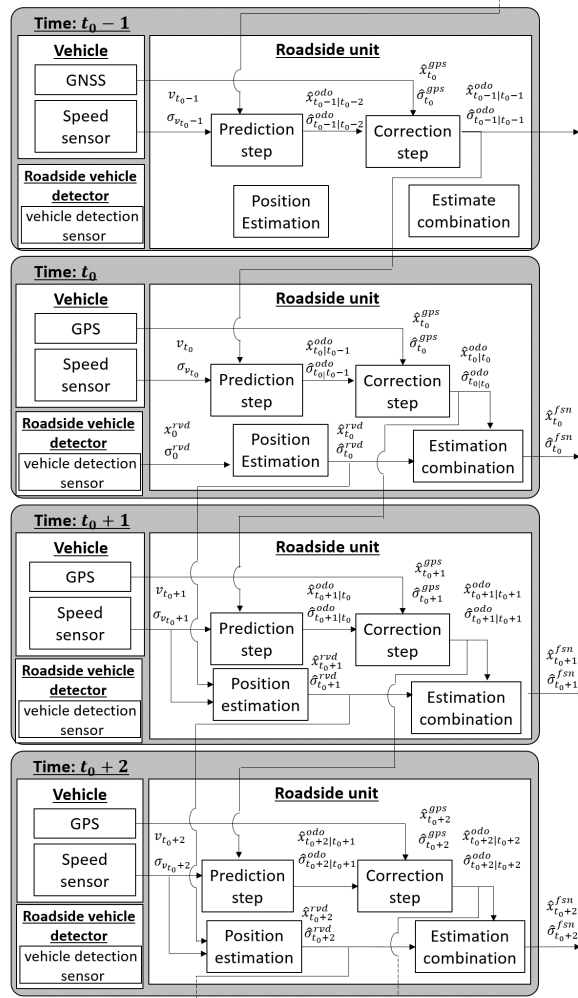
Fig. 12. Flow diagram showing the steps performed to estimate the vehicle position.



Fig. 13. Overview of our communication delays compensation method.

km/h to m/s.

The roadside unit receives the position of the vehicle's center from the roadside vehicle detector at time $t_0$ s, then combines this with the velocity information received from the vehicle and the Kalman filters' prediction step to estimate the vehicle position. The specific equations are presented below.

Position estimation:

$$\hat{x}_t^{rvd} = \hat{x}_{t-1}^{rvd} + \frac{5}{18} v_t dt, \tag{6}$$

$$\left(\hat{\sigma}_t^{rvd}\right)^2 = \left(\hat{\sigma}_{t-1}^{rvd}\right)^2 + \left(\frac{5}{18}\sigma_{v_t} dt\right)^2. \tag{7}$$

where, $\hat{x}_t^{rvd}$ m and $\hat{\sigma}_t^{rvd}$ m are the average and standard deviation of the vehicle position predicted by the Kalman filters at timestep $t$ s, respectively; $v_t$ km/h and $\sigma_{v_t}$ km/h are the average and standard deviation of the vehicle velocity at timestep $t$ s; and $\frac{5}{18}$ is a term used for converting a vehicle velocity from km/h to m/s.

The vehicle position (average $\hat{x}_{t_0}^{rvd}$ m and standard deviation $\hat{\sigma}_{t_0}^{rvd}$ m) at timestep $t_0$ s is defined as the position received

from the roadside vehicle detector; hence, it is given as follows:

$$\hat{x}_{t_0}^{rvd} = x_0^{rvd}, \tag{8}$$

$$\hat{\sigma}_{t_0}^{rvd} = \sigma_0^{rvd}. \tag{9}$$

where $x_0^{rvd}$ m and $\sigma_0^{rvd}$ m are the average and standard deviation, respectively, of the vehicle position received from the roadside vehicle detector.

Finally, the two vehicle position estimates are combined using the statistical approach proposed by Duffin [12], which is based on Bayes' rule and Kalman filters. This approach simply combines the two Gaussian distribution.

Estimate combination:

$$\hat{x}_t^{fsn} = \hat{x}_{t|t}^{odo} + \frac{\left(\hat{\sigma}_{t|t}^{odo}\right)^2}{\left(\hat{\sigma}_{t|t}^{odo}\right)^2 + \left(\hat{\sigma}_t^{rvd}\right)^2} \left(\hat{x}_t^{rvd} - \hat{x}_{t|t}^{odo}\right), \tag{10}$$

$$\left(\hat{\sigma}_t^{fsn}\right)^2 = \left\{ 1 - \frac{\left(\hat{\sigma}_{t|t}^{odo}\right)^2}{\left(\hat{\sigma}_{t|t}^{odo}\right)^2 + \left(\hat{\sigma}_t^{rvd}\right)^2} \right\} \left(\hat{\sigma}_{t|t}^{odo}\right)^2. \tag{11}$$

where, $\hat{x}_t^{fsn}$ m and $\hat{\sigma}_t^{fsn}$ m are the average and standard deviation of the vehicle position obtained by combining the two estimates.

### B. Communication delay compensation

Communication delays inevitably occur when exchanging information over wireless networks; thus, they must be considered when estimating the vehicle positions. Here, we used

Kagami et al.'s approach, which involves modified Kalman filters that account for the communication delays [14]. This method retroactively calculates prior positions and velocities when older data are received up to a predetermined maximum communication delay. However, the modified Kalman filters model proposed by Kagami et al. is a multidimensional state–space model. Hence, in this study, the communication delay compensation is only used a concept, and the Kalman filters model is changed to a one-dimensional (1D) constant velocity (CV) model.

Figure 13 illustrates our method of communication delay compensation. Here, $L$ s is the maximum communication delay. $t - L$ s denotes the $L$ timesteps after the vehicle passed through the roadside vehicle detector. The roadside units did not receive any position and velocity information from the vehicle at timesteps $t - 1$ and $t$ s. The data is received at timestep $t - 2$ s, and the roadside unit uses this to retroactively calculate the vehicle positions and velocities at timesteps $t - 1$ and $t$ s, working back from the present time to the predetermined maximum communication delay.

The vehicle position is estimated using Eqs. (1)–(11) at each time. The vehicle position is estimated based on the estimate from the previous time step and each time vehicle position and velocity information. Finally, the two vehicle position estimates are combined.

## IV. EVALUATION EXPERIMENTS

Our vehicle position estimation method is evaluated in several simulations in this section. First, our vehicle position estimation method is evaluated using the data created from the GNSS position error models and the communication delay model and conducted with and without communication delays. Next, our system is evaluated using a dataset created based on the actual vehicle information and communication delays in the simulations. We used Matlab R2019a herein. Table I shows the personal computer (PC) specifications used for the simulation.

### A. Evaluation using models

*1) Environment without communication delays:* In this experiment, the position estimation accuracy was evaluated in an environment where the communication delays were assumed to be negligible. Our proposed method labeled as "Fusion" was compared with one that simply applies Kalman filters to the in-vehicle sensor data (i.e., "GNSS only").

In this simulation, a vehicle drove from the starting point at $x = -100$ m toward the merging point at a speed of 80 km/h. The roadside vehicle detector was installed at $x = 0$ m. The standard deviations of the vehicle's center position and velocity were set to s $\sigma_0^{rvd} = 0.5$ m and $\sigma_{v_t} = 5$ km/h, respectively. The vehicle sent its position and velocity to the roadside unit every 100 ms. The roadside unit also estimated the vehicle position every 100 ms. The vehicle acquired its position from a GNSS device. We considered two models of the GNSS position error, namely a Gaussian white noise model and a Gauss-Markov random process model, used in

### TABLE I
### SPECIFICATIONS OF THE PC USED FOR THE SIMULATIONS.

| | |
|---|---|
| CPU | Intel Core i9-9900X at 3.50GHz |
| Memory | 64 GB |
| Storage | Samsung MZVLB1T0HALR-00000 |

### TABLE II
### PARAMETERS OF THE GAUSSIAN WHITE NOISE MODEL.

| GNSS error | $\sigma_w$ |
|---|---|
| Low | 3 |
| Medium | 6 |
| High | 9 |

### TABLE III
### PARAMETERS OF THE GAUSS–MARKOV RANDOM PROCESS MODEL.

| | $\sigma_g$ | $\sigma_r$ | $\beta$ |
|---|---|---|---|
| Case 1 | 0.2020 | 0.0027 | 1/600 |
| Case 2 | 0.1030 | 0.3160 | 1/600 |

a previous study [6]. These models will be described below. The simulation was repeated six times for each GNSS position error model.

Gaussian white noise model:
This is given by the following equations:

$$x_t^{gps} = x_t + w_t. \tag{12}$$

where, $x_t$ is the actual vehicle position, and $w_t$ is the Gaussian white noise, i.e., $w_t \sim N(0, \sigma_w)$. Here $\sigma_w$ is set as in Table II. $\sigma_w$ in Table II is set from a trial experiment using a general-purpose GNSS sensor. UBX-M8030-KT was used for a general-purpose GNSS sensor. The specifications of it denotes in Table IV. When position data were acquired using the sensor at multiple points in Nagoya University, the most low value was $\sigma_w \approx 3$, the most high value was $\sigma_w \approx 9$. Thus the $\sigma_w$ is set as 3 in the low GNSS error and the $\sigma_w$ is set as 9 in the high GNSS error. As an intermediate value between the high and low GNSS error, $\sigma_w$ is set as 6 in the medium GNSS error.

Gauss–Markov random process model:
This is given by the following equations [15]:

$$m_t = e^{-\beta dt} m_{t-1} + g_t, \tag{13}$$
$$n_t = m_t + r_t, \tag{14}$$
$$x_t^{gps} = x_t + 0.9 n_t. \tag{15}$$

Here, represents time–correlated noise with time constant $\beta$ and Gaussian white noise $g_t$, i.e., $g_t \sim N(0, \sigma_g)$. In addition, $n_t$ is the total noise composed of $m_t$ and uncorrelated noise $r_t$, i.e., $r_t \sim N(0, \sigma_r)$. As in the previous study [6], $\sigma_g$, $\sigma_r$, and $\beta$ were set as in Table III. The GNSS error in Case 2 was worse than that of Case 1.

*2) Environment with communication delays:* In this experiment, our communication delay compensation method was evaluated by comparing the performance our method, called "Fusion with DC" with those of the other two methods.

The first method (i.e., "GNSS only without DC") estimated the vehicle position by applying Kalman filters to the in-vehicle sensor data without compensating for the communication delays. This method only used the most recent position

TABLE IV
SPECIFICATIONS OF THE GNSS SENSORS USED IN THE
EXPERIMENT.

| AQLOC-VCX (Mitsubishi Electric) | |
|---|---|
| Receiver type | GPS, QZSS, Galileo |
| Horizontal position accuracy | 12.0 cm |
| Internal antenna | Dielectric antenna (59x59x33 mm) |
| UBX-M8030-KT (u-blox) | |
| Receiver type | GPS, QZSS, GLONASS |
| Horizontal position accuracy | 2.0 m |
| Internal antenna | Dielectric antenna (25x25x4 mm) |



Fig. 14. Experimental vehicle and sensor installation.



Fig. 15. Vehicle position acquired by GNSS sensors.

and velocity information received, thereby ignoring older delayed data. The second method (i.e., "GNSS only with DC") was similar, but with communication delay compensation.

In this method, the communication delays were represented by a Gaussian white noise (i.e., $N(\bar{d}, \sigma_{\bar{d}}^2)$) $\bar{d}$ was set to 96.130 ms following that in a previous study [11], and $\sigma_{\bar{d}}$ was set to 2 ms). We considered three possible maximum communication delays, namely 0.10, 0.12, and 0.14 s, and repeated each simulation six times.

*B. Evaluation using actual data*

In this experiment, the proposed system was also evaluated using a dataset created based on actual vehicle information and communication delays. The actual vehicle information were obtained when driving through a main lane of the airport west interchange in Tokyo, Japan. The dataset, including the communication delays, was created by actual transmission from a certain terminal to the other terminal via LTE communication. Accordingly, the performance of our proposed method ("Fusion") was compared with those of the other three methods. The first method ("RVDS only") estimated the vehicle position by applying a linear uniform motion to the time and speed, at which the vehicle passed the roadside vehicle detector. The second method ("GNSS only with CD") estimated the vehicle position by applying Kalman filters to the in-vehicle sensor data with communication delay compensation. The third method ("GNSS only without CD") estimated the vehicle position by applying Kalman filters to the in-vehicle sensor data without communication delay compensation. In this simulation, the standard deviation of the vehicle's center position, vehicle position, and velocity were set to s $\sigma_0^{rvd} = 0.5$ m, $\sigma_0^{gps} = 2$ m, and $\sigma_{v_t} = 5$ km/h, respectively. The vehicle sent its position and velocity to the roadside unit every 100 ms, and the roadside unit estimated the vehicle position every 100 ms.

*1) Acquisition of vehicle information:* The vehicle position and velocity were acquired using two sensors. Table IV lists the sensor specifications. AQLOC-VCX is a high-precision GNSS sensor that can acquire quasi-zenith satellites information. This sensor includes an IMU and can obtain the vehicle position and velocity by inputting the vehicle speed pulse and the back pulse. Therefore, the position information obtained from it were used as the correct vehicle position data, while the velocity information were used for the position estimation. UBX-M8030-KT is a general-purpose GNSS sensor. The position information obtained from it were used herein for
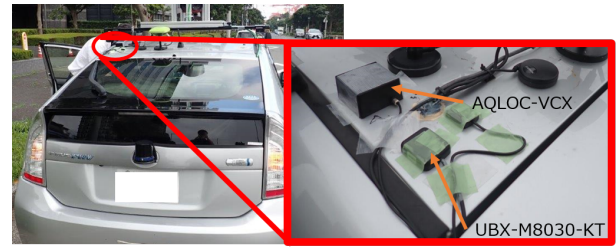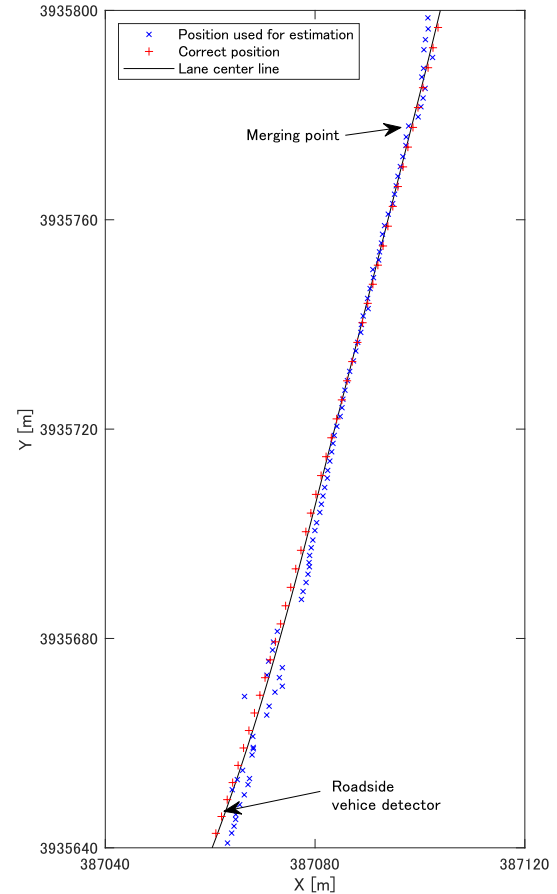
the position estimation. These sensors were mounted on the vehicle celling (Figure 14).

Figure 15 shows the vehicle position acquired by the GNSS sensors. The vehicle position was expressed in the UTM coordinate system. The lane center line was extracted from a high-precision map created using the mobile mapping system. The correct vehicle position was along the lane center line. Meanwhile, the position used for the estimation was far from the lane center line because of noise.

Our proposed position estimation method used a 1D CV model and considered lateral movement, but not vertical movement. The position information of the GNSS outputs were 2D. In this simulation, the position information were projected to the lane center line, and we only considered the lateral movement. Figure 16 shows how the position error changed
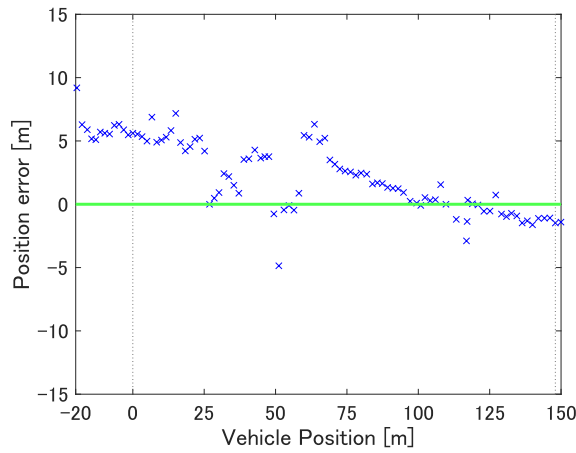
Fig. 16.   Position error versus true position.
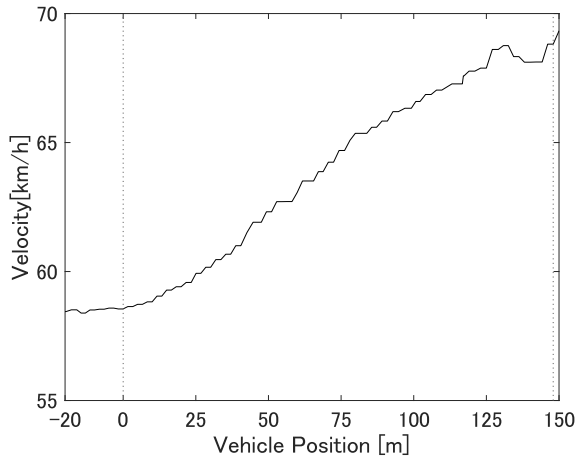


Fig. 17.   Vehicle velocity versus true position.
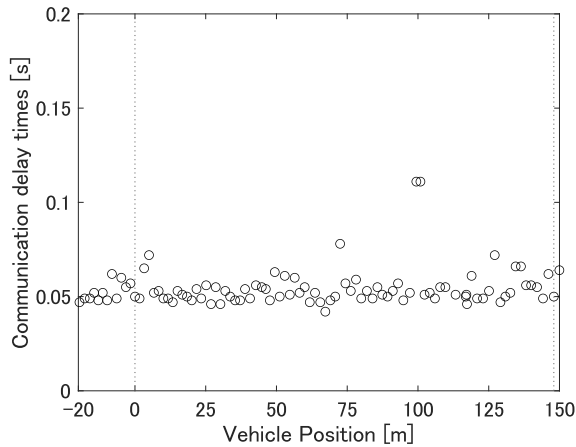


Fig. 18.   Communication delay times versus true position.

with the vehicle position. The roadside vehicle detector was installed at $x = 0$ m. The merging point was at $x = 148$ m. The vehicle estimation section started from $x = -20$ m to $x = 150$ m. The horizontal axis represents the position, whereas the vertical axis represents the position error, namely the difference between the correct vehicle position and the vehicle position used for estimation. When the position error

was positive (negative), the vehicle's estimated position was ahead (behind) its true position. The vehicle position used for the estimation had a large position error near the roadside vehicle detector and a small position error near the merging point.

Figure 17 exhibits how the vehicle velocity changed. The horizontal axis represents the true vehicle position, whereas the vertical axis represents the vehicle velocity at that position. The vehicle accelerated from the roadside vehicle detector toward the merging point.

*2) Addition of communication delays:* The information obtained from the GNSS sensors did not include communication delays. To contain the actual communication delays, the vehicle information were transmitted between the terminals that were time synchronized using the ntp server via LTE communication. The communication delays were then recorded. The communication carrier used in the experiments was b-mobile, mobile virtual network operator of Docomo. Docomo is the main communication carrier in Japan.

Figure 18 represents the state of the communication delays. The horizontal axis represents the actual vehicle position, whereas the vertical axis represents the communication delay times. The average value of the communication delays were approximately 0.05 s. The maximum value is 0.111 s, whereas the minimum value was 0.042 s. Therefore, the maximum communication delay time was set to 0.12 s in this experiment.

## V. RESULTS

### A. Evaluation using models

*1) Environment without communication delays:* Table V presents the simulation results. Figures 19-23 show how the position error changed with the vehicle position. The horizontal axis represents the true vehicle position, while the vertical axis represents the position error, namely the difference between the true and estimated vehicle positions. When the position error was positive (negative), the vehicle's estimated position was ahead (behind) its true position.

Table V shows that the average and standard deviation of the position error were both lower for "Fusion" than for "GNSS only." When the vehicle passed through the roadside vehicle detector, the position error of the "Fusion" method sharply dropped, becoming much lower than that of the "GNSS only" (Figures 19-23). This result demonstrates that the proposed position estimation method can be significantly more accurate than "GNSS only."

*2) Environment with communication delays:* Table VI shows the simulation results. Figures 24-26 depict how the position error changed with the vehicle position. The horizontal axis represents the true vehicle positions, whereas the vertical axis represents the position error defined as before.

Table VI shows that both the average and standard deviation of the position error were lower for our "Fusion with DC" method compared to the other approaches. The errors were lower for the "GNSS only with DC" than for the "GNSS only without DC" and significantly lower for "Fusion with DC" than for the other methods when the maximum communication

TABLE V
COMPARISON OF POSITION ESTIMATION METHODS.

| Gaussian white noise model (Low GNSS position error) | Average | Standard deviation |
|---|---|---|
| GNSS only | 0.298 m | 0.182 m |
| Fusion | 0.238 m | 0.030 m |
| Gaussian white noise model (Medium GNSS position error) | Average | Standard deviation |
| GNSS only | 0.360 m | 0.290 m |
| Fusion | 0.260 m | 0.052 m |
| Gaussian white noise model (High GNSS position error) | Average | Standard deviation |
| GNSS only | 0.467 m | 0.461 m |
| Fusion | 0.274 m | 0.085 m |
| Gauss–Markov random process model (Case 1) | Average | Standard deviation |
| GNSS only | 1.681 m | 0.907 m |
| Fusion | 1.280 m | 0.591 m |
| Gauss–Markov random process model (Case 2) | Average | Standard deviation |
| GNSS only | 0.691 m | 0.467 m |
| Fusion | 0.477 m | 0.315 m |
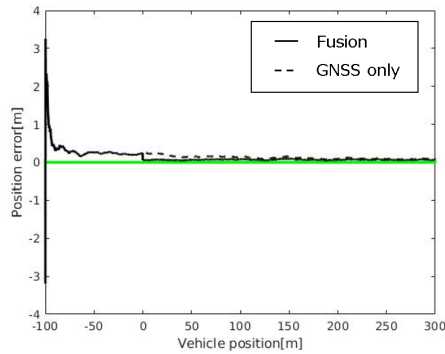


Fig. 19. Position error versus true position (Gaussian white noise model with a low GNSS position error).
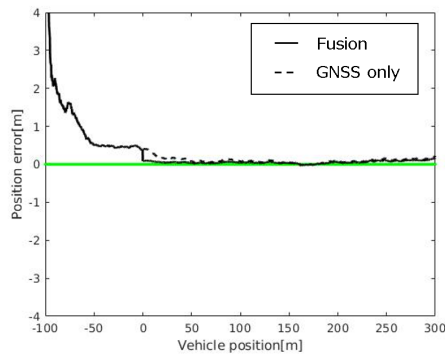


Fig. 20. Position error versus true position (Gaussian white noise model with a medium GNSS position error).

delay time was 0.14 s. We believe this was because the amount of data that had to be discarded due to not being received within the maximum communication delay time decreased as the maximum communication delay time increased.

