Methods for Massive, Reliable, and Timely Access for Wireless Internet of Things (IoT)

Kalør, Anders Ellersgaard

DOI (link to publication from Publisher):
10.54337/aau478415180

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

METHODS FOR MASSIVE, RELIABLE, AND TIMELY ACCESS FOR WIRELESS INTERNET OF THINGS (IoT)

BY
ANDERS ELLERSGAARD KALØR

DISSERTATION SUBMITTED 2022

AALBORG UNIVERSITY
DENMARK

Methods for Massive, Reliable, and Timely Access for Wireless Internet of Things (IoT)

Ph.D. Dissertation

Anders Ellersgaard Kalør

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7B
DK-9220 Aalborg

Dissertation submitted:	February 28, 2022
PhD supervisor:	Prof. Petar Popovski Department of Electronic Systems Aalborg University
PhD committee:	Associate Professor Tatiana Kozlova Madsen (chair) Aalborg University, Denmark
	Professor Wolfgang Kellerer Technical University of Munich, Germany
	Professor Roy D. Yates The State University of New Jersey, USA
PhD Series:	Technical Faculty of IT and Design, Aalborg University
Department:	Department of Electronic Systems

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-936-3

Published by:
Aalborg University Press
Kroghstræde 3
DK – 9220 Aalborg Ø
Phone: +45 99407140
aauf@forlag.aau.dk
forlag.aau.dk

Printed in Denmark by Rosendahls, 2022

Abstract

Despite the enormous advances in wireless communication technologies over the past decades, providing efficient wireless connectivity to the Internet of Things (IoT) remains a central challenge towards in future generations of wireless networks. Part of the reason for this can be attributed to the diverse set of requirements and characteristics of IoT applications, which are fundamentally different from the broadband applications that have driven the development of current wireless technologies. As a result, supporting IoT requires new methods and protocols that are designed specifically for IoT. The initial steps towards native support for IoT were taken in 5G with the division between services for ultra-reliable and low latency communications (URLLC), massive machine type communications (mMTC), and enhanced mobile broadband (eMBB). However, this strict division covers only a subset of the characteristics of IoT, and there are still many open problems that need to be addressed before the full range of IoT applications can be supported.

This thesis puts forth a number of methods for providing connectivity for IoT. The methods are designed to address and take advantage of IoT applications with heterogeneous requirements and of correlation between IoT devices. Furthermore, a subset of the methods also assess whether statistical learning and feedback can be used to further enhance the performance. The proposed methods target three central categories of IoT applications. We first consider URLLC, and propose schemes for handling heterogeneous reliability and latency requirements. We also study how machine learning can be used to take advantage of correlated blockages in a millimeter-wave URLLC scenario. Then, we consider applications with a massive number of devices, where we focus on the random access procedure. We present schemes that take into account correlation in the device activations, and also study how to provide feedback to a massive number of devices after a random access procedure. Finally, we study connectivity applications in which the destination node needs to have a timely overview of processes observed by IoT devices. To this end, we use the age of information (AoI) metric and analyze how correlation in the observations impact the AoI. Related to this, we also propose a scheme that implicitly learns the correlation of the observations. The results demonstrate that although designing connectivity solutions for the heterogeneous characteristics of IoT is challenging, taking into account the specific characteristics can lead to significant efficiency gains compared to providing generic connectivity services.

Resumé

På trods af de enorme fremskridt i trådløs kommunikationsteknologi gennem de seneste årtier, er det fortsat en central udfordring at levere effektiv trådløs kommunikation til Internet of Things (IoT) frem mod de fremtidige generationer af trådløst netværk. En den af forklaringen på dette kan tilskrives den brede vifte af krav til og karakteristika ved IoT-applikationer, som er fundamentalt forskellige fra bredbåndsapplikationer, der har drevet udviklingen af de nuværende trådløse teknologier. Som følge heraf kræver understøttelsen af IoT nye kommunikationsmetoder og -protokoller, der er designet specifikt til IoT. De første skridt mod understøttelsen af IoT var taget i 5G, med inddelingen i tjenester til ultra-pålidelig og lav-latens kommunikation (URLLC), massiv maskine-til-maskine kommunikation (mMTC), og forbedret mobilt bredbånd (eMBB). Denne opdeling dækker dog kun en delmængde af karakteristikkerne for IoT, og der er stadig mange åbne problemer, der skal løses, før hele spektret af IoT-applikationer understøttes.

Denne afhandling præsenterer en række metoder til at levere kommunikation til IoT. Metoderne er designet til at adressere og drage fordel af IoT-applikationer med heterogene krav, samt af korrelation mellem IoT-enheder. Ydermere vurderer en delmængde af metoderne også, hvorvidt statistisk læring og feedback kan bruges til at forbedre præstationen yderligere. De foreslåede metoder er rettet mod tre centrale kategorier af IoT-applikationer. Vi undersøger først URLLC og foreslår systemer til at håndtere heterogene pålideligheds- og latenstidskrav. Vi studerer også, hvordan maskinlæring kan bruges til at drage fordel af korrelerede blokeringer i et millimeterbølgebaseret URLLC-scenarie. Derefter undersøger vi applikationer med et enormt antal enheder, hvor vi fokuserer på random access-proceduren. Vi præsenterer systemer, der tager højde for korrelation i enhedsaktiveringerne og studerer også, hvordan man giver feedback til et massivt antal enheder efter en random access-procedure. Til sidst studerer vi kommunikationsapplikationer, hvor destinationen skal have et rettidigt overblik over processer observeret af IoT-enheder. Til dette formål bruger vi Age of Information (AoI)-metrikken og analyserer, hvordan korrelation i observationerne påvirker AoI. I forbindelse med dette foreslår vi også et system, der implicit lærer korrelationen af observationerne. Resultaterne viser, at selvom det er udfordrende at designe kommunikationsløsninger til de heterogene karakteristika ved IoT, kan det føre til betydelige effektivitetsforbedringer at tage højde for de specifikke IoT egenskaber sammenlignet med at levere generiske kommunikationstjenester.

Contents

List of Papers

The following papers constitute the main body of the Ph.D. thesis and are included in Part II:

[A] A. E. Kalør, R. Guillaume, J. J. Nielsen, A. Mueller, and P. Popovski, "Network slicing in industry 4.0 applications: Abstraction methods and end-to-end analysis," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 12, pp. 5419–5427, 2018.

[B] A. E. Kalør, P. Popovski, "Ultra-reliable communication for services with heterogeneous latency requirements," in *2019 IEEE Globecom Workshops (GC Wkshps)*, 2019, pp. 1–6.

[C] A. E. Kalør, O. Simeone, and P. Popovski, "Prediction of mmWave/THz link blockages through meta-learning and recurrent neural networks," *IEEE Wireless Communications Letters*, vol. 10, no. 12, pp. 2815–2819, 2021.

[D] A. E. Kalør, O. A. Hanna, and P. Popovski, "Random access schemes in wireless systems with correlated user activity," in *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2018, pp. 1–5.

[E][1] K. Stern, A. E. Kalør, B. Soret, and P. Popovski, "Massive Random Access with Common Alarm Messages (extended version)," in *Proceedings of 2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 1–5.

[F] A. E. Kalør, R. Kotaba, and P. Popovski, "Common message acknowledgments: massive ARQ protocols for wireless access," submitted to *IEEE Transactions on Communications*, 2022.

[G] A. E. Kalør and P. Popovski, "Minimizing the age of information from sensors with common observations," *IEEE Wireless Communications Letters*, vol. 8, no. 5, pp. 1390–1393, 2019.

[H] A. E. Kalør and P. Popovski, "Timely monitoring of dynamic sources with observations from multiple wireless sensors," submitted to *IEEE/ACM Transactions on Networking*, 2021.

[1] Paper E is an augmented version of the paper accepted for ISIT, which includes all proofs, and which was submitted as supplementary material for ISIT.

In addition to the papers mentioned above, the following papers and patents, which are not included in the thesis, have been submitted before and during the Ph.D. studies:

[1] A. E. Kalør, M. I. Agurto, N. K. Pratas, J. J. Nielsen, and P. Popovski, "Statistical multiplexing of computations in C-RAN with tradeoffs in latency and energy," in *proc. 2017 IEEE International Conference on Communications Workshops (ICC Workshops)*, 2017, pp. 772–777.

[2] P. Danzi, A. E. Kalør, Č. Stefanović, and P. Popovski, "Analysis of the communication traffic for blockchain synchronization of iot devices," in *proc. 2018 IEEE International Conference on Communications (ICC)*, 2018, pp. 1–7.

[3] P. Danzi, A. E. Kalør, Č. Stefanović, and P. Popovski, "Delay and communication tradeoffs for blockchain systems with lightweight iot clients," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2354–2365, 2019.

[4] P. Danzi, A. E. Kalør, Č. Stefanović, and P. Popovski, "Repeat-authenticate scheme for multicasting of blockchain information in IoT systems," in *proc. 2019 IEEE Globecom Workshops (GC Wkshps)*, 2019, pp. 1–7.

[5] P. Danzi, A. E. Kalør, R. B. Sørensen, A. K. Hagelskjær, L. D. Nguyen, Č. Stefanović, and P. Popovski, "Communication aspects of the integration of wireless IoT devices with distributed ledger technology," *IEEE Network*, vol. 34, no. 1, pp. 47–53, 2020.

[6] L. D. Nguyen, A. E. Kalør, I. Leyva-Mayorga, and P. Popovski, "Trusted wireless monitoring based on distributed ledgers over NB-IoT connectivity," *IEEE Communications Magazine*, vol. 58, no. 6, pp. 77–83, 2020.

[7] M. Shehab, A. K. Hagelskjær, A. E. Kalør, P. Popovski, and H. Alves, "Traffic prediction based fast uplink grant for massive IoT," in *proc. 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2020, pp. 1–6.

[8] I. Leyva-Mayorga, F. Chiariotti, Č. Stefanović, A. E. Kalør, and P. Popovski, "Slicing a single wireless collision channel among throughput- and timeliness-sensitive services," in *proc. 2021 IEEE International Conference on Communications (ICC)*, 2021, pp. 1–6.

[9] J. Holm, A. E. Kalør, F. Chiariotti, B. Soret, S. K. Jensen, T. B. Pedersen, and P. Popovski, "Freshness on demand: Optimizing age of information for the query process," in *proc. 2021 IEEE International Conference on Communications (ICC)*, 2021, pp. 1–6.

[10] F. Chiariotti, I. Leyva-Mayorga, Č. Stefanović, A. E. Kalør, and P. Popovski, "Spectrum slicing for multiple access channels with heterogeneous services," *Entropy*, no. 6, 2021.

[11] C. Zheng, M. Egan, L. Clavier, A. E. Kalør, and P. Popovski, "Stochastic resource optimization of random access for transmitters with correlated activation," *IEEE Communications Letters*, vol. 25, no. 9, pp. 3055–3059, 2021.

[12] I. Nikoloska, J. Holm, A. E. Kalør, P. Popovski, and N. Zlatanov, "Inference over wireless iot links with importance-filtered updates," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1089–1098, 2021.

[13] A. E. Kalør, D. Michelsanti, F. Chiariotti, Z. H. Tan, and P. Popovski, "Remote anomaly detection in industry 4.0 using resource-constrained devices," in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2021, pp. 251–255.

[14] R. Kotaba, A. E. Kalør, P. Popovski, I. Leyva-Mayorga, B. Soret, M. Guillaud, and L. G. Ordóñez, "How to identify and authenticate users in massive unsourced random access," *IEEE Communications Letters*, vol. 25, no. 12, pp. 3795–3799, 2021.

[15] ——, "Unsourced random access with authentication and joint downlink acknowledgements," in *2021 55th Asilomar Conference on Signals, Systems, and Computers*, 2021, accepted for publication.

[16] C. Zheng, M. Egan, L. Clavier, A. E. Kalør, and P. Popovski, "Stochastic resource allocation for outage minimization in random access with correlated activation," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, 2022, accepted for publication.

[17] F. Chiariotti, I. Leyva-Mayorga, Č. Stefanović, A. E. Kalør, and P. Popovski, "RAN slicing performance trade-offs: Timing versus throughput requirements," submitted to *IEEE Open Journal of the Communications Society*, 2021. Preprint: `https://arxiv.org/abs/2103.04092`.

[18] P. Popovski, F. Chiariotti, K. Huang, A. E. Kalør, M. Kountouris, N. Pappas, and B. Soret, "A perspective on time towards wireless 6G," submitted to *Proceedings of the IEEE*, 2021. Preprint: `https://arxiv.org/abs/2106.04314`.

[19] P. Popovski, F. Chiariotti, V. Croisfelt, A. E. Kalør, I. Leyva-Mayorga, L. Marchegiani, S. R. Pandey, and B. Soret, "Internet of things (IoT)

connectivity in 6G: An interplay of time, space, intelligence, and value," 2021. Preprint: https://arxiv.org/abs/2106.04314.

[20] F. Chiariotti, J. Holm, A. E. Kalør, B. Soret, S. K. Jensen, T. B. Pedersen, and P. Popovski, "Query age of information: Freshness in pull-based communication," *IEEE Transactions on Communications*, 2022, accepted for publication. Preprint: https://arxiv.org/abs/2105.06845.

[21] P. Popovski, A. E. Kalør, B. Soret, A. K. Hagelskjær, R. Kotaba, and I. Leyva-Mayorga, "Joint data source identification and authentication," European patent, submitted 2021.

Acknowledgment

When I started my PhD it was with the hope that it would help me to better understand the theory behind communications and its relation to practical systems, to learn the role of research in engineering disciplines, and to develop my own creativity within the field of communication engineering. Now, several years later, I have advanced much more than I had ever imagined I would.

No small measure of this achievement is due to my supervisor Petar, who, in addition to introducing me to the joys and pitfalls of research, provided me with a lot of freedom and involved me in numerous projects and works that were well beyond the scope of my own PhD project. I would like to also express my gratitude to my current and former colleagues in the Connectivity Section, especially those that I have been fortunate to share office with. A special thanks also goes to my co-authors and collaborators (more than forty so far), who have been critical to my ideas and provided countless of inspiring ideas. I would like to thank Osvaldo Simeone and his group for hosting me at King's College London. I am grateful for Osvaldo's immense patience and his detailed comments and suggestions. A special acknowledgment also goes to my family and friends for the many times they have pulled me out of my rabbit hole.

Finally, I would like to thank the members of my assessment committee for taking their time to evaluate my work, the various sources that have funded my research in the past years, and the Independent Research Fund Denmark (IRFD) for providing me with the opportunity to continue my research for another two years.

Anders E. Kalør
Aalborg University, February 28, 2022

Part I

Introductory Chapters

1 Introduction

1.1 Motivation

Providing wireless connectivity to machines is a vital step towards ubiquitous connectivity, where everyone and everything is connected everywhere and at any time. Not only does this enable long-promised technologies such as the Internet of Things (IoT) and self-driving cars, it also provides opportunities for new vertical sectors, such as industrial manufacturing, to leverage wireless technologies and replace wired network infrastructures with more flexible alternatives [1]. However, the characteristics of IoT are fundamentally different from those of traditional mobile broadband that the majority of the current systems have been designed to support. As a result, many of the techniques that have revolutionized wireless communication throughout the past decades need to be redesigned with the characteristics of IoT in mind, since scaling the current systems by increasing the bandwidth, adding more antennas, etc. will not suffice to support these emerging applications. The first steps towards this happened with 5G with the introduction of massive machine type communications (mMTC) and ultra-reliable low latency communications (URLLC) as new target use cases in addition to the traditional mobile broadband. However, there are still many open problems that need to be solved before the promise of ubiquitous connectivity can be realized [2].

Some of the exemplary characteristics of IoT include small packet sizes, strict latency and/or reliability requirements and a massive number of devices. A given IoT application may not share all these characteristics, but only a subset of them. In fact, many of the characteristics are conflicting from a communication theoretic perspective in the sense that some combinations of the requirements are hard to support in practice due to the constraints imposed by the wireless channel. The quintessential example of this is the one of providing both very low latency and ultra high reliability, as targeted by the URLLC use case. Both low latency and ultra high reliability can easily be achieved separately, but achieving both at the same time comes at the cost of a very large penalty in the data rate. Similarly, serving a massive number of uncoordinated devices, as targeted by mMTC, while providing low latency, high reliability or high data rate is also hard. The basic trade-offs are illustrated in Fig. 1.1 for mMTC and URLLC, as well as for mobile broadband for comparison.

Another characteristic of IoT is some degree of correlation and predictability in the transmitted data or in the activation patterns introduced by the applications and environment. Although this characteristic does not relate to a

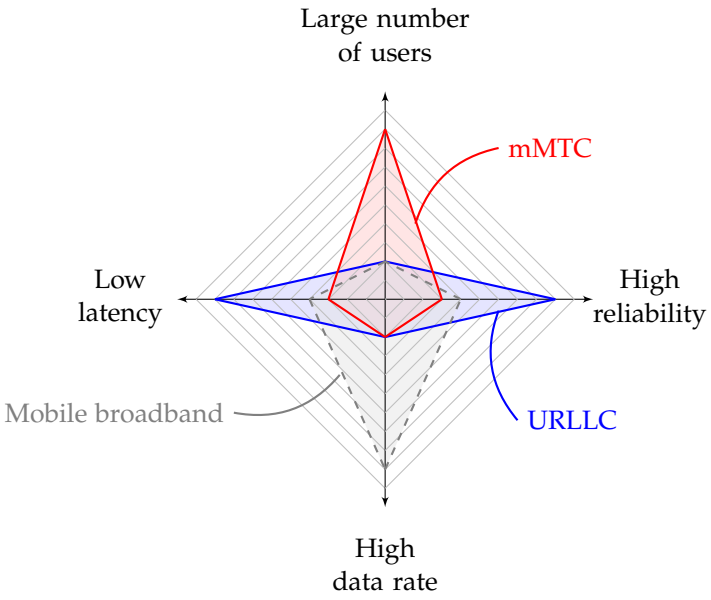


Fig. 1.1: The trade-offs between latency, reliability, massiveness and data rate for mMTC, URLLC, and mobile broadband.

specific service requirement, it is crucial to transform the other requirements into operational design choices. As an example, correlation in the data may happen when a set of devices report various measurements of slowly varying processes, such as temperature and humidity, to a destination node. In this case, a reasonable objective is to ensure that the destination node has a *timely* picture of the processes at any time. This notion of timeliness is different from the traditional latency or delay metrics that are used to characterize communication systems, which measure delay on a per-packet basis, independently of the underlying process and the inter-packet delay. This has led to the introduction of age of information (AoI) [3] as a generic metric of the age of the information at the destination node, which has recently sparked a large number of works.

To illustrate how correlation can also impact the activation pattern, consider a group of devices that can be in either a "regular" state or in an "alarm" state, triggered for instance by some external event. In the regular state, the devices transmit uncorrelated packets at a regular basis (e.g., measurements). However, upon a certain event, many or all devices transition into the alarm state, in which they all try to transmit the same message at the same time. Such an event can happen for instance in power outages, and has been observed in IoT street lights that are triggered to transmit when they are turned on/off [4]. Supporting the large number of activations that

happen under such an event is challenging, and designing a communication system to support the "peaks" in the activation pattern requires significant over-provisioning of the wireless resources. However, since the devices are triggered by the same event, most of the transmissions are unlikely to provide additional information for the receiver, and it might be better to reduce the number of transmitting devices. In an even more extreme case, the fact that the devices transmit the same message at the same time may even be exploited to increase the reliability by combining the signals transmitted by all devices [5].

The examples illustrated above suggest that providing massive, reliable and timely connectivity to applications with diverse characteristics, such as correlation and heterogeneous latency and reliability requirements, represents a central challenge in supporting IoT. However, targeting these application-specific characteristics comes at the cost of extreme complexity, since it is infeasible to design dedicated systems for each application. Nevertheless, they reveal the need for characterizing and assessing the potential gains of such systems, so that informed decisions can be made about how communication systems for IoT should be designed.