Figures 24-26 depict that the position error was always negative for the "GNSS only without DC" method because the latest information received from the vehicle was out-of-data due to communication delays. The fact that the errors
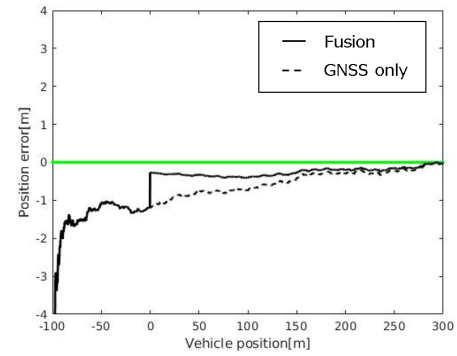


Fig. 21. Position error versus true position (Gaussian white noise model with a high GNSS position error).
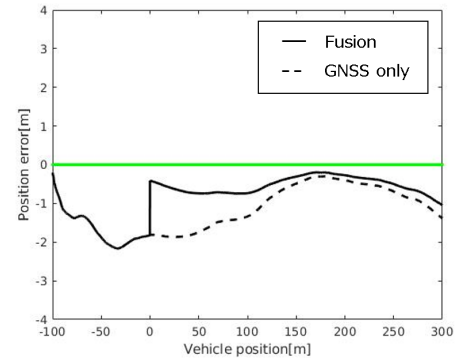


Fig. 22. Position error versus true position (Gauss-Markov random process model, Case 1).
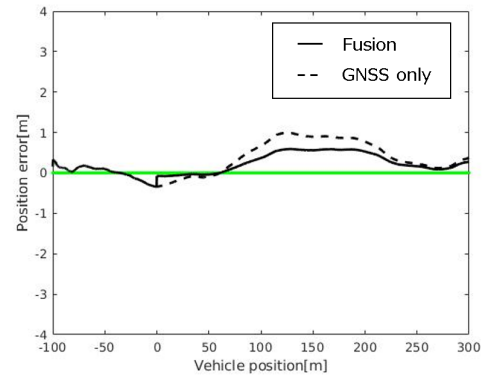


Fig. 23. Position error versus true position (Gauss-Markov random process model, Case 2).

were smaller for both "GNSS only with DC" and "Fusion with DC" confirmed that our communication delay compensation method performed well. In addition, the fact that the position errors were lower for our "Fusion with DC" method than for "GNSS only with DC" confirmed that our proposed method can more accurately estimate the vehicle position than "GNSS only" in an environment with communication delays.

### B. Evaluation using actual data

Table VII lists the simulation results. Figure 27 shows how the position error changed with the vehicle position. Table VII

TABLE VI
EVALUATION OF OUR COMMUNICATION DELAYS
COMPENSATION METHODS.

| | Average | Standard deviation |
|---|---|---|
| GNSS only without DC | -2.420 m | 0.612 m |
| Maximum communication delay of 0.10 s | Average | Standard deviation |
| GNSS only with DC | -0.471 m | 0.527 m |
| Fusion with DC | -0.248 m | 0.065 m |
| Maximum communication delay of 0.12 s | Average | Standard deviation |
| GNSS only with DC | -0.410 m | 0.411 m |
| Fusion with DC | -0.246 m | 0.059 m |
| Maximum communication delay of 0.14 s | Average | Standard deviation |
| GNSS only with DC | -0.380 m | 0.362 m |
| Fusion with DC | -0.234 m | 0.063 m |
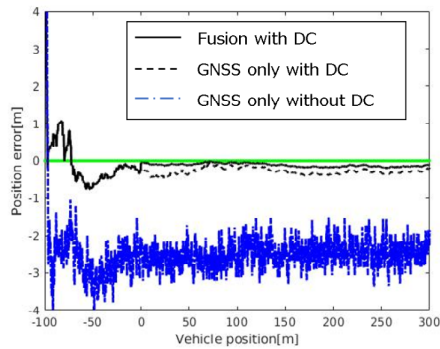


Fig. 24. Evaluation of our communication delay compensation method (maximum communication delay of 0.10s).

TABLE VII
COMPARISON OF POSITION ESTIMATION METHODS.

| | Position error at the merging point | Average | Standard deviation |
|---|---|---|---|
| Fusion | -1.037 m | 0.576 m | 1.707 m |
| RVDS only | -14.420 m | -8.035 m | 6.722 m |
| GNSS only with DC | -0.862 m | 2.181 m | 2.840 m |
| GNSS only without DC | -2.513 m | 0.468 m | 2.889 m |

summarizes the position error at the merging point, and the average values and standard deviation of the position errors after passing the roadside vehicle detector until reaching the merging points. The horizontal axis represents the true vehicle position, whereas the vertical axis represents the position error, namely the difference between the true and estimated vehicle positions. The vehicle's estimated position was ahead (behind) its true position when the position error was positive (negative).

Table VII shows that "GNSS only with DC" method had the smallest position error at the merging point. The position estimation method with the smallest average value was "GNSS only without DC," and the method with the smallest standard value was "Fusion." The value of "RVDS only" was large in all items.

Figure 27 also depicts that the position errors of "Fusion" and "RVDS only" were very small when the vehicle passed through the roadside vehicle detector and the position can accurately be obtained. The positon error of "RVDS only" gradually left behind its true position because the vehicle
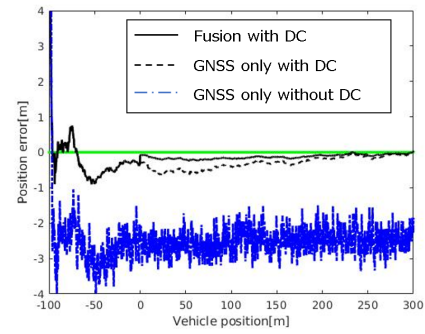


Fig. 25. Evaluation of our communication delay compensation method (maximum communication delay of 0.12s).
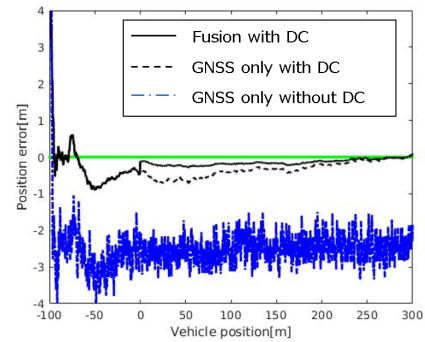


Fig. 26. Evaluation of our communication delay compensation method (maximum communication delay of 0.14s).
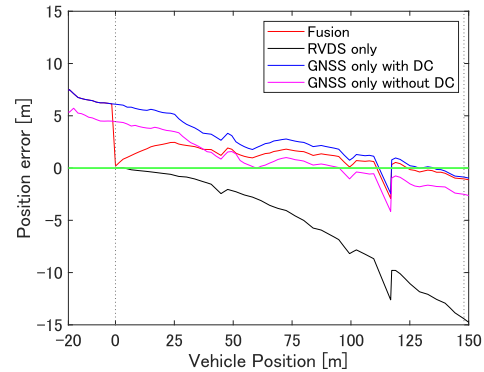


Fig. 27. Position error versus true position.

accelerated after passing through the roadside vehicle detector (Figure 17). Finally, the fact that the estimated position were always behind for the "GNSS only with DC" method than for the "GNSS only without DC" method confirmed that our communication delay compensation method performed well.

## VI. DISCUSSION

This section discusses the advantage of the proposed method based on the evaluation results.

### A. Advantages of the proposed method

The position estimation using only the roadside vehicle detector information can accurately estimate the vehicle position near the roadside vehicle detector. However, an accurate

position estimation cannot be performed when the vehicle velocity changes after passing the roadside vehicle detector. Vehicle velocity changes often occur to deal with vehicles merging into the main lane; thus, vehicle speed information are necessary for an accurate position estimation. In this respect, the proposed method and the position estimation method using only the in-vehicle sensors have an advantage.

Merging support systems create merging support information based on the position information of the vehicles approaching the merging point. Therefore, the position information near the merging point are more important for the merging support system than at the merging point. Furthermore, the merging support system may provide strange merging support information if the standard deviation is large, even if the average of the position error is small after passing the roadside vehicle detector until reaching the merging points. Thus, the average and standard deviation of the position error must be small after passing the roadside vehicle detector until reaching the merging points. The evaluation experiment confirmed that the proposed method can accurately estimate the vehicle position near the roadside vehicle detector because of the information from the roadside vehicle detector. Consequently, we confirmed that the average and standard deviation of the position error after passing the roadside vehicle detector until reaching the merging points become relatively small. Therefore, the proposed method has an advantage over the position estimation method using only in-vehicle sensor information in terms of support system reliability.

The method without the communication delay compensation estimates the position based on the past information of the GNSS due to the communication delays. The evaluation experiment showed that the estimated position was always behind the position estimated using the method with the communication delay compensation. Thus, communication delay compensation is useful in the environment with communication delays.

### B. Maximum communication delay time

In the evaluation using the actual data, we set the maximum communication delay time based on the record of the communication delay times. Communication delays are influenced by various factors, such as the number of vehicles and building. The time required for the position estimation increases as the maximum communication delay time increases. The maximum communication delay time must be set considering the time constrain of the merging support system and the communication delay time in various situations.

## VII. CONCLUSION

In this study, our system was evaluated in several simulations using the various models and actual data. We demonstrated that the proposed method can estimate vehicle positions more accurately than that using only the in-vehicle sensors and roadside vehicle detector. We also confirmed that our communication delay compensation method can perform well. Our method can estimate the vehicle positions accurately in

environments with communication delays; hence, it is more suitable for managing traffic and controlling merging at junctions.

We will explore several topics in the future work. The proposed method can be applied while the delay occurs according to a specific distribution. The communication delay distribution tendency drastically changes with the number of increasing vehicles. Thus, a method that can handle it, even if the communication delay distribution tendency is changed, must be considered. We would like to evaluate whether or not the proposed method works correctly by incorporating it into the merging support system proposed in the previous research.

## REFERENCES

[1] S. Yamada, Y. Watanabe and H. Takada, "A Vehicle Position Estimation Method Combining Roadside Vehicle Detector and In-Vehicle Sensors," in Proceedings of the Nineteenth International Conference on Networks(ICN2020), Feb 23–27, 2020, Lisbon, Portugal, pp.7–14, ISBN: 978-1-61208-770-2.

[2] T. Abbas, L. Bernado, A. Thiel, C. F. Mecklenbräuker, and F. Tufvesson, "Measurements based channel characterization for vehicle-to-vehicle communications at merging lanes on highway," in Proceedings of the 5$^{th}$ International Symposium Wireless Vehicular Communications(WiVeC), June 2–3, 2013, Dresden, Germany, Jun. 2013, pp.1–5, ISBN: 978-1-4673-6339-6, URL: https://ieeexplore.ieee.org/abstract/document/6698241 [retrieved: 12, 2019].

[3] B. N. Janson, W. Awad, J. Robles, J. Kononov and B. Pinkerton, "Truck accidents at freeway ramps: data analysis and high-risk site identification," Journal of Transportation and Statistics, vol. 1,Jan. 1998, pp. 75–92, ISSN: 1094-8848.

[4] H. Cui et al., "Early ramp warning using vehicle behavior analysis," Soft Computing, vol. 22, March, 2018, pp. 1421–1432, ISSN: 1432-7643.

[5] V. Milanes, J. Godoy, J. Villagra and J. Perez, "Automated On-Ramp Merging System for Congested Traffic Situations," IEEE Transactions on Intelligent Transportation Systems, vol. 12, Jun. 2011, pp. 500–508, ISSN: 1524-9050.

[6] F. C. Chou, S. E. Shladover, and G. Bansal, "Coordinated merge control based on V2V communication," in Proceedings of the Vehicular Networking Conference (VNC) December 8–10, 2016, Ohio, USA, Dec. 2016, pp.1–8, ISBN: 978-1-5090-5197-7, ISSN: 2157-9865 , URL:https://ieeexplore.ieee.org/document/7835933 [retrieved: 12, 2019].

[7] S. Hirai et al., "AHS Safety Service Utilizing an ITS On-Board Unit for Driving Support in Merging Sections," World Congress on ITS, pp. 1–8, October, 2007.

[8] "Start of Field Operational Tests of Autonomous Driving in Tokyo Waterfront Area," URL: http://www.soumu.go.jp/menu_news/s-news/01kiban14_02000404.html [retrieved: 12, 2019].

[9] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey," IEEE, vol. 18, pp. 45–50, August, 2004.

[10] Y. Morales and T. Tsubouchi, "DGPS, RTK-GPS and StarFire DGPS performance under tree shading environments," in Proceedings of IEEE international conference on integration technology(ICIT), March 20–24, 2007, Shenzhen, China, pp. 519–524, ISBN:1-4244-1091-6, URL:https://ieeexplore.ieee.org/abstract/document/4290370 [retrieved: 12, 2019].

[11] K. C. Deya, A. Rayamajh, M. Chowdhury, P. Bhavsar and J. Martin, "Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in a heterogeneous wireless network - Performance evaluation," Transportation Research Part C, vol. 68, pp. 168–184, April, 2016.

[12] W. Anderson and R. Duffin, "Series and parallel addition of matrices," Journal of Mathematical Analysis Applications, vol. 26, pp. 576–594, 1969.

[13] "Start of joint research on information provision to support autonomous driving," URL: http://www.nilim.go.jp/lab/bcg/kisya/journal/kisya20180119.pdf [retrieved: 12, 2019].

[14] S. Kagami and M. Ishikawa, "A sensor selection method considering communication delays," in Proceedings of IEEE International Conference on Robotics and Automation(ICRA) April 26–May 1, 2004, LA, USA, pp.206–211, ISBN: 0-7803-8232-3, ISSN: 1050-4729, URL: https://ieeexplore.ieee.org/document/13071529 [retrieved: 12, 2019].

[15] J. Rankin, "An error model for sensor simulation GPS and differential GPS," in Proceedings of of IEEE Position Location and Navigation Symposium(PLANS'94) April 11–15, 1994, Las Vegas, USA, pp.260–266, ISBN: 0-7803-1435-2, URL: https://ieeexplore.ieee.org/document/303322 [retrieved: 12, 2019].

# Near-Ground IEEE 802.11 b/g/n Coverage Design for Precision Agriculture

Marta Botella-Campos[1], José Miguel Jiménez[1], Sandra Sendra[1], Jaime Lloret[1]

[1]*Instituto de Investigación para la Gestión Integrada de zonas Costeras (IGIC)*
*Universitat Politècnica de València,* València, Spain
marbocam@teleco.upv.es, jojiher@dcom.upv.es, sansenco@upv.es, jlloret@dcom.upv.es

*Abstract*—**Due to the broad range of options that wireless systems offer; Wi-Fi products are increasingly being used in agriculture environments to improve farming practices and better control the output of the production. However, the foliage has proven to harm radio-frequency propagation as well as decreasing the coverage area of Wireless Sensor Networks (WSNs). Therefore, near-ground channel characterization can help in avoiding high antennas and vegetation. Nevertheless, theoretical models tend to fail forecasting near-ground path losses. A comprehensive analysis of the influence of rural environments can help obtain better near-ground WSN performance and coverage in precision agriculture. Hence, this paper aims at determining how field components such as, soil, grass, and trunks affect radio-links in near-ground scenarios. To do this, we measure the Received Signal Strength (RSSI), the Signal to Interference Ratio (SIR) and the Round-Trip Time (RTT) of a Wireless Local Area Network (WLAN) at different distances and compare the results with 6 prediction models: the Free-Space Propagation Model, the Two-Ray Ground Reflection Model, the One-Slope Log-Normal Model, the Two-Slope Log-Normal Model, the Okumura-Hata Model, and the Walfisch-Ikegami Model. The experiment was carried out by collecting experimental data at four different locations, i.e., an orange tree plantation with mature trees, an orange tree plantation with young trees, a persimmon field, and a field without vegetation, to compare and contrast the influence of field components on signal losses.**

*Keywords—Wireless Coverage; IEEE 802.11; Precision agriculture; Propagation Losses; Propagation Model.*

## I. INTRODUCTION

This paper is an extended version of the paper presented by M. Botella-Campos et al. in [1].

Smart Farming (SF) emphasizes the use of Information and Communication Technologies (ICTs) to leverage the farm management cycle. Improving the production capacity does not only enhance business efficiency. It also allows to increase production and reduce the environmental impact. Since the world population is expected to reach 9.8 billion by 2050, human societies are facing the challenge of providing nourishment and livelihoods, while addressing the effects of climate change [2]. Smart farming applies measures that are ecologically meaningful and site-specific, focusing on implementing auto-piloted harvesters and other farm machinery [3].

The Internet of Things (IoT) and Cloud Computing are expected to move forward in farming management development by introducing these technologies into machinery and production systems [4]. The gathered information will then be sent via different technologies such as, IEEE 802.11 standards, Bluetooth, Zigbee, LoRa, 6LoWPAN, 3G, 4G, etc., depending on the amount of data to be transmitted [5]. Although IoT systems usually deal with small amounts of data to be transmitted through short distances, in some cases it could be required to send higher amounts of data and include images to monitor the status of the plants. The two main storage systems used to save the gathered information from the sensors are traditional databases or clouds. The most used databases are MySQL and SQL, while the ThingSpeak platform is the most used in cloud systems [5].

Wireless Sensor Networks (WSNs) are needed to monitor environmental conditions and provide decision-making information and are composed of a group of spatially dispersed sensors to monitor and record environmental conditions such as, humidity, temperature, soil moisture, etc. WSNs are made up of four parts: a wireless sensor node, a gateway node, a wireless communication network, and a server [6]. Nowadays, sensor nodes have evolved to be small devices with sensing, communication, and computing devices. However, each node can only monitor a specific part of the field. Thus, the coverage area is a key problem since all nodes among a WSN must be autonomous to cooperatively pass data through the network to a main location. Moreover, its topology can vary enormously depending on the field.

Many WSN applications generally use IEEE 802.11 g/n standard because it allows distances up to approximately 300 meters in outdoor environments (when there is free space between devices) [7]. This allows a maximum raw data throughput of 54 Mbps or 600 Mbps, depending on the standard used. Likewise, the radio-frequency band can vary from 2.4 GHz to 5 GHz using Modulation Code Keying (CCK), Direct-sequence Spread Spectrum (DSSS) or Orthogonal Frequency-Division Multiplexing (OFDM) modulation schemes. Moreover, the use of this standard will reduce the cost of the deployment of the nodes as well as other time-consuming actions when processing the data.

In 2019, Bacco et al. [8] conducted a survey on SF research activities to state the achieved results and current investigations within EU territory. In this study, challenges impeding the adoption of recent technologies and techniques

were highlighted. Although the current use of sensor nodes and analytic techniques is boosting Decision Support Systems (DSSs) in farms, the lack of diffusion programs is preventing areas affected by the digital divide from incorporating ICTs. Nevertheless, technology is expected to have an increasing role in agriculture so that operations, such as, planting and harvesting, may be automatized. Moreover, the availability of real-time data will allow finer control of pesticides and other chemicals. However, none of this will be possible without supporting policies to address poor telecommunication infrastructures and reduced digital skills.

As for Precision Agriculture (PA), Lindblom et al. [9] conducted a review on agricultural DSSs within the frame of the ongoing Swedish project. This project intends to identify the scientific disciplines and other competences that need to work together in developing technology for agricultural DSS. Therefore, the discussion is focused on the importance of considering in-land processes to design suitable WSNs. However, the lack of active participation in agricultural research and development processes is preventing the development of new practices and behaviours for more sustainable farming.

The effect of vegetation on radio-wave signals can be of great deal when designing such networks. Many studies have strived to evaluate and characterise land components that may affect signal strength. Still, the lack of experimental data has prevented scientists from developing generalized procedures to assess the foliage effect on attenuation. The International Telecommunication Union (ITU) recommends a series of propagation models depending not only on frequency ranges but also different link geometries and vegetation types [10]. Nonetheless, relying on prediction models in near-ground scenarios may lead on over or underestimating the networking capacities of terrestrial systems.

This paper aims to study near-ground wireless coverage in rural environments to ease multi-hop routing design. To this end, the Received Signal Strength Indicator (RSSI), the Signal to Interference Ratio (SIR), and the Round-Trip Time (RTT) of a wireless signal were measured at two orange fields, a persimmon field, and a land with no vegetation. This study attempts to determine how near-ground radio-links are affected by field components such as, grass, soil, and trunks, and examine how accurate path loss estimation models are in this type of scenarios. In all cases, we used an access point and a laptop to take measurements at different distances, 30 cm above the ground at 2.437 GHz.

The rest of this paper is structured as follows. Section II presents some related work. In Section III, several well-known propagation models are explained. The methodology and materials used in the experiment are presented in Section IV. In Section V, the experimental results are analysed. A comparison with previous works discussed in Section VI. Finally, the main conclusions and future work are exposed in Section VII.

## II. RELATED WORK

Few technical works characterize near-ground radio-frequency propagation. In this section, some of the related works are discussed.

In [11], Torabi et al. proposed a near-ground prediction model to facilitate accurate WSN simulations using the principles of the Fresnel zones. In this study, the effects of antenna height, frequency, polarization, and electrical and geometrical properties of the terrain were studied. The accuracy of the proposed model was verified by comparing the theoretical results with near-ground measurements carried out in outdoor open areas at 300 MHz and 868 MHz. The results of this study showed that antenna height was by far the most influential parameter on network connectivity. Moreover, the wireless connection was proven to be sensitive to the reflection coefficient in near-ground situations.

In 2014, Yildiz et al. [12] investigated the impact of path loss models on near-ground WSN lifetime. In this study, researchers performed a comparative analysis on the effects of path loss models and proposed a novel Mixed Integer Program (MIP) to maximize WSN lifetime. By designing an effective energy dissipation system, and using empirically validated characteristics of Mica2 motes, this investigation managed to characterize the parameter space through numerical evaluations of the MIP model at four different frequencies: 315 MHz, 433MHz, and 868/916 MHz. The analysis of the results demonstrated that theoretical models, such as, the Free-Space Model and the Two-Ray Model, can lead to overestimations on WSN lifetime and should be avoided in such scenarios.

Tang et al. [13] studied a near-ground WSN at 470 MHz in four different scenarios to obtain the corresponding path loss models. To do this, measurements were taken on a flat concrete road, flat grass and two derived scenarios placing the transmitter directly on the ground. Three different antenna heights were used: 5 cm, 50 cm and, 1 m, and the RSSI was measured every meter at a distance up to 10 m, every 2 m at a distance of up to 20 m and every 5 meters at a distance of up to 50 m. The results showed that when antenna height is lower than 50 cm, prediction models tend to inaccurately forecast path loss and thus, network connectivity.

Klaina et al. [14] performed a narrowband characterization of near-ground channels for WSNs at 5G-IoT bands. In this study, RSSI signal was measured at two different heights: 0.2 and 0.4 m, to test the effects of ground coverings in path loss for three frequency ranges: 868 MHz, 2.4 GHz, and 5.8 GHz. To fit the signal strength decay caused by agriculture fields, an experimental measurement campaign was carried in agriculture fields with three types of ground: soil, short and tall grass. The path loss was estimated with a proposed narrowband radio channel model: the three-slope log-distance model. The analysis of the results demonstrated that the difference between the Free-Space model and the measured path loss is reduced with the increase in frequency. Moreover, lowering antenna heights increased attenuation.

In 2017, Klaina et al. [15] presented a narrowband radio channel model operating under near-ground conditions. To do

this, a WSN based on ZigBee was designed to analyse the effects caused by soil and grass fields. In this case, radio communications were made at 868 MHz, 2.5 GHz, and 5.8 GHz. In order to estimate signal quality, RSSI was measured and compared to path loss. Finally, they concluded that the ground has no effects on RF propagation except in the cases where antenna heights were 40 cm or less. However, signal levels decreased in the presence of grass fields and soil.

Masadan et al. [16] studied the foliage effect in terms of attenuation and its contribution to the pathloss and link budget estimations. In this study, researchers quantified the path loss through 5 tree types in Malaysia for different path crossings (trunk, tree-top and branches) at 915 MHz. The analysis of the results showed that the RSSI values obtained in Line of Sight (LOS) scenarios were lower than expected due to tropical climate environment, as well as the size of the trees and density of the crowns. As for Non-Line of Sight (NLOS) scenarios, the obstructions caused diffractions, scattering of energy, and reflections that lead to larger attenuations.

In [17], Wang et al. depicted a statistical model for near-ground channels based on experimental data collected through three different scenarios at 2.4 GHz. The main objective of this study was to develop a WSN to collect data in military explosive research. To do this, sensor nodes with a 3 cm antenna height were fixed to the ground to resist damages from detonations. Different propagation models were applied to predict path loss and compare the results with the performance of the obtained model. The main conclusion of this research was that antenna height determines the breakpoint distance of the nodes.

In 2011, Lloret et al. [18] presented a WSN that uses image processing to detect bad leaves in vineyards and sends an alarm to the farmer. In this case, wireless communications are made through IEEE 802.11 a/b/g/n standard to allow long-distance connections. Although the proposed system does not identify the cause of the deficiency, it detects bad leaves and notifies it to the farmer who can then decide what actions need to be taken. This solution provides a cost-effective sensor based on IP routers that have been adapted to fulfil this purpose. The designed WSN takes into account both sensing and radio coverage areas to allow low bandwidth consumption and higher scalability. The system to detect bad leaves goes through a 5-stage process before the node decides whether an alarm needs to be sent.

In 2015, Szajna et al. [19] characterized path loss and near-ground channels at 2.45 GHz on forested areas covered by snow. This study aimed to investigate the impact of antenna height and distance between nodes on path loss and special correlation. To do this, measurements were carried out in two different scenarios: a multi-purpose sports facility and a forested area covered by 15 cm of snow. In this case, antenna heights varied from 0 to 130.8 cm and the distance between the nodes varied in steps of 15.24 m and up to 79.2 m. The analysis of the results showed that reducing antenna heights increased path loss and reduced spatial correlation.

Luciani et al. [20] described a study done on near-ground node range at different heights in Wi-Fi crowded environments. The designed WSN used IEEE 802.15.4 standard to avoid direct Wi-Fi interference. Signal quality and range were determined by collecting RSSI data of three nodes at increasing node separation distance until signal loss. To perform the tests, measures were taken at three different heights: 15 cm, 30 cm, and 100 cm, at three different scenarios. The results of this experiment showed that prediction models failed to accurately forecast path loss. Moreover, ground-loss proved to be a major issue that determines node range and thus, must be taken into account when designing WSNs.

Sangodoyin et al. [21] presented a near-ground channel model to achieve precision ranging and localization of ultrawideband (UWB) propagation channels. This experiment was performed using a self-built channel sounder with an arbitrary waveform generator and a high-bandwidth sampling oscilloscope. In this case, antenna heights ranged from 10 cm up to 2 m above ground to determine its effects on signal strength. The results showed that the distance-dependent path loss was highly dependent on antenna heights. Moreover, under near-ground situations, frequency-dependent path loss exponent and shadowing variance increased.

Though many of these studies have attempted to characterize near-ground wireless systems in rural environments, the wide range of parameters to take into account, together with the randomness introduced by foliage and its different geometries in fields, have prevented researchers from stablishing a generalized procedure to assess the design and planning of WSN.

For these reasons, in this work we present a site-specific study to guarantee the performance of near-ground radio-links in fruit plantations. To do this, measurements of RSSI, SIR, and RTT were collected at two orange tree plantations with different tree sizes, as well as a persimmon plantation and a land with no vegetation, to stablish the accuracy of six prediction models and test how distinct vegetation environments affect radio-links.

## III. PROPAGATION MODELS

In this section, several propagation models are presented to predict the average signal strength drop and assess the level of accuracy that can be achieved in near-ground WSN scenarios. Thus, this section is divided into six different subsections.

### A. Free-Space Model

The Free-space propagation model is the simplest way to calculate radio-signals propagation. From [12], we can extract the Free-Space propagation model based on Friis Transmission Formula. This equation is usually used when there are no obstacles in the LOS, and it is given by equation (1).

$$P_r = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2} \tag{1}$$

where:

$P_t$: transmitter power, in watts.

$G_t$: transmitter antenna gain.

$G_r$: receiver antenna gain.

$\lambda$: wavelength.

d: distance, in meters, between transmitter and receiver.

However, it is possible to calculate the losses between a transmitter and a receiver in terms of the frequency with equation (2).

$$FSPL\ (dB) = 20 \log\left(\frac{4\pi\ d\ f}{c}\right) - G_T - G_R \quad (2)$$

where:

d: distance, in meters, between transmitter and receiver.

f: frequency in Hz.

c: speed of light in the vacuum (meters per second).

$G_T$: transmitter antenna gain, in dBi.

$G_R$: receiver antenna gain, in dBi.

### B. Two-Ray Ground Reflection Model

The Two-Ray Ground Reflection Model predicts path losses between a Tx and a Rx when they are both in line-of-sight but have different antenna heights. This way, the received signal has two components: the line-of-sight component and the multipath component which is given by ground reflected waves. From [12], the given equation for the Two-Ray Model can be expressed by equation (3).

$$P_r = P_t G_t G_r \frac{h_t{}^2 h_r{}^2}{d^4} \quad (3)$$

where:

$P_t$: transmitter power, in watts.

$G_t$: transmitter antenna gain.

$G_r$: receiver antenna gain.

$h_t$: transmitter antenna height, in meters.

$h_r$: receiver antenna height, in meters.

d: distance, in meters, between transmitter and receiver.

Nevertheless, from the work in [13], we can tell that when radio-waves propagate near-ground in line-of-sight conditions, the path loss can be described by the plane-earth path loss formula, given by equation (4).

$$PL(dB) = 40 \log(d) - 20 \log(h_r) - 20 \log(h_t) \quad (4)$$

where:

d: distance, in meters, between transmitter and receiver.

$h_t$: transmitter antenna height, in meters.

$h_r$: receiver antenna height, in meters.

### C. One-Slope Log-Normal Model

The log-distance path loss model is a statistical model that takes into consideration object blockage, environmental clutter, and other changes to predict path loss. From [12], the log-normal model can be described by equation (5).

$$PL\ (d) = PL(d_0) + 10n \log\left(\frac{d}{d_0}\right) + X_\sigma \quad (5)$$

where:

PL (d): path loss at distance d, in dB.

PL ($d_0$): path loss, in dB, at reference distance of 1 meter (FSPL at 1 meter).

n: path loss factor (n = 2).

$X_\sigma$: zero mean Gaussian distributed variable with standard deviation $\sigma$.

$\sigma$: linear regression of measured data.

However, from reference [22] we can express One-Slope Log-Normal Model by equation (6).

$$PL\ (d) = FSPL(f, 1\ m) + 10n \log\left(\frac{d}{1\ m}\right) \quad (6)$$

where:

PL (d): path loss at distance d, in dB.

FSPL (f, 1 m): free space path loss, in dB, at a reference distance of 1 meter.

n: path loss factor (n = 2).

d: distance, in meters, between transmitter and receiver.

### D. Two-Slope Log-Normal Model

Although the Two-Slope Model is not often utilized in near-ground propagation scenarios, experimental studies on WSNs have shown that its use may result on better representations of the collected data [12]. From [20], this path loss estimation model can be expressed by equation (7).

$$PL(d_i) = \begin{cases} L_0 + 10\ n_1\ \log_{10}\left(\dfrac{d_i}{d_0}\right) & d_i \leq d_b \\[2ex] L_{b+1} + 10\ n_2\ \log_{10}\left(\dfrac{d_i}{d_{b+1}}\right) & d_i > d_b \end{cases} \quad (7)$$

where:

PL (di): path loss at di distance (in meters), in dB.

db: breakpoint distance, in meters.

$L_0$, $L_{b+1}$: path losses before and after the breakpoint, respectively.

$n_1$, $n_2$: path loss factor (n = 2).

The breakpoint distance indicates a change rate of the path loss and can be calculated by equation (8) in LOS conditions [17].

$$d_b = \frac{4h_t h_r}{\lambda} \qquad (8)$$

In this case, the estimated breakpoint distance for the given dataset was 8.64 meters.

### E. Okumura-Hata Model

The Okumura-Hata path loss estimation model is an empirical formulation typically used for the frequency range of 150 MHz to 1500 MHz. However, this radio propagation model identifies the signal attenuation caused by reflections, diffractions, and the scattering of energy [16]. From [23], the Okumura-Hata path loss can be calculated by equation (9).

$$L_b = 69.55 + 26.16 \log(f) - 13.82 \log(h_b) - a(h_m) \qquad (9)$$
$$+ [44.9 - 6.55 \log(h_b)] \log(d) + C$$

where:

    f: frequency, in MHz.

    $h_b$, $h_m$: Tx and Rx antenna heights in meters, respectively.

    d: distance between transceiver and receiver, in kilometres.

The function a(hm) is dependent on the environment. In the case of rural areas, this correction factor corresponds to the same as in urban areas [23] and is described by equation (10).

$$a(h_m) = (1.1 \log(f) - 0.7)h_m - (1.56 \log(f) - 0.8) \qquad (10)$$

As for the factor C, equation (11) formulates its value for rural areas [23].

$$C = -4.78[\log(f)]^2 + 18.33 \log(f) - 40.98 \qquad (11)$$

### F. Walfisch-Ikegami Model

This empirical model is considered to have a high accuracy in urban environments when the distance between the Tx and the Rx is relatively small [23-24]. In LOS scenarios, the Walfisch-Ikegami estimation model can be described by equation (12).

$$L_{LOS}(dB) = 42.6 + 26 \log(d) + 20 \log(f) \qquad (12)$$

where:

    d: distance between Tx and Rx in the range of 20 m to 5000 m, in kilometres.

    f: frequency in MHz (800-2000 MHz).

Other studies have determined that, when antenna heights are lower than 50 cm, the One-Slope Model tends to estimate path losses better than other models [13]. However, other researchers state that the use of these theoretical models can lead to overestimations of the networking capacities and should be avoided [12]. In the following sections, we will compare these six models with collected data to evaluate their performance and verify their accuracy in near-ground scenarios.

### IV. SCENARIO DESCRIPTION AND TOOLS USED

This section describes the devices used to perform the experiments, as well as the setup. Therefore, this section is segmented in four different subdivisions.

### A. Place of measurement

In order to evaluate the path loss caused by vegetation in near-ground radio-wave signals, we sought out different plantations with no walls: an orange tree plantation of mature trees, an orange tree plantation of young trees, and a persimmon plantation of mature trees. Furthermore, we collected data at a land with no vegetation in order to compare and contrast the attenuation introduced by field components.

### B. Hardware used

To perform this experiment, we used Linksys WRT320N-EZ router as a Tx configured to work at 2.437 GHz (channel 6) with IEEE 802.11 b/g/n standard [25]. This router has three internal antennas with 1.5 dBi of antenna gain and an RF power of 17 dBm. The Rx was ASUS Gaming Notebook GL753V, which has a 2.8 GHz Intel Core i7-7700 HQ processor, 16 GB of memory. Wireless connections are made with Intel Dual Band Wireless Wifi Bluetooth Card 7265NGW that uses the IEEE 802.11 ac standard and has two antennas of 5 dBi of gain.

### C. Software used

The measurements were made using the software Vistumbler [26] to scan the wireless network and measure both the SIR and the RSSI. As for the latency of the connection, it was measured by sending a ping signal through MS-DOS commands to the gateway.

### D. Set-up of the experiment

Both Tx and Rx were positioned along the same line, 30 cm above the floor to measure the SIR and the RSSI. The evaluation of the path loss of RF signals was made by taking measurements in four different scenarios.

- Scenario 1: Measurements were made on a field with no vegetation, collecting data every meter 30 cm above the ground.

- Scenario 2: Measurements were made at an orange tree plantation with mature trees, with data being collected every meter 30 cm above the ground.

- Scenario 3: Measurements were made at an orange tree plantation of young trees (3-year-old), with data being collected every 2 meters, 30 cm above the ground.

- Scenario 4: Measurements were made at a persimmon plantation, with data being collected every 2.5 meters, 30 cm above the ground.

Fig. 1 illustrates the set-up of the experiment and the vegetation geometry of Scenario 2. In order to be able to perform comparisons of the signal strength, measurements were made at the same distances in Scenario 1. Fig. 2 shows the set-up in Scenario 2. Fig. 3 exemplifies the geometry of Scenario 3 and the established set up. In Fig. 4 Scenario 3 is displayed. As it can be seen, the size of the trees is relatively small. As for Scenario 4, its geometry and set up are illustrated in Fig. 5. Fig. 6 shows the persimmon trees of Scenario 4. The noise floor in all cases was -80 dBm. Additionally, measurements were taken three times at each point.

## V. EXPERIMENTAL RESULTS

In this section, the accuracy of the chosen prediction models will be verified by comparing them to near-ground measurements. First, the measured data will be examined and then prediction models will be discussed and compared with collated data.

Fig. 7 shows the RSSI levels measured in the chosen scenarios at 2.437 GHz. As the figure shows, the RSSI from Scenario 2 fluctuates much more than the one from Scenario 1. This can be due to the random distribution of vegetation, as well as the presence of trunks. Moreover, the absorption of energy in Scenario 2 may be caused by the presence of grass. In the case of Scenario 3, the RSSI signal is quite stable up until 31 meters and does not reach levels lower than -80 dBm for the first 37 meters. This can be due to the cleanliness of this land as well as the low height of these trees. As this figure shows, the coverage area of Scenario 3 is larger than in

Scenario 2. Finally, the signal levels of Scenario 4 are lower than in Scenario 2, which means that persimmon trees have a greater effect on the strength of the signal than mature orange trees. However, the RSSI signal does not exceed the established noise floor up until 23.5 meters, which is the same that happened in Scenario 2.

Fig. 8 shows the SIR measured in all four scenarios. In this case, we can observe that mature orange trees from Scenario 2 do not introduce interferences up until 15 meters. However, the collected SIR data shows that mature orange trees have little effect on the quality of the signal. In Scenario 3, the interferences appear after 11 meters and its SIR levels fluctuate much more than in Scenario 2 after 19 meters and surpass the 60% after 23.5 meters. This can be caused by the reflections on the ground, as well as the short height of the trees. As for Scenario 4, interferences appear after 6 meters. This may be due to the geometry of the land shown in Figure 5. As the figure shows, the SIR levels rapidly drop. Though the signal levels do not fluctuate as much as in the rest of scenarios, its levels exceed 60% after 23.5 meters, which is the same that happened in Scenario 3, and errors may appear depending on the modulation used.

Fig. 9 shows the RTT measured during the experiment. In this case, the time delays vary far more in Scenario 2 than in Scenario 1. In the case of Scenario 3, time delays are generally lower than in Scenario 2, although the measured RSSI in Scenario 3 is higher than in Scenario 2. As for Scenario 4, persimmon trees seem to have a greater effect on time delays even though RSSI levels are higher than in Scenario 3.
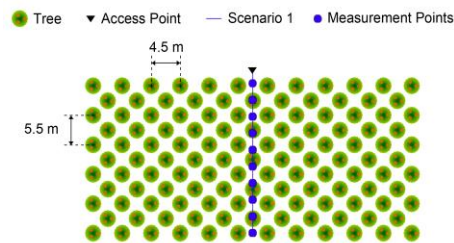


Figure 1. Vegetation geometry and measurement points of Scenario 2.
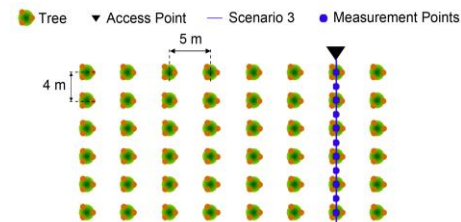


Figure 2. Vegetation geometry and measurement points of Scenario 3.



Figure 3. Scenario 2.



Figure 4. Scenario 3.
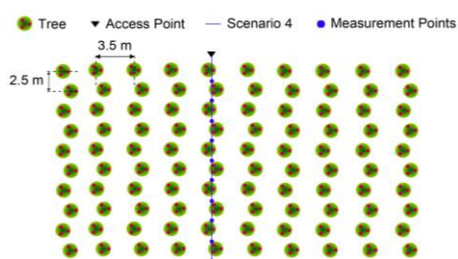
Figure 5. Vegetation geometry and measurement points of Scenario 4.



Figure 6. Scenario 4.
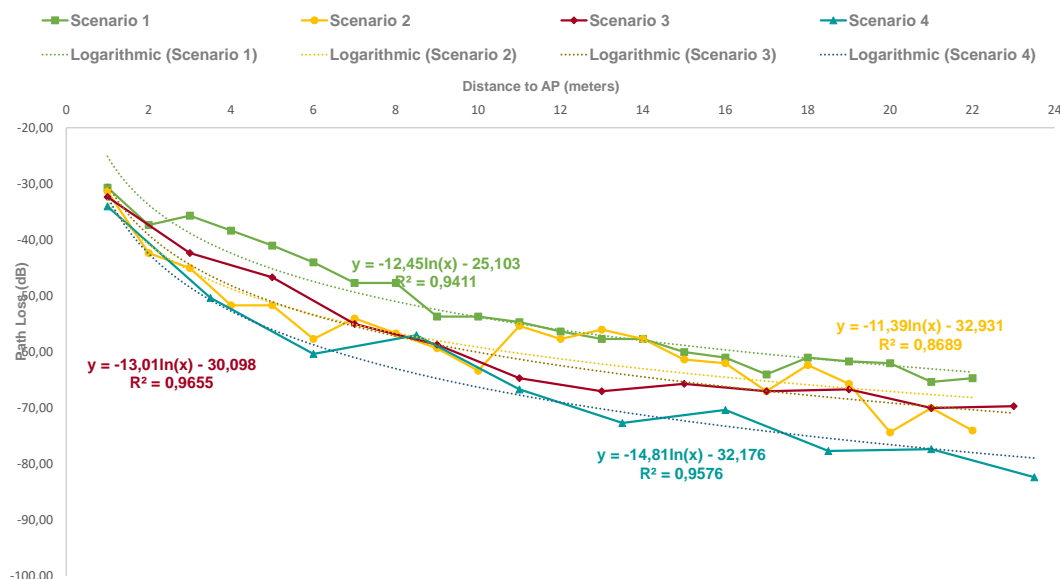


Figure 7. Measured Received Signal Strength Indicator.
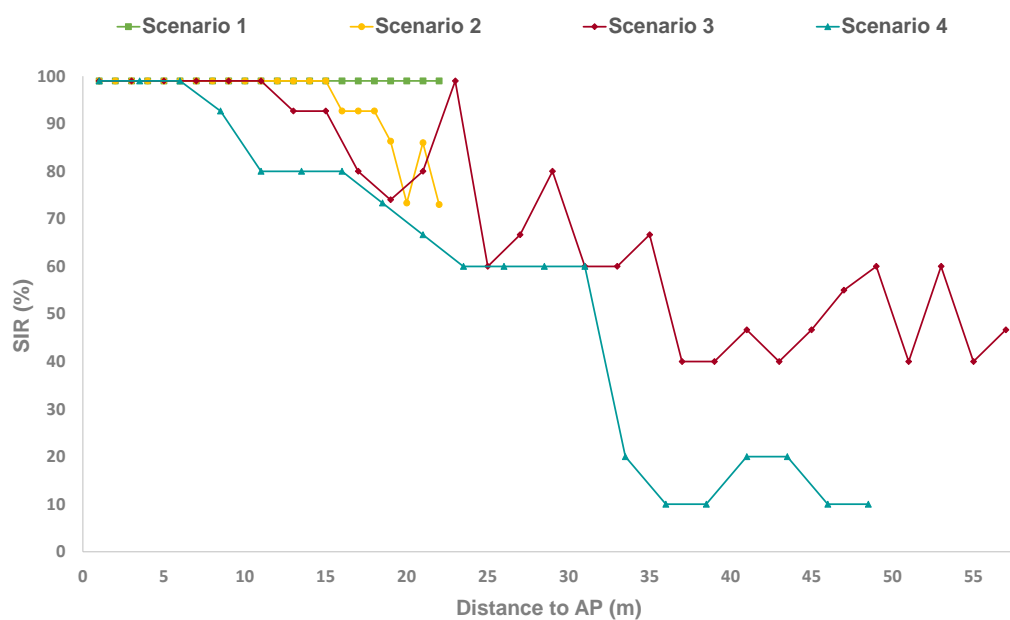