1.2 Thesis Objectives and Methodology

The aim of this thesis is to characterize and address a small subset of the challenges discussed in the previous section. To be concrete, it aims to provide partial answers to the following three research questions:

Q1: How can a communication system be designed to support and take advantage of the diverse and heterogeneous timing requirements for IoT?

Q2: How can the correlation among IoT devices be exploited to improve the efficiency of the communication systems?

Q3: How can learning and feedback be used to support massive, reliable, and timely connectivity?

The research questions will be addressed through a number of scientific papers that propose and analyze algorithms and communication schemes for the mentioned scenarios. When beneficial, the analysis will be supported by numerical simulation results. Because IoT spans a wide range of applications with diverse requirements, each research question will be studied in the context of certain IoT use cases, which are considered relevant for the given question (see Fig. 1.2 for an overview).

The first research question, Q1, will primarily be studied from the perspective of supporting URLLC, since the latency and reliability requirements of URLLC necessitate that the system can provide strict guarantees to each

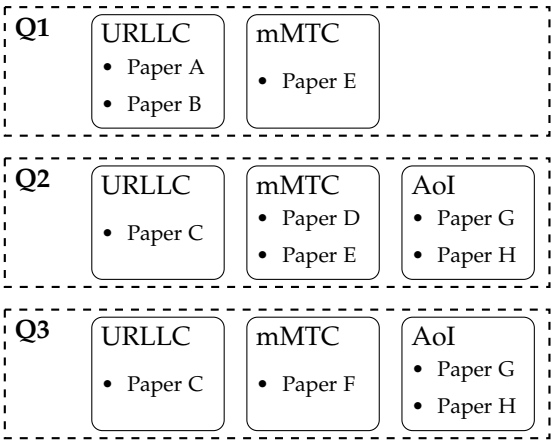


Fig. 1.2: Relation between research questions, scenarios, and papers included in the thesis.

service that it supports. Furthermore, we will consider some aspects of Q1 in context of mMTC. The second research question, Q2, will be studied both for URLLC, mMTC, and AoI with various types of correlation. For URLLC, we will study how correlations in signal blockage events caused by moving objects can be used to increase reliability. In mMTC, we will study how correlated activations influence random access mechanisms, and how knowledge of the correlations can be used to increase the transmission success probability. Finally, we will also study the impact of data correlation on the AoI, and propose scheduling mechanisms to minimize the AoI when the observations are correlated.

The last research question, Q3, concerns the design and use of reliability-enhancing mechanisms that can operate "on top" of or in parallel with the schemes studied in Q1 and Q2. To limit the scope, we restrict the research question to (statistical/machine) learning and feedback, which represent mechanisms and technologies that have been shown to be beneficial for communication systems. Q3 will be studied for URLLC in the context of learning-aided blockage prediction, for mMTC in the context of providing feedback for massive random access, and for AoI where we will learn scheduling policies that take into account correlation between devices.

1.3 Thesis Outline

The thesis is divided into two parts. The remainder of Part I is organized as follows. Chapter 2 covers the relevant theoretical background on URLLC, massive random access, and AoI. Chapter 3 contains extensive summaries of the individual papers included in the thesis, and a conclusion and discus-

sion of future work is given in Chapter 4. The contributions, in the form of submitted and published scientific papers, are included in Part II, which comprises the main part of the thesis. The papers are ordered according to the IoT use case that they consider. As mentioned, Papers A, B, and C are related to URLLC, Papers D, E, and F are related to mMTC, and particularly to massive random access, and finally Papers G and H are related to AoI.

2 Wireless Connectivity for the Internet of Things (IoT)

In this chapter, we provide a brief overview of the theory behind the core topics that underlie the contributions, namely URLLC (Section 2.1), massive random access (Section 2.2), and AoI (Section 2.3).

2.1 Ultra-reliable and Low Latency Communication

Ultra-Reliable and Low Latency Communication (URLLC) refers to the group of wireless connectivity services that aim to deliver very high reliability and low latency connectivity for applications such as industrial manufacturing and automation, robotics, connected cars, etc. In this context, latency refers to the time from transmission of a packet until it has been successfully received at the destination, and reliability refers to the probability that a packet is successfully delivered before the deadline. It should be noted that the reliability and latency requirements should be treated jointly, since, in theory, an arbitrary reliability can be achieved by transmitting with sufficiently low rate provided that one has enough time, and a latency requirement is meaningless without a reliability constraint. Furthermore, as in any other practical communication system, there are typically additional constraints, such as power and frequency limitations, that also need to be obeyed, but these are not distinctive for URLLC.

Although URLLC does not have a strict definition, it frequently refers to systems with reliability requirements in the range from 99%–99.9999%, latency requirements from a few milliseconds down to hundreds of microseconds. Taking into account physical limitations of wireless communications, these requirements typically imply small packet sizes and a modest number of devices (e.g., up to 100, often less). In addition to reliability and latency, it is useful to also introduce *survival time* as the time that an application can operate without connectivity, and *availability* as the probability that the fraction of time the quality of service is provided (i.e., the latency, reliability and survival time. To illustrate the difference, consider an application that transmits periodically every millisecond, and has a survival time of 1 millisecond, so that it can tolerate single packet losses but not two packet losses in a row. Therefore, if the availability requirement is 99.99%, and if packet losses occur independently, this implies that the packet reliability must be at least $1 - \sqrt{10^{-8}} = 99.99\%$. However, because packet losses typically not are independent, and because it may be difficult for the base station to provide

Table 2.1: Selection of URLLC use cases defined by 3GPP [7]

Scenario	Latency	Availability	Survival time	Payload size
Motion control 1	500 us	99.999%–99.99999%	500 us	50 bytes
Motion control 2	1 ms	99.999%–99.999999%	1 ms	40 bytes
Cooperative carrying of fragile work pieces	5 ms	99.999%–99.999999%	5 ms	250–500 bytes
Emergency stop	8 ms	99.999999%	16 ms	40–250 bytes
Smart grid load control	50 ms	99.9999%	—	100 bytes

feedback within the latency requirement, it may be necessary to support a much higher reliability target in order to meet the availability requirement. These definitions, amongst others, are widely used by the 3rd Generation Partnership Project (3GPP) in relation to URLLC [6]. The requirements for a selection of use cases defined by the 3GPP are listed in Table 2.1 [7].

A natural question that arises from the requirements defined above is how to model the reliability of wireless systems under latency constraints. Although this is largely an open question since typical communication models are not constructed to be accurate when it comes to rare events (see e.g., [8–10]), in this thesis we will primarily be concerned with the flat block-fading channel where a transmitted signal $\mathbf{x} \in \mathbb{C}^n$ is received at the destination as

$$\mathbf{y} = h\mathbf{x} + \mathbf{w}, \tag{2.1}$$

where $h \in \mathbb{C}$ is the instantaneous channel coefficient distributed according to a distribution $p(h)$ and $\mathbf{w} \in \mathbb{C}^n$ is additive white Gaussian noise with elements drawn independently from $\mathcal{CN}(0, N_0)$. Without loss of generality, the channel coefficient is assumed to be normalized, i.e. $\mathbb{E}[|h|^2] = 1$. Furthermore, due to the strict latency constraints, we will assume that the device does not have time to estimate the channel coefficient h, and thus needs to design its transmission schemes based on the long-term statistics $p(h)$. This assumption is reasonable when the transmissions are sporadic, as opposed to deterministic or periodic. Under this model, the transmission reliability is given as the outage probability, defined as

$$p_{\text{out}}(R) = \Pr\left(\log_2(1 + |h|^2\gamma) < R\right), \tag{2.2}$$

where R is the transmission rate measured in bits/s/Hz and $\gamma = \frac{\mathbb{E}[|x|^2]}{N_0}$ is the average received signal-to-noise ratio (SNR). Using this definition, we can define the *outage capacity* as the maximal capacity that achieves a reliability of at least ϵ

$$C_\epsilon = \sup\{R : p_{\text{out}}(R) < \epsilon\}. \tag{2.3}$$

It should be noted that the definition in Eq. (2.2) is valid as $n \to \infty$, whereas URLLC packets generally are short as indicated in Table 2.1. Nevertheless, for most practical fading distributions $p(h)$ the fading coefficient h is much more dominant than the noise, and thus Eq. (2.2) serves as a good metric for the impact of the channel on the reliability [11].

A typical assumption is that h is Rayleigh distributed, in which case the outage capacity can be expressed as

$$C_\epsilon = \log_2 \left(1 - \ln(1-\epsilon)\gamma\right). \tag{2.4}$$

When ϵ is small $-\ln(1-\epsilon) \approx \epsilon$, and thus $C_\epsilon \approx \log_2 \left(1+\epsilon\gamma\right) \approx \epsilon\gamma$ [12]. Thus, it can be seen that in the case of Rayleigh fading, adding one nine to the reliability, i.e., reducing ϵ by a factor of ten, requires approximately a ten-fold reduction in the rate. This illustrates the high cost of providing URLLC service, and suggests that URLLC should be addressed by designing services that target the individual requirements of each application, as opposed to giving all applications the same URLLC service designed to meet the most strict requirements.

The previous discussion reveals that the random channel coefficient h is the limiting factor when it comes to meeting the reliability targets envisioned in URLLC. One way to reduce the impact of h is to transmit the signal across parallel channels and achieve channel diversity. This can be done by e.g., transmitting the signal in different coherence blocks (i.e., at different times or frequencies), so that the channels experienced by the transmissions are effectively independent. By transmitting across L independent channels, the rate supported by the channel is the average rate supported by the L channels, so the outage probability can be expressed as [12]

$$p_{\text{out}}(R) = \Pr\left(\frac{1}{L}\sum_{l=1}^{L}\log_2(1+|h_l|^2\gamma) < R\right). \tag{2.5}$$

The fact that the rates from the individual channels are averaged provides a so-called "hardening" effect against bad channels, which reduces the outage probability. Another way to achieve diversity is to use multiple antennas at the transmitter or the receiver. Equipping the transmitter with L antennas leads to the case in which the channels, as opposed to the rates, are averaged [12]

$$p_{\text{out}}(R) = \Pr\left(\log_2\left(1+\frac{1}{L}\sum_{l=1}^{L}|h_l|^2\gamma\right) < R\right). \tag{2.6}$$

Similarly, by having L antennas at the receiver, the receiver can combine the signals received by each antenna and obtain a gain in both diversity and

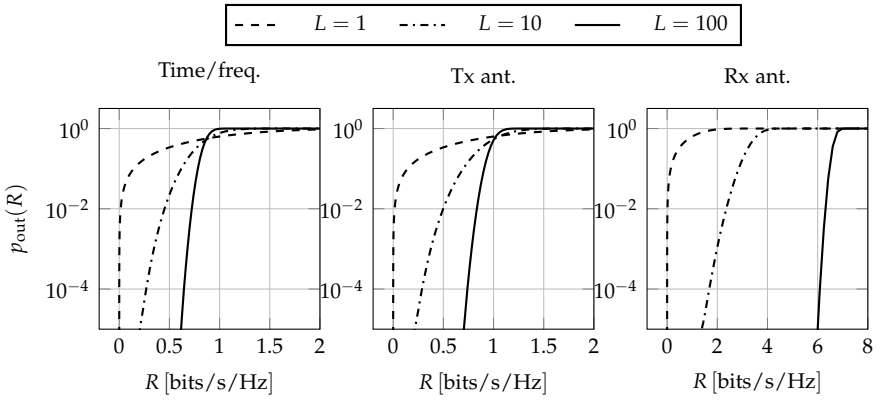


Fig. 2.1: Comparison of outage probabilities achieved with L diversity branches through time/frequency diversity, transmitter antenna diversity, and receiver antenna diversity. Note how the curves become steeper and more concentrated around their mean values as L increases.

power, resulting in an outage probability of [12]

$$p_{\text{out}}(R) = \Pr\left(\log_2\left(1 + \sum_{l=1}^{L} |h_l|^2 \gamma\right) < R\right). \tag{2.7}$$

Note that the power gain is primarily due to the larger physical size of the antenna, which allows it to capture more power. Under the assumption that the diversity branches (antennas or time/frequency slots) are independent, adding an additional antenna at the receiver yields highest reduction in the outage probability, and adding an antenna to the transmitter provides a higher gain than adding a time or frequency slot, as illustrated in Fig. 2.1. However, in practice there are often other benefits to time/frequency diversity that are not captured by the model above, such as low cost and complexity, etc. Furthermore, in practice there is often a high degree of correlation between the signals received by the antennas, which result in diminishing gains as the number of antennas are increased. The combination of these aspects means that a system would typically combine several diversity schemes to achieve robustness while balancing cost and complexity.

2.2 Massive Random Access

One of the use cases that has recently received much attraction is massive MTC (mMTC), where the number of potentially active devices is massive, e.g., up to 300,000 within a single cell as envisioned by some IoT scenarios [1]. However, although mMTC shares some characteristics with URLLC,

such as small packet sizes, mMTC does not provide strict reliability and latency guarantees, and transmissions are typically infrequent and sporadic of nature (e.g., one transmission every two hours per device [13]). Because of the massive number of devices, the set of potentially active devices (i.e., the ones that have data to transmit) at any given time is very large. As a result, the central challenge in supporting mMTC lies in designing the random access scheme, where the active devices in an uncoordinated fashion either signal to the base station that they have data to transmit, or directly transmit their data.

In the first case, where the active devices start by transmitting a signal to indicate their activity, the base station needs to first detect the active devices, a process referred to as activity detection. Following the detection of a set of active devices, the base station sends as feedback to each device a pointer to dedicated wireless resources in which they can each transmit their data without interference. Due to the overhead of this three-step mechanism, it is mainly beneficial when the devices have large amounts of data to transmit. Nevertheless, several activity detection methods have been designed for the massive access scenario, e.g., based on advanced signal processing techniques, such as compressed sensing [14].

The second case of random access schemes, where devices directly transmit their data, can be seen as a generalization of the first scheme, where the devices are allowed to transmit their data rather than some activity indicator signal. Schemes within this category are often variants of slotted ALOHA [15], inspired by the scheme used in the ALOHA System [16]. In slotted ALOHA, the wireless medium is divided into recurrent slots, in which the active devices transmit their data. To simplify the analysis, it is often assumed that the arrivals is Poisson distributed, and that the receiver is unable to recover packets transmitted in slots with more than one transmission [15]. This assumption gives rise to the so-called collision channel, in which the throughput, measured in packets per slot, is given as

$$T = \lambda e^{-\lambda}, \tag{2.8}$$

where λ is the average number of transmissions in a slot. The throughput expression in Eq. (2.8) is maximized when $\lambda = 1$, achieving a throughput of $1/e \approx 0.37$ packets per slot. This reveals the notoriously low spectral efficiency of slotted ALOHA, since on average only slightly more than one third of the slots are used. To increase the throughput, a number of schemes have been proposed that make use of repeated transmissions combined with more advanced signal processing in the form of successive interference cancellation. These techniques allow the throughput to be increased significantly [17, 18]. By additionally introducing packet-level coding, the throughput can reach close to one packet per slot [19] or even higher with multipacket reception capabilities [20].

Due to the significance of ALOHA-like random access, a number of attempts has been made to characterize the fundamental information-theoretic limits of random access. Two prominent recent models targeting the massive access scenario are the many-access channel [21] and unsourced random access (URA) channel [22]. The main difference between these two models is that the many-access channel assumes that the devices use individual codebooks, whereas the URA model assumes that all devices use the same codebook. Although this difference may seem small, it means that the many-access channel includes both identification and message recovery, whereas the URA model only considers message recovery. This has a significant impact in the asymptotic regime where the number of potentially active devices grows towards infinity, since the blocklength required for device identification in many-access channel also grows to infinity. On the other hand, URA can work with a fixed blocklength in the asymptotic regime as long as the number of *active* devices remains fixed, and, contrary to the many-access channel, includes slotted ALOHA as a valid scheme.

The original URA model proposed by Polyanskiy in [22] assumes that the number of active devices is fixed and denoted by K_a, and considers the memoryless and *permutation invariant* channel $P_{Y|X_1,\ldots,X_{K_a}} : \mathcal{X}^{K_a} \to \mathcal{Y}$, where $\mathcal{X}$ denotes the set of codewords. Furthermore, a random-access code for M messages ($M \geq K_a$) and a given blocklength n is defined by an encoder

$$f : [M] \to \mathcal{X}^n, \tag{2.9}$$

where $[M] = \{0, 1, \ldots, M-1\}$, and a corresponding decoder that outputs a list of K_a messages

$$g : \mathcal{Y}^n \to \{\widehat{\mathcal{X}} \subseteq [M] \mid |\widehat{\mathcal{X}}| = K_a\}. \tag{2.10}$$

A valid encoder and decoder satisfies the average reliability constraint

$$\frac{1}{K_a} \sum_{j=1}^{K_a} \Pr(E_j) \leq \epsilon, \tag{2.11}$$

where E_j denotes the event that that a message W_j transmitted by device j is either not contained in the message list generated by the decoder, or collides with a message from another device (i.e., a device $i \neq j$ transmits message $W_i = W_j$). A typical example of a channel is the Gaussian multiple access channel, studied in [22] and defined as

$$Y = X_1 + \ldots + X_{K_a} + Z, \tag{2.12}$$

where $Z \sim \mathcal{N}(0, 1)$ and the codewords are subject to an energy constraint $\|f(W_j)\|_2^2 \leq nP$. The model has subsequently been extended to other channels, including the Rayleigh fading channel [23, 24].

One limitation of the URA model is that it assumes that the number of devices K_a is fixed and known to the decoder, which is typically not the case in practice, where the set of active devices is random, as in e.g., slotted ALOHA. This also has the consequence that a collision error implies a false positive message, and thus the model fails to properly characterize the trade-off between misdetections and false positives. This inconvenience has recently been addressed in [25, 26], which extend the URA model to the case where K_a is random, and which rigorously characterize the trade-off between misdetections and false positives. Note that for linear channels the URA problem is equivalent to the well-known compressed sensing problem, but with a very large sensing matrix ($n \times M$), which makes conventional compressed sensing methods infeasible [27].

In addition to providing information-theoretic bounds on random access, the URA model has inspired a large number of practical schemes. Notable schemes include schemes based on coded compressed sensing [27–29], sparse regression codes [30], tensor based modulation [31], as well as schemes that split codewords into a preamble and a data part [32].

2.3 Age of Information

Age of Information (AoI) was introduced in [33, 34] as a metric for the *timeliness* of sensor information at the destination in vehicular networks, and further formalized in [3]. Contrary to the traditional metric of packet delay/latency, AoI characterizes the *age* of the most recent packet received at the destination. Formally, let $\tau_1, \tau_2, \ldots,$ denote the time instances at which packets $1, 2, \ldots$ are generated at the sensor device, and denote by τ_i' the time at which the i-th packet is received at the destination node (clearly, $\tau_i' \geq \tau_i$). The instantaneous AoI at time t, denoted by $\Delta(t)$, is then defined as

$$\Delta(t) = t - \max\{\tau_i | \tau_i' \leq t\}, \tag{2.13}$$

and the long-term average AoI is

$$\mathbb{E}[\Delta] = \lim_{t\to\infty} \frac{1}{t} \int_0^t \Delta(t)\, dt. \tag{2.14}$$

Obviously, if a packet is lost in the transmission, we can define $\tau_i' = \infty$ so that it does not contribute to reducing the AoI. The AoI definition in Eq. (2.13) results in the characteristic sawtooth curve illustrated in Fig. 2.2, where $\Delta(t)$ increases linearly with time between packet receptions, and is reduced to the age of the new packet when a new packet is received.

To gain intuition about the meaning of AoI as compared to e.g., delay, it is instructive to consider a first-come first-served queue as depicted in Fig. 2.3.