Figure 8. Measured Signal to Interference Ratio.
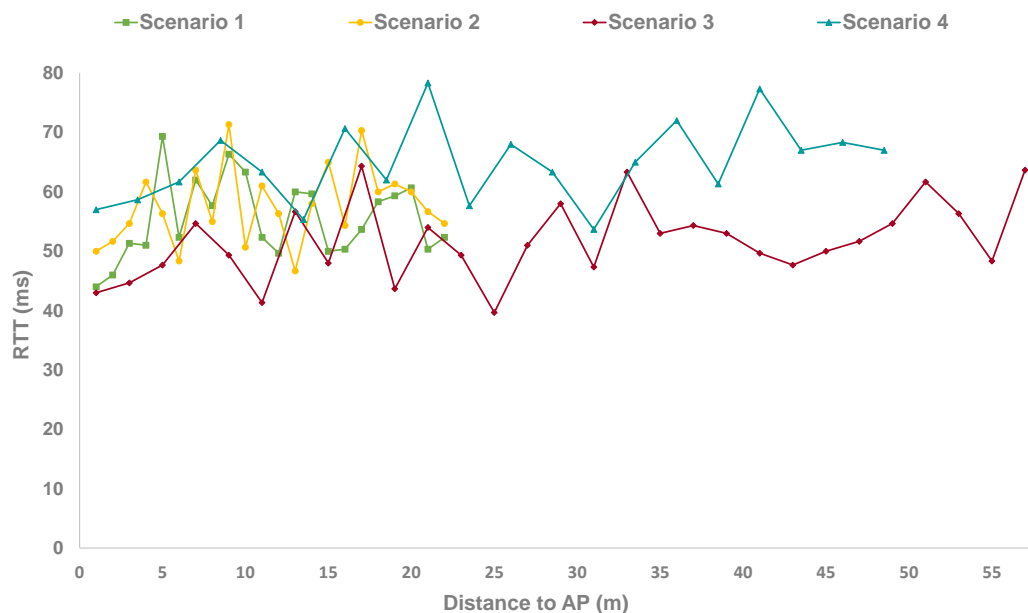
Figure 9. Measured Round Trip Time.
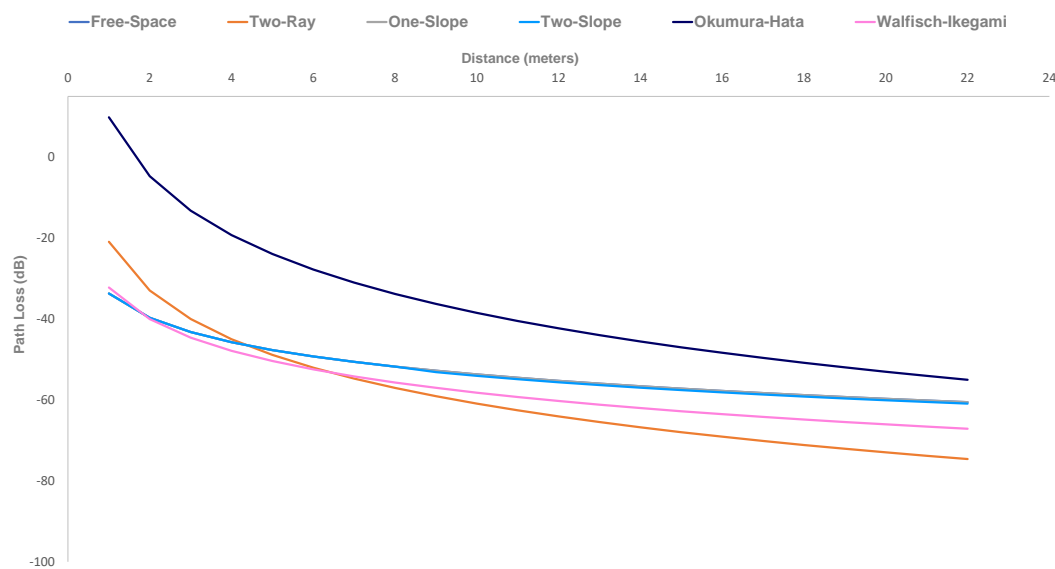


Figure 10a. Comparison of path loss models.
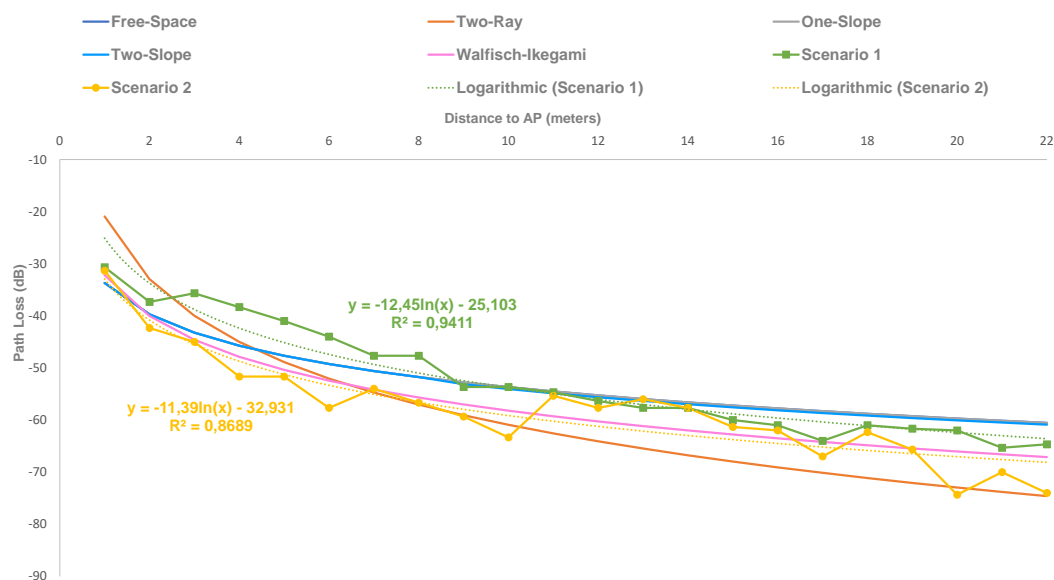
Figure 10b. Comparison of path loss models with collated data from Scenario 1 and Scenario 2.
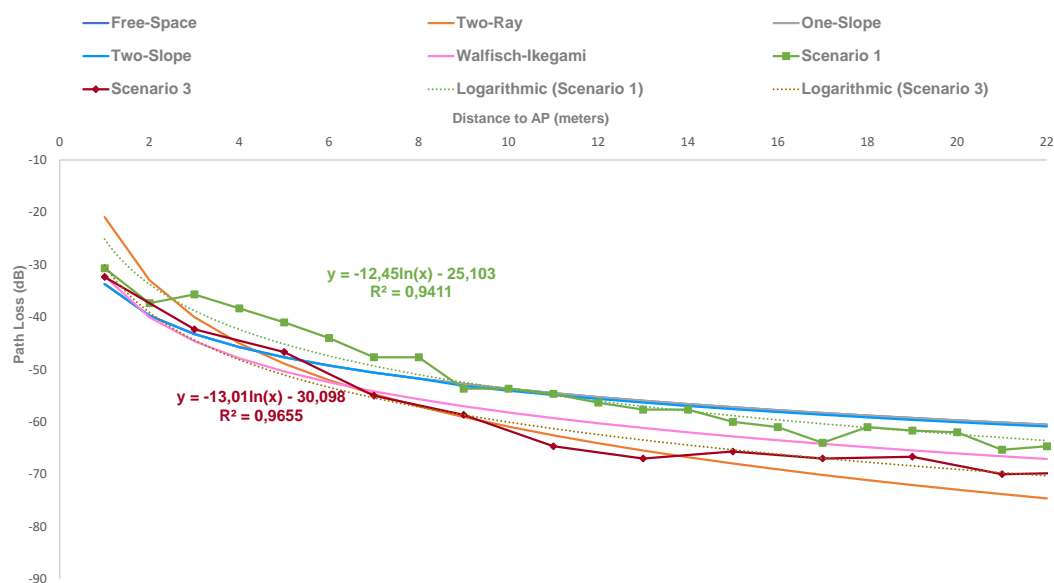


Figure 10c. Comparison of path loss models with collated data from Scenario 1 and Scenario 3.
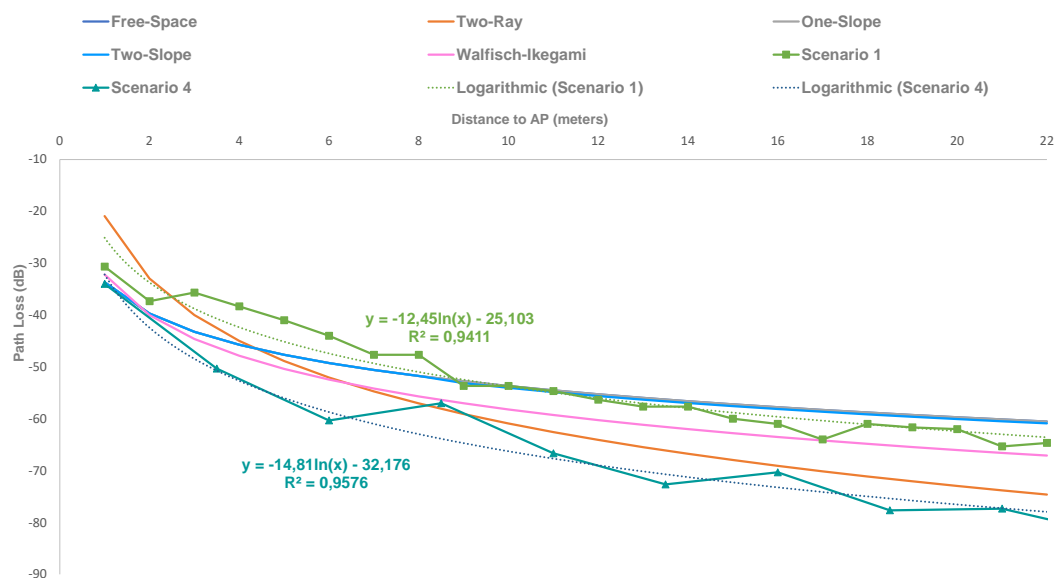
Figure 10d. Comparison of path loss models with collated data from Scenario 1 and Scenario 4.
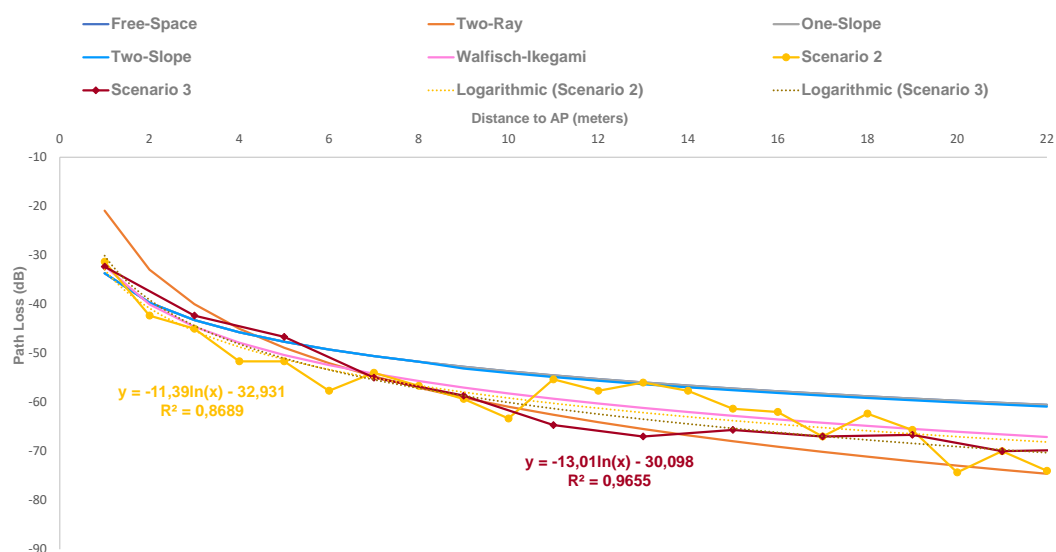


Figure 10e. Comparison of path loss models with collated data from Scenario 2 and Scenario 3.

Figure 10f. Comparison of path loss models with collated data from Scenario 2 and Scenario 4.

Finally, we compared the selected prediction models by plotting them in Fig. 10a. In this figure, Two-Slope Model overlaps both Free-Space Model and One-Slope Model. Attending to the collected data curves from Fig. 7, the path loss is higher in all three scenarios with vegetation than in Scenario 1, which proves that vegetation does in fact introduce attenuations on wireless signal. As Fig. 10a shows, the Okumura-Hata Model failed to predict the attenuation correctly and will be taken out from the graphs for a more fluent reading.

Fig. 10b shows the collected data from Scenario 1 and Scenario 2, together with the selected prediction models. In this case, the Walfisch-Ikegami prediction model is the only path loss model that accurately predicts attenuation for Scenario 1. Furthermore, the collected data from Scenario 1 shows a greater path loss than one the predicted by the Free-Space Model and the Log-Normal Shadowing Models.

In Fig. 10c, the collected data from Scenario 1 and Scenario 3 were plotted, as well as the beforementioned prediction models. However, none of the selected models succeeded foreseeing attenuation for Scenario 3. In this case, the plotted data of this scenario shows a higher attenuation than Walfisch-Ikegami Model, but lower than the path loss predicted by the Two-Ray Model.

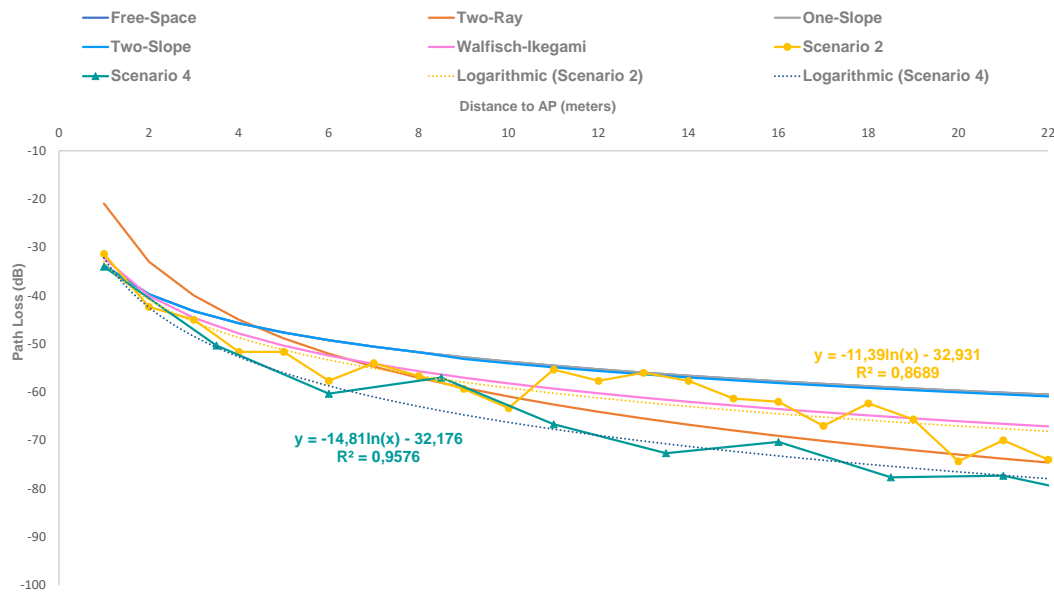In Fig. 10d, the collected data from Scenario 1 and Scenario 4 was plotted together with prediction models. As for the previous case, none of the selected path loss models managed to predict attenuation correctly. Persimmon trees seem to introduce much more losses than what prediction models estimated.

Fig. 10e illustrates the collected data from Scenario 2 and Scenario 3 along with the propagation models. When comparing the trendline of the path losses from both scenarios, one can clearly see that young orange trees introduce higher attenuations on the radio-wave signal than mature orange

trees. The reason for this is that, although young orange trees have lighter foliage, the treetop is in line with both the Tx and the Rx. As for the previous cases, none of the selected propagation models managed to forecast path losses, which highlights the need of finding ways to predict attenuation.

Finally, Fig. 10f shows the collected data from Scenario 2 and Scenario 4 together with the selected prediction models. Although none of the selected prediction models managed to predict attenuation, it can be highlighted that the attenuation introduced by persimmon trees is much higher than the one introduced by mature orange trees.

## VI. COMPARISON WITH PREVIOUS WORKS

In this section, we will discuss previous studies that have attempted to characterize near-ground wireless communications in rural environments in order to establish how frequency ranges and different testbeds may influence the results.

A comparative of similar studies is presented in TABLE I. The research in [11] demonstrated that wireless connections were fairly sensitive to the reflection coefficient in near-ground situations, achieving 101 dB of attenuation in over 14 meters. Moreover, the study performed in [12] verified that theoretical propagation models fail to characterize near-ground wireless communications. In this case, path loss reached 102 dB of attenuation in a radio network of 175 m. In [13], a near-ground WSN with a transmitter placed directly on the ground was presented, showing that prediction models fail to accurately forecast path loss when antenna heights are lower than 50 cm, although in this case, path loss surpassed 100 dB after almost 200 meters. Furthermore, the studies carried in [14-15] to design a WSN based on ZigBee under near-ground conditions showed that grass fields and soil affect signal strength, but still reached 120 meters of distance to the access point with good coverage. In [16], Okumura-Hata,

Log-Normal Shadowing and foliage models were used as a reference to test their performance when comparing the predicted attenuation in tropical environments. However, none of these models managed to accurately predict path loss in tropical environments, which demonstrates the importance of characterizing foliage attenuation in different environments. In this case, path loss exceeded 100 dB after 13 meters. The statistical model described in [17] demonstrated that antenna height determines coverage area and that propagation models fail to accurately forecast path loss. However, path loss only surpassed 100 dB after 100 m. In [18], the presented solution did manage to detect bad leaves. However, in this case, vegetation loss was not introduced into the power balance formula and no information was given about the amount of attenuation. The study performed in [19] concluded that reducing antenna heights increased path loss, though this investigation was carried out in forested areas covered by

snow where attenuation reached 96 dB after almost 40 m. Moreover, the study depicted in [20] demonstrated that ground-loss is a major issue when determining node range. Nevertheless, this experiment was performed in Wi-Fi crowded environments and after 30 m attenuation reached 100 dBm. The experiment performed in [21] to characterize near-ground UWB propagation channels showed that the node range is highly dependent on antenna heights and only reached 75 dB after 200 meters.

As it can be seen on TABLE I, our work expands the knowledge on this area by including the most common prediction models used in near-ground scenarios, and other propagation models that identify reflections, diffractions, and scattering. Furthermore, we attempted at testing these prediction models by measuring at different heights than the ones employed in other existing works.

TABLE I.    COMPARATIVE OF TESTBEDS OF STUDIES ON THE EFFECTS OF VEGETATION ON NEAR-GROUND WIRELESS SIGNALS

| Authors | Year | Frequency | Tx height | Rx height | Models | | | | | Path Loss |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | *Free-Space* | *Two-Ray* | *One-Slope* | *Two-Slope* | *Others* | |
| Torabi et al. [11] | 2015 | 300 MHz, 868 MHz | 13 cm, 0.87 m, 1.15 m, 1.55 m, 2 m | 0.4 – 1.8 m | - | YES | - | - | - | 32 - 101 dB |
| Yildiz et al. [12] | 2014 | 315 MHz, 433 MHz, 868/916 MHz | < λ | < λ | YES | YES | YES | YES | - | 31 – 102 dB |
| Tang et al. [13] | 2019 | 470 MHz | 5 cm, 50 cm, 1 m | 5 cm, 50 cm, 1 m | YES | YES | YES | YES | Walfisch-Ikegami | 57 – 115 dB |
| Klaina et al. [14] | 2018 | 868 MHz, 2.4 GHz, 5.8 GHz | 0.2 m, 0.4 m | 0.2m, 0.4 m | YES | - | - | - | Three-Slope Log-Normal | 0 – 37 dB |
| Klaina et al. [15] | 2017 | 868 MHz, 2.5 GHz, 5.8 GHz | 20 cm, 40 cm | 20 cm, 40 cm | YES | - | - | - | - | 31 – 90 dBm |
| Masadan et al. [16] | 2019 | 915 MHz | 0.65 - 4.5 m | 0.19-1.4 m | YES | - | YES | YES | Okumura-Hata | 83 – 104 dB |
| Wang et al. [17] | 2012 | 2.4 GHz | 3 cm, 1 m | 1 m, 2 m | YES | YES | YES | YES | - | 40 – 109 dB |
| Lloret et al. [18] | 2011 | 2.44 GHz | 6 m | 6 m | YES | - | - | - | - | - |
| Szajna et al. [19] | 2015 | 2.45 GHz | 0 cm, 86.4 cm, 130.8 cm | 0 cm, 86.4 cm, 130.8 cm | - | - | - | YES | - | 60 – 96 dB |
| Luciani et al. [20] | 2013 | 2.48 GHz | 15 cm, 30 cm, 100 cm | 15 cm, 30 cm, 100 cm | - | YES | - | - | - | 60 – 100 dBm |
| Sangodoyin et al. [21] | 2016 | 3–10 GHz | 10 cm, 20 cm, 50 cm, 200 cm | 10 cm, 20 cm, 50 cm, 200 cm | - | - | - | - | Distance & Frequency Dependent Pathloss Models | 20 – 75 dB |
| Our proposal | 2020 | 2.4 GHz | 30 cm | 30 cm | YES | YES | YES | YES | Okumura-Hata Walfisch-Ikegami | 31 - 82 dB |

## VII. CONCLUSION AND FUTURE WORK

In this paper, we attempted to determine how near-ground radio-waves are affected by field components such as, grass, soil, and trunks. To this end, we performed an experiment at four different scenarios: a land with no vegetation, an orange tree plantation with mature trees, an orange tree plantation with young trees, and a persimmon field, where measurements were taken 30 cm above the ground.

In this case study, we analysed the signal quality by measuring the RSSI, the SIR and the RTT of a wireless signal and compared the collated data with six different path loss prediction models. The results showed that, in near-ground scenarios, the RSSI tends to fluctuate much more in the presence of vegetation. In other terms, the geometry of the trees and the presence of grass produce a scattering of energy and a higher number of reflections and refractions. However, the interference was only noticeable at the persimmon field, where noise was introduced from 6 meters. As for the selected prediction models, Walfisch-Ikegami Model managed to accurately predict attenuation for Scenario 1. However, none of the presented path loss models managed to accurately forecast path loss for Scenario 3 and Scenario 4, and Okamura-Hata Model failed capturing the effect of vegetation.

As future work, we would like to include in the experimental test different types of plantations and agriculture environments, such as, vineyards [18]. Moreover, it would be interesting to test other propagation models to verify their accuracy in near-ground scenarios. Another important point for future researches would be introducing simulation models to effectively design and plan wireless networks in near-ground scenarios with vegetation. Additionally, it could be interesting to perform these practical experiments with other technologies such as, LoRa [27], Zigbee and Sigfox which are currently being used in farming activities and compare them with the results of IEEE 802.11 standard.

### ACKNOWLEDGMENT

### REFERENCES

[1] M. Botella-Campos, J. M. Jiménez, S. Sendra, and J. Lloret. "Near-Ground Wireless Coverage Design in Rural Environments". The Fifth International Conference on Advances in Sensors, Actuators, Metering and Sensing (ALLSENSORS 2020) IARIA, March 2020, pp. 14-19, ISSN: 2519-836X, ISBN: 978-1-61208-766-5.

[2] UN DESA | United Nations Department of Economic and Social Affairs (2017), "World population projected to reach 9.8 billion in 2050, and 11.2 billion in 2100" [online] Available at: https://www.un.org/development/desa/en/news/population/world-population-prospects-2017.html (accessed Feb 27, 2020).

[3] H. Bach and W. Mauser, "Sustainable Agriculture and Smart Farming". Earth Observation Open Science and Innovation. ISSI Scientific Report Series, 2018, vol. 15, pp. 261-269.