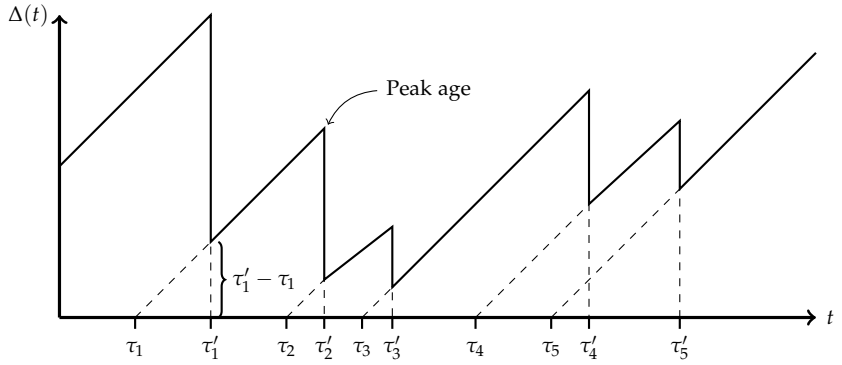


Fig. 2.2: Illustration of the AoI process (solid line). Packets are generated at times $\tau_1, \tau_2, \ldots$ and received at the destination node at times $\tau_1', \tau_2', \ldots$.

Fig. 2.3: Illustration of a first-come first-served queue.

If the rate of arriving packets is high, the delay that a packet experiences is also likely to be high as a result of queuing, and conversely, when the arrival rate is low, the delay is likely to be low. However, contrary to the traditional per-packet delay metric, a low arrival rate also results in a low AoI due to the long duration between packet receptions. This trade-off between arrival rate and queuing time was the initial motivation of AoI, and has been characterized in numerous scenarios, including a single source [3, 35, 36], multiple sources/streams [37–39], as well as in general queuing networks [40–42]. Furthermore, several studies consider also alternative queuing disciplines, such as the last-come first-served discipline, which generally achieves lower AoI than first-come first-served [37, 39, 43–46].

In this thesis, we will primarily be concerned with a discrete definition of the AoI, which can be defined recursively as

$$\Delta[t+1] = \begin{cases} \min(t - \tau_i, \Delta[t]) + 1 & \text{if update } i \text{ is received at time } t, \\ \Delta[t] + 1 & \text{otherwise,} \end{cases} \tag{2.15}$$

for $t = 0, 1, 2, \ldots$ and $t[0] = 0$. This setup has been considered in numerous works, including discrete queues [35, 47], design of random access schemes [48–50], and scheduling problems with multiple sensors [51, 52]. Scheduling problems share the same fundamental trade-off as the queuing problems mentioned earlier, since scheduling can be seen as a way to control how the service process of the queues at each device.

In addition to the classical AoI, a number of variants have also been defined, most notably the peak age [44] and cost of update delay (CoUD) [53]. The peak age characterizes the peaks of the AoI curve, i.e., the AoI measured immediately prior to a new update, and while it can sometimes be more analytically tractable than AoI, it still captures the main properties of AoI. The peak age has for instance been used with finite-blocklength information theory to characterize the trade-off between packet length, decoding reliability and timeliness [54], and in ALOHA-based random access schemes [55, 56]. The CoUD, sometimes also referred to as value of information (VoI), generalizes the concept of AoI by considering arbitrary increasing cost functions of the age (AoI represents the special case in which the cost function is the identity function). CoUD has for instance been used in networked control systems [57] and vehicular networks [58]. Another variant of AoI is the Query AoI (QAoI) metric [59, 60], that takes into account not only how data are generated, but also how data are accessed at the destination node. For instance, if data is accessed at periodic intervals, then the data need to be fresh only at those instants. Comprehensive surveys on AoI with more examples can be found in e.g., [61, 62].

3 Summaries of Contributions

This chapter provides summaries of each paper included in Part II. We start by presenting the papers related to URLLC in Section 3.1, then the papers related to massive random access in Section 3.2, and finally, in Section 3.3, the papers related to AoI.

3.1 Ultra-Reliable Low Latency Communication

Paper A: "Network Slicing in Industry 4.0 Applications: Abstraction Methods and End-to-End Analysis"

In this paper, we propose network slicing as a way to address diverse end-to-end requirements in an industrial scenario comprising several physical networks, which are virtualized or abstracted into generic connectivity services. Network slicing refers to the concept of dividing a single network into virtual isolated communication services, i.e., *slices*, that offer a specific quality of service, such as a given latency and reliability. For example, one network slice may provide an URLLC service for critical devices, while another slice, sharing the same underlying physical network resources, provides a service for less critical mMTC-like traffic. Because of the ability to virtualize the physical network into distinct services, network slicing is often highlighted as a key technology to support heterogeneous service types in wireless systems.

We assume that the network is hierarchically structured and consists of periodic, cyclic communication at the factory units which interconnects controllers, sensors and actuators, and a packet switched network at the factory level, which connects the factory units to computation and storage resources, the cloud, etc. We focus on the latency and reliability aspects, and based on the assumed network structure we devise a framework that uses network calculus to guarantee the end-to-end latency of the individual slices. Similarly, the reliability is guaranteed by analyzing the probability that any of the individual links in a connection will fail. We illustrate the applicability of the framework by applying it to an example scenario, which also validates that it is suitable for industrial networks.

Paper B: "Ultra-Reliable Communication for Services with Heterogeneous Latency Requirements"

While Paper A abstracted the physical layer of the individual links, this paper studies the problem of supporting heterogeneous applications from the

perspective of the physical layer in a wireless system, which can be seen as an implementation of network slicing at the physical layer. Specifically, we study the problem of providing connectivity to two categories of sporadic applications, which both have strict reliability requirements but different latency requirements. The motivation behind this scenario is the observation that high reliability comes at a very high cost when combined with a strict latency requirement. In particular, when the application requires very low latency, the only source of diversity is frequency, and the device needs to transmit across a sufficient number of resources to ensure that the probability of outage is low. On the other hand, when the latency is less strict, say, two or three frames instead of one, it is possible to also spread the transmissions across time, which introduces several benefits. First, it is possible to have non-orthogonal transmission between the two services without risking that all resources of either service are contaminated by transmissions that belong to the other service. Second, the base station can provide feedback between the frames, so that the device only needs to transmit in the second and third frames if the supported rate in the first frame is insufficient.

We study the gains of such a scheme, and compare the spectral efficiency to (1) the case in which both applications are given the same service based on the most strict requirement, and (2) the proposed scheme without feedback. To assess the efficiency gains of the proposed scheme, we compute the ratio between the average supported rate and transmission rate required to achieve the target reliability (i.e., the outage capacity). A high ratio indicates low spectral efficiency, since the *excess capacity* is high. On the other hand, a ratio close to one the indicates that the excess capacity is low and hence the system is spectrally efficient. The results show that although the non-orthogonal scheme forces the rates of the individual applications to be reduced, the fact that it allows for more diversity significantly increases the reliability. Furthermore, they show that although feedback further increases the spectral efficiency, the gain is relatively small, suggesting that the non-orthogonal division provides the most significant gains. In conclusion, the paper illustrates the importance of considering individual application requirements, as opposed to providing a "one-size-fits-all" URLLC service designed to support applications with the most strict requirements.

Paper C: "Prediction of mmWave/THz Link Blockages Through Meta-Learning and Recurrent Neural Networks"

This paper studies how machine learning can be used to exploit spatial correlation in mmWave/THz URLLC scenarios to detect blockages. mmWave/THz systems are generally attractive in for certain industrial URLLC use cases due to their large bandwidth, wide coherence bandwidth, and short coherence time, which allows for short symbol durations and in turn low latency [63].

However, due to the high directivity of mmWave/THz systems, they have significantly less space diversity than systems that operate at lower frequencies. In particular, mmWave/THz systems are susceptible to signal blockages, which lead to sudden drops in the signal power and outages. Being able to predict such blockages allows the communication system to adapt before the actual blockage, e.g., by changing control strategy or by changing communication interface to a more reliable link.

In the paper, we consider a scenario in which devices located at fixed locations transmit periodically to a base station. In addition to the static devices, we assume that a number of objects move around in the area and generate temporary signal blockages. The overall idea is that this movement causes blockages to occur with a spatio-temporal correlation, since signal paths that are physically close are likely to be blocked by the same object within a relatively short time interval. As a result, blockages and drops in the signal power observed to one device can be used to predict blockages of other devices. We focus on the detection problem, and apply machine learning and recurrent neural networks to learn the patterns and predict blockages. Furthermore, to avoid the need for large amounts of training data for a specific configuration, we apply meta-learning to learn efficient initialization points of the neural network so that that they can converge more quick than if the initialization was random. We show that the recurrent neural network is able to learn the blockage patterns, and that meta-learning can reduce the detection time, especially when the number of training samples is small. Although we do not evaluate specific mitigation techniques, the fact that the blockages can be predicted suggests that the spatio-temporal correlation can be exploited to increase reliability, and that machine learning is a promising method for implicitly learning such correlations in context of blockage prediction.

3.2 Massive Random Access

Paper D: "Random Access Schemes in Wireless Systems with Correlated User Activity"

In this paper, we consider a slotted ALOHA random access scenario, where the devices, contrary to the traditional assumption, transmit in a correlated but uncoordinated fashion. In particular, we assume that there are K slots in a frame and N devices, and that the active devices within a given frame are drawn independently from the joint distribution $\Pr(x_1, x_2, \ldots, x_N)$, where $x_n = 1$ if device n is active, otherwise $x_n = 0$. Because the aim of the paper is to show how such correlation can be exploited, we assume that the activation distribution is known a priori.

To exploit the activation correlation, we consider the problem of assigning transmission slots to the devices, so that they will transmit in these slots whenever they are active. Thus, the scheme can be seen as a semi-scheduled random access method. We propose two heuristic algorithms that aim to minimize collisions by assigning the slots in such a way that devices that are likely to transmit at the same time are assigned to different slots, while devices that are unlikely to transmit at the same time are assigned to different slots. We propose two heuristic algorithms that produce such assignments. The algorithms are based on upper and lower bounds on the pairwise correlation, and while this means that they cannot exploit higher order correlations, the assumption significantly reduces the complexity of the algorithms. Finally, we also consider variants of the algorithms, which, in addition to assigning slots, also specify the probability with which a device should discard its packet instead of transmitting it. This is beneficial when the load is high, since it reduces the risk of collisions. The proposed algorithms are evaluated in an alarm scenario, where physical events are assumed to activate devices within a given radius of the event. We show that they increase the throughput, i.e., the number of non-colliding transmissions, significantly compared to the classical slotted ALOHA.

The idea developed in the paper has been extended in [64–66], which are not included in the thesis. Paper [64] considers a time-dependent variant of the problem where devices are assumed to activate according to a known hidden Markov model, and transmission grants are given to the devices that are expected to be active. In [65], the problem of estimating the correlation is considered, and algorithms based on stochastic optimization are proposed to maximize the throughput and the sum-rate. A similar approach is taken in [66] with the aim to minimize the outage probability.

Paper E: "Massive Random Access with Common Alarm Messages (extended version)"

Paper E considers a scenario similar to that in Paper D, but instead of slotted ALOHA, it studies the problem from an information theoretic perspective within the framework of unsourced random access. We analyze a scenario in which the devices can transmit either regular messages, or, in case of an external alarm event, alarm messages. While the regular messages transmitted by the devices are drawn independently by each device, upon the event of an alarm, all devices that observe the event transmit the same message drawn from a separate message set. We derive an achievability bound under the constraint that the alarm message must be detected with high probability, and use it to study the trade-off between spectral efficiency and reliability. We show that alarm messages can be transmitted reliably when the number of regular messages is small, while the probability of false positive alarm

messages becomes significant when the interference caused by the regular messages is high, leading to low spectral efficiency.

Paper F: "Common Message Acknowledgments: Massive ARQ Protocols for Wireless Access"

Unlike the previous papers which studies the uplink phase in a massive access scenario, this paper studies general problem of providing message acknowledgments after the completion of the uplink phase. The motivation for studying this problem is twofold. First, modern random access methods tend to decode devices in batches, e.g., after each frame, which means that many devices need to be acknowledged at the same time, allowing for the construction of a compressed common feedback message. Second, the fact that a random subset of the devices are active means that an acknowledgment message requires not only a single bit per device as in the case of coordinated access, but needs to acknowledge either the source address or the full message, which requires a significant number of bits. To illustrate, suppose we want to send acknowledgments to 50 devices, which each has a 64-bit address. A naïve message construction would be to concatenate the addresses, which requires a total of 3200 bits. On the other hand, there are only $\binom{2^64}{50}$ different messages that acknowledges 50 devices, suggesting that only $\log_2 \binom{2^64}{50} \approx 2985$ bits are needed. The paper aims to investigate various methods for jointly encoding the acknowledgments so that the total message length is minimized.

We consider both error-free encoding and erroneous encoding, and derive bounds on the required message length for both cases, which show that introducing a small fraction of false positive acknowledgments leads to a significant reduction in the message length. We then analyze the performance of various practical schemes with false positives, and show that some schemes can achieve a performance that is very close to the theoretical bounds. Furthermore, we study the impact of the false positives on the reliability in a scenario with retransmissions, and show that the reduced message length allows the rate to be reduced (assuming a fixed number of available channel symbols), which in turn increases not only the reliability of the acknowledgment message but also the probability that a device transmission succeeds within the maximum number of allowed transmissions.

3.3 Age of Information

Paper G: "Minimizing the Age of Information From Sensors With Common Observations"

In this paper, we study how correlated observations can be exploited from the perspective of AoI. We consider a scenario comprising a set of sources that generate events according to independent Poisson processes. Each event is observed by a set of sensor devices with a given probability, and each device may observe multiple sources. The goal is to schedule the devices so that the average AoI of the sources at the destination node is minimized. As in several of the previous papers, the scenario is motivated by scenarios such as environmental sensing, where the ultimate interest is not in the data at the individual devices, but in the phenomenon that they observe. We propose two scheduling policies for the problem. In the first, we assume that the observation model is known to the scheduler, and propose a myopic (greedy) scheduling policy that minimizes the expected AoI in the next time slot. In the second, we assume that the observation model is unknown and needs to be learned first. Inspired by the contextual bandit problem, we propose to use a generalized linear model of the expected reduction in AoI, and show that it performs close to the optimal policy. Overall, the paper shows that by taking into account the correlation, the AoI can be significantly reduced compared to alternative policies.

Paper H: "Timely Monitoring of Dynamic Sources with Observations from Multiple Wireless Sensors"

This paper extends Paper G by allowing the sources to have states, which influence the probability that they are observed by the sensor devices. The motivation behind this is to capture dynamic sources, such as vehicles. As exemplified in the paper, the model can for instance be used in a scenario where the sensor devices represent cameras with fixed fields of view, while sources are vehicles that move around between the cameras. The source states represent the position of the vehicles, and they can only be observed by a given camera when they are in a certain state representing the camera's field of view. In contrast to Paper G, we consider the long-term optimal scheduling policy, obtained by optimizing an average-cost Markov decision process. We first consider the case in which the source states are observable, and prove that the model has a stationary optimal policy, which can be found using relative value iteration. We also derive a closed-form expression for the average AoI obtained using a random policy, which serves as a baseline scheme. In addition to the case with observable source states, we also consider the more

realistic situation where the source states are unobservable. We consider two variants of the problem. In the first, the source state is revealed when the source is observed. In the second, the source identities are never revealed, which means that both the source states and their AoI are unobservable. Due to the complexity required to obtain optimal policies for these two variants, we derive approximate but computationally efficient policies. We evaluate all proposed schemes and characterize the penalty incurred by partial observability. To assess the potential benefit of having sensor devices that observe multiple source states, we consider a scenario that comprises both sensors that observe single individual states and sources that observe multiple states, but with lower probability. The results indicate that, despite the lower observation probability, sensors that observe multiple source states are particularly beneficial in the partially observable case, where there is uncertainty about the actual states of the sources.

4 Conclusions and Future Work

The contributions presented in this thesis cover a broad range of topics related to massive, reliable and timely connectivity for IoT. The focus has been centered around three research questions, namely how to support the diverse and heterogeneous timing requirements of IoT (Q1), how to exploit correlation among IoT devices (Q2), and how to use learning and feedback to support massive, reliable, and timely connectivity (Q3). In the following, we conclude on each of the three research questions based on the findings in the thesis and discuss potential future directions.

Q1: How can a communication system be designed to support and take advantage of the diverse and heterogeneous timing requirements for IoT?

We have studied the problem of supporting diverse and heterogeneous requirements in Paper A, where we proposed a framework based on deterministic network calculus to guarantee the latency of network slices in URLLC. As more devices get connected, often with various technologies, such frameworks become necessary to ensure that end-to-end latency guarantees are met. Nevertheless, it requires the reliabilities of the underlying technologies to be specified, which, as discussed in Section 2.1, remains an open problem that needs to be address before the proposed framework can be used with wireless URLLC.

In Papers B and E, we have given examples of how wireless systems, by taking advantage of diverse application requirements, can significantly improve the efficiency of the systems, compared to the case where the systems are designed to support a single service that satisfies the requirements of all applications at the same time. This idea of tailoring the wireless services to the requirements of individual applications represents a promising direction towards supporting the wide range of applications envisioned for future wireless systems, as opposed to "just" dividing the services into URLLC, mMTC, and mobile broadband. A potential future direction of this could be to work towards designing a number of generic building blocks that can be combined in different ways to tailor the service to a given application. This would require not only identifying the individual building blocks, but also characterizing the performance of the numerous systems that can be built by combining blocks. For this, the ideas explored in Paper A may be useful.

Q2: How can the correlation among IoT devices be exploited to improve the efficiency of the communication systems?

We have considered several types of device correlation, namely in the channel (Paper C), in random access activation processes (Papers D and E), and in sensor observations (Papers G and H). Each of these papers challenges the typical assumption that devices act independently, which has been dominant in the research that has influenced the wireless systems deployed today. We have shown that this correlation can be challenging to handle for a system designed by assuming independence, and that by taking the correlation into account, the system efficiency can be significantly improved for all the considered types of correlation.

The work on exploiting correlation for blockage prediction in Paper C is a promising idea for providing high reliability with mmWave systems. However, while predicting blockages represents a significant part of a system that is robust against blockages, the design of methods that adapt the communication scheme when a blockage is predicted represents an equally important aspect that has not been considered in this thesis. Papers D and E demonstrate how correlation can be used in the design of random access mechanisms. However, while the gains are large, the considered scenarios are somewhat constructed and artificial, and obtaining good correlation models for realistic IoT scenarios is required to assess the practical gains. Another interesting direction for future work would be to consider temporal correlation, i.e., correlation that happens across frames, which may be more suited for the correlation that occurs in practice, where devices may not be fully synchronized. Along the same lines, generalizing the alarm model considered in Paper E to the case where messages are not drawn independently, but from a joint distribution similar to the one considered in Paper G, would also be a step towards more practically applicable schemes. Finally, the models considered in the AoI studies in Papers G and H could be generalized to cases where sources observe e.g., noisy observations of the sources, similar to the problem considered for two sensors in [67]. However, a rigorous study of such a scenario would require the inclusion of distributed source coding, which complicates the problem. Nevertheless, it would represent an interesting application of the closely related CoUD metric.

Q3: How can learning and feedback be used to support massive, reliable, and timely connectivity?

We have studied Q3 from the perspective of learning in Paper C for blockage prediction, and in Papers G and H for AoI scheduling. The blockage prediction scheme for URLLC proposed in Paper C is an example of a machine learning mechanism that can be added on top of other schemes, e.g., the one

considered in B. Although the proposed scheme does not itself increase reliability, it provides information that can be used by the system to increase the overall reliability by continuously adapting based on the predicted imminent blockages. The mitigation technique may vary depending on the application; for instance, some applications may want to change to a more robust secondary interface, while control applications may want to change the control algorithm to a more robust one. Integrating such information provided by the connectivity service with application-layer algorithms, such as a robotic controller, is an interesting and promising direction towards not only building more intelligent applications, but also towards enabling reliable applications on top of imperfect communication services.