[4] D. Pivoto, P. D. Waquil, E. Talamini, C. Pauletto, S. Finocchio, V. F. Dalla Corte, and G. de Vargas Mores, "Scientific development of smart farming technologies and their application in Brazil". Information Processing in Agriculture, 2018, vol. 5, no. 1, pp. 21-32.

[5] L. Garcia, L. Parra, J. M. Jimenez, and J. Lloret, " IoT-Based Smart Irrigation Systems: An Overview on the Recent Trends on Sensors and IoT Systems for Irrigation in Precision Agriculture", Sensors (Basel), 2020, vol. 20, no. 4, pp. 1042.

[6] Q. Li and N. Liu, "Monitoring are coverage optimization algorithm based on nodes perceptual mathematical model in wireless sensor networks". Computer Communications, 2019. (In Press). Available at: https://www.sciencedirect.com/science/article/abs/pii/S01403664193151 17 (accessed Feb 27, 2020).

[7] S. Sendra, P. Fernandez, C. Turro, and J. Lloret. "IEEE 802.11a/b/g/n Indoor Coverage and Performance Comparison". In proc. of the 2010 6th International Conference on Wireless and Mobile Communications, Valencia, Spain, Sept. 23-25, 2010.

[8] M. Bacco, P. Barsocchi, E. Ferro, A. Gotta, and M. Ruggeri. "The Digitisation of Agriculture: a Survey of Research Activities on Smart Farming". Array, 2019, vol. 3-4, article 100009.

[9] J. Lindblom, C. Lundström, M. Ljung, and A. Jonsson, "Promoting sustainable intensification in precision agriculture: review of decision support systems development and strategies". Precision Agriculture, 2017, vol. 18, no. 3, pp. 309-331.

[10] Recommendation UIT-R P.833-9. Attenuation in vegetation. Available at: https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.833-9-201609-I!!PDF-E.pdf (accessed August 12, 2020).

[11] A. Torabi and S. A. Zekavat, "A Rigorous Model for Predicting the Path Loss in Near-Ground Wireless Sensor Networks," 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), Boston, MA, September 6–9, 2015, pp. 1-5.

[12] H. U. Yildiz, S. Kurt and B. Tavli, "The Impact of Near-Ground Path Loss Modeling on Wireless Sensor Network Lifetime," 2014 IEEE Military Communications Conference, Baltimore, MD, October 6-8, 2014. pp. 1114-1119.

[13] W. Tang, X. Ma, J. Wei, and Z. Wang, "Measurement and Analysis of Near-Ground Propagation Models under Different Terrains for Wireless Sensor Networks". Sensors, 2019, vol. 19, no. 8, pp.1901.

[14] H. Klaina, A. Vazquez Alejos, O. Aghzout, and F. Falcone. "Narrowband Characterization of Near-Ground Radio Channel for Wireless Sensors Networks at 5G-IoT Bands", Sensors, 2018, vol. 18, no. 8, doi: 10.3390/s18082428

[15] H. Klaina, A. Alejos, O. Aghzout, and F. Falcone. "Characterization of Near-Ground Radio Propagation Channel for Wireless Sensor Network with Application in Smart Agriculture", 4th International Electronic Conference on Sensors and Applications (ECSA-4), Online, Nov. 15-30, 2017.

[16] N. A. B. Masadan, M. H. Habaebi, and S. H. Yusoff. "Long range channel characteristics through foliage", Bulletin of Electrical Engineering and Informatics, 2019, vol. 8, no. 3, pp. 941-950.

[17] D. Wang, L. Song, X. Kong, and Z. Zhang. "Near-Ground Path Loss Measurements and Modeling for Wireless Sensor Networks at 2.4 GHz". International Journal of Distributed Sensor Networks, 2012, vol. 8, no. 8, 969712.

[18] J. Lloret, I. Bosch, S. Sendra, and A. Serrano, "A Wireless Sensor Network for Vineyard Monitoring That Uses Image Processing". Sensors (Basel), 2011, vol. 11, no. 6, pp. 6165-6196.

[19] A. Szajna, M. Athi, A. Rubeck, and S. Zekavat, "2.45 GHz Near Ground Path Loss and Spatial Correlation for Open Indoor and Snowy Terrain," 2015 IEEE 82nd Vehicular Technology Conference (VTC2015-Fall), Boston, MA, September 6–9, 2015. pp. 1-5.

[20] D. P. Luciani and A. Davis, "RSSI based range analysis of near-ground nodes in Wi-Fi crowded environments," 2013 IEEE International Conference on Technologies for Homeland Security (HST), Waltham, MA, November 12-14, 2013, pp. 693-697.

[21] S. Sangodoyin, S. Niranjayan, and A. F. Molisch, "A Measurement-Based Model for Outdoor Near-Ground Ultrawideband Channels," in IEEE Transactions on Antennas and Propagation, 2016. vol. 64, no. 2, pp. 740-751.

[22] S. Sun, T. A. Thomas, T. S. Rappaport, H. Nguyen, I. Z. Kovacs, and I. Rodriguez, "Path Loss, Shadow Fading, and Line-of-Sight Probability Models for 5G Urban Macro-Cellular Scenarios," 2015 IEEE Globecom Workshops (GC Wkshps), San Diego, CA, December 6-10, 2015. pp. 1-7.

[23] A. F. Molisch, "Channel Models," in Wireless Communications, IEEE, 2011, pp. 125-143, doi: 10.1002/9781119992806.ch7.

[24] A. Zreikat and M. Djordjevic. "Performance Analysis of Path Loss Prediction Models in Wireless Mobile Networks in Different Propagation Environments", The 3rd World Congress on Electrical Engineering and Computer Systems and Science (EECSS'17), Rome, Italy, June 5-6, 2017.

[25] Linksys.com (2020). Linksys WRT320N Dual-Band Wireless-N Gigabit Router Frequently Asked Questions. [online] Available at: https://www.linksys.com/us/support-article?articleNum=137128 (accessed February 27, 2020).

[26] Vistumbler.net. (2020). Vistumbler - Open Source WiFi scanner and channel scanner for windows. [online] Available at: https://www.vistumbler.net/ (accessed February 27, 2020).

[27] R. Vega-Rodríguez, S. Sendra, J. Lloret, P. Romero-Díaz and J. L. Garcia-Navas, "Low Cost LoRa based Network for Forest Fire Detection," 2019 Sixth International Conference on Internet of Things: Systems, Management and Security (IOTSMS), October 22-25, 2019, Granada, Spain, 2019, pp. 177-184.

# Performance Comparison of Wireless pH Sensor Modules for Application to Health Monitoring

## Short Paper

Lan Zhang, Jian Lu, and Ryutaro Maeda

Device Technology Research Institute
National Institute of Advanced Industrial Science and Technology (AIST)
Tsukuba, Ibaraki 305-8564, Japan
e-mail: chou-ran@aist.go.jp
e-mail: jian-lu@aist.go.jp
e-mail: maeda-ryutaro@aist.go.jp

*Abstract*— **The paper presents a compact pH sensor system and its extension to health monitoring, particularly for monitoring the urine of infants. Three types of sensing electrodes, namely, plate, comb, and pillar, are fabricated and evaluated to determine the suitable approach for different application requirements. Moreover, to realize a highly compact pH sensor system with a biofriendly interface that offers good user experience, a highly compact transmission module and new sensing materials are introduced to optimize the sensor structure and measurement system. Sensing electrodes fabricated with different materials are characterized comprehensively; two types of wireless transmission boards are proposed and compared. Temperature calibration is adaptably performed for the wider application of the proposed pH sensor system. The results indicate that the performance of the proposed compact pH system with a wireless transmission function is adequate and it can monitor the urine condition of infants in real time. In addition, the proposed pH sensor system can be used for healthcare, well-being, and medical applications.**

*Keywords-pH; field effect transistor; health condition; calibration; radio frequency; Bluetooth.*

## I. INTRODUCTION

Infant diseases, also known as childhood illnesses, can infect children several months after birth. Serious infant diseases include pertussis, Candida albicans infection, and measles, which may lead to death, if not detected in time [1] [2]. Parents are becoming increasingly concerned about their children's health. The detection and prevention of infant diseases is a critical task for doctors as well as researchers. Hence, the related products for infant health monitoring, which are in great demand worldwide, need to be developed as soon as possible [3].

The detection of physicochemical and biological parameters is generally based on three main types of media: blood, saliva, and urine [4]. A typical process for biological detection involves professional staff, such as nurses, who obtain samples from the target person, after which these samples are collected and sent to the designated laboratory for analysis. In general, such tests require several hours to days, according to the procedural difficulty. The real-time monitoring of pathological parameters can efficiently indicate the health condition of infants and provide a reference to doctors for early treatment. However, blood specimen collection is an invasive sampling procedure, and after the sampling process, the target needs to hold a small gauze pad over the puncture site for a few minutes to stop the bleeding. Although saliva sampling is a noninvasive method, it is still challenging because the suitability of the collection method for the analyte of interest has not been sufficiently investigated. Among these physicochemical media, real-time monitoring of urine is the easiest approach. This is because a urine sensor can be placed in the diaper and can monitor the urine condition continuously without affecting the normal life of the infant [5][6].

Of late, there have been several developments in silicon-based microelectromechanical system (MEMS) technology, such as the realization of highly compact sizes and high performances [7][8][9][10]. Using the MEMS process, we have fabricated a highly compact pH sensor that has already been applied not only in chemical engineering but also in other fields such as agriculture and industry [11][12]. In this study, previous as well as latest work on the fabrication and evaluation of the improved pH sensor electrodes and wireless transmission approaches has been summarized. Moreover, extension of the application of the proposed sensor for health monitoring has been discussed. We modify the pH sensor system and casing method and optimize the sensing material of the electrode with a biofriendly surface such that it can be suitable for monitoring the urine of infants in the diaper. The settings of the sensors placed in the diapers of newborns are not well-developed for neonatal health monitoring. For monitoring neonatal conditions, nurses usually need to use traditional sensors (temperature or pH sensors) to measure the urine in the diapers every few hours. The efficiency of this method is very low, and precision is hard to guarantee. The sensor system developed in this paper, using MEMS technology and a wireless transmission method, can greatly improve work efficiency, save time, and enhance measurement precision.

Fig. 1 depicts the wireless pH sensor system and its potential applications as well as the proposed pH sensor electrodes. The sensor system mainly includes the sensing

electrodes and measurement and transmission PCBs (inset images). The electrodes are connected to the gate terminal of a metal-oxide-semiconductor field-effect transistor (MOSFET) on the measurement board for measuring the pH value of the target solution [11]. The electrodes were fabricated using the MEMS technique for realizing high-dimension and performance homogeneity. To determine the appropriate approach for health monitoring, we fabricated and evaluated three types of sensing electrodes, the plate, comb, and pillar type, to detect the urine in the diaper. Different sensing materials such as indium tin oxide (ITO), stannic oxide ($SnO_2$), tantalum pentoxide ($Ta_2O_5$), and antimony (Sb) were evaluated and compared. Two types of wireless transmission boards based on the radio frequency and Bluetooth techniques were proposed and evaluated (Fig. 1, left). Moreover, temperature calibration was adaptably performed for practical application of the proposed pH sensor system. Based on a smart phone Android operating system with an easy-to-use user interface, a barrier-free operating system was developed. The measured data can be obtained through a wireless transmission function. After stability testing through field measurements, several potential applications can be considered (Fig. 1, right). The compact sensing system unit can be placed in a diaper for real-time monitoring of the health of children or elders. Moreover, the compact sensor can be packaged with a biofriendly surface. The ultrasmall low-power wireless sensor node can also be implanted into the body of a pet animal to measure its body fluid condition. The rest of the paper is organized as follows. Section II presents experimental setup and procedure; Section III presents measurement results and Section IV introduces the user interface of the proposed pH sensor based on an android smartphone. Finally, Section V summarizes the paper.
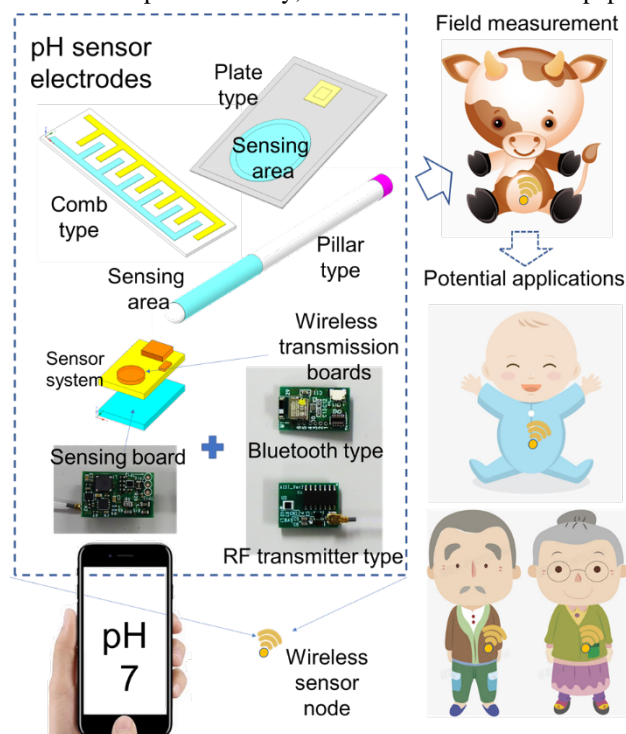


Figure 1. Wireless pH sensor system and its potential applications.
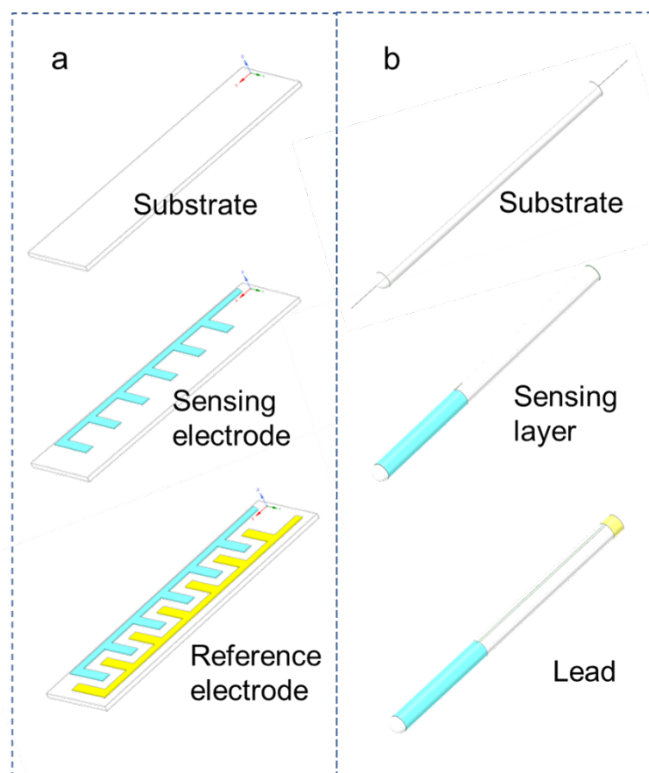


Figure 2. (a) Comb and (b) pillar pH electrode fabrication sequence.

## II. EXPERIMENTAL SETUP AND PROCEDURE

### A. Design and fabrication of the pH-sensing electrodes

We designed, fabricated, and evaluated plate, comb, and pillar type sensing electrodes. The fabrication process of a plate-type sensing electrode has been presented in our previous work [11]. Similar processes can be used to fabricate the comb and pillar structures. The comb electrode was fabricated through a micromachining process using three photomasks. Fig. 2a illustrates the comb electrode fabrication sequence. The device substrate was fabricated by polishing a glass wafer with a thickness and diameter of 200 µm and 100 mm, respectively. A 100-nm sensing material layer was initially deposited on the surface of the glass substrate through a sputtering process. The sensing electrodes were then patterned through photolithography and an etching process. Finally, using the sputtering and liftoff processes, the reference electrode was generated and patterned (see Fig. 2a). Fig. 2b shows the fabrication sequence of the pillar-type electrode. By sputtering with uniform rotation, a 150-nm-thick sensing material layer was deposited on the cylindrical surface of a 5-mm diameter glass pillar. Then, using photolithography, sputtering, and liftoff, the lead and connecting electrodes were generated on the pillar electrode (see Fig. 2b). A highly compact reference electrode (R2K712, Toyorika Co.) was selected for the pH sensor system.

### B. Fabrication of the measurement system

An 18 mm × 12 mm measurement board was fabricated, which included a sensing unit, the related power supply units, and a standard inter-integrated circuit (I²C) interface. In our previous work, the measurement board of the pH sensing system and the test philosophy have been described [11]. The n-type channel of the MOSFET is the sensing unit of the measurement system, which is connected to the sensing electrode for measuring the H⁺ density of the measured solution.

In the power supply units, DC/DC converters and regulators are used for the voltage drain and provide the reference voltage between the reference electrode and ground. I²C (INA 231, Texas Instruments Inc.) has 16-bit analog-to-digital converters for recording the measured analog data.

### C. Design and fabrication of the wireless transmission system

Two types of control and transmission modules, a radio frequency integrated circuit (RFIC) and Bluetooth board, are introduced here. The specification and performance of the RFIC for animal testing are mentioned in our previous work [13]. An RFIC with a low-power type Si4010, which is a fully integrated crystal-less RF transmitter with an embedded microcontroller unit (CIP-51 8051), was selected. The transmission approach included a transmitter but not a transceiver because power consumption is crucial in health monitoring devices. The transmitter method can efficiently reduce the standby current by a factor of 100 [13], and the power consumption of the system can be reduced by several orders correspondingly.

Fig. 3 displays the front and back images of the Bluetooth transmission board as well as its circuit diagram. A four-layer PCB with a thickness of 1.6 mm was employed, and the fabricated Bluetooth transmission board was compact at 19 mm × 12 mm. The supply voltage was designed to be 3 V such that it can be provided by a standard cell battery with a nominal voltage of 3 V. A standard I²C interface was set in connector CN3 to control the measurement board of the sensor system. The largest component on the transmission board was the Bluetooth chip (MDBT42V-512KV2, Raytac Co.), with high-performance and excellent connectivity (Fig. 3a). To realize a compact transmission module, we used the least number of components possible. Typically, a crystal oscillator, three capacitors, and three capacitor networks (CNs) were assembled on the transmission board.

### D. Experimental apparatus

Sputter equipment SME-200E (Ulvac Co.) was used to deposit the material films on the wafer substrate. Mask aligner SUSS MA6/BA6 (SUSS MicroTec Co.) was used to implant the lithograph for generating the designed pattern on the photoresist. Four-point resistance processor Sigma-5 (NPS Inc.) was used to measure the sheet resistance of the fabricated sensing electrode. The topographies of the sensing material layers were measured using an atomic force microscope (AFM) SPA-500 (SII Nanotechnology Inc.). A mixed signal oscilloscope DLM2024 (Yokogawa Electric Co.) was employed to record the flow current on the shunt resistor. A cabinet chamber LHL-114 (ESPEC Corp.) was utilized to provide a varying ambient temperature environment. Digital anatomy 3D printer J750 (Stratasys. Co.) was employed to fabricate the casing of the pH sensor module for field measurement.

## III. MEASUREMENT RESULTS

### A. Configuration comparison of the proposed sensor electrodes

Using the AFM, four-point resistance probe, and optical microscope, the mechanical properties of the fabricated sensing electrodes were investigated and characterized comprehensively. Table 1 shows the property comparisons of the proposed sensing electrodes. The comb- and pillar-sensing electrodes were fabricated using ITO; $Ta_2O_5$ and $SnO_2$ were examined as sensing materials for the plate-type electrode. The sizes of the proposed comb, plate, and pillar sensing electrodes were 15 mm × 4 mm, 12 mm × 9 mm, and Φ5 mm × 30 mm, respectively. Correspondingly, the contact area of these sensing electrodes were 39.0 mm², 28.3 mm², and 471 mm², respectively. As the pillar-type sensing electrode has the largest contact area under a similar electrode size, it exhibits high sensitivity, but function failure can occur due to surface contamination. Moreover, compared to the traditional two-dimensional MEMS technique, in the fabrication process of a pillar electrode, a uniform speed rotating device is required in the sputter equipment to coat the sensing film on the surface with homogeneous thickness.
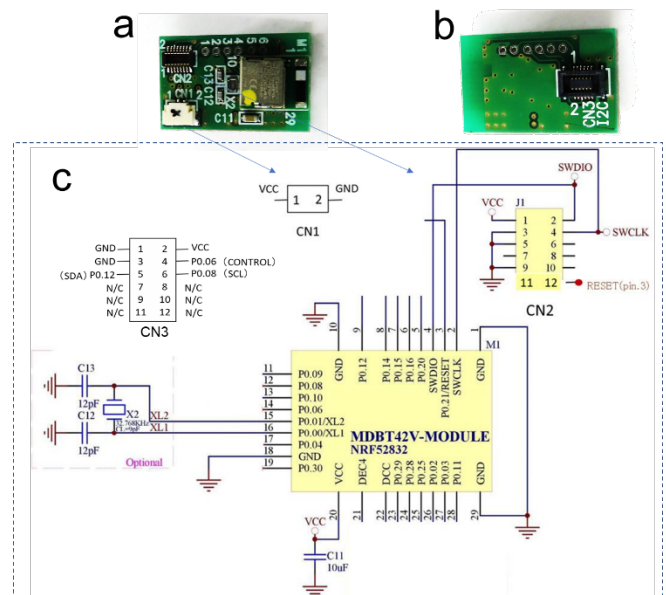


Figure 3. (a) and (b): fabricated Bluetooth transmission board; (c): circuit diagram of the Bluetooth transmission module.

TABLE I. COMPARISON OF THE SENSING ELECTRODES

| Sensor type | Comb | Plate | Pillar |
|---|---|---|---|
| Electrode size | 15 mm × 4 mm | 12 mm × 9 mm | Φ5 mm × 30 mm |
| Fabrication method | Photolithography Sputtering Lift-off/Etching | Photolithography Sputtering Lift-off/Etching | Sputtering (uniform speed rolling device) |
| Contact area | 39.0 mm$^2$ | 28.3 mm$^2$ | 471 mm$^2$ |
| Sheet resistance | 10 Ω/sq (Film on the SiO$_2$ substrate) | SnO$_2$:0.85 Ω/sq Ta$_2$O$_5$:0.91 Ω/sq (Films on the metallic under-electrode) | 10-30 Ω/sq (Films on the glass substrate) |
| Surface homogeneity [14] | 1.1 nm (Ra) | SnO$_2$:1.2 nm (Ra) Ta$_2$O$_5$:2.5 nm (Ra) | Unmeasured |

The surface roughness of the different sensing materials was measured as an arithmetical average value (Ra). For the surface topographies of the fabricated Ta$_2$O$_5$, SnO$_2$, and ITO, it was 2.5 nm, 1.2 nm, and 1.1 nm, respectively. The sensing films on the electrodes retained their smoothness with fewer morphological defects after the MEMS etching process. The fabricated sensing electrode had a uniform surface and was suitable for application in electronic devices. In some cases, sensing films have rough grain surfaces, which can completely obscure the material intrinsic charge transport properties [15].