We also used learning in Papers G and H to assist AoI-minimizing scheduling policies. In Paper G, machine learning was used to directly learn a scheduling policy for a scenario with unknown correlation, whereas in Paper H we assumed that the correlation was known, but used statistical learning to deal with partially observable sensor states. In both cases, we demonstrated that learning is a promising technique that enables the system to exploit the type of correlation that can be found in IoT scenarios. While the works here were limited to AoI, similar ideas could also be applied to AoI variants like CoUD and QAoI, or even to other IoT scenarios where it might be beneficial to take correlation into account in the scheduling process.

In regard to the use of feedback, we have studied methods for providing message acknowledgments for massive access in Paper F. While we have focused on the downlink only, the proposed methods can readily be combined with a wide range of random access schemes, including the random access schemes studied in Papers D and E. We have shown that the proposed acknowledgment encoding scheme, despite introducing a small fraction of false positives, substantially increases the reliability compared to several naïve encoding schemes. The paper represents a promising direction of future research towards the design of retransmission schemes for the massive access problem, which can lead to increased reliability and spectral efficiency of mMTC. Potential future work includes adapting the scheme to the characteristics considered in Q1 and Q2. For Q1, the scheme could be designed to better suit heterogeneous activation probabilities, i.e., where some devices are more active than others, so that the false positives are more likely for devices that are unlikely to be active. Similarly, for Q2 the scheme could be designed so that devices that are likely to transmit together are acknowledged together, allowing for an even higher encoding efficiency.

References

[1] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5g: physical and mac-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, 2016.

[2] P. Popovski, F. Chiariotti, V. Croisfelt, A. E. Kalør, I. Leyva-Mayorga, L. Marchegiani, S. R. Pandey, and B. Soret, "Internet of things (IoT) connectivity in 6G: An interplay of time, space, intelligence, and value," *arXiv preprint arXiv:2112.00491*, 2021.

[3] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *2012 Proceedings IEEE INFOCOM*, 2012, pp. 2731–2735.

[4] R. B. Sørensen, J. J. Nielsen, and P. Popovski, "Machine learning methods for monitoring of quasiperiodic traffic in massive iot networks," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7368–7376, 2020.

[5] K. Stern, A. E. Kalør, B. Soret, and P. Popovski, "Massive random access with common alarm messages," in *proc. 2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 1–5.

[6] 3GPP, "Service requirements for the 5G system," 3rd Generation Partnership Project (3GPP), TS 22.261, Dec. 2021, v18.5.0.

[7] ——, "Service requirements for cyber-physical control applications in vertical domains," 3rd Generation Partnership Project (3GPP), TS 22.104, Dec. 2021, v18.3.0.

[8] P. C. F. Eggers, M. Angjelichinoski, and P. Popovski, "Wireless channel modeling perspectives for ultra-reliable communications," *IEEE Transactions on Wireless Communications*, vol. 18, no. 4, pp. 2229–2243, 2019.

[9] V. N. Swamy, P. Rigge, G. Ranade, B. Nikolić, and A. Sahai, "Wireless channel dynamics and robustness for ultra-reliable low-latency communications," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 4, pp. 705–720, 2019.

[10] M. Angjelichinoski, K. F. Trillingsgaard, and P. Popovski, "A statistical learning approach to ultra-reliable low latency communication," *IEEE Transactions on Communications*, vol. 67, no. 7, pp. 5153–5166, 2019.

[11] W. Yang, G. Durisi, T. Koch, and Y. Polyanskiy, "Quasi-static multiple-antenna fading channels at finite blocklength," *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4232–4265, 2014.

[12] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge university press, 2005.

[13] ITU-R, "Guidelines for evaluation of radio interface technologies for IMT-2020," International Telecommunication Union (ITU), Tech. Rep. M.2412-0, Nov. 2017.

[14] L. Liu, E. G. Larsson, W. Yu, P. Popovski, C. Stefanovic, and E. de Carvalho, "Sparse signal processing for grant-free massive connectivity: A future paradigm for random access protocols in the internet of things," *IEEE Signal Processing Magazine*, vol. 35, no. 5, pp. 88–99, 2018.

[15] L. G. Roberts, "Aloha packet system with and without slots and capture," *SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, p. 28–42, apr 1975. [Online]. Available: https://doi.org/10.1145/1024916.1024920

[16] N. Abramson, "Development of the alohanet," *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 119–123, 1985.

[17] E. Casini, R. De Gaudenzi, and O. Del Rio Herrero, "Contention resolution diversity slotted aloha (crdsa): An enhanced random access schemefor satellite access packet networks," *IEEE Transactions on Wireless Communications*, vol. 6, no. 4, pp. 1408–1419, 2007.

[18] G. Liva, "Graph-based analysis and optimization of contention resolution diversity slotted aloha," *IEEE Transactions on Communications*, vol. 59, no. 2, pp. 477–487, 2011.

[19] E. Paolini, G. Liva, and M. Chiani, "Coded slotted aloha: A graph-based method for uncoordinated multiple access," *IEEE Transactions on Information Theory*, vol. 61, no. 12, pp. 6815–6832, 2015.

[20] C. Stefanovic, E. Paolini, and G. Liva, "Asymptotic performance of coded slotted aloha with multipacket reception," *IEEE Communications Letters*, vol. 22, no. 1, pp. 105–108, 2018.

[21] X. Chen, T.-Y. Chen, and D. Guo, "Capacity of gaussian many-access channels," *IEEE Transactions on Information Theory*, vol. 63, no. 6, pp. 3516–3539, 2017.

[22] Y. Polyanskiy, "A perspective on massive random-access," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 2523–2527.

[23] S. S. Kowshik, K. Andreev, A. Frolov, and Y. Polyanskiy, "Energy efficient random access for the quasi-static fading mac," in *2019 IEEE International Symposium on Information Theory (ISIT)*, 2019, pp. 2768–2772.

[24] ——, "Energy efficient coded random access for the wireless uplink," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4694–4708, 2020.

[25] K.-H. Ngo, A. Lancho, G. Durisi, and A. G. i. Amat, "Massive uncoordinated access with random user activity," in *2021 IEEE International Symposium on Information Theory (ISIT)*, 2021, pp. 3014–3019.

[26] ——, "Massive uncoordinated access with random user activity," *arXiv preprint arXiv:2103.09721*, 2022.

[27] V. K. Amalladinne, J.-F. Chamberland, and K. R. Narayanan, "A coded compressed sensing scheme for unsourced multiple access," *IEEE Transactions on Information Theory*, vol. 66, no. 10, pp. 6509–6533, 2020.

[28] V. K. Amalladinne, A. K. Pradhan, C. Rush, J.-F. Chamberland, and K. R. Narayanan, "Unsourced random access with coded compressed sensing: Integrating amp and belief propagation," *IEEE Transactions on Information Theory*, pp. 1–1, 2021.

[29] A. Fengler, S. Haghighatshoar, P. Jung, and G. Caire, "Non-bayesian activity detection, large-scale fading coefficient estimation, and unsourced random access with a massive mimo receiver," *IEEE Transactions on Information Theory*, vol. 67, no. 5, pp. 2925–2951, 2021.

[30] A. Fengler, P. Jung, and G. Caire, "Sparcs for unsourced random access," *IEEE Transactions on Information Theory*, vol. 67, no. 10, pp. 6894–6915, 2021.

[31] A. Decurninge, I. Land, and M. Guillaud, "Tensor-based modulation for unsourced massive random access," *IEEE Wireless Communications Letters*, vol. 10, no. 3, pp. 552–556, 2021.

[32] A. Fengler, O. Musa, P. Jung, and G. Caire, "Pilot-based unsourced random access with a massive mimo receiver, interference cancellation, and power control," *IEEE Journal on Selected Areas in Communications*, pp. 1–1, 2022.

[33] S. Kaul, M. Gruteser, V. Rai, and J. Kenney, "Minimizing age of information in vehicular networks," in *2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2011, pp. 350–358.

[34] S. Kaul, R. Yates, and M. Gruteser, "On piggybacking in vehicular networks," in *2011 IEEE Global Telecommunications Conference - GLOBECOM 2011*, 2011, pp. 1–5.

[35] Y. Sun, E. Uysal-Biyikoglu, R. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," in *IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications*, 2016, pp. 1–9.

[36] R. Talak and E. H. Modiano, "Age-delay tradeoffs in queueing systems," *IEEE Transactions on Information Theory*, vol. 67, no. 3, pp. 1743–1758, 2021.

[37] N. Pappas, J. Gunnarsson, L. Kratz, M. Kountouris, and V. Angelakis, "Age of information of multiple sources with queue management," in *2015 IEEE International Conference on Communications (ICC)*, 2015, pp. 5935–5940.

[38] E. Najm and E. Telatar, "Status updates in a multi-stream m/g/1/1 preemptive queue," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018, pp. 124–129.

[39] R. D. Yates and S. K. Kaul, "The age of information: Real-time status updating by multiple sources," *IEEE Transactions on Information Theory*, vol. 65, no. 3, pp. 1807–1827, 2019.

[40] R. Talak, S. Karaman, and E. Modiano, "Minimizing age-of-information in multi-hop wireless networks," in *2017 55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2017, pp. 486–493.

[41] A. M. Bedewy, Y. Sun, and N. B. Shroff, "The age of information in multihop networks," *IEEE/ACM Transactions on Networking*, vol. 27, no. 3, pp. 1248–1257, 2019.

[42] R. D. Yates, "The age of information in networks: Moments, distributions, and sampling," *IEEE Transactions on Information Theory*, vol. 66, no. 9, pp. 5712–5728, 2020.

[43] S. K. Kaul, R. D. Yates, and M. Gruteser, "Status updates through queues," in *2012 46th Annual Conference on Information Sciences and Systems (CISS)*, 2012, pp. 1–6.

[44] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Transactions on Information Theory*, vol. 62, no. 4, pp. 1897–1910, 2016.

[45] R. D. Yates, "Age of information in a network of preemptive servers," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018, pp. 118–123.

[46] A. M. Bedewy, Y. Sun, and N. B. Shroff, "Minimizing the age of information through queues," *IEEE Transactions on Information Theory*, vol. 65, no. 8, pp. 5215–5232, 2019.

[47] A. Kosta, N. Pappas, A. Ephremides, and V. Angelakis, "The age of information in a discrete time queue: Stationary distribution and non-linear age mean analysis," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1352–1364, 2021.

[48] J. Sun, Z. Jiang, B. Krishnamachari, S. Zhou, and Z. Niu, "Closed-form whittle's index-enabled random access for timely status update," *IEEE Transactions on Communications*, vol. 68, no. 3, pp. 1538–1551, 2020.

[49] H. Chen, Y. Gu, and S.-C. Liew, "Age-of-information dependent random access for massive iot networks," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2020, pp. 930–935.

[50] D. C. Atabay, E. Uysal, and O. Kaya, "Improving age of information in random access channels," in *IEEE INFOCOM 2020 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2020, pp. 912–917.

[51] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Scheduling policies for minimizing age of information in broadcast wireless networks," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2637–2650, 2018.

[52] I. Kadota, A. Sinha, and E. Modiano, "Scheduling algorithms for optimizing age of information in wireless networks with throughput constraints," *IEEE/ACM Transactions on Networking*, vol. 27, no. 4, pp. 1359–1372, 2019.

[53] A. Kosta, N. Pappas, A. Ephremides, and V. Angelakis, "The cost of delay in status updates and their value: Non-linear ageing," *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4905–4918, 2020.

[54] R. Devassy, G. Durisi, G. C. Ferrante, O. Simeone, and E. Uysal, "Reliable transmission of short packets through queues and noisy channels under latency and peak-age violation guarantees," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 4, pp. 721–734, 2019.

[55] A. Munari and G. Liva, "Information freshness analysis of slotted aloha in gilbert-elliot channels," *IEEE Communications Letters*, vol. 25, no. 9, pp. 2869–2873, 2021.

[56] A. Munari, F. Lázaro, G. Durisi, and G. Liva, "An age of information characterization of frameless ALOHA," *arXiv preprint arXiv:2112.00491*, 2021.

[57] O. Ayan, M. Vilgelm, M. Klügel, S. Hirche, and W. Kellerer, "Age-of-information vs. value-of-information scheduling for cellular networked control systems," in *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, 2019, pp. 109–117.

[58] M. Giordani, T. Higuchi, A. Zanella, O. Altintas, and M. Zorzi, "A framework to assess value of information in future vehicular networks," in *Proceedings of the 1st acm mobihoc workshop on technologies, models, and protocols for cooperative connected cars*, 2019, pp. 31–36.

[59] J. Holm, A. E. Kalør, F. Chiariotti, B. Soret, S. K. Jensen, T. B. Pedersen, and P. Popovski, "Freshness on demand: Optimizing age of information for the query process," in *proc. 2021 IEEE International Conference on Communications (ICC)*, 2021, pp. 1–6.

[60] F. Chiariotti, J. Holm, A. E. Kalør, B. Soret, S. K. Jensen, T. B. Pedersen, and P. Popovski, "Query age of information: Freshness in pull-based communication," *IEEE Transactions on Communications*, 2022, accepted for publication. Preprint: https://arxiv.org/abs/2105.06845.

[61] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: A new concept, metric, and tool," *Foundations and Trends in Networking*, vol. 12, no. 3, pp. 162–259, 2017.

[62] R. D. Yates, Y. Sun, D. R. Brown, S. K. Kaul, E. Modiano, and S. Ulukus, "Age of information: An introduction and survey," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 5, pp. 1183–1210, 2021.

[63] R. Ford, M. Zhang, M. Mezzavilla, S. Dutta, S. Rangan, and M. Zorzi, "Achieving ultra-low latency in 5g millimeter wave cellular networks," *IEEE Communications Magazine*, vol. 55, no. 3, pp. 196–203, 2017.

[64] M. Shehab, A. K. Hagelskjær, A. E. Kalør, P. Popovski, and H. Alves, "Traffic prediction based fast uplink grant for massive IoT," in *proc. 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2020, pp. 1–6.

[65] C. Zheng, M. Egan, L. Clavier, A. E. Kalør, and P. Popovski, "Stochastic resource optimization of random access for transmitters with correlated

activation," *IEEE Communications Letters*, vol. 25, no. 9, pp. 3055–3059, 2021.

[66] ——, "Stochastic resource allocation for outage minimization in random access with correlated activation," in *2022 IEEE Wireless Communications and Networking Conference (WCNC)*, 2022, accepted for publication.

[67] J. Hribar, M. Costa, N. Kaminski, and L. A. DaSilva, "Using correlated information to extend device lifetime," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2439–2448, 2019.

Part II

Papers

Paper A

Network Slicing in Industry 4.0 Applications: Abstraction Methods and End-to-End Analysis

Anders E. Kalør, René Guillaume, Jimmy J. Nielsen, Andreas Mueller, and Petar Popovski

Published in
IEEE Transactions on Industrial Informatics, vol. 14, no. 12, pp. 5419–5427, 2018, doi: 10.1109/TII.2018.2839721.

The layout has been revised.

Abstract

Industry 4.0 introduces modern communication and computation technologies such as cloud computing and Internet of Things to industrial manufacturing systems. As a result, many devices, machines and applications will rely on connectivity, while having different requirements to the network, ranging from high reliability and low latency to high data rates. Furthermore, these industrial networks will be highly heterogeneous as they will feature a number of diverse communication technologies. Current technologies are not well suited for this scenario, which requires that the network is managed at an abstraction level which is decoupled from the underlying technologies. In this paper, we consider network slicing as a mechanism to handle these challenges. We present methods for slicing deterministic and packet-switched industrial communication protocols which simplifies the manageability of heterogeneous networks with various application requirements. Furthermore, we show how to use network calculus to assess the end-to-end properties of the network slices.

1.1 Introduction

The fourth industrial revolution, known as Industry 4.0, brings cyber-physical systems and Internet of Things (IoT) to industrial manufacturing systems [1, 2]. Furthermore, the number of interconnected physical devices will increase drastically, and they will continuously interact with local cloud services in order to act intelligently and flexibly. This introduces numerous challenges to industrial networks, which have traditionally been very static and strongly isolated [3, 4]. First, it is expected that many new technologies, comprising both wired and wireless connections, will gradually be introduced into production lines resulting in a very heterogeneous network [5]. Secondly, the network will have to serve a wide range of applications with different Quality-of-Service (QoS) requirements, ranging from traditional closed-loop control systems to event-driven sensors and Augmented Reality (AR) displays. For instance, control and alarm systems may require a delivery reliability in the order of $1 - 10^{-9}$ and end-to-end latencies in the range of 0.5–5 ms, while at the same time interactive applications require high data rates and moderate latencies [6, 7]. Finally, the increased system complexity also poses a challenge in managing the network and in particular the end-to-end QoS. This necessitates programmability of the network, as well as a framework for analyzing the end-to-end network characteristics [3, 8].

In this paper, we consider end-to-end network slicing as an architecture for handling the network complexity. Network slicing refers to the use of Software-Defined Networking (SDN) and Network Function Virtualization (NFV) to *slice* a network into logically isolated sub-networks [9]. Each network slice may have certain properties such as latency and reliability guar-

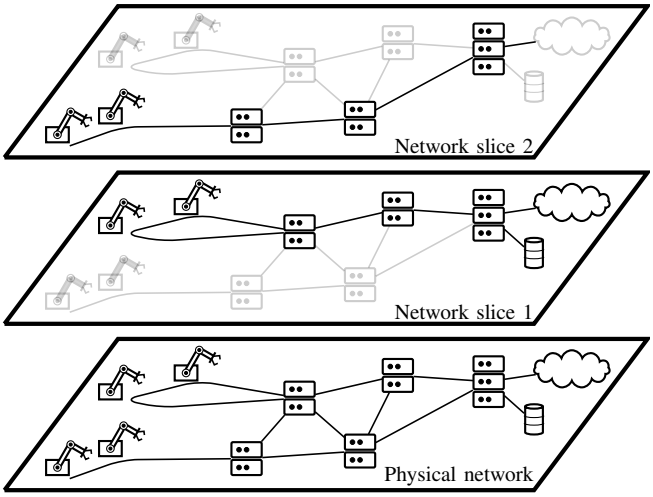


Fig. A.1: The network slicing concept where each network slice contains a subset of the physical network resources.

antees, and appears to applications as a single unified network in an abstract form that is independent of the specific underlying communication technologies. Furthermore, network slices may be constructed with restrictive access control e.g. that traffic in a certain slice cannot leave the internal network. As an example, a network slice may be constructed to offer very low latency communication from a group of wireless sensor devices to a cache in the edge of the network, or a database in the cloud. To guarantee a low latency service, communication and buffer resources along a path are allocated according to the expected aggregate data arrival from the sensors, and the traffic is served with high priority using preemptive packet schedulers. At the same time, in the same physical network, another network slice may reserve resources for specific service-oriented architectures, such as OPC UA [10], or for a high throughput best-effort service between another end-device and the factory cloud, e.g. to allow for downloading firmware updates while being logically isolated from critical control traffic. This is illustrated in Fig. A.1.

Network slicing has previously been studied in the context of industrial systems as a way to manage the increasing complexity of manufacturing networks [11–13]. However, the concept has mainly been treated from an architectural point of view, and the problem of how to slice the physical networks has received less attention. Outside the industrial domain, network slicing has been studied extensively in the context of Internet protocols and 5G systems, where it is considered an essential technology for handling heterogeneous application requirements [14–17]. Unfortunately, these methods are targeted best-effort Internet protocols and cannot be directly applied

to industrial networks which are currently dominated by very specialized and often deterministic communication protocols [18]. However, due to the advantages of SDN in regard to manageability and flexibility, several SDN-based architectures for industrial networks and cyber-physical systems have been proposed recently, see e.g. [8, 19, 20] and references therein. Analysis of end-to-end QoS in SDN-based industrial networks using network calculus has been treated in [21–23] which also study algorithms for queue allocation and admission control in priority queue networks. However, while queuing networks are increasingly made available to industrial systems, e.g. using Ethernet TSN [3], it is unlikely that all networks in a manufacturing system will be replaced at once. Hence compatibility with traditional and current technologies, such as fieldbuses and cyclic master/slave industrial Ethernet protocols, as well as interoperability between technologies, is of vital importance [3, 5, 24]. This requires that the problem is approached from an abstraction level which is decoupled from the specific protocols, so that it can function across heterogeneous subsystems.

This paper presents methods for slicing both cyclic and switched industrial communication protocols in an abstract setting which is independent of the specific implementation of the underlying protocols. Using network calculus, we demonstrate how worst-case end-to-end properties of the network slices can be calculated, both for a specific use case and in a general setting, and compare it to simulation results. The remainder of the paper is organized as follows. Section 1.2 introduces methods for slicing industrial networks. Section 1.3 describes a personalized medicine manufacturing system, which is used as basis for an end-to-end analysis in Section 1.4. Finally, Section 1.5 presents numerical and simulation results, and the paper is concluded in Section 1.6.

1.2 Network Slicing for Industry 4.0

Industrial networks are often structured hierarchically as illustrated in Fig. A.2 [25]. The factory units contain the individual devices such as actuators, sensors, etc., usually in a master/slave configuration connected through a deterministic and cyclic communication link. The factory units are interconnected by a factory network, which may also connect the devices to a local cloud, or to an external infrastructure such as the Internet. In addition, the factory network may contain computation and storage resources which can be used by the devices in the network. The factory network may be based on real-time Ethernet, in the presence of strict real-time requirements, or regular switched Ethernet and TCP/IP technologies that exploit statistical multiplexing and provide high throughput and interoperability with general-purpose hardware [25, 26]. The communication technologies that are used at the dif-

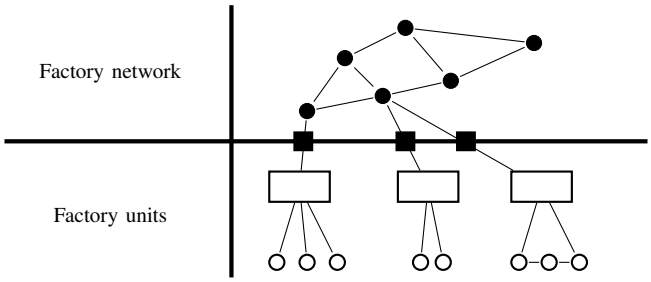


Fig. A.2: Hierarchical industrial network architecture. White circles are factory unit devices that are controlled by master devices indicated by rectangles. The master devices are connected to the factory network through gateways (solid squares). The factory network consists of switches and computation/storage nodes (solid circles).

ferent levels, and even at the different units, are not necessarily interoperable, and may require gateways in order to exchange information [27]. Even if a gateway interconnects two real-time communication technologies, it is likely to introduce queuing since the cycles of the networks and the gateway may not be synchronized.

1.2.1 Slicing Factory Unit Networks

The factory unit networks are comprised of cyclic master/slave communication technologies with a fixed cycle time, and each cycle contains a number of pre-allocated and fixed-sized telegrams. The components in the factory units usually comprise sensors and actuators which periodically interacts with a controller in a closed-loop control system. However, as manufacturing systems evolve towards cyber-physical systems, it is likely that sensors and actuators will also become more intelligent and generate traffic sporadically, e.g. only transmit when a sensor value exceeds a certain threshold, or in case of anomalies [28]. Furthermore, the requirements to the network are likely to be different, e.g. infrequent aperiodic sensor readings may require higher reliability than periodic sensor readings transmitted several times per millisecond. In the following paragraphs we discuss three slicing schemes which provide different trade-offs in isolation, latency and reliability and utilization: Static telegram allocation, shared telegrams and telegram overwriting. To ease the presentation, we consider a scenario consisting of one deterministic application that transmits R_d telegrams of size N_d in every cycle, and K applications that exhibit a stochastic transmission behavior where the kth application in a given cycle transmits a random number of telegrams, R_k, of fixed size N_k. We do not cover the case where applications transmit telegrams of random size, since this is not a common scenario in practice, and because it requires applications to be able to read the frames before writing their data in order to

detect telegram boundaries, which is often not feasible due to the short cycle times. Instead, the telegram size may be considered as an upper bound on the amount of data that need to be transmitted.

In a traditional configuration, the static telegram allocation, applications are assigned telegrams in each cycle according to the data that they will transmit. The deterministic application will then have allocated $R'_{\mathrm{d}} = R_{\mathrm{d}}$ telegrams of size N_{d}, and the reliability is entirely defined by transmission errors introduced by the communication link. For the stochastic applications, suppose R'_k telegrams are allocated to application k. Neglecting transmission and other error sources, and assuming that excess telegrams are not buffered but dropped, the reliability of the scheme, denoted by ζ_k, is the probability that a telegram is among the R'_k that are transmitted:

$$\zeta_k = 1 - \sum_{r=R'_k}^{\infty} \left(1 - \frac{R'_k}{r}\right) \Pr(R_k = r). \tag{A.1}$$

Furthermore, ignoring propagation delays, the latency experienced by the application is uniform in the cycle duration. This scheme provides a very high degree of isolation due to the separation of resources between applications. However, this comes at the cost of low utilization, since the resources are left empty during a cycle where less than R'_k telegrams are transmitted. Especially transmission patterns that exhibit strong burstiness and have strict requirements to ζ_k, are likely to result in low utilization.

In scenarios where the applications have low transmission rates, it may be beneficial to have multiple applications sharing the same telegrams in order to exploit the statistical multiplexing. However, as argued previously, it is usually not possible for an application to read the contents of a frame before writing to it, which means that there is a risk of overwriting telegrams from other applications. Suppose the telegrams of the K applications have equal length, $N_k = N'$ for all K, and denote the total number of allocated telegrams by $R' = \sum_k R'_k$ and the total number of transmitted telegrams by $R = \sum_k R_k$. We assume that an application writes to a random telegram (uniformly distributed), and that two writes to the same telegram result in failure of both transmissions. Furthermore, we ignore the fact that some applications may be more likely to succeed, e.g. due to being located closer to the master device in a ring topology. Under these assumptions, a transmission is successful if no other application writes to the same telegram:

$$\zeta_k = \frac{1}{\Pr(R > 0)} \sum_{r=1}^{\infty} \left(1 - \frac{1}{R'}\right)^{r-1} \Pr(R = r), \tag{A.2}$$

where we have normalized to condition on the event that at least one transmission occur ($R > 0$).

While the above scheme increases the utilization by increasing the number of applications that share the same resources, it still results in a low utilization for low arrival rates, especially if the reliability requirements are strict. A way to improve the utilization in these cases is to allow applications with strict reliability requirements to overwrite telegrams allocated to other applications. For instance, a closed-loop control system with periodic feedback may be operational during short interruptions in the feedback. In a system with both closed-loop control and applications with sporadic transmissions, the sporadic transmissions can overwrite the feedback transmissions without causing failure of the control system. Suppose $R'_{\mathrm{d}} = R_{\mathrm{d}}$ telegrams of size N_{d} are allocated to the control traffic. We retain the notation of N and R introduced in the previous scheme, and assume that $N \leq N_{\mathrm{d}}$ so that the telegram size of the sporadic traffic can be contained within a control traffic. Under this scheme, the probability that an arbitrary control telegram is overwritten by any of the R sporadic telegrams is given by

$$\zeta_{\mathrm{d}} = \sum_{r=0}^{\infty} \left(1 - \frac{1}{R_{\mathrm{d}}}\right)^{r} \Pr(R = r). \tag{A.3}$$

Similar to the previous scheme, a sporadic transmission also fails if two or more transmissions overwrite the same control telegram:

$$\zeta_k = \frac{1}{\Pr(R > 0)} \sum_{r=1}^{\infty} \left(1 - \frac{1}{R_{\mathrm{d}}}\right)^{r-1} \Pr(R = r). \tag{A.4}$$

1.2.2 Factory Networks

Factory networks are typically based on conventional (switched) or real-time Ethernet. In conventional Ethernet the frames are queued at each link, while real-time Ethernet usually supports both strict real-time traffic and conventional Ethernet traffic through different channels. Since the methods covered in the previous section can be directly applied to the strict real-time functionality of the protocols, we here focus on conventional Ethernet traffic, but we do not distinguish between whether it is served by conventional Ethernet links or by the Ethernet channel in real-time Ethernet technologies. We slice the networks by assigning a dedicated egress queue to each individual slice in each hop in the network. To handle the latency requirements in industrial scenarios, we employ a strict priority scheduler between the queues. Priority schedulers are widely supported by networking hardware through management APIs such as OpenFlow [29]. Several frameworks for analyzing the end-to-end latency in queuing networks have been proposed in literature, most notably queuing theory [30], stochastic network calculus (SNC) [31] and deterministic network calculus (DNC) [32]. Queuing theory seeks probabilistic quantities of queues such as the waiting time distribution and the mean

number of items in the queue. While such results, in particular distributions, are useful for analyzing the end-to-end guarantees in a network, the calculations are often intractable for traffic and service time distributions that are not memoryless, which limits the range of applicable scenarios. SNC extends the number of applicable scenarios by seeking bounds on the latency distribution instead of exact results. However, while many traffic arrival distributions can be used in SNC, it falls short when traffic models are deterministic, such as periodic arrivals that are prominent in industrial networks. DNC provides worst-case bounds on the latency, and supports a wide range of traffic models as long as the arrivals can be upper bounded by some function. As the factory unit networks enforces an upper bound on the arrivals in each cycle, DNC is well suited for industrial applications. For this reason, we will focus on modeling the latency using DNC. For simplicity, we restrict the presentation of DNC to the case where traffic arrival are bounded by affine functions. A thorough and more general discussion of DNC can be found in e.g. [32, 33].

The theory of DNC is based on arrival and service curves which bound the cumulative number of bytes that arrive to and are served by a queue. Example curves, chosen for illustrative purposes, are illustrated in Fig. A.3, where the dashed line $A'(t)$ is a periodic arrival curve that is bounded by the solid affine curve $A(t)$, and $S(t)$ and $D(t)$ are the service and departure curves, respectively. $W(t)$ is the waiting time experienced by the data arriving at time $t = 1.5$. We assume that the cumulative number of bytes generated by an application per time unit, say seconds, is upper bounded by an affine arrival function

$$A(t) = [\alpha t + \beta]_+, \tag{A.5}$$

where $[x]_+ = \max(x, 0)$. The parameters $\alpha \geq 0$ and $\beta \geq 0$ are referred to as rate and burst parameters, respectively. For example, the arrivals from an application that generates a frame of size N periodically every M seconds would be bounded by the affine function parameterized by $\alpha = N/M$ and $\beta = N$.

Similar to the arrivals, we assume that the rate at which bytes are extracted from a queue (typically modeling the serialization of frames) is lower bounded by the affine service function

$$S(t) = [\sigma t - \rho]_+, \tag{A.6}$$

where $\sigma \geq 0$ is the minimum service rate and $\rho \geq 0$ accounts for service given to bursts of higher priority. For the system to be stable we require $\sigma \geq \alpha$. The waiting time, i.e. the time a frame spends in the queue while waiting for service, is bounded by [33]

$$W(t) \leq \frac{\rho + \beta}{\sigma}. \tag{A.7}$$

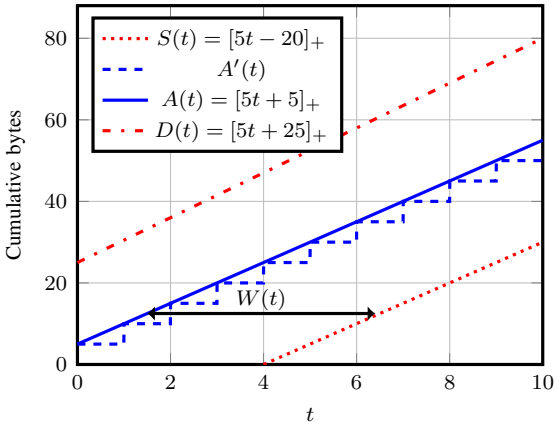


Fig. A.3: Graphical representation of the quantities in DNC. Curves are chosen for illustrative purposes.

If frames from multiple applications arrive to the same queue, the aggregate arrival rate is parameterized by $\alpha = \sum_j \alpha_j$ and $\beta = \sum_j \beta_j$.

A single queue as considered above usually does not suffice when a wide range of application requirements are present, since some frames may need to receive higher priority. To analyze prioritization queuing using DNC, we need to introduce the concept of leftover service, which refers to the minimum service that is left to a flow after flows of higher priorities have been served. To simplify the analysis, we ignore the impact of frame blocking. This simplification may be justified by frame preemption mechanisms, e.g. introduced in Ethernet TSN [34]. If two arrival curves $A_1(t) = [\alpha_1 t + \beta_1]_+$ and $A_2(t) = [\alpha_2 t + \beta_2]_+$ are served by the same server $S(t)$, but $A_1(t)$ is prioritized higher than $A_2(t)$, then the leftover service for $A_2(t)$ is given as [33]

$$S_{\text{lo}}(t) \geq [(\sigma - \alpha_1)t - \rho - \beta_1]_+ , \tag{A.8}$$

while $A_1(t)$ is served by the entire service given by $S(t)$. Notice that DNC is not restricted to prioritization queueing, and several other scheduling schemes can be analyzed as well [33].

Finally, in order to obtain results across several queues in series, we need to obtain bounds on the departures of a queue. It can be shown that the departures, $D(t)$, from a system are bounded by [32]

$$D(t) \leq \left[\alpha t + \beta + \frac{\alpha\rho}{\sigma}\right]_+ , \tag{A.9}$$

which is again an affine bounded arrival function, but with the burst increased by $\alpha\rho/\sigma$.

Table A.1: End-to-end requirements considered in the use case

Application	Source	Dest.	Type	Mean period	Size	Latency req.	Reliability req.	Priority
Control feedback	Robot	Master	Periodic	1 ms	128 B	1 ms	$1 - 10^{-4}$	1
Device alarms	Any	Cloud	Poisson	60 s	32 B	5 ms	$1 - 10^{-6}$	1
Patient info request	Master	Cloud	Periodic	200 ms	128 B	10 ms	$1 - 10^{-4}$	2
Scale readings	Scale	Cloud	Periodic	200 ms	512 B	100 ms	$1 - 10^{-6}$	3
AR stream	AR display	Cloud	Periodic	20 ms	20 kB	20 ms	$1 - 10^{-2}$	3

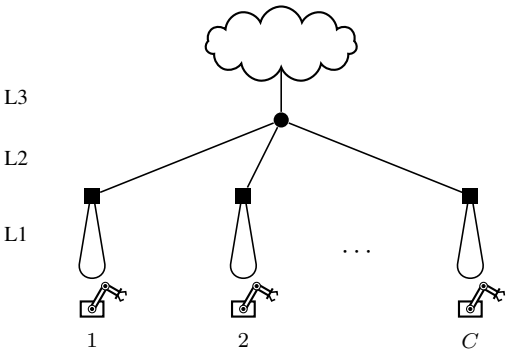


Fig. A.4: Personalized medicine manufacturing network consisting of C factory units and three link levels: L1, L2 and L3.

1.3 Use Case: Personalized Medicine Manufacturing

To study how network calculus can be applied to analyze the end-to-end delays in a real manufacturing system, we define a simple Industry 4.0 use case for a system that produces personalized medicine. The system is described in an abstract way, which is independent of the specific communication technologies that are used. The system consists of C identical master/slave factory units which communicate using a cyclic industrial protocol. Each factory unit network is connected to a gateway that provides access to a factory network based on conventional switched Ethernet. The factory network is connected to the cloud through a switch (Fig. A.4). We denote the three link levels by L1, L2 and L3.

Each of the C factory units contains a number of devices which are used to dispense a drug product into a container and to supervise the process. The amount of dispensed drug is gathered from the cloud in real-time, and the weight of the resulting product is stored in the cloud. The drug is dispensed by a pipetting machine mounted on a robotic arm, which is part of a closed-loop control system executed by the master device. Furthermore, the individual devices may raise alarms if an anomaly is detected, and the pro-

cess can be followed from a AR display, which continuously streams video to the cloud.

We assume that the master/slave network within the factory units are based on 100 Mbit Ethernet with a cycle time of $\tau = 1$ ms, and that the factory network is based on switched 100 Mbit Ethernet with prioritization queuing. The frame delivery reliability of both the industrial and switched Ethernet links is $\zeta_{\text{link}} = 1 - 10^{-9}$. For simplicity, we only consider uplink traffic from the devices to the master device or the cloud. The traffic characteristics and latency/reliability requirements are listed in Table A.1, along with the considered priority scheme where a lower number represents higher priority. We note that finding an optimal priority scheme is a complex problem, and that the priorities used here are chosen mainly for illustrative purposes.

The use case is very general as it has only few assumptions to the underlying communication network: That the factory units deploy a cyclic master/slave communication protocol with reserved resources, and that the factory network is based on switched Ethernet with preemption and prioritization queuing. As a result, the same use case could apply both to wired and wireless factory units, and to several communication technologies.

1.4 End-to-End Latency Analysis

In this section, we apply the network slicing methods described in Section 1.2 to the use case, and analyze end-to-end latency and reliability properties. For the reasons of clarity, our discussion and analysis will be at first focused to the device alarms and the patient info requests in the concrete use case. We will then provide directions for generalizing the analysis by applying suitable abstractions to other use cases. However, a formal description of the abstraction process is outside of the scope for this article.

For simplicity, we ignore propagation delays as well as potential overhead added by protocol headers in the network. To deal with the complexity involved in modeling multi-hop networks, we employ a conservative approach and consider each link independently. This method is conservative since bursts are considered at each link, and hence it does not comply with the "pay burst only once" principle [32]. Throughout the analysis we use milliseconds as the time unit.

1.4.1 Use Case Analysis

We first consider the device alarms which are transmitted to the cloud. Suppose that we at the device level, L1, allocate R_{alarms} bytes in each cycle according to one of the proposed schemes so that we obtain a worst-case latency equal to the cycle time, τ, of 1 ms with probability $\zeta_{\text{L1,alarms}}$. R_{alarms} enforces

a limit to the number of bytes that arrive to link L2 at the factory network. Specifically, at most R_{alarms} bytes arrive in each cycle, so that the arrivals are bounded by $A_{\text{L2,alarms}}(t) = [R_{\text{alarms}}t + R_{\text{alarms}}]_+$ bytes, where t denotes the elapsed time in milliseconds. Since device alarms have highest priority, the queue is served by the 100 Mbit link with rate $\sigma = 0.125 \cdot 10^5$ bytes/ms and $\rho = 0$. It follows from (A.7) that the waiting time at link L2 is bounded by $W_{\text{L2,alarms}} \leq R_{\text{alarms}}/(0.125 \cdot 10^5)$ ms.