### B. Comparison of the electrical properties of the pH sensing materials

To characterize the electrical properties of the sensing materials and determine the performances of the fabricated sensor electrodes, we evaluated the output sensitivity of the ITO, SnO$_2$, and Ta$_2$O$_5$-based pH sensor electrodes. Their outputs were linear, proportional to the pH value changes, with sensitivities ranging from 0.29–0.53 mV/pH [16]. Some researchers have proposed Sb-based pH sensors for application in biosensing for better biocompatibility [17]. Therefore, in this study, we also compared the sensitivity of a typical ITO-pH sensing electrode and Sb electrode. The range of the measured pH solution was set to 4.5–7.5 because urine is weakly acidic with a pH range of 4.6–8.0 and an average of 6.0 under normal dietary conditions [18]. Fig. 4 compares the output of a typical ITO-based pH sensor electrode and an Sb-based one. The Sb sensing electrode maintains a linear output proportional to the changes in the pH value. Moreover, its output slope is similar to that of the ITO-based pH sensor; the relative error between the two sensors does not exceed 5 % on average. The minuscule difference between the slopes of the fitting curves shows that

the Sb–pH sensor electrode has a relatively high sensitivity (Fig. 4, fitting curves). Long-term stability (months of testing of deviation and drift) evaluation is in progress and will be reported in future works. After evaluating the electrical properties, we can reasonably determine the advantages and disadvantages of the sensing materials. In addition, the measured sensitivity data can be used to calibrate the proposed sensors, after which the sensor module can be used to perform field measurements to establish its stability and reliability.

### C. Temperature calibration of the sensor system

The developed MEMS sensor will be used in different working conditions, and several environmental factors such as the environmental noise, humidity, and temperature may affect the sensor function. Environmental noise and humidity can be easily removed using packaging with good sealing; the temperature is critical and is generally eliminated through calibration. Therefore, temperature calibration of the proposed compact pH sensor needs to be completed before field measurements. Fig. 5 shows the voltage output of the proposed pH sensor with respect to the temperature variation. A typical ITO sensing electrode was used for demonstration. The measurement and transmission boards were placed in the cabinet chamber, and the temperature range was set from 20–60 °C. We found that the proposed pH sensor system was temperature dependent, i.e., in the measured solution with a stable pH value, the sensor voltage output increased when the environmental temperature decreased. The reason for this is that there are many electrical components, and the working resistances, in particular, are easily influenced by the environmental temperature changes. However, the output voltage deviation has a linear relationship with the temperature change; therefore, we can conclude that the calibration, i.e., output voltage, can be resolved using a first-degree polynomial equation with the temperature variation.
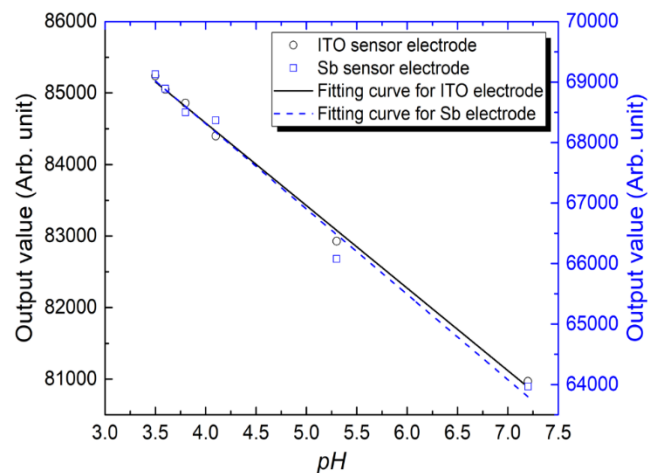


Figure 4. Comparison of output values of typical ITO- and Sb-based pH sensor electrodes.
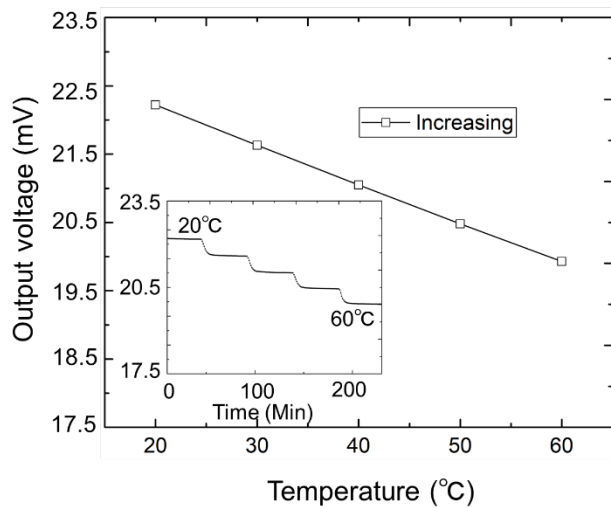
Figure 5. Temperature calibration of the proposed sensor module. Inset is the continuous record of the sensor voltage output with respect to the temperature change.

### D. Power consumption of the measurement systems

The lifetime of the power supply is one of the critical issues for health monitoring sensors. Hence, we measured the power consumption of the proposed transmission approaches. To record the input current flow into the sensor's measurement and transmission circuits, a 1-ohm shunt resistor was connected between the power supply and sensor system. An oscilloscope was used to continuously measure and record the corresponding voltage on the shunt resistor. Fig. 6 compares the input current flow of pH sensors with the RFIC transmitter and Bluetooth transmission system.

The results show that the RFIC transmitter sensor system can implement each data acquisition under a measurement average of 128 times and transmit in 250 ms. The power consumption was calculated to be $7.0 \times 10^{-4}$ mAh for a one-time test. Correspondingly, the Bluetooth device had a capacity charge, stable process before the measurement operation under a 32-times data average, and the power consumption was calculated to be $2.8 \times 10^{-4}$ mAh. To minimize the package size of the sensor system for user comfort, a CR1220 battery has been developed for providing the power supply in future, considering the balance between the sensor system size and battery capacity. The lifetimes of the coin cell battery for the RFIC wireless and Bluetooth sensors can be calculated and are estimated to be more than three months under a 10-min-interval measurement condition, which is a sufficient sampling rate for the real-time monitoring of the target pH value.

### E. Long-term measurement of the sensor output with ambient temperature variation

The proposed compact pH sensor system has several potential applications. Typically, the sensor prototype is placed in a diaper for monitoring the urine conditions of infants. Generally, a diaper will be used and changed within a few hours; thus, a relatively long-term measurement should be verified. Fig. 7 shows the long-term measurement results of the proposed pH sensor output and the environmental ambient temperature. The tested data were collected and recorded by a PC-based receiver for more than two days. The proposed pH sensor could continuously measure a stable solution for 48 h with a reasonable output and maintain a stable baseline. Simultaneously, a digital thermal infrared temperature sensor was loaded on the transmission board; after temperature calibration of the output voltage, the proposed sensor exhibited a highly linear output performance.
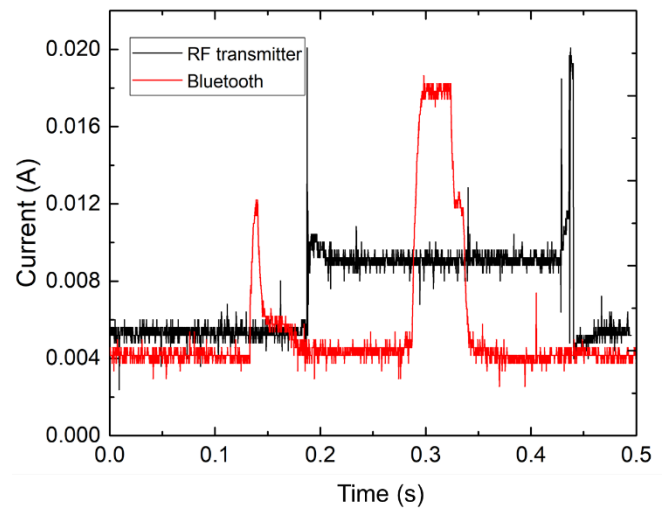


Figure 6. Input current flow comparison between pH sensors with the RFIC wireless and Bluetooth transmission systems.
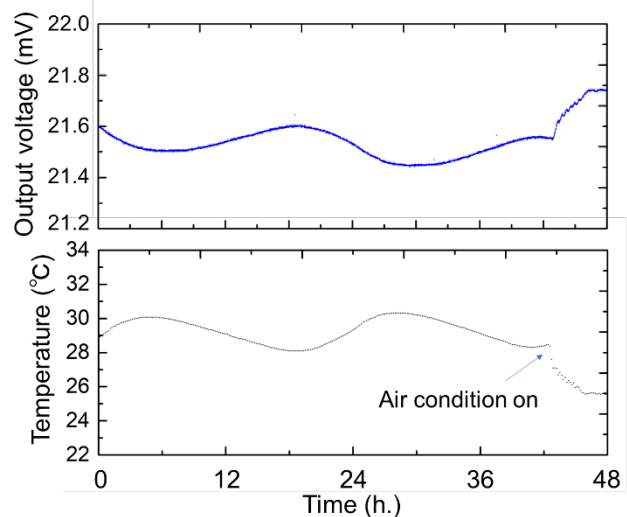


Figure 7. Long-term measurement of the proposed pH sensor voltage output at ambient environmental temperature.

## IV. DEVELOPMENT OF A USER INTERFACE FOR THE PH SENSOR BASED ON AN ANDROID SMARTPHONE

### A. Sensor module casing using a 3D printer for field measurement

A digital anatomy 3D printer (J750) was used for fabricating the sensor module casing using a biofriendly material. Fig. 8 depicts the casing of the pH sensor module for field measurements. The size of the sensor module case was 26 mm × 17 mm × 13 mm (inset in Fig. 8). A CR1220 (Panasonic. Co.) cell battery was placed in the battery case to provide power to the sensor system. Two signal-line wires are connected to the sensing board with the sensor electrodes. During field measurement, the sensor electrodes were placed in a layer of superabsorbent polymers, where the sensor can contact the target solution adaptably (Fig. 8). The plate and comb sensing electrodes are thin and are therefore suitable for this application. A smartphone was used to monitor the pH value via Bluetooth transmission.

### B. Development of the pH sensor user interface based on an Android smartphone

Using the integrated development environment of SEGGER Embedded Studio 4.30c, a customized Android application with a friendly user interface was coded in C-language. The application program, mainly comprising the pH sensing function control and Bluetooth transmission, was successfully developed. Fig. 9 shows the user interface of the pH sensor system; a Pixel 4 smart phone (Google Co.) was used as the terminal device.
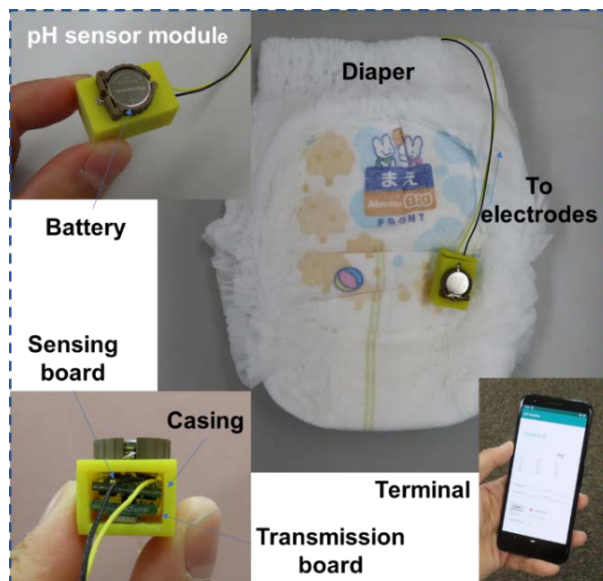


Figure 8. Casing of the pH sensor module and field measurement.
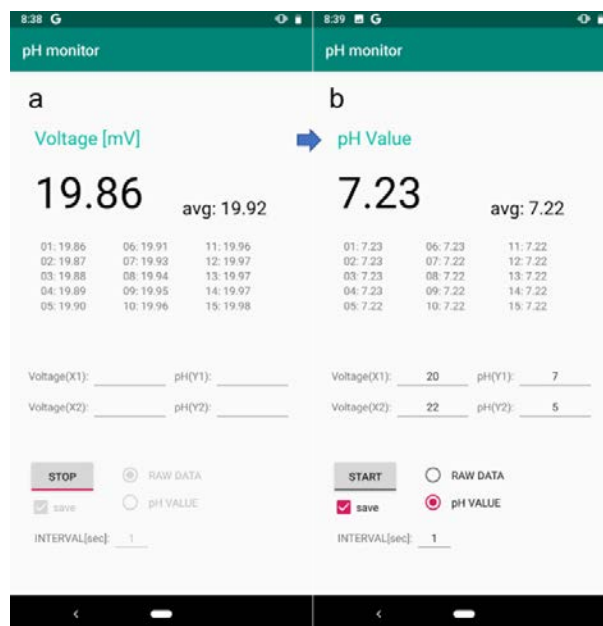


Figure 9. User interface of the wireless pH sensor system.

Fig. 9a displays the calibration page of the pH sensor system, where the primitive raw data is the voltage value of the shunt resistance. Because the calculation formula is embedded in the program, the user can calibrate the pH sensor using two different pH values with a standard reference solution or two liquids with known pH values. The conversion formula between the voltage and pH values is written in the program. Therefore, the pH sensor system can be used to measure the target solution, similar to a commercial pH sensor (Fig. 9b). In addition, the interval time can be adjusted from 1–100 s. The latest 15 data points can be directly observed on the screen, and all the data are saved in the memory with a time stamp.

## V. CONCLUSION AND FUTURE WORK

In this study, a highly compact pH sensor system was developed and the extension of its application for health monitoring was discussed. Three types of sensing electrodes, plate, comb, and pillar, were fabricated and evaluated to determine the appropriate approach for different application requirements. The results indicated that the pillar type, which had the largest contact area, was suitable for complicated testing environments, whereas the comb type, which was the most compact, was suitable for use in limited space. Different sensing materials such as ITO, $SnO_2$, $Ta_2O_5$, and Sb were characterized and evaluated comprehensively. The advantages and disadvantages of these sensing materials were determined to establish their suitability for different application requirements. The $SnO_2$ sensing electrode exhibited the highest sensitivity for pH detection and the least output voltage, reducing the power consumption. Moreover, two types of wireless transmission boards were proposed and compared. Both had low power

consumption, suitable for long-term measurement. The proposed compact pH sensor system with adequate performance can be used in healthcare, well-being, and medical applications. Future study in this area will include further optimization of the sensor system to make it smaller and more comfortable to wear. Empirical and field experiments are also expected to be implemented as soon as possible.

REFERENCES

[1] L. Zhang, J. Lu, and R. Maeda, "Performance comparison of pH sensor module with wireless transmission function," The Fifth International Conference on Advances in Sensors, Actuators, Metering and Sensing (ALLSENSORS 2020), IARIA Press Mar. 2020, pp. 18-19, ISSN: 2519-836X/ ISBN: 978-1-61208-766-5.

[2] Information on diseases & conditions for parents with infants & toddlers (Ages 0-3). [Online]. Available from: http://www.cdc.gov/ 2020.07.25

[3] A. Symon, N. Hassan, H. Rashid, I. Ahmed, and S. Reza, "Design and development of a smart baby monitoring system based on Raspberry Pi and Pi camera," 4th International Conference on Advances in Electrical Engineering (ICAEE 2017), IEEE Press, Sept. 2017, pp. 117-122, doi: 10.1109/ICAEE.2017.8255338.

[4] L. Manjakkal, S. Dervin, and R. Dahiya, "Flexible potentiometric pH sensors for wearable systems," RSC Adv., vol. 10, pp. 8594–8617, Feb. 2020, doi: 10.1039/D0RA00016G.

[5] M. Obatake, T. Muraji, S. Satoh, E. Nishijima, and C. Tsugawa, "Uninary sulfated bile acids: A new simple urine test for cholestasis in infants and children," J. Pediatr. Surg., vol. 37, pp. 1707–1708, Dec. 2002, doi: 10.1053/jpsu.2002.36701.

[6] P. Li, L. Ma, and S. Wong, "Is bag urine culture useful in monitoring urinary tract infection in infants?," J. Paediatr. Child Health, vol. 38, pp. 377–381, Aug. 2002, doi: 10.1046/J.1440-1754.2002.00010. x.

[7] G. Ciuti, L. Ricotti, A. Menciassi, and P. Dario, "MEMS sensor technologies for human centred applications in healthcare, physical activities, safety and environmental sensing: A review on research activities in Italy," Sensors, vol. 15, pp. 6441–6468, Mar. 2015, doi: 10.3390/S150306441.

[8] R. Bogue, "Recent developments in MEMS sensors: A review of applications, markets and technologies," Sens. Rev., vol. 33, pp. 300–304, Sept. 2013, doi: 10.1108/SR-05-2013-678.

[9] X. Hu, Y. Wei, Y. Chen, and C. Liu, "Design of the smart pH sensor based on ion selection electrode," Sensors & Transducers, vol. 26, Special Issue, pp. 92-100, Mar. 2014, ISSN: 2306-8515, e-ISSN 1726-5479.

[10] S. Lozovoy, A. Kukla, and A. Pavluchenko, "Investigation of metrological performance of the ISFET-based pH sensors," Sensors & Transducers, vol. 27, Special Issue, pp. 225-232, May 2014, ISSN: 2306-8515, e-ISSN 1726-5479.

[11] L. Zhang, J. Lu, H. Okada, H. Nogami, T. Itoh, and S. Arai, "Low-power highly sensitive pH sensor with µ-dots protective structures for monitoring rumen in cows in real-time," IEEE Sens. J., vol. 17, pp. 7281–7289, Sept. 2017, doi: 10.1109/JSEN.2017.2757701.

[12] L. Zhang, J. Lu, H. Okada, H. Nogami, and T. Itoh, "Development of ITO- and FET-based cow rumen sensor for long-term pH value monitoring," 2016 Symposium on Design, Test, Integration and Packaging of MEMS/MOEMS (DTIP 2016), IEEE Press, May 2016, pp. 92-96, doi: 10.1109/DTIP.2016.7514846.

[13] J. Lu, L. Zhang, D. Zhang, S. Matsumoto, H. Hiroshima, R. Maeda, M. Sato, A. Toyoda, T. Gotoh, and N. Ohkohchi, "Development of implantablewireless sensor nodes for animal husbandry and medtech innovation," Sensors, vol. 18, pp. 979–1–12, Apr. 2018, doi: 10.3390/s18040979.

[14] L. Zhang, J. Lu, R. Maeda, and H. Nogami, "Research on the performance of solid-State pH sensor affected by the sensing materials," The Fourth International Conference on Advances in Sensors, Actuators, Metering and Sensing (ALLSENSORS 2019), IARIA, Feb. 2019, pp. 1-2, ISSN: 2519-836X, ISBN: 978-1-61208-691-0.

[15] Y. Diao, L. Shaw, Z. Bao, and S. Mannsfeld, "Morphology control strategies for solution-processed organic semiconductor thin films," Energy Environ. Sci., vol. 7, May 2014, pp. 2145–2159, doi: 10.1039/C4EE00688G.

[16] L. Zhang, J. Lu, R. Maeda, and H. Nogami, "Research on the performance of solid-state pH sensor affected by the sensing materials," The Fourth International Conference on Advances in Sensors, Actuators, Metering and Sensing (ALLSENSORS 2019), IARIA, Feb. 2019, pp. 1-2, ISSN: 2519-836X, ISBN: 978-1-61208-691-0.

[17] T. Kazuyoshi, A. Saad Mohammad, K. Daiki, K. Kagemasa, and K. Minoru, " Development of one electrode type pH sensor measuring in microscopic Region," Int. J. Appl. Electrom., vol. 52, pp. 1417–1424, Dec. 2016, doi: 10.3233/JAE-162139.

[18] What is the normal pH range of urine? Available from: http://www.medicalnewstoday.com/ 2020.12.09

# Machine Learning-based Classification and Generation of Vibrotactile Information

Satoshi Saga*, Shotaro Agatsuma†, Simona Vasilache†, and Shin Takahashi†

* Kumamoto University, Kumamoto, JAPAN. 8608555. Email: saga@saga-lab.org

† University of Tsukuba, Tsukuba, JAPAN. 3058573. Email: {agatsuma@iplab, simona@, shin@}cs.tsukuba.ac.jp

*Abstract*—In the field of tactile displays, many researchers are developing systems that employ recorded tactile information as an input signal for tactile display. Various tactile information has been recorded from real textures and presents high-quality tactile sensations via the displays. However, collecting, classifying, and generating large amounts of tactile information data under many different conditions with complicated sensors are difficult to realize. Thus, we developed a method of collecting accelerations in haptic behaviors using wireless microcomputers and implemented a Convolutional Neural Network-based classification method of tactile information. We had succeeded in classifying 30 types of data with an accuracy of 88.9%. Furthermore, we proposed to generate unrecorded data under various conditions from recorded data. We construct a data generation model using a Generative Adversarial Network. The model generates unrecorded three-axis, time-series acceleration data from recorded acceleration data obtained by stroking real objects. To evaluate the quality of the data generated, we presented generated vibrotactile information to users via a tactile display. We revealed that the generated data were indistinguishable from real data. Besides mixing and generating data of two or more classes, we generated new, unrecorded data with mixed features of the original classes.

*Keywords*–*Tactile Information; Machine Learning; Convolutional Neural Network; Generative Adversarial Network.*

## I. INTRODUCTION

Today, many researchers are developing tactile display systems that employ recorded tactile information as an input signal, and these systems present high-quality tactile sensations. To enhance this kind of displaying method, it is necessary to collect and classify various recorded vibration types. Our research proposed a solution to collecting and generating haptic information without complicated devices [1]. In this approach, we collect, classify, and generate only acceleration as tactile information. By using only acceleration data, we collect the information easier than conventional research. Furthermore, employing machine learning-based classification and generation methods, we propose a consistent handling approach of the information for tactile displays.

Many kinds of researches on the collection and classification of recorded tactile information have been performed. To ensure high-quality tactile display, it is necessary to collect and analyze data under various conditions. However, multiple conditions were not addressed in the following works [2], [3]. For example, Strese et al. [3] collected six types of physical data (accelerations, pressures, temperatures, images, sounds, and magnetic field powers) for 108 textures, using

a pen-type device. However, there are many more than 108 textures in the real world, and not all conditions, such as stroking directions, contact angles, or pressure force, were explored. However, most of these researches collect tactile information under limited experimental environments using a device that has many sensors. Therefore, it is difficult to collect haptic information outside of the experiment environment, for example, in daily behavior.

On the other hand, we collect, classify, and generate only acceleration as tactile information. Using a wireless microcomputer with an accelerometer, we collect the information easier than conventional research. To classify haptic information, we implemented a classifier using machine learning. The classifier classifies the collected tactile information. It is also used as a search engine for surface material retrieval of the newly collected information to expand the database. We attached a ZigBee-based wireless microcomputer with an accelerometer to the experimenter's finger or pen and performed stroking of various objects. We collected 30 types of accelerations in stroking haptic behaviors. As a machine learning method, we used the Convolutional Neural Network (CNN) to classify the haptic information with high precision and we succeeded in classifying 30 types of data with an accuracy of 88.9%.