The aggregate arrival from all C factory units arrive to the high-priority queue to link L3. The arrivals from each factory unit are bounded by (A.9) as $D_{\text{L2,alarms}}(t) \leq [R_{\text{alarms}}t + R_{\text{alarms}}]_+$ bytes. It follows that the arrivals to link L3 are bounded by $A_{\text{L3,alarms}}(t) = CD_{\text{L2,alarms}}(t)$, which yields a waiting time of $W_{\text{L3,alarms}} \leq CR_{\text{alarms}}/(0.125 \cdot 10^5)$ ms. The worst-case end-to-end latency, L_{alarms}, is then given by the sum of the cycle time τ and the waiting times at L2 and L3:

$$L_{\text{alarms}} \leq \tau + \frac{R_{\text{alarms}}}{0.125 \cdot 10^5}(1 + C). \tag{A.10}$$

Similarly, the end-to-end reliability is the product of reliabilities at each link. The reliability of link L1 is given by the packet delivery probability of the network slicing scheme and the reliability of the Ethernet link, i.e. $\zeta_{\text{L1,alarms}}\zeta_{\text{link}}$, while links L2 and L3 each has reliability ζ_{link}. Hence, the end-to-end reliability is

$$\zeta_{\text{alarms}} = \zeta_{\text{L1,alarms}}(\zeta_{\text{link}})^3. \tag{A.11}$$

Following the same procedure for the patient info requests and allocating $R_{\text{req}} = 128$ bytes at L1 and taking the periodicity of 200 ms into account, the arrivals are bounded by $A_{\text{L2,req}}(t) = [R_{\text{req}}/200t + R_{\text{req}}]_+$. The leftover service remaining after the device alarms in the high-priority queue have been served is given by (A.8) as

$$S_{\text{L2,req}}(t) = [(0.125 \cdot 10^5 - R_{\text{alarms}})t - R_{\text{alarms}}]_+, \tag{A.12}$$

$$S_{\text{L3,req}}(t) = [(0.125 \cdot 10^5 - CR_{\text{alarms}})t - CR_{\text{alarms}}]_+. \tag{A.13}$$

Similarly, the patient info requests arriving to link L3 are bounded by (A.9):

$$A_{\text{L3,req}}(t) = [\alpha_{\text{L3,req}}t + \beta_{\text{L3,req}}]_+ \tag{A.14}$$

with

$$\alpha_{\text{L3,req}} = CR_{\text{req}}/200, \tag{A.15}$$

$$\beta_{\text{L3,req}} = CR_{\text{req}} + \frac{CR_{\text{req}}/200R_{\text{alarms}}}{0.125 \cdot 10^5 - R_{\text{alarms}}}. \tag{A.16}$$

As a result, the worst-case end-to-end latency is

$$L_{\text{req}} \leq \tau + \frac{R_{\text{alarms}} + R_{\text{req}}}{0.125 \cdot 10^5 - R_{\text{alarms}}} + \frac{CR_{\text{alarms}} + \beta_{\text{L3,req}}}{0.125 \cdot 10^5 - CR_{\text{alarms}}}, \tag{A.17}$$

with reliability

$$\zeta_{\text{req}} = (\zeta_{\text{link}})^3. \tag{A.18}$$

1.4.2 Generalization

The analysis can be generalized to systems with the same hierarchical network structure by considering the factory unit and factory networks separately, and by using the fact that the factory unit network defines an upper bound on the arrivals to the factory network. Consider a factory unit application which transmits at most R bytes periodically every ηth cycle according to one of the proposed slicing schemes, and let τ denote the cycle time. An application transmission succeeds with a reliability ζ_{d} which is given by the slicing scheme, as described in Section 1.2, and the reliability of the underlying technology. R and τ dictates the affine arrival bound which arrives to the factory network as

$$A(t) = \left[\frac{R}{\eta\tau}t + R\right]_+, \tag{A.19}$$

with reliability ζ_{d}. At the factory network, suppose the frame is routed through the links $P = \{p_1, p_2, \ldots, p_n\}$. Each link p_i has delay W_{p_i} and reliability ζ_{p_i} so that the worst-case end-to-end delay is

$$L = \tau + \sum_{p_i \in P} W_{p_i}, \tag{A.20}$$

with reliability

$$\zeta = \zeta_{\text{d}} \prod_{p_i \in P} \zeta_{p_i}. \tag{A.21}$$

The link reliabilities ζ_{p_i} are independent of the traffic and completely described by the physical link characteristics. However, the queuing times W_{p_i} depend on the traffic characteristics at each link, and must be calculated in the order in which the links are traversed by the traffic.

To obtain the queuing times, consider an arbitrary link in the path, which is equipped with a number of prioritization queues. Let $q^{(i)} = \{f_i^{(1)}, f_i^{(2)}, \ldots, f_i^{(N)}\}$ be the flows sharing queue i at the link, and let each flow $f_i^{(l)}$ be characterized by the arrival curve $A_i^{(l)}(t) = [\alpha_i^{(l)} t + \beta_i^{(l)}]_+$. Furthermore, define the strict queue prioritization order $\epsilon_1, \epsilon_2, \ldots, \epsilon_K$, so that queue j is served before k if $\epsilon_j < \epsilon_k$. The minimum service given to a flow $f_j^{(l)} \in q^{(j)}$ is obtained from (A.8) where $A_1(t)$ is the aggregate of all other flows with higher or equal priority as $f_j^{(l)}$. Using the notation introduced

here, the minimum service given to $f_j^{(l)}$ is parameterized by

$$\sigma_j^{(l)} = \sigma + \alpha_j^{(l)} - \sum_{k\in[1,K]:\epsilon_k\leq\epsilon_j} \sum_{f_k^{(i)}\in q^{(k)}} \alpha_k^{(i)}, \tag{A.22}$$

$$\rho_j^{(l)} = \rho - \beta_j^{(l)} + \sum_{k\in[1,K]:\epsilon_k\leq\epsilon_j} \sum_{f_k^{(i)}\in q^{(k)}} \beta_k^{(i)}, \tag{A.23}$$

where σ and ρ describe the total service available to the queues at the node. The worst-case queuing delay and the departures are readily given by (A.7) and (A.9) as

$$W_j^{(l)} \leq \frac{\rho_j^{(l)} + \beta_j^{(l)}}{\sigma_j^{(l)}}, \tag{A.24}$$

$$D_j^{(l)}(t) \leq \left[\alpha_j^{(l)} t + \beta_j^{(l)} + \frac{\alpha_j^{(l)}\rho_j^{(l)}}{\sigma_j^{(l)}}\right]_+ . \tag{A.25}$$

By using $D_j^{(l)}(t)$ as arrival bound at the next link, the procedure can be repeated until the waiting times at all links have been obtained.

The proposed framework can be automated and used even in large-scale networks with many link levels. However, it assumes that the traffic has been assigned queue priorities, which is itself a difficult problem especially for large networks with many traffic sources. The problem of assigning traffic to queues is outside the scope of this paper, but we remark that methods proposed in the literature for DNC (e.g. [21–23]) can be directly applied to the factory network using the framework presented here. However, extending the methods to include allocation of computational and storage resources represents a challenge since current techniques for this problem consider a very generic latency model, where links have constant delay but are capacity constrained, which is not directly applicable to real networks [12, 35].

1.5 Numerical Results

In this section, we investigate the end-to-end latencies and reliabilities derived in the previous section for the overwriting and prioritization queuing slicing schemes presented in Section 1.2.

We first consider the resources for the device alarms within in each factory unit network. Since the number of alarms in each cycle is random, we can either reserve a fixed number of resources in each cycle, or we can allow alarms to overwrite the cyclic control traffic, which has a lower reliability requirement. Allocating a fixed number of resources results in a low

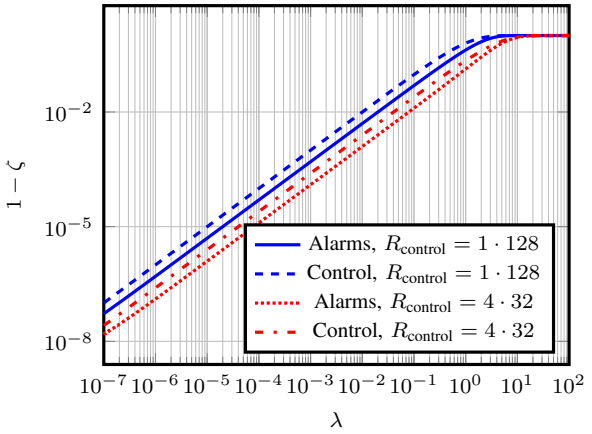


Fig. A.5: End-to-end failure rate of control and device alarm traffic in the network slicing scheme based on overwriting for various alarm arrival rates.

utilization, while overwriting control traffic introduces a decrease in the reliability of the control traffic. Since the rate of alarms is very low compared to the cycle time, the overwriting scheme is a promising approach for this use case. In the overwriting scheme, the reliability of the alarm traffic is given by (A.11) with $\zeta_{\text{L1,alarms}}$ obtained from (A.4). Similarly, the reliability of the control traffic, which only uses link L1, may be obtained as $\zeta = \zeta_{\text{L1,control}}\zeta_{\text{link}}$ with $\zeta_{\text{L1,control}}$ obtained from (A.3). The resulting reliability is illustrated in Fig. A.5, where the end-to-end frame failure rates $(1 - \zeta)$ for the alarm and control traffic are shown for different values of the mean alarm arrivals per cycle, λ. We consider the two cases where the control traffic is composed of either 1 telegram of 128 bytes, and of 4 telegrams of 32 bytes, denoted by $R_{\text{control}} = 1 \cdot 128$ and $R_{\text{control}} = 4 \cdot 32$. We assume that entire telegrams are overwritten in the overwriting scheme and that each telegram contains an integrity check so that it can be detected whether the original data has been overwritten. Therefore, only one alarm can be delivered per cycle when the control traffic is transmitted in a single telegram, and a single alarm results in complete failure of the control information. In the other case, if the control traffic is divided into 4 telegrams, then up to 4 alarms can be transmitted during a cycle, and one alarm transmission only causes failure of a single control telegram. This results in a lower failure rate as can be seen in the figure.

At the expected device alarm inter-arrival time of 60 s ($\lambda \approx 1.7 \cdot 10^{-4}$) from the use case, the reliability requirement of the control traffic ($\zeta = 1 - 10^{-6}$) is satisfied in the case where the control traffic is transmitted in 4 telegrams. Furthermore, the delivery reliability of the device alarm traffic at this point is also sufficient, and thus the overwriting slicing scheme would be a reasonable

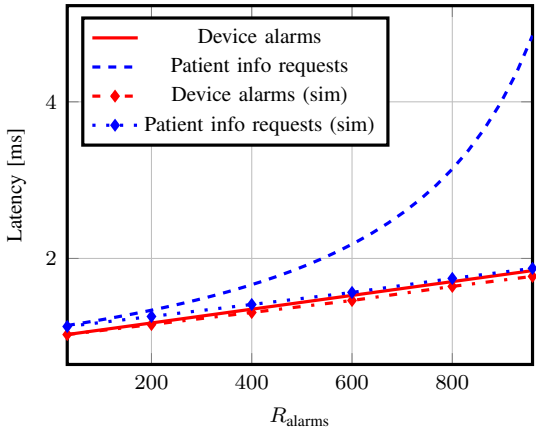


Fig. A.6: End-to-end latency of device alarms and patient info requests for various alarm arrival rates.

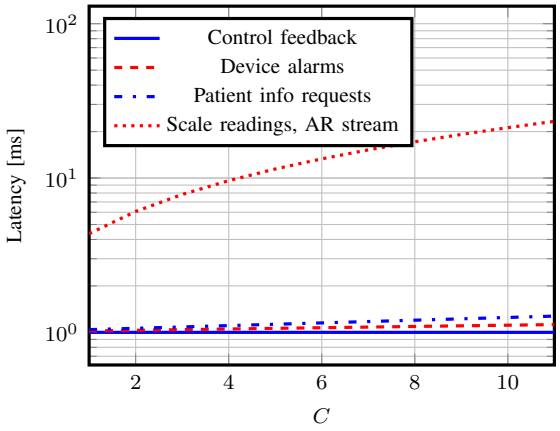


Fig. A.7: End-to-end latency of the applications in the use case using the given prioritization scheme.

choice since it results in high utilization (100%). By comparison, if 32 bytes were allocated in each cycle only to the device alarms, it would on average only be used once every 60 seconds, yielding a utilization of approximately 0.02%, and would in addition occupy 32 bytes more of the frame than the overwriting scheme.

The telegrams that are destined to the cloud need to be forwarded by the gateways to the switched network, where the telegrams are served according to their priority. Obviously, applications that are given high priority influence the latency of the applications with lower priority. This is illustrated in Fig. A.6 for the patient info requests for various values of R_{alarms}. The sim-

ulation results are obtained by simulating the scenario with periodic arrivals under the assumption that the factory units are synchronized, and observing the maximum experienced latency. Due to an increased serialization time, an increasing R_{alarms} causes an increase in the latency experienced by both the device alarms, as evident from (A.10), and the patient info requests, as described by (A.17). Since an increased number of arrivals also results in an increased burst size, the patient info request latency obtained using network calculus increases more than the device alarm latency. Furthermore, in a system with more queuing priorities, the impact of bursts would be amplified at each prioritization queue all the way to the queue with lowest priority. However, as the simulation results show, this tendency does not occur in practice. This is due to the affine approximation of the periodic arrivals used in the network calculus calculations. In particular, the affine approximation assumes that high priority bytes arrive during two cycles, which interrupts the low priority traffic and hence introduces an additional delay. In the case with periodic arrivals, the low priority traffic is served without interruptions from the point where the high priority traffic has been served until the next cycle.

We now consider the question of how the end-to-end latency is affected by the number of factory units in the network, C. Fig. A.7 shows the end-to-end latencies obtained using the network calculus methodology for all applications in the use case, where we assume that the control traffic is transmitted as $4 \cdot 32$ bytes. Notice that the total number of generated bytes exceeds the capacity of the links when C reaches 12, which is why only values of $C < 12$ are considered. The scale readings and the AR stream share the same latency as they share the same priority and follow the same path in the network. As can be seen, the latency requirements for control feedback, device alarms and the patient info requests are satisfied for all the considered values of C. However, the AR stream latency requirement fails when C reaches 10. Although it is expected that the high data rate applications with strict latency requirements will fail first, it also states the limit of deterministic network calculus, since the relatively low reliability requirement of the AR stream has not been exploited. In fact, one could be greedy in this situation and drop AR packets as long as the reliability requirement is satisfied in order to reduce the data rate.

1.6 Conclusion

Industry 4.0 introduces a wide range of new requirements to industrial networks demanded by applications that interact with cloud services. This introduces several challenges in regard to the management and control of the network, and traditional technologies such as SDN are not well suited for

the heterogeneous industrial networks. This paper investigates how network slicing can be used to deal with this complexity by introducing programmability and flexibility to industrial networks. We have presented methods for slicing cyclic and switched industrial communication protocols, and analyzed their trade-offs in utilization, reliability and isolation. Furthermore, we have introduced an Industry 4.0 use case that illustrates how network slicing can be used to handle the diverse requirements. Based on the use case, we have demonstrated how deterministic network calculus can be used to systematically analyze end-to-end latencies of network slices. The presented work can be used to apply existing techniques and algorithms such as queue allocation and admission control to the industrial domain.

References

[1] R. Drath and A. Horch, "Industrie 4.0: Hit or hype?" *IEEE industrial electronics magazine*, vol. 8, no. 2, pp. 56–58, 2014.

[2] N. Jazdi, "Cyber physical systems in the context of industry 4.0," in *IEEE International Conference on Automation, Quality and Testing, Robotics*, 2014, pp. 1–4.

[3] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[4] J. Q. Li, F. R. Yu, G. Deng, C. Luo, Z. Ming, and Q. Yan, "Industrial internet: A survey on the enabling technologies, applications, and challenges," *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1504–1526, 2017.

[5] T. Sauter, "The three generations of field-level networks—evolution and compatibility issues," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 11, pp. 3585–3595, 2010.

[6] 3GPP, "Service requirements for next generation new services and markets," 3rd Generation Partnership Project (3GPP), TS 22.261 v16.0.0, 06 2017.

[7] P. Schulz, M. Matthe, H. Klessig, M. Simsek, G. Fettweis, J. Ansari, S. A. Ashraf, B. Almeroth, J. Voigt, I. Riedel, A. Puschmann, A. Mitschele-Thiel, M. Müller, T. Elste, and M. Windisch, "Latency critical iot applications in 5g: Perspective on the design of radio interface and network architecture," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 70–78, 2017.

[8] E. Molina and E. Jacob, "Software-defined networking in cyber-physical systems: A survey," *Computers & Electrical Engineering*, 2017.

[9] NGMN Alliance. (2015, 2) 5G white paper. Accessed: 22/09/2017. [Online]. Available: http://www.ngmn.org/5g-white-paper.html

[10] S.-H. Leitner and W. Mahnke, "OPC UA–service-oriented architecture for industrial applications," *ABB Corporate Research Center*, 2006.

[11] H. P. Huth and A. M. Houyou, "Resource-aware virtualization for industrial networks: A novel architecture combining resource management, policy control and network virtualization for networks in automation or supervisory control and data acquisition networks," in *2013 International Conference on Data Communication Networking (DCNET)*, 7 2013, pp. 1–7.

[12] W. Mandarawi, A. Fischer, A. M. Houyou, H.-P. Huth, and H. de Meer, *Constraint-Based Virtualization of Industrial Networks*. Springer International Publishing, 2016, pp. 567–586.

[13] Y. W. Ma, Y. C. Chen, and J. L. Chen, "SDN-enabled network virtualization for industry 4.0 based on iots and cloud computing," in *2017 19th International Conference on Advanced Communication Technology (ICACT)*, 2 2017, pp. 199–202.

[14] Y. Yiakoumis, K.-K. Yap, S. Katti, G. Parulkar, and N. McKeown, "Slicing home networks," in *Proceedings of the 2nd ACM SIGCOMM workshop on Home networks*. ACM, 2011, pp. 1–6.

[15] N. Nikaein, E. Schiller, R. Favraud, K. Katsalis, D. Stavropoulos, I. Alyafawi, Z. Zhao, T. Braun, and T. Korakis, "Network store: Exploring slicing in future 5g networks," in *Proceedings of the 10th International Workshop on Mobility in the Evolving Internet Architecture*. ACM, 2015, pp. 8–13.

[16] M. R. Sama, X. An, Q. Wei, and S. Beker, "Reshaping the mobile core network via function decomposition and network slicing for the 5g era," in *Wireless Communications and Networking Conference Workshops (WCNCW), 2016 IEEE*. IEEE, 2016, pp. 90–96.

[17] 3GPP, "Study on architecture for next generation system," 3rd Generation Partnership Project (3GPP), TR 23.799 v14.0.0, 12 2016.

[18] P. Gaj, J. Jasperneite, and M. Felser, "Computer communication within industrial distributed environment—a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 182–189, 2013.

[19] J. Wan, S. Tang, Z. Shu, D. Li, S. Wang, M. Imran, and A. V. Vasilakos, "Software-defined industrial internet of things in the context of industry 4.0," *IEEE Sensors Journal*, vol. 16, no. 20, pp. 7373–7380, 10 2016.

[20] D. Li, M. T. Zhou, P. Zeng, M. Yang, Y. Zhang, and H. Yu, "Green and reliable software-defined industrial networks," *IEEE Communications Magazine*, vol. 54, no. 10, pp. 30–37, 10 2016.

[21] J. W. Guck and W. Kellerer, "Achieving end-to-end real-time quality of service with software defined networking," in *2014 IEEE 3rd International Conference on Cloud Networking (CloudNet)*, 10 2014, pp. 70–76.

[22] J. W. Guck, M. Reisslein, and W. Kellerer, "Function split between delay-constrained routing and resource allocation for centrally managed QoS in industrial networks," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2050–2061, 12 2016.

[23] J. W. Guck, A. V. Bemten, and W. Kellerer, "DetServ: Network models for real-time QoS provisioning in SDN-based industrial environments," *IEEE Transactions on Network and Service Management*, vol. 14, no. 4, pp. 1003–1017, 12 2017.

[24] D. Schulz, "Network models for the industrial intranet," in *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, 2016, pp. 1–10.

[25] K. C. Lee, S. Lee, and M. H. Lee, "Worst case communication delay of real-time industrial switched ethernet with multiple levels," *IEEE Transactions on Industrial Electronics*, vol. 53, no. 5, pp. 1669–1676, 2006.

[26] K. C. Lee and S. Lee, "Performance evaluation of switched ethernet for real-time industrial communications," *Computer standards & interfaces*, vol. 24, no. 5, pp. 411–423, 2002.

[27] T. Sauter and M. Lobashov, "How to access factory floor information using internet technologies and gateways," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 4, pp. 699–712, 2011.

[28] L. L. Bello, E. Bini, and G. Patti, "Priority-driven swapping-based scheduling of aperiodic real-time messages over ethercat networks," *IEEE Transactions on Industrial Informatics*, vol. 11, no. 3, pp. 741–751, 6 2015.

[29] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69–74, 2008.

[30] L. Kleinrock, *Queueing systems, volume 2: Computer applications*. wiley New York, 1976, vol. 66.

[31] M. Fidler and A. Rizk, "A guide to the stochastic network calculus," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 92–105, 2015.

[32] J.-Y. Le Boudec and P. Thiran, *Network calculus: a theory of deterministic queuing systems for the internet*. Springer Science & Business Media, 2001, vol. 2050.

[33] A. Van Bemten and W. Kellerer, "Network calculus: A comprehensive guide," Technische Universität München—Lehrstuhl für Kommunikationsnetze, Tech. Rep. 201603, 03 2016.

[34] IEEE 802.1 Time-Sensitive Networking Task Group. Accessed: 19/05/2017. [Online]. Available: http://www.ieee802.org/1/pages/tsn.html

[35] S. Vassilaras, L. Gkatzikis, N. Liakopoulos, I. N. Stiakogiannakis, M. Qi, L. Shi, L. Liu, M. Debbah, and G. S. Paschos, "The algorithmic aspects of network slicing," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 112–119, 8 2017.

Paper B

Ultra-Reliable Communication for Services with Heterogeneous Latency Requirements

Anders E. Kalør and Petar Popovski

Published in
Proceedings of IEEE Globecom Workshops (GC Wkshps), pp. 1–6, 2019,
doi: 10.1109/GCWkshps45667.2019.9024507.

The layout has been revised.

Abstract

Ultra-reliable communication (URC) is often studied with very strict and homogeneous latency requirements, commonly referred to as ultra-reliable low-latency communication (URLLC). However, in many scenarios the tolerated latencies may vary across users, and treating all users equally may lead to unnecessary over-provisioning of resources. In this paper, we study URC with orthogonal and non-orthogonal access in uplink scenarios where users have heterogeneous latency requirements. Users with strict latency requirements are given resources that are localized in time, while users with less strict latency are given resources that are spread across time and with intermediate feedback. We show that exploiting differences in the tolerated latency can lead to both a significant increase in reliability, and to more efficient use of resources.

2.1 Introduction

Ultra-reliable communication (URC) plays a central role in the support of emerging wireless applications, such as industrial automation, smart grids, and virtual reality, where required packet error rates are in the range of 10^{-9}–10^{-5} [1]. In many cases, URC is blended with strict latency requirements in the order of a few milliseconds, which has led to the introduction of Ultra-Reliable Low-Latency Communication (URLLC), one of the three defining pillars in 5G along with enhanced Mobile Broadband (eMBB) and massive Machine-Type Communication (mMTC) [2]. As a result, URLLC has received much attention during the past years as part of the research activities towards 5G. However, URLLC represents a very demanding regime, as the stringent latency requirement restricts both the degrees of diversity that can be used to ensure high reliability, and prevents the use of intermediate feedback from the receiver [3]. In particular, URLLC requires significant over-provisioning in order to ensure that the transmission will succeed with high probability under various channel conditions, and is expensive in terms of spectral efficiency [4].

However, several URC use cases have less stringent latency requirements than those usually studied under URLLC, such as remote health monitoring and disaster-and-rescue scenarios (see e.g. [1, 5, 6]). This opens the possibilities for using more degrees of diversity and more coordination. Examples could be the ability to spread transmissions across time, acquire channel state information (CSI) to allow for precoding, or to provide intermediate feedback (e.g. stop-feedback) to the transmitter to schedule resources with higher granularity, thus reducing the resource overhead. Common to all of these methods is that they introduce a delay in the communication, and hence cannot be considered for the traditional URLLC use case. Furthermore, it is likely that a single base station will serve applications with heterogeneous

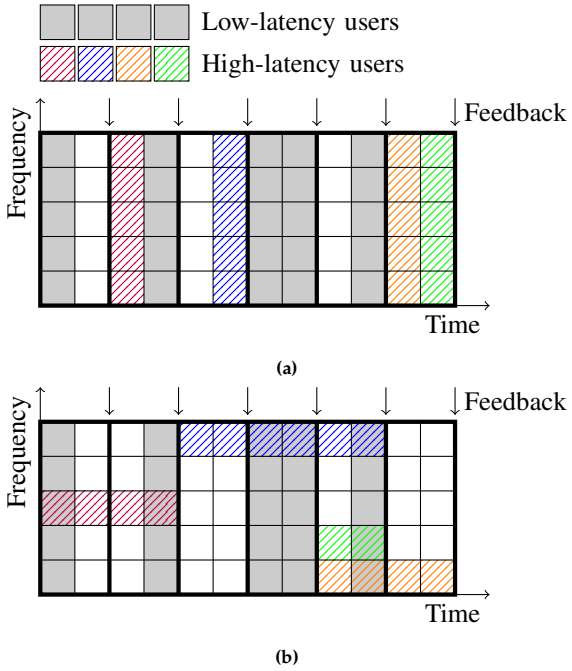


Fig. B.1: Example allocation for (a) Orthogonal access scheme with equal treatment to users with strict (gray) and moderate (hatched) latency requirements (b) Non-orthogonal access scheme with different allocations to the user groups.

latency requirements. Architectures that support heterogeneous services in the same network have been widely studied in the literature through the concept of network slicing [7]. However, most focus has been on the co-existence of eMBB, URLLC and mMTC [8, 9], while the case with diverse latency requirements within the ultra-reliable regime has received less attention. Nevertheless, due to the high over-provisioning required in URLLC regime, it is generally desirable to exploit the additional delay that can be tolerated by some applications.

Motivated by this observation, in this paper we study the scenario in which a base station serves URC devices with heterogeneous, but still moderate latency requirements. We limit the focus to the feedback aspect, i.e. the case where some of the users can tolerate latencies that allow for (short) intermediate feedback during the transmission, while other users cannot. We study various feedback schemes in settings with orthogonal and non-orthogonal access, and quantify the gains in terms of rate and spectral efficiency that can be achieved by exploiting the feedback.

To illustrate the overall idea, consider a wireless interface between a number of ultra-reliable users and a base station, comprising frequency and time

resources as depicted in Fig. B.1. The resources colored in gray are occupied by users that require very low latency, and hence must be localized in time, while the hatched resources are users with less strict latency requirements that can span several time slots. Feedback is given after every second time slot. In Fig. B.1a both user groups are treated equally as URLLC users and multiplexed orthogonally. There is no use of feedback, and hence each user is allotted sufficient (dedicated) resources to cope with potentially bad channel conditions in order to ensure high reliability. On the other hand, Fig. B.1b illustrates the idea of multiplexing the users that can tolerate higher latency non-orthogonally with the low-latency users. Furthermore, the base station transmits a stop-feedback signal as soon as the transmission has completed, so as to limit the resource overhead and the interference to the low-latency users. In addition, due to the non-orthogonal multiplexing, the total number of users that can be supported is larger, and the time diversity allows for spreading the interference across multiple low-latency users, thereby gaining diversity with respect to interference. In summary, the scenario in Fig. B.1b has both better utilization of resources and more diversity, thus facilitating the ability to provide high reliability.

The remainder of the paper is organized as follows. Section 2.2 presents the system model, and in Section 2.3 we present the different scheduling and feedback schemes that we consider. The schemes are evaluated in Section 2.4 and finally the paper is concluded in Section 2.5.

2.2 System model

We consider the uplink in a system comprising N users and a single base station. The air interface is divided into time slots, and each time slot is further divided into S minislots and F frequency channels, as illustrated in Fig. B.2. One frequency channel and one time slot constitute a radio resource, which represents the minimum granularity of the scheduler and is assumed to be within the time/frequency coherence interval of the channel. Due to the latency requirements, which are relatively strict for all applications that we consider, we assume that the transmitters have no information about the channel, while the base station has full CSI, acquired through an idealized estimation process during the URC transmission. To satisfy the strict reliability requirements, we assume that the users are pre-assigned radio resources, so that the interference can be controlled. Each user accesses its assigned resources in a grant-free manner and is active with probability p.

We consider two groups of users which have identical reliability requirements but different latency requirements. The common reliability requirement is given in terms of a minimal probability, ϵ, of successful transmission within their tolerated latency. Regarding the latency requirements, the users

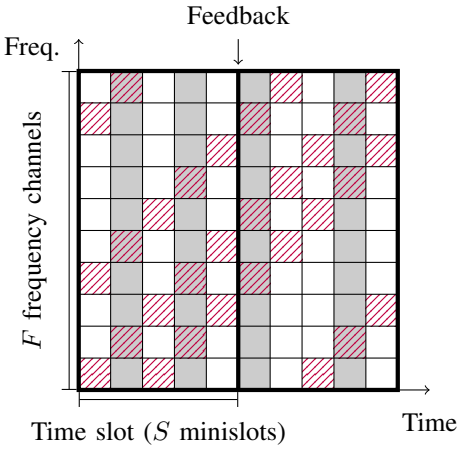


Fig. B.2: Frame structure considered in the system model, illustrated in a non-orthogonal multiple access scenario. URC-LL users (solid gray cells) transmit in a single minislot across several frequency channels, while URC-HL users (hatched cells) transmit across several time slots, here in a diagonal pattern. The base station provides feedback after each time slot.

in the first group, which we refer to as low-latency users (URC-LL), have very strict latency requirements and must transmit within a single minislot. On the other hand, the users in the second group have less stringent latency requirements, and need to finish within two time slots ($2S$ minislots). We refer to the second group as high-latency users (URC-HL). The fundamental difference between the two user groups is that the base station can provide feedback to the users in second group between the two time slots. The feedback schemes, which are assumed instantaneous and error free, are outlined in detail in the next section.

We study both orthogonal and non-orthogonal transmissions, and assume a Rayleigh block fading channel. We further assume that the users employ frequency hopping between the minislots so that they experience independent channel coefficients. Denoting the symbols transmitted by user n in frequency channel f and minislot s as $\mathbf{X}_{s,f}^{(n)}$ the signal received at the base station in slot s, f reads

$$\mathbf{Y}_{s,f} = \sum_{n=1}^{N} \delta_{s,f}^{(n)} H_{s,f}^{(n)} \mathbf{X}_{s,f}^{(n)} + \mathbf{Z}_{s,f}, \tag{B.1}$$

where $\delta_{s,f}^{(n)}$ is a Bernoulli random variable indicating whether user n is active in the slot, $H_{s,f}^{(n)} \sim \mathcal{CN}(0, \Gamma)$ is the channel coefficient of user n in the slot, and $\mathbf{Z}_{s,f} \sim \mathcal{CN}(0, \mathbf{I})$ is additive noise.

We remark that it may be beneficial to multiplex the URC users non-

orthogonally with eMBB as discussed in [8]. However, since we are only concerned about the URC use case, we assume that the resources are dedicated to URC and note that superimposing eMBB traffic would merely add uncorrelated interference to the URC users.

To quantify the performance of the schemes we define the following metrics. First, we consider the maximum per-user rate, denoted by r_{LL} and r_{HL} for URC-LL and URC-HL, respectively, that provide a certain reliability ϵ. To indicate the utilization of the resources, we introduce the ratio between the average rate supported by the channel, $\mathbb{E}[R]$, and the maximum rate, C_ϵ, required to satisfy the reliability requirement ϵ for the respective scheme. Mathematically, this is expressed as

$$\frac{\mathbb{E}[R]}{C_\epsilon} = \frac{\mathbb{E}[R]}{\sup\{r \mid \Pr(E) \leq 1-\epsilon\}}, \tag{B.2}$$

where $\Pr(E)$ is the probability of error. In general, it is desirable for the number to be small, as this reflects a small resource overhead. Notice that the ratio can be less than one if the rate distribution is asymmetric. Although the data packets are small, the finite blocklength effects are known to have little impact on the outage capacity [10], and thus we study the scenario in the infinite blocklength regime.

2.3 Scheduling and feedback schemes

We study a total of three transmission policies as illustrated for two URC-LL and two URC-HL users in Fig. B.3. The first is orthogonal access without feedback, which reflects the situation of treating both user groups equivalently according to the most strict requirements. The remaining two policies are based on non-orthogonal access; one without feedback, and one with stop-feedback.

Since we consider the per-user error probability, and the users are assumed to use frequency hopping so that they experience independent channel realizations, we omit the dependency on the slot and user in the channel coefficients. Instead, we denote them by H_k where k indicates the resource index (frequency and time). When necessary, we distinguish between URC-LL and URC-HL users using the subscripts LL and HL, e.g. $H_{\mathrm{LL},k}$, $H_{\mathrm{HL},k}$.

2.3.1 Orthogonal access without feedback

The orthogonal access scheme reflects the standard grant-free transmission, in which users are assigned dedicated resources that they access if they have data to transmit. Furthermore, in line with the majority of the current research, no distinction is made between the low-latency and high-latency

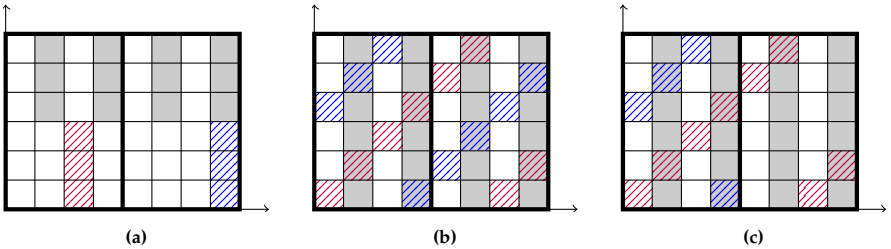


Fig. B.3: The scheduling and feedback schemes that we consider, illustrated with four URC-LL users (solid gray) and two URC-HL users (hatched): (a) Orthogonal access without feedback and (b–c) non-orthogonal access. (b) is without feedback, (c) with stop-feedback.

users. The situation is illustrated in Fig. B.3a, where both URC-LL and URC-HL users are assigned 3 frequency slots within the same minislot.

We denote the number of frequency resources assigned to each user by K. The reliability of each user is then given by

$$\Pr(E) = \Pr\left[\frac{1}{K}\sum_{k=1}^{K}\log_2\left(1+|H_k|^2\right) < r\right], \tag{B.3}$$

where r is the transmission rate and $|H_k|^2$ are independent exponentially distributed random variables with mean Γ.

For a given reliability requirement ϵ, the rate r can be determined using Monte Carlo simulation by setting Eq. (B.3) equal to ϵ. However, this can be difficult to compute for small ϵ, which are usually of interest for URC. As an alternative, the rate may instead be selected conservatively using the bound derived in Appendix A as

$$r = \max_{t>0}\frac{1}{tK}\log_2(\epsilon) - \frac{1}{t}\log_2\left(\mathbb{E}\left[(1+|H_k|^2)^{-t}\right]\right), \tag{B.4}$$

where the expectation can be calculated using Monte Carlo simulation and $t > 0$ can be optimized to maximize the rate.

2.3.2 Non-orthogonal without feedback

We now turn our attention to non-orthogonal allocations. The motivation for this is twofold. First, non-orthogonal access is beneficial when the access probability p is relatively low, since the probability of unused resources is lower. Secondly, the non-orthogonality allows for higher frequency and time diversity gain, as each resource can serve multiple users. We first consider the case without feedback. For simplicity, we assume that each resource is allocated to one URC-LL and one URC-HL user, as shown in Fig. B.3b. However, the resources could as well be shared among users of the same

group, which is likely to be beneficial if the activation probability is low. Although Fig. B.3b illustrates a diagonal frequency hopping pattern for the URC-HL users, the analysis is valid as long as each channel resource is not shared by multiple URC-HL users, and each frequency channel is used at most once by the same URC-HL user within a time slot.

We assume that the base station initially attempts to decode the URC-LL users while treating the URC-HL users as noise, since URC-LL users receive all their channel resources within one minislot. The URC-LL users that are successfully decoded are subsequently cancelled and the URC-HL users are decoded. Notice that even if the decoding of a URC-LL user fails it may still be possible for the base station to decode the URC-HL users by treating the URC-LL interference as noise. Denoting by $\delta_{\mathrm{HL},k}$ the Bernoulli random variable indicating whether the URC-HL assigned to resource k in the current minislot is active, the error probability of the URC-LL user is

$$\Pr\left(E_{\mathrm{LL}}\right) = \Pr\left[\frac{1}{K_{\mathrm{LL}}}\sum_{k=1}^{K_{\mathrm{LL}}}\log_2\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right) < r_{\mathrm{LL}}\right], \tag{B.5}$$

where the subscripts LL and HL are used to distinguish between the URC-LL and URC-HL users, respectively. As in the orthogonal case, the rate can be selected according to the bound in Appendix A as

$$r_{\mathrm{LL}} = \max_{t>0}\ \frac{1}{tK_{\mathrm{LL}}}\log_2\left(\epsilon\right) - \frac{1}{t}\log_2\left(\mathbb{E}\left[\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right)^{-t}\right]\right). \tag{B.6}$$

Due to the interference cancellation procedure, the URC-HL users can experience three scenarios in each resource: (i) The URC-LL user is not active, (ii) the URC-LL user is active, successfully decoded and cancelled, and (iii) the URC-LL user is active but not decoded. The resulting error probability can be written

$$\Pr\left(E_{\mathrm{HL}}\right) = \Pr\left[\frac{1}{K_{\mathrm{HL}}}\sum_{k=1}^{K_{\mathrm{HL}}}\log_2\left(1+\frac{|H_{\mathrm{HL},k}|^2}{1+\delta_{\mathrm{LL},k}(1-\gamma_{\mathrm{LL},k})|H_{\mathrm{LL},k}|^2}\right) < r_{\mathrm{HL}}\right], \tag{B.7}$$

where $\delta_{\mathrm{LL},k}$ is the binary variable indicating whether the URC-LL user is active in resource k, and $\gamma_{\mathrm{LL},k} = 1$ if the URC-LL user is successfully decoded, otherwise $\gamma_{\mathrm{LL},k} = 0$. Notice that even though the URC-LL and URC-HL users have the same reliability requirements, due to the interference cancellation procedure it is not optimal to select $r_{\mathrm{LL}} = r_{\mathrm{HL}}$ and $K_{\mathrm{LL}} = K_{\mathrm{HL}}$. Instead, the rates depend on both the activation probability of the interfering users, as well as the number of resources assigned to URC-LL and URC-HL. For this reason, deriving bounds on the rate for URC-HL is challenging. However, a valid bound can be obtained by assuming that the URC-HL user is decoded

first, while treating the URC-LL transmissions as interference. This gives the rate

$$r_{\mathrm{HL}} = \max_{t>0} \frac{1}{tK_{\mathrm{HL}}} \log_2(\epsilon) - \frac{1}{t} \log_2 \left(\mathbb{E}\left[\left(1 + \frac{|H_{\mathrm{HL},k}|^2}{1 + \delta_{\mathrm{LL},k} |H_{\mathrm{LL},k}|^2} \right)^{-t} \right] \right). \tag{B.8}$$

2.3.3 Non-orthogonal with stop-feedback

The non-orthogonal scenario with stop-feedback is equivalent to the previous case without feedback, with the addition of a feedback signal from the base station after the initial time slot, that indicates to the URC-HL users whether their transmission has completed successfully. Consequently, the URC-LL users may experience less interference if a URC-HL user succeeds already within the first time slot (see Fig. B.3c). This in turn slightly increases the reliability of URC-LL. We denote the number of resources given in the first and second time slots by $K_{\mathrm{HL}}^{(1)}$ and $K_{\mathrm{HL}}^{(2)}$, respectively, so that $K_{\mathrm{HL}}^{(1)} + K_{\mathrm{HL}}^{(2)} = K_{\mathrm{HL}}$. The probability that a URC-HL user succeeds after the first time slot is

$$\Pr\left(E_{\mathrm{HL}}^{(1)}\right) = \Pr\left[\frac{1}{K_{\mathrm{HL}}} \sum_{k=1}^{K_{\mathrm{HL}}^{(1)}} \log_2 \left(1 + \frac{|H_{\mathrm{HL},k}|^2}{1 + \delta_{\mathrm{LL},k}(1 - \gamma \mathrm{LL}, k) |H_{\mathrm{LL},k}|^2} \right) < r_{\mathrm{HL}} \right]. \tag{B.9}$$

As a result, the expected number of resources allocated to a URC-HL user is $K_{\mathrm{HL}}^{(1)} + \Pr(E_{\mathrm{HL}}^{(1)}) K_{\mathrm{HL}}^{(2)}$. This reflects the central advantage of feedback, namely a higher granularity in the resource assignment, which in turn results in less resource overhead.