Furthermore, realistic surface reproduction is challenging because touching is bidirectional. If the object surface, physical characteristics, or stroking speed differ between the contactor and the contacted object, the induced phenomena will vary. By recording real texture data in many conditions, such as stroking directions, contact angles, or pressure force, you may prepare several real object data. However, an ideal complete dataset would be unimaginably large. Several researches have used GANs (Generative Adversarial Networks) [4] to generate data for tactile displays to solve this problem. Ujitoko et al. [5] employed a GAN for generating time-series data equivalent to real texture. The model consists of an encoder and a generator, and the encoder transformed texture images into labeled vectors. Then the generator generated spectrograms by using the recoded accelerations and the labels. The spectrograms were transformed into tactile signals for pen-type vibrotactile displays. The model generated nine types of high-quality, one-axis time-series data for simple (i.e., pen-type) vibrotactile displays. However, their proposed system requires high computational cost because the size of the model is large.

Our model is more straightforward than the above model. We generate three-axis acceleration data available for more types of situations (e.g., displaying, analyzing, and recogniz-

ing the vibrotactile signals) than one-axis data. Our method eliminates the need to collect vast vibrotactile signal data from various real objects. Instead of the data collection, unrecorded vibrotactile stimulations are created by employing existing recorded data of real textures. This method reduces the collection cost of real data and greatly expands the utility of limited recorded data for vibrotactile displays. We generated new data with the aid of a GAN. GANs generate images that find many applications in super-resolution [6] and audio synthesis; some sounds are very similar to the human voice [7]. GANs can generate high-quality time-series data. Our data generation model is based on WaveGAN [7], which was developed for audio synthesis. We generated nine types of time-series data based on real textures. By using the tablet-mounted vibrotactile display developed by Saga et al. [8], we performed a user study to evaluate whether the generated stimuli were realistic. Besides, we explored whether it was possible to mix the characteristics of two textures by combining two label data types in the input.

Our principal contribution is that we proposed a solution to collecting and classifying haptic information without complicated devices. In this approach, we collected only acceleration as haptic information. Using a wireless microcomputer that has an accelerometer, we collected haptic information easier than conventional research. Besides, by employing a CNN-based machine learning approach, we realized an accurate classifier for 30 types of textures. Furthermore, we generated unrecorded time-series data using a GAN. Our model has a simpler architecture than an earlier model [5] and requires fewer computational resources. We generate three-axis time-series data to display, analyze, and recognize the vibrotactile signals.

The structure of this paper is as follows. This section describes the purpose of our research and our approach. In Section II, we propose a solution to collecting and classifying haptic information without complicated devices. We propose a generation method of unrecorded time-series data using a GAN and describe our proposed GAN model's system architecture and generated data in Section III. Section IV deals with the user study. Section V describes the user study. Section VI presents a preliminary experiment on multi-label (merged) data generation. Section VII draws conclusions and describes our future work.

## II. Collection and classification of vibrotactile information

In recent years, several methods for classifying tactile information with several sensor inputs are realized by using machine learning technology. We also considered that it is possible to achieve similar classification with fewer sensors while incorporating such human tactile movements. Then we have proposed a tactile information classification system using only acceleration sensors [9]. Usually, texture is evoked by active human tactile movement and friction between texture and finger. In other words, the acceleration sensor attached near the texture captures information, including both his active motion and the vibration generated from the texture. Therefore, we focused on this acceleration and proposed gathering

and classifying the acceleration information collected by the wireless sensors attached near the texture.

### A. Collection of acceleration by a wireless sensor network

We focus on the acceleration between the user and the texture during the stroking operation, construct a wireless sensor network attached to the texture, and collect and classify the induced acceleration information between them. The system consists of the sensor based on a power-saving compact microcomputer with an accelerometer (Figure 1), and a computer enabling signal identification by machine learning.



Figure 1. Overview of ZigBee microcomputer. Left: TWE-Lite-2525A [10]. ZigBee microcomputer: it includes a 3-axes accelerometer, a battery cell, and a communication module. Right: Overview of data collection.



Figure 2. Packet structure: $\Delta t$ shows a time, and $x_i, y_i, z_i$ shows consecutive measured values.

The collected tactile information is transmitted to the computer by ZigBee wireless communication and then classified. ZigBee available in Japan uses the 2.4 GHz band, and at this frequency, a maximum of 250 Kbps, the stable transmission is possible. For this reason, the transfer rate is 144 Kbps, so it is difficult to transmit at a high cycle due to the restrictions of ZigBee communication. Therefore, although the maximum measurement cycle in the initial firmware is 33 Hz, we adopted a method in which ten consecutive measurement values are stored in one packet and transmitted (Figure 2). As a result, we achieved the transmission of the 3-axis acceleration sensor's measured values at 330 Hz on ZigBee communication. By attaching these sensors to a finger or a pen, we record the human's active movement and the vibration caused by friction between the texture and the finger.

### B. Design of classifier by machine learning

In this section, we introduce a method of classifying the acceleration information collected by machine learning. We used a CNN (Convolutional Neural Network), which is widely used in machine learning for images, and constructed a 13-layered network (Figure 3). As an input to this network, we used a 200-point continuous time series with information on three axes of $x$, $y$, and $z$. These sequences were randomly extracted from the collected acceleration information. This division extraction can

increase the learning pattern and improve the generalization performance of the model. The output of this model represents the probability that the input data belongs to each class. We used Tensorflow [11] to build this CNN model. Tensorflow is a machine learning library developed by Google. Besides, the model is based on VGG [12], which is a typical configuration of CNN, and the number of convolutions in the convolution layer is reduced to match our input information. To improve accuracy, we increased the number of convolutional layers. A convolution filter with a size of $1\times5$ was used. The number of filters was 64 for the first and second convolution layers, 128 for the third and fourth layers, and 256 for the fifth and sixth layers. Also, the ReLU function [13] was used as the activation function.

Moreover, generalization performance was improved by doubling the number of filters in the pooling layer. In the pooling layer, max-pooling of $1\times2$ was used to absorb small data errors. The cross-entropy error was used for the loss function. Adam [14] was used as the weight optimization algorithm. To suppress over-learning, Batch Normalization [15] was applied after calculating each layer's activation function. Also, we used Dropout [16] between the fully connected second layer and the output layer to suppress over-fitting.
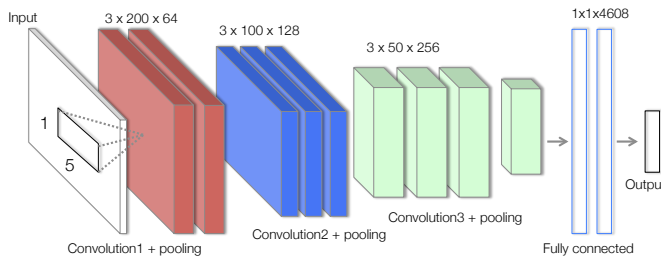


Figure 3. Composition of our CNN model.

### C. The evaluation result of tactile information classification accuracy

Here, we show the results of verifying the classification accuracy with the collected tactile information. The ZigBee microcomputer was attached to a finger or pen, and the experimenter collected tactile information by stroking the surface of objects made of various materials. The experiment conditions are shown in Figure 1. The object used in this experiment is shown in Figure 4.

Each texture to be examined is a planar object about 100 mm in length and width. "Carpet1", "Carpet2", and "Carpet3" are part of carpets made from different materials. "Sponge-g" and "Sponge-y" are the front and the back of a household sponge. "Sponge-b" is made of styrofoam. "Stonetile1", "Stonetile2", and "Stonetile3" are stone tiles made from different materials. "Whitetile1", "Whitetile2", and "Whitetile3" are white tiles with different textures. "Woodtile1", "Woodtile2", and "Woodtile3" are wood plates of different textures. Haptic information is collected by stroking these objects with the finger/pen at an almost constant speed. At the surface of
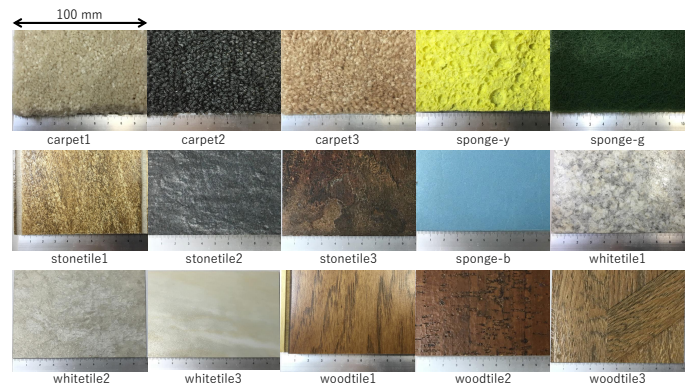


Figure 4. 15 textures. These are plate-shaped objects with 70–100 mm length and 100–130 mm width.

each object, acceleration was collected during the stroking movement of going back and forth for three minutes. By performing this operation three times per object, acceleration data for nine minutes per object was collected. Thus, there are 30 types of combinations between the objects and contactors in the experiment. Moreover, the stroking speed also matters. For each combination, the experimenter stroked at velocities of 100 mm/s, 200 mm/s, and 400 mm/s.

The collected data is classified and evaluated by machine learning using the CNN described above. The input of CNN is $3\times200$ acceleration data, and the output is $1\times30$, which is a probability vector representing the class to which the input data belongs. At the time of classification, to confirm the generalization performance of the model created by this CNN, all data was divided into ten parts and 10-fold cross-validation was performed. Table I shows the confusion matrix at 400 mm/s, and Table II shows the classification accuracy for each stroking speed. As can be seen from the tables, each class diagonal component shows a value close to 1, indicating that 30 types of texture information can be classified with high accuracy. The average of all classes was 93.2%. Besides evaluating the effect of different stroke conditions (stroking speed: 100, 200, and 400 mm/s) on the same texture, we combined them in one class and classified the 30 types of textures. The result shows 88.9% of classification accuracy. From these results, we concluded that there is a possibility that the textures can be classified regardless of the stroking speed.

## III. Generation of unrecorded tactile information

As we have introduced so far, some researchers have used machine learning based on collected data to classify the textures [2], [3], [9]. However, there are enormous amounts of conditions of the combination between textures and stroking motions, and these studies cannot cover all of these combinations. For example, Strese et al. [3] collected data on various conditions with a pen-type device, although the acute angle against the object is fixed. Thus the data that can be

TABLE I. Confusion matrix of 30 kinds of data classification under 400 mm/s movement.

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y | Z | AA | AB | AC | AD |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| A: carpet1-pen | 0.94 | 0 | 0 | 0 | 0.02 | 0 | 0.03 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| B: carpet1 | 0 | 0.92 | 0 | 0 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 |
| C: carpet2-pen | 0 | 0 | 0.92 | 0 | 0.05 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| D: carpet2 | 0 | 0 | 0 | 0.97 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 |
| E: carpet3-pen | 0.02 | 0 | 0.02 | 0 | 0.95 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| F: carpet3 | 0 | 0.04 | 0 | 0.03 | 0 | 0.9 | 0 | 0.01 | 0 | 0.01 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| G: sponge-b-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0.96 | 0 | 0.01 | 0 | 0.01 | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| H: sponge-b | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.92 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.05 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0.01 |
| I: sponge-g-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.98 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| J: sponge-g | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.94 | 0 | 0.03 | 0 | 0 | 0 | 0.01 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| K: sponge-y-pen | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0.03 | 0 | 0.95 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| L: sponge-y | 0 | 0.04 | 0 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.91 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| M: stonetile1-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.98 | 0.01 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| N: stonetile1 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0 | 0.95 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0.01 |
| O: stonetile2-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0.86 | 0.01 | 0.02 | 0 | 0.02 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0.08 | 0 |
| P: stonetile2 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.92 | 0.01 | 0 | 0.03 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.02 | 0 | 0 | 0.07 | 0 |
| Q: stonetile3-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0.9 | 0 | 0.01 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | 0.02 | 0 | 0 | 0.07 | 0 |
| R: stonetile3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | 0.94 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.02 | 0 | 0 | 0 | 0.01 | 0 |
| S: whitetile1-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0.87 | 0.01 | 0 | 0.02 | 0 | 0.03 | 0.04 | 0 | 0 | 0 | 0.01 | 0 |
| T: whitetile1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.03 | 0 | 0 | 0.01 | 0 | 0.85 | 0.01 | 0 | 0 | 0.03 | 0 | 0 | 0 | 0.06 | 0 | 0 |
| U: whitetile2-pen | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.01 | 0 | 0 | 0.97 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 |
| V: whitetile2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0 | 0 | 0 | 0 | 0.95 | 0 | 0.01 | 0 | 0 | 0 | 0.01 | 0 | 0 |
| W: whitetile3-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.99 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 |
| X: whitetile3 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0.02 | 0 | 0 | 0.04 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.89 | 0.01 | 0.02 | 0.01 | 0 | 0 | 0.01 |
| Y: woodtile1-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.97 | 0 | 0 | 0 | 0.01 | 0 |
| Z: woodtile1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0.01 | 0.01 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.94 | 0 | 0 | 0.01 | 0 |
| AA: woodtile2-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.04 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.94 | 0 | 0.01 | 0 |
| AB: woodtile2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0.99 | 0 | 0 |
| AC: woodtile3-pen | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.08 | 0.01 | 0.02 | 0 | 0.03 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.86 | 0 |
| AD: woodtile3 | 0 | 0 | 0 | 0 | 0 | 0 | 0.01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0.02 | 0.03 | 0 | 0 | 0 | 0 | 0 | 0 | 0.94 |

TABLE II. Result summary. "Mixed" is the result of classifying 30 kinds of texture, each of which include data under three different speed.

| Stroking speed (mm/s) | 100 | 200 | 400 | Mixed |
|---|---|---|---|---|
| Classification accuracy (%) | 89.0 | 88.0 | 93.2 | 88.9 |

obtained may change if the angle is changed. In this way, it is unrealistic to comprehensively collect data from an actual object because of the significant number of conditions to consider. The existence of data of conditions that cannot be collected also means that tactile information of objects under those conditions cannot be classified. Therefore, we consider this problem and propose a new method to replace the method of collecting data directly from an object. Instead of an exhaustive direct data collection method, machine learning is used to generate alternative data from the minimum amount of collected data [1]. As a result, the data collection cost can be minimized. Besides, by adjusting the machine learning model, it is possible to generate data that will replace the data that has not been collected. As a first step in realizing the proposed method, we implemented data generation based on machine learning that focuses on the stroking movement's acceleration data.

### A. Tactile data generation by GAN

We used the GAN [4] as a data generation method. GAN is a machine learning method mainly for generating unknown images, but it is being applied to various fields such as image resolution enhancement, image property synthesis, and voice synthesis. In the field of speech synthesis, previous research has succeeded in synthesizing speech that is almost the same as that produced by a human. On the other hand, the number of studies on processing tactile data using GAN is still small. Ujitoko et al. [5] proposed a GAN model that generates vibration data corresponding to a texture image This model consists of an "Encoder" network and a "Generator" network. The Encoder network converts the texture image into label data. In the Generator network, data is generated using GAN, which is trained with label data and acceleration data collected in advance. Using this method, Ujitoko et al. generated data for nine classes of textures.

In order to generate more effective data, we constructed a machine learning model for vibration data generation based on WaveGAN [7], which is a GAN for speech synthesis. However, WaveGAN does not support multi-class generation and is not suitable for generating various types of data. To deal with this problem, we combined the method of Conditional GAN [17]. In Conditional GAN, by adding label data to the training data and making it learn, the data corresponding to the label data can be specified and generated at the time of data generation. We introduced multi-class generation by introducing Conditional GAN into our GAN. As the label data, we used one-hot vectors, a type of vector having the same length as the number of classes to be trained and having only 0 or 1 as elements. Besides, we processed acceleration data on three axes in consideration of expandability. To prevent the axes' features from being convolved during learning, each axis was set to be convolved independently in the time direction during learning.

The structure of the data generation model is shown in Table III. "Generator" and "Discriminator" in the table indicate the neural network layer structure that constitutes the model. "Input" indicates the input layer, and "Output" indicates the output layer. Each layer between the input layer and the output layer is a hidden layer. In the hidden layer shown here, values propagate from the input layer side to the output layer side. "Kernel Size" shows the shape of the kernel for each convolutional layer, and "Output Shape" shows the shape of the output data for each layer.

### B. Evaluation of tactile information generation model

Using the constructed model, data generation, and reproducibility verification experiment of the generated data was performed. The wired accelerometer collected the acceleration data to record higher frequency data (1 kHz) during the stroking movement on each texture instead of using the wireless microcontroller. For training data, we used triaxial

TABLE III. Proposed GAN structure. The left shows the configuration of "Generator" and the right shows the configuration of "Discriminator".

| Generator | Kernel Size | Output Shape |
|---|---|---|
| Input : Uniform(-1,1)+$C$ | | $(n,\quad 100+C)$ |
| Dense | $(100+C,\quad 49152)$ | $(n,\quad 49152)$ |
| Reshape | | $(n,\quad 3, 16, 1024)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 3, 16, 1024)$ |
| Trans Conv2D (Stride = (1, 4)) | (1, 25, 512, 1024) | $(n,\quad 3, 64, 512)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 3, 64, 512)$ |
| Trans Conv2D (Stride = (1, 4)) | (1, 25, 256, 512) | $(n,\quad 3, 256, 256)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 3, 256, 256)$ |
| Trans Conv2D (Stride = (1, 4)) | (1, 25, 128, 256) | $(n,\quad 3, 1024, 128)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 3, 1024, 128)$ |
| Trans Conv2D (Stride = (1, 4)) | (1, 25, 64, 128) | $(n,\quad 3, 4096, 64)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 3, 4096, 64)$ |
| Trans Conv2D (Stride = (1, 4)) | (1, 25, 1, 64) | $(n,\quad 3, 16384, 1)$ |
| Output : Tanh | | $(n,\quad 3, 16384, 1)$ |

| Discriminator | Kernel Size | Output Shape |
|---|---|---|
| Input : Training data or Generated data | | $(n,\quad 3, 16384, 1+C)$ |
| Conv2D (Stride = (1, 4)) | (1, 25, 1+$C$, 64) | $(n,\quad 64, 4096, 64)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 64, 4096, 64)$ |
| Phase Shuffle | | $(n,\quad 64, 4096, 64)$ |
| Conv2D (Stride = (1, 4)) | (1, 25, 64, 128) | $(n,\quad 64, 1024, 128)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 64, 1024, 128)$ |
| Phase Shuffle | | $(n,\quad 64, 1024, 128)$ |
| Conv2D (Stride = (1, 4)) | (1, 25, 128, 256) | $(n,\quad 64, 256, 256)$ |
| Phase Shuffle | | $(n,\quad 64, 256, 256)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 64, 256, 256)$ |
| Conv2D (Stride = (1, 4)) | (1, 25, 256, 512) | $(n,\quad 64, 64, 512)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 64, 64, 512)$ |
| Phase Shuffle | | $(n,\quad 64, 64, 512)$ |
| Conv2D (Stride = (1, 4)) | (1, 25, 512, 1024) | $(n,\quad 3, 16, 1024)$ |
| LeakyReLU ($\alpha = 0.2$) | | $(n,\quad 3, 16, 1024)$ |
| Reshape | | $(n,\quad 49152)$ |
| Output : Dense | (49152, 1) | $(n,\quad 1)$ |



Figure 5. Textures used in the experiment.
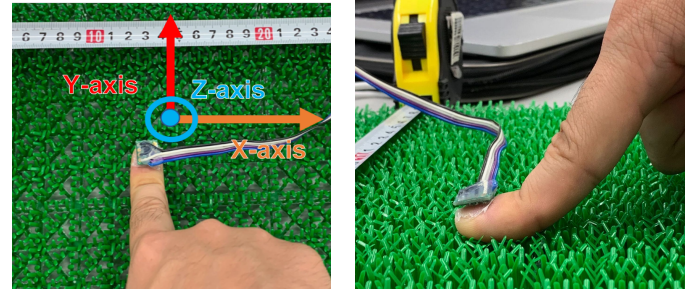


Figure 6. Overview of data acquisition.

acceleration data. An experimenter attached the acceleration sensor to his finger and rubbed the texture in one direction. For recording texture information, we used nine types of textures to collect the data. The texture used for the collection is shown in Figure 5. "Artificial Grass" is the texture of artificial grass with many protrusions, and "Cloth" is the texture of smooth cloth. "Carpet" is the texture of a hard carpet, and "Cork Sheet" is a plate-shaped cork. "Punched Plastic Sheet" is a texture with a lot of punch holes on a smooth board, and "Tile" is a texture of tiles arranged regularly. "Luncheon Mat 01", "Luncheon Mat 02", and "Luncheon Mat 03" are luncheon mats with different surface materials. The overview of the data acquisition is shown in Figure 6.

The hyperparameters used for learning are shown in Table IV. In case the length of the collected data was less than 16,384 points during learning, data was created by repeating each collected data 10 times, and 16,384 points were randomly extracted from the data around 40,000 points and learned. The learning amount was 40 epochs. The generated data is 16,384 points of 3-axis time series data.

TABLE IV. The hyperparameters in our model.

| Name | Value |
|---|---|
| Batch size | 64 |
| Phase Shuffle | 2 |
| Loss | WGAN-GP |
| WGAN-GP $\lambda$ | 10 |
| Generator updates per discriminator | 2 |
| Optimizer | Adam |
| | ($\alpha = 1 \times 10^{-4}, \beta_1 = 0.5, \beta_2 = 0.9$) |

The created model can generate nine classes of texture data. To evaluate the reproducibility of the generated data, we took the spectrogram of the generated data and visualized the data. Short-time Fourier transform (STFT) was performed on the generated data and the training data for spectrogram conversion. At this time, a Hamming window ($N$=256) was used, and the hop size was set to 128. The spectrogram values are normalized from 0 to 1.

An example of the spectrogram of the data generation result is shown in Figure 7. The settings for the spectrogram generation are shown in Table IV. By comparing the spectrograms, you can see the similarity between them. It was difficult to distinguish between training data and generated data for the three textures given in the example. Thus, the data generation that captures the characteristics of the training data was successful. Figures 8 and 9 show the spectrograms of generated data, including the other 6 classes.

## IV. TACTILE DISPLAY EXPERIMENT USING GENERATED DATA

In this section, we describe a tactile presentation experiment using generated data. To evaluate the data generated by our model in more detail, we conducted an experiment to present a tactile sensation to the user using the generated data. In this experiment, data were generated using the WaveGAN-based model described in the previous section. The nine classes of collected data described in the previous section were used as training data for generating the data.