While the rates for URC-HL users are the same as in the case without feedback, the URC-LL users can support slightly higher rate. However, including this into the calculation of the bounds is challenging due to the dependence between the interference experienced by the URC-LL users and the rate of the URC-LL users. Hence, we will resort to using the same bounds as in the case without feedback for both URC-HL and URC-LL users.

2.4 Numerical evaluation

In this section we present numerical results to illustrate the reliabilities under the schemes described in the previous section. Since the dependency between successful decoding of URC-LL and URC-HL render the error probability calculations difficult, we approximate the results using Monte Carlo simulations.

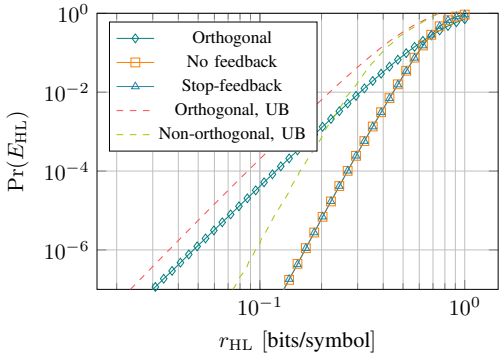


Fig. B.4: Transmission error probabilities for various rates of the URC-HL users.

We consider a scenario of a total of $N = 20$ users, divided into 10 URC-LL and 10 URC-HL users. The air interface contains $F = 10$ frequency channels and each time slot is comprised of $S = 5$ minislots. The activation probability is $p = 0.5$ and the channel gains are normalized to $\Gamma = 1$. In the case of orthogonal access, each user is assigned 5 frequency resources so that the users occupy a total of two time slots. In the non-orthogonal schemes, each user is given 10 resources. More specifically, the URC-LL users are each allocated a dedicated time slot, i.e. $K_{\mathrm{UL}} = 10$, while each URC-HL user is assigned one frequency resource in each minislot, so that their resources are equally divided between the two time slots i.e. $K_{\mathrm{HL}}^{(1)} = K_{\mathrm{HL}}^{(2)} = 5$.

The error probabilities for the various schemes and the rate bounds are shown for URC-HL and URC-LL users in Figs. B.4 and B.5, respectively. In both cases, the non-orthogonal schemes support higher rates than the orthogonal in the high-reliability region. This indicates that despite the interference caused by non-orthogonality, the fact that twice as many resources can be assigned to each user results in higher reliability. For URC-HL, the scheme without feedback and the scheme with stop-feedback result in the same error probabilities, as stop-feedback only impacts the URC-HL users that have successful transmissions. However, in the case with URC-LL users the stop-feedback scheme result in higher reliability than the other schemes due to the reduced interference experienced in the second time slot.

The utilization of the schemes are shown for URC-HL in Fig. B.6. Again, the two non-orthogonal schemes outperform the orthogonal due to the higher number of resources and hence improved average channel conditions. For large target error probabilities, ϵ_{HL}, the non-orthogonal schemes perform equivalently and the utilization ratio tends towards zero. However, as the target reliability increases, the stop-feedback scheme becomes more efficient as less users are given excess resources.

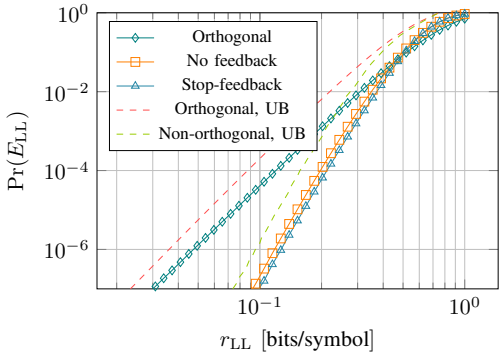


Fig. B.5: Transmission error probabilities for various rates of the URC-LL users.

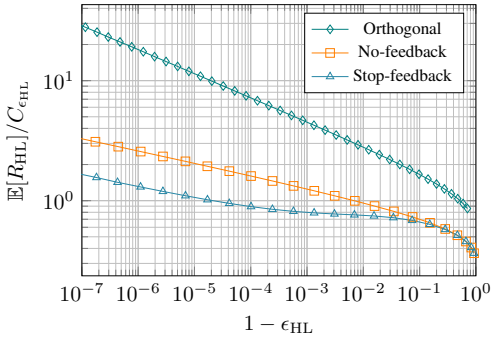


Fig. B.6: Relative average given rates for various target error probabilities for URC-HL.

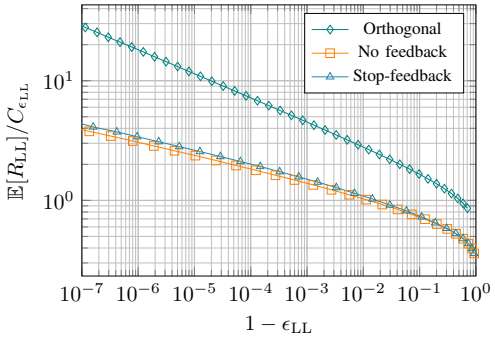


Fig. B.7: Relative average given rates for various target error probabilities for URC-LL.

The case with URC-LL is shown in Fig. B.7, where it can be seen that due to the lack of feedback, all schemes exhibit high overhead in the high reliability region. While the overhead for the orthogonal scheme is largest, the stop-feedback scheme is slightly higher than the one without feedback. This indicates that, despite users in this scheme can transmit with higher rate to achieve a certain ϵ_{LL} (see Fig. B.5), the average rate overhead is even larger.

2.5 Conclusion

In this paper we have investigated ultra-reliable communication in a scenario where some users require very low latency, while others can tolerate higher latencies. We have studied how the increased time diversity and the use of intermediate feedback to the high-latency users can help supporting ultra-reliable communication with both orthogonal and non-orthogonal access. More specifically, we show that the ability to use stop-feedback can lead to higher reliability and better resource utilization. Furthermore, even without feedback non-orthogonal access outperforms orthogonal access in the ultra-reliable regime both in terms of reliability and resource efficiency due to increased time, frequency and interference diversity gains. This suggests that adapting the use of the radio resources to the latency requirements is highly beneficial in the ultra-reliable regime.

Future research can be in the direction of studying the use of power control as well as more sophisticated feedback schemes that exploit the information that the base station has obtained during the initial time slot, such as channel estimations. Furthermore, the model could be generalized e.g. to allow for more general resource allocations, heterogeneous reliability requirements, and eMBB traffic.

A Derivation of lower bound on rate

We derive the lower bounds on the rate for the non-orthogonal case in Eq. (B.5), from which the orthogonal access can be obtained by setting $\delta_{\mathrm{LL},k} = 0$. By fixing the error probability as $\Pr(E_{\mathrm{LL}}) = \epsilon_{\mathrm{LL}}$ and by following

the same procedure as in [8] we obtain

$$\epsilon_{\mathrm{LL}} = \Pr\left[\sum_{k=1}^{K_{\mathrm{LL}}} \log_2\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right) < K_{\mathrm{LL}} r_{\mathrm{LL}}\right] \tag{B.10}$$

$$= \Pr\left[-t\log_2\left(\prod_{k=1}^{K_{\mathrm{LL}}}\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right)\right) < -K_{\mathrm{LL}} r_{\mathrm{LL}} t\right] \tag{B.11}$$

$$= \Pr\left[\prod_{k=1}^{K_{\mathrm{LL}}}\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right)^{-t} < 2^{-K_{\mathrm{LL}} r_{\mathrm{LL}} t}\right] \tag{B.12}$$

$$\leq \frac{\mathbb{E}\left[\prod_{k=1}^{K_{\mathrm{LL}}}\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right)^{-t}\right]}{2^{-K_{\mathrm{LL}} r_{\mathrm{LL}} t}} \tag{B.13}$$

$$= \frac{\mathbb{E}\left[\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right)^{-t}\right]^{K_{\mathrm{LL}}}}{2^{-K_{\mathrm{LL}} r_{\mathrm{LL}} t}}. \tag{B.14}$$

Here, (B.11) is obtained by moving the summands inside the logarithm, and then multiplying both sides of the inequality by $-t$. In (B.12) we have raised both sides to the power of two, and (B.13) follows from the Markov inequality. Using the fact that the terms inside the expectation are independent and identically distributed we arrive at the expression in (B.14). The lower bound on the rate for a given ϵ_{LL} can then be obtained by rewriting the expression as

$$r_{\mathrm{LL}} \geq \frac{1}{tK_{\mathrm{LL}}}\log_2(\epsilon_{\mathrm{LL}}) - \frac{1}{t}\log_2\left(\mathbb{E}\left[\left(1+\frac{|H_{\mathrm{LL},k}|^2}{1+\delta_{\mathrm{HL},k}|H_{\mathrm{HL},k}|^2}\right)^{-t}\right]\right), \tag{B.15}$$

where the expectation can be approximated using Monte Carlo simulation and $t > 0$ can be optimized so as to maximize the rate.

Acknowledgment

This work has been in part supported the European Research Council (ERC) under the European Union Horizon 2020 research and innovation program (ERC Consolidator Grant Nr. 648382 WILLOW) and Danish Council for Independent Research (Grant Nr. 8022-00284B SEMIOTIC)

References

[1] P. Popovski, J. J. Nielsen, Č. Stefanović, E. d. Carvalho, E. Ström, K. F. Trillingsgaard, A. Bana, D. M. Kim, R. Kotaba, J. Park, and R. B. Sørensen, "Wireless

access for ultra-reliable low-latency communication: Principles and building blocks," *IEEE Network*, vol. 32, no. 2, pp. 16–23, Mar. 2018.

[2] ITU-R, *M.2410-0—Minimum requirements related to technical performance for IMT-2020 radio interface(s)*, Feb. 2017.

[3] H. Chen, R. Abbas, P. Cheng, M. Shirvanimoghaddam, W. Hardjawana, W. Bao, Y. Li, and B. Vucetic, "Ultra-reliable low latency cellular networks: Use cases, challenges and approaches," *IEEE Communications Magazine*, vol. 56, no. 12, pp. 119–125, Dec. 2018.

[4] J. Sachs, G. Wikström, T. Dudda, R. Baldemair, and K. Kittichokechai, "5G radio network design for ultra-reliable low-latency communication," *IEEE Network*, vol. 32, no. 2, pp. 24–31, Mar. 2018.

[5] P. Popovski, "Ultra-reliable communication in 5G wireless systems," in *Proc. IEEE Int. Conf. 5G Ubiq. Connect.*, Nov. 2014, pp. 146–151.

[6] J. Åkerberg, M. Gidlund, and M. Björkman, "Future research challenges in wireless sensor and actuator networks targeting industrial automation," in *Proc. IEEE INDIN*, Jul. 2011, pp. 410–415.

[7] H. Zhang, N. Liu, X. Chu, K. Long, A. Aghvami, and V. C. M. Leung, "Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 138–145, Aug. 2017.

[8] P. Popovski, K. F. Trillingsgaard, O. Simeone, and G. Durisi, "5G wireless network slicing for eMBB, URLLC, and mMTC: A communication-theoretic view," *IEEE Access*, vol. 6, pp. 55 765–55 779, 2018.

[9] A. Anand, G. De Veciana, and S. Shakkottai, "Joint scheduling of URLLC and eMBB traffic in 5G wireless networks," in *Proc. IEEE INFOCOM*, Apr. 2018, pp. 1970–1978.

[10] W. Yang, G. Durisi, T. Koch, and Y. Polyanskiy, "Quasi-static SIMO fading channels at finite blocklength," in *Proc. IEEE ISIT*, Jul. 2013, pp. 1531–1535.

Paper C

Prediction of mmWave/THz Link Blockages Through Meta-Learning and Recurrent Neural Networks

Anders E. Kalør, Osvaldo Simeone, and Petar Popovski

Published in
IEEE Wireless Communications Letters, vol. 10, no. 12, pp. 2815–2819, 2021,
doi: 10.1109/LWC.2021.3118269.

The layout has been revised.

Abstract

Wireless applications that rely on links that offer high reliability depend critically on the capability of the system to predict link quality within a given time interval. This dependence is especially acute at the high carrier frequencies used by mmWave and THz systems, where the links are susceptible to blockages. Predicting blockages with high reliability requires a large number of data samples to train effective machine learning modules. With the aim of mitigating data requirements, we introduce a framework based on meta-learning*, whereby data from distinct deployments are leveraged to optimize a shared initialization that decreases the data set size necessary for any new deployment. Predictors of two different events are studied: (1) at least one blockage occurs in a time window, and (2) the link is blocked for the entire time window. The results show that an RNN-based predictor trained using meta-learning is able to predict blockages after observing fewer samples than predictors trained using standard methods.*

3.1 Introduction

Due to its extreme bandwidth and delivery of high rates, highly directional millimeter wave (mmWave) and THz communication links are attractive options for many wireless applications including virtual reality (VR) [1] and Industrial Internet-of-Things (IIoT) [2]. However, the susceptibility of directional links to blockages makes it challenging to deploy the technology for time-sensitive applications with high reliability requirements. For instance, many IIoT applications, such as motion control systems, rely on frequent periodic transmissions of short packets that must be delivered with low latency (sub-millisecond to few milliseconds) and high reliability (up to $1 - 10^{-8}$) [3]. Moreover, the system must be down only for a short duration known as the *survival time*. The impact of blockages needs to be reduced by implementing reactive or preventive mechanisms in the communication system, such as searching for alternative communication paths or adopting a more robust control strategy to avoiding unnecessary system shutdowns. While reactive mechanisms bring a detection delay, preventive measures, such as beam training, incur high overhead as they need to be repeated at reoccurring intervals.

To achieve both low latency and low overhead in applications such as VR and IIoT, recent studies have considered the problem of *blockage prediction* with the aim of predicting when blockages will occur, and to proactively initiate countermeasures. Due to the lack of mathematical models for blockages, the predictors are often based on machine learning models, which use metrics from the communication link as well as external features such as visual and location information [4]. A range of works have applied neural networks to the problem of using sub-6 GHz channels to predict blockages

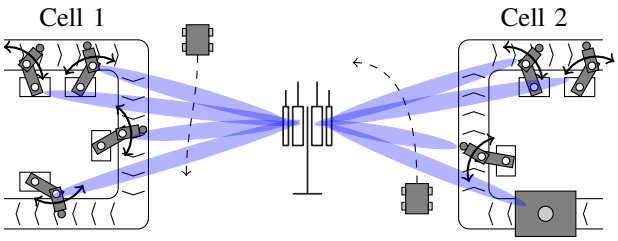


Fig. C.1: An example scenario comprising two manufacturing cells with robots that are connected to a base station through mmWave links. The mmWave links are subject to blockages from the robotic arms, as well as by moving vehicles.

in the mmWave bands [5–7]. The use of Recurrent Neural Networks (RNNs) to predict beam blockages in scenarios with user mobility has been studied in [8–11], which learn the spatial and/or temporal correlation of blockages. Furthermore, visual features has been shown to increase the accuracy of such predictors [6, 12]. Other works predict blockages using diffraction effects, etc. [13], and [14] uses survival analysis to predict the probability of experiencing a blockage based on a recent observation window.

One of the main challenges associated with machine learning predictors is that they rely on large amounts of training samples from their specific deployment in order to accurately learn the dynamics of the blockages. In this paper, we address this problem using *meta-learning*, which has previously been applied successfully to wireless systems [15]. Meta-learning allows one to exploit observations from a set of deployments to optimize the *inductive bias*, allowing predictors, such as a neural networks, to be trained for a new deployment using much fewer samples than would otherwise be required.

As an example, consider an indoor industrial scenario as depicted in Fig. C.1, where blockages are predominantly due to the movement of agents such as robots and conveyor belts. Although the blockages are experienced differently in various cells, they are likely to share the same characteristics, e.g., to have similar duration and attenuation. Capturing these characteristics using a shared inductive bias, allows us to accelerate the training of predictors for new cells.

In order to account for the survival time or other latency constraints, we consider the problem of predicting whether a link will be blocked within a given time window. Two types of events are predicted: (i) occurrence of at least one blockage within the time window, and (ii) blockage through the entire time window. These predictors are motivated by different applications, as (i) is suitable for applications that are intolerant to any packet errors and (ii) is appropriate if the application is sensitive to consecutive packet losses.

3.2 System Model and Problem Definition

We consider a wireless system comprising a generic manufacturing cell with K spatially co-located devices, see Fig. C.1. Within the cell, communication is organized in periodic cycles that involve transmissions between each of the K devices and a base station (BS) through a high-rate mmWave link. The link model uses block fading with memory introduced by the physical environment through blockages [16].

The cell environment is described by a random variable π, which is assumed to be drawn independently from an unknown distribution $\pi \sim p(\pi)$. For example, in the evaluation scenario described in Section 3.4.1 and depicted in Fig. C.2a, we will use π to describe parameters such as the location of the devices and the number and speed of blockage objects.

In the cell characterized by variables π, the signals received by the BS from device k, or vice versa, in cycle t are

$$\mathbf{y}_{k,t} = h_{k,t}\mathbf{x}_{k,t} + \mathbf{w}_{k,t}, \tag{C.1}$$

where $\mathbf{x}_{k,t}$ is the packet transmitted with normalized power $E[\|\mathbf{x}_{k,t}\|_2^2] = 1$ (the dimension is not relevant for our discussion), $\mathbf{w}_{k,t} \sim \mathcal{CN}(\mathbf{0}, \mathbf{I})$ is additive white Gaussian noise, and $h_{k,t}$ is the scalar fading coefficient. The fading coefficients $h_{k,t}$ are assumed to be realizations from a random process $h_{k,1:t} = h_{k,1}, h_{k,2}, \ldots, h_{k,t}$ that depends on π. The channel processes are generally statistically dependent across time steps and devices in the cell. Specifically, the K fading processes $\mathbf{h}_{1:t} = (h_{1,1:t}, \ldots, h_{K,1:t})$ are characterized by an unknown joint distribution $p(\mathbf{h}_{1:t}|\pi)$.

We assume that the application requirements of the communication link can be satisfied as long as the instantaneous SNR, $\gamma_{k,t} = |h_{k,t}|^2$, is above a certain threshold γ_0, which is assumed to be the same for all devices. Conversely, communication fails when $\gamma_{k,t} \leq \gamma_0$.

Our goal is to predict future blockages between the BS and a device indexed by k based on previous observations of the channel and, possibly, on side information. The sequence of observations is denoted $\mathbf{o}_{1:t}$, where each observation vector $\mathbf{o}_t$ may contain, among others, SNR estimates, device locations, and