In this experiment, two items are investigated. One is to distinguish between tactile presentation using training data and generated ones. If the participant cannot distinguish the two
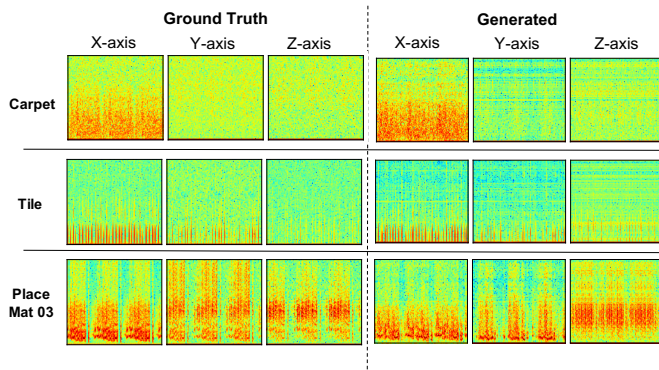
Figure 7. Spectrograms for each labeled class of collected data: Left shows learning data and right shows generated data.
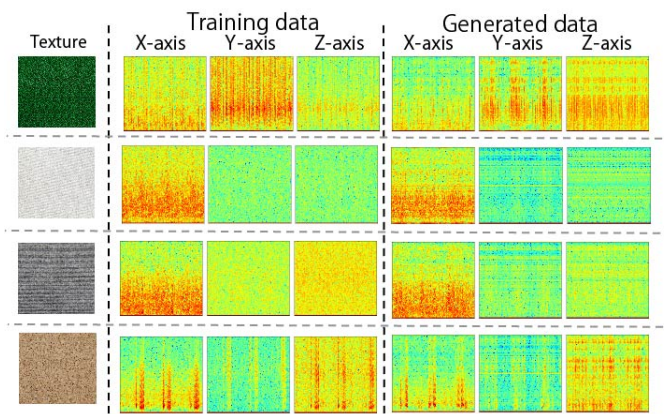


Figure 8. Spectrogram of generated data based on collected data (1). The texture here is the same as the textures shown in Figure 5
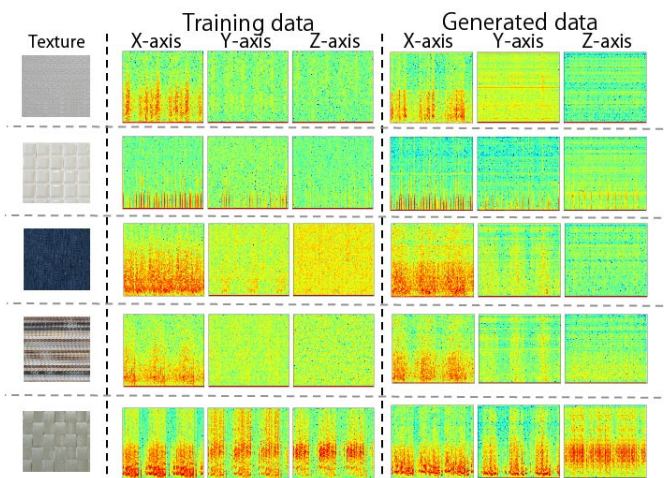


Figure 9. Spectrogram of generated data based on collected data (2). The texture here is the same as the textures shown in Figure 5

data, it indicated that valid data has been generated. The other is a comparison of reality between them. After stroking the actual texture and virtual texture of the two data, we ask the participants to evaluate the reality of the tactile presentation of each data. If the effective data is generated, the evaluation values between the training data and the generated one will show similar values. Based on the results of the experiment, we evaluate whether the model can generate effective data generation or not. For the specific procedure of the evaluation experiment, we decided to use the method of Ujitoko et al. [5]. The tablet-mounted vibrotactile display developed by Saga et al. [8] was used for this tactile presentation.

The participants were ten university students (8 males, 2 females, all in their 20s) in the experiment. The Ethics Review Committee has approved this experiment of the University of Tsukuba (Review approval number, 2019R299). At the beginning of the experiment, the participants filled out a consent form.

### A. Experiment procedure

Tactile stimulation is presented in the two rectangle areas, A and B, on the tactile display, and the participant is asked to stroke the tactile display along the area. A moving target was displayed on the touchscreen and showed a 500 mm/s of stroking movement during the experiment. The participants were asked to adjust their stroking movement to follow the target. Either tactile sensation derived from training data or the generated data is presented in A or B. We ask the participants to answer which stimulation is the tactile presentation derived from the generated data. Figure 10 shows an overview of the experiment. Figure 11 shows the displaying interface in this experiment.



Figure 10. Overview of tactile presentation experiment.

The area the generated data is displayed in was randomly determined for each trial. After the experiment is completed, an evaluation is performed based on the value of a 100 mm long Visual Analog Scale (VAS) [18]. Using the VAS, we collect
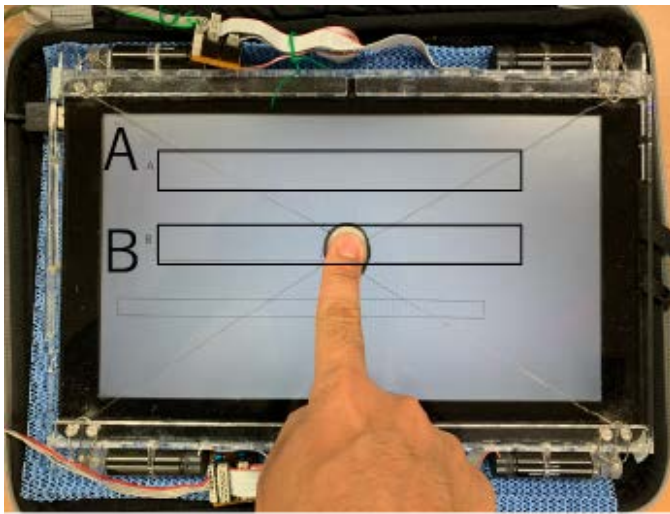
Figure 11. Tactile display during the experiment.



Figure 12. Correct answer rate of distinguishing task between generated data and training data for each texture.

the evaluated answer of each stimulus from the participants. Ten trials were performed for each texture, using these two surveys as one trial. A questionnaire survey was conducted to obtain the participants' opinions about the experiment after completing all trials. It took about one hour to complete the experiment.

### B. Results and discussion

In the following, we describe the experimental results. Figure 12 shows the correct answer rate of the experiment that distinguishes the tactile presentation of the generated data from the training one. These values are the average of the results of all participants. If the result of the correct answer rate is close to 50%, it shows that the participants cannot distinguish between the generated data and the training data. That is, it shows the realization of effective data generation that reflects the characteristics of the training data.

As you can see in Figure 12, the value of any texture is almost 50%. Therefore, the participants could not distinguish between the presentations in the training and the generated data. In the post-experimental questionnaire, almost all participants answered that they could not distinguish between the presentation by the training data and the generated data. From these results, we confirmed the possibility that our proposed model can generate data close to the actual acceleration data. From the detailed result of each texture, most participants showed a correct answer rate of 40% to 60%. Especially for the results of the "Carpet" texture, 7 out of 10 people showed a correct answer rate of 50%. This result indicates that our model may be especially useful for data generation based on rough texture data such as "Carpet" .

Next, Figure 13 shows the results for the presentation's reality values by the generated data and the training data. These values are the average of the results of all participants. The closeness of the evaluation results between reality values of the generated data and the training data shows effective data
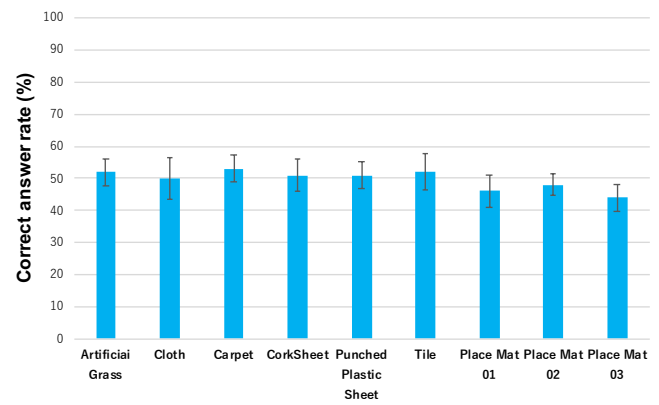
generation achievement. Figure 13 shows that the presentation's reality values by the generated data and the training data are almost the same for all textures. To verify whether there is a significant difference between the results of the generated data and the training data, Student's t-test was performed on the pair of data for each texture, and no significant difference was found for all textures ($p > 0.05$). From this result, by generated data, we succeeded in presenting tactile sensation with the same degree of realism as that by training data. In the previous study of Ujitoko et al. [5], they found a significant difference in some textures. Thus, our method had generated higher quality data than the previous study.

Looking at Figure 13, we obtained values between 50% and 60% for textures other than "Cloth" and "Tile". According to an experiment conducted by Saga et al. [8], the reality values were between 50% and 70% by using the recorded vibration.

Our results agreed with the results of Saga et al. [8], and we found that the generated data could reproduce the performance of the recorded data sufficiently. Here we consider two textures with low reality values. The display we used this time effectively reproduces a rough tactile sensation because it presents vibration to the fingertips but is not suitable for reproducing a smooth tactile sensation like "Cloth". Since "Cloth" has the texture of cloth, the tactile display used this time had difficulty in presenting it. In the future, it is necessary to investigate using a tactile display that excels in smooth tactile sensation.

Regarding the "Tile" results, because the change in acceleration was small, the participants felt little vibration. The reason for the slight difference in acceleration is the shallow unevenness of the Tile. Although the reality value was low in the result of "Tile" this time, we considered improving this value by using more precise tactile displays. Since there was almost no difference in reality between the generated data and the training data, and data close to the training data can be generated for "Tile" (Figure 7), we confirmed that the data generation was successful.
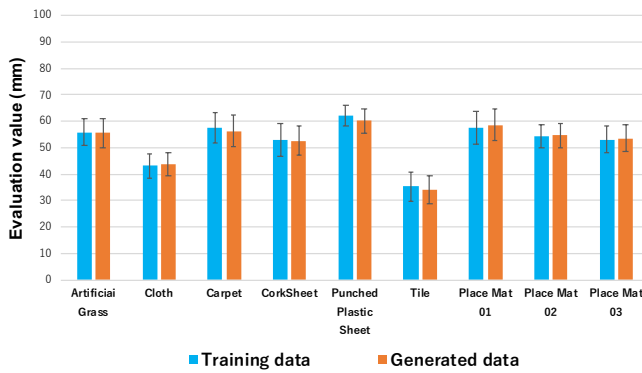
Figure 13. Reality evaluation value for each texture in tactile presentation experiment. The blue graph results from the tactile presentation using the training data, and the orange graph results from the tactile presentation using the generated data.

## V. DATA GENERATION BY MERGING TWO CLASSES

This section confirm the data generation ability of an unknown class by the created 9-class generation model. The unrecorded data was generated by merging two classes and specifying two input labels instead of specifying one class for the created model. Figure 14 shows the schematic diagram of the label synthesis of tactile information. We will look at the result of merging the classes of "TIle" and "Place Mat 03" and specifying them. When "Tile" is specified in the generation model, the elements of "Tile" are specified as 1 in the input vector. For an unrecorded class generation, we set the element of "Tile" between 0.0 and 1.0, and "Place Mat 03" to 1. Figure 15 shows the spectrogram of each generated data. The spectrograms of "Tile" and "Place Mat 03" are shown on the left side of Figure 15, and the spectrogram of generated data with a merged element is shown on the right. It can be seen that as the element is increased from 0.0 to 1.0, the features are highly mixed, especially on the X-axis. In other words, it can be seen that unknown data can be generated by changing the input elements.
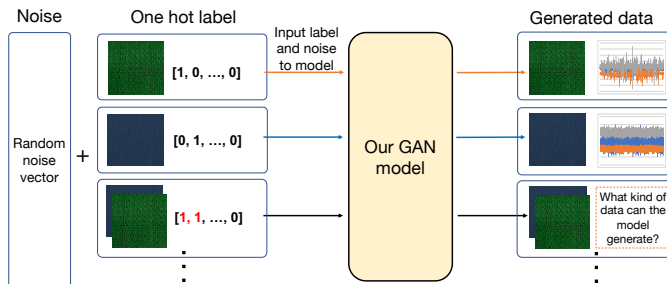


Figure 14. The schematic diagram of the label synthesis of tactile information

In addition, to investigate whether the difference of the texture used for data generation affects the result, we generated
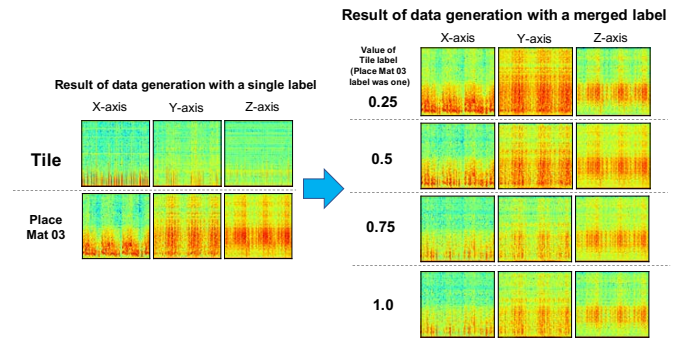


Figure 15. Spectrogram of each generated data. Left shows single class, right shows mixed class spectrograms.

the 3-axis acceleration data obtained from the nine classes of texture shown in Figure 5. This time the scale of the input vector is fixed to 1.0. Thus the element of "Tile" was set in the range between 0.0 to 1.0, and the other element was set in the range between 1.0 to 0.0. The following Figures 16, 17, 18 show the results. Figure 16 shows the results for combinations of "Tile" and "ArtGrass", "Tile" and "Cloth", and "Tile" and "Carpet". Figure 17 shows the results for "Tile" and "Cork", "Tile" and "Punched Plastic Sheet", and "Tile" and "Place Mat 01". Figure 18 shows the results for "Tile" and "Place Mat 02", and "Tile" and "Place Mat 03".
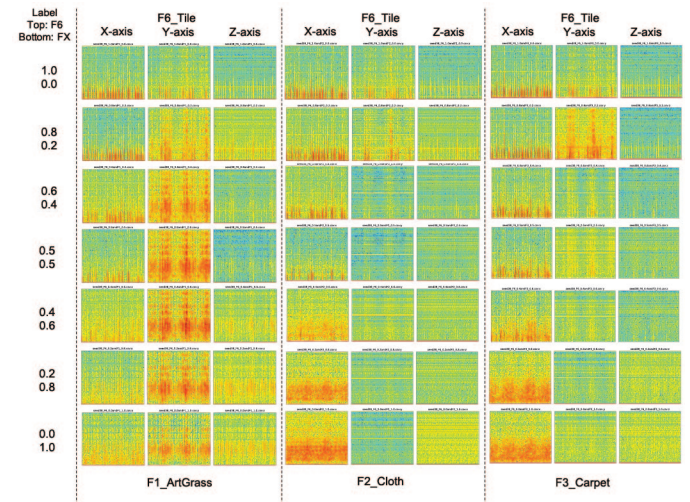


Figure 16. The generated data's spectrograms combined the "Tile" label and the label of another texture (1). This figure shows the results for combinations of "Tile" and "ArtGrass", "Tile" and "Cloth", "Tile" and "Carpet".

In Figures 16, 17, and 18, the upper number shows the element value of Tile, and the number below is the element value of the other texture. From the results, it can be seen that the data that strongly reflects the feature of the texture with the higher label value is generated in any combination. Similar to the result in Figure 15, this tendency is remarkable,
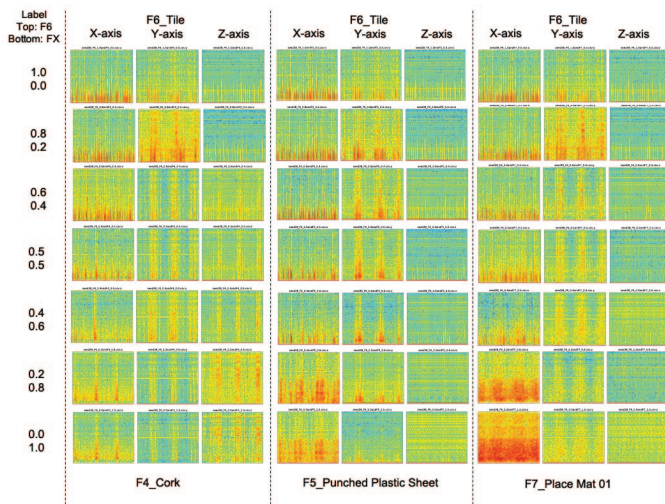
Figure 17. The generated data's spectrograms combined the "Tile" label and the label of another texture (2). This figure shows the results for combinations of "Tile" and "Cork", "Tile" and "Punched Plastic Sheet", "Tile" and "Place Mat 01".
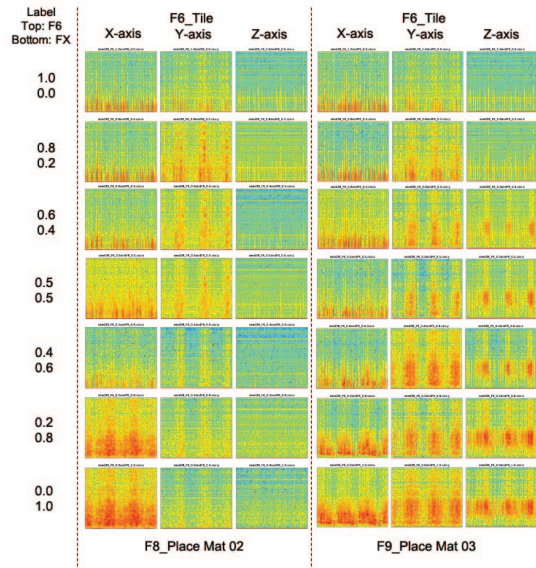


Figure 18. The generated data's spectrograms combined the "Tile" label and the label of another texture (3). This figure shows the results for combinations of "Tile" and "Place Mat 02", "Tile" and "Place Mat 03".

especially on the X-axis. From the result, we found that data synthesis is possible even when the texture is other than Place Mat 03. When the two textures' label values are the same, the "Tile" texture characteristics are often strongly represented in the generated data. The "Tile" texture feature is that the strong and weak regions of the spectrum are finely repeated, but this feature appears in the generated data even when the "Tile" label's value is small. The "Tile" feature appears in the generated data from the case where the "Tile" label element is over 0.4 in all the results except the "Cloth" and "Carpet" results in Figure 16. From this, we found a feature that easily influences the generated data when the labels are combined.

From the result of the generation experiment described in this section, we found that new data was generated with mixed features of two textures by manipulating the label input to the GAN model. We also found that changing the rate of each element can control the interpolated characteristics of each texture. By applying the method, it will be possible to design and generate unrecorded information. For example, by inputting the numerical value of the speed or the pressure, the model can generate the data accordingly. In the future, we plan to conduct a detailed investigation of how the product will be affected when data is generated with variable numerical labels. Based on the results, we will consider constructing a new GAN model that assumes variable numerical labels.

## VI. CONCLUSION AND FUTURE WORK

This paper proposed a solution for collecting and generating haptic information without complicated devices [1]. In this approach, we collect, classify, and generate only acceleration as tactile information. By using only acceleration data, we collect the information easier than in conventional research. Furthermore, employing machine learning-based classification and generation methods, we propose a consistent handling approach of the information for tactile displays. By using the ZigBee-based microcomputers and implementing a CNN-based classification method of haptic information, we succeeded in classifying 30 types of data with an accuracy of about 88.9%.

Furthermore, we developed a method to generate unrecorded data under conditions differing from those at the initial recording time. We constructed a data generation model using a GAN. The model makes simple calculations and generates unknown data from recorded acceleration data obtained by stroking real objects. The model can generate three-axis, time-series data. To evaluate the quality of the data generated, we devised a string-based tactile display and presented generated vibrotactile information to users. Users reported that the generated data were indistinguishable from real data.

Moreover, using GAN, which is based on the method of voice generation, as the method of generating tactile information, we realized effective data generation with a simpler machine learning configuration than previous studies. By creating a vibration data generation model using GAN and generating 3-axis data, we succeeded in generating information close to the actual acceleration sensor's information. To evaluate the quality of the generated data, we devised a string-based

tactile display and presented generated vibrotactile information to participants. The participants reported that the generated data were indistinguishable from real data. Besides mixing and generating data of two or more classes, we generated unrecorded data with mixed features of the original classes.

In the future, we aim to construct a system that enables a lot of tactile information processing by making these collection, classification, and generative models more versatile.

#### References

[1] Shotaro Agatsuma, Junya Kurogi, Satoshi Saga, Simona Vasilache, and Shin Takahashi. Simple Generative Adversarial Networkto Generate Three-axis Time-series Data for Vibrotactile Displays. In *Proceedings of International Conference on Advances in Computer-Human Interactions, ACHI 2020*, pp. 19–24, March 2020.

[2] Arsen Abdulali and Seokhee Jeon. Data-Driven Modeling of Anisotropic Haptic Textures: Data Segmentation and Interpolation. In *Haptics: Perception, Devices, Control, and Applications: 10th International Conference, EuroHaptics 2016, London, UK*, pp. 228–239. Springer International Publishing, 2016.

[3] Matti Strese, Yannik Boeck, and Eckehard Steinbach. Content-based surface material retrieval. In *World Haptics Conference (WHC), 2017 IEEE*, pp. 352–357. IEEE, 2017.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pp. 2672–2680, 2014.

[5] Yusuke Ujitoko and Yuki Ban. Vibrotactile signal generation from texture images or attributes using generative adversarial network. In *International Conference on Human Haptic Sensing and Touch Enabled Computer Applications*, pp. 25–36. Springer, 2018.

[6] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4681–4690, 2017.

[7] Jesse Engel, Kumar Krishna Agrawal, Shuo Chen, Ishaan Gulrajani, Chris Donahue, and Adam Roberts. GANSynth: Adversarial neural audio synthesis. In *International Conference on Learning Representations*, 2019.

[8] Satoshi Saga and Ramesh Raskar. Simultaneous geometry and texture display based on lateral force for touchscreen. In *Proceedings of IEEE World Haptics 2013*, pp. 437–442, Apr. 2013.

[9] Shotaro Agatsuma, Shinji Nakagawa, Tomoyoshi Ono, Satoshi Saga, Simona Vasilache, and Shin Takahashi. Classification method of rubbing haptic information using convolutional neural network. In *Proceedings of International Conference, HCI International 2018*, pp. 159–167, July 2018.

[10] Mono Wireless Inc. TWE-Lite-2525A. https://mono-wireless.com/jp/products/TWE-Lite-2525A/index.html (accessed on 2020.11.20).

[11] Google Inc. Tensorflow. https://tensorflow.org/ (accessed on 2020.11.20).

[12] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*, 2015.

[13] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 315–323, 2011.

[14] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*, 2015.

[15] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning*, pp. 448–456, 2015.

[16] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, Vol. 15, No. 1, pp. 1929–1958, 2014.

[17] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[18] Kathryn A Lee, Gregory Hicks, and German Nino-Murcia. Validity and reliability of a scale to assess fatigue. *Psychiatry Research*, Vol. 36, No. 3, pp. 291–298, 1991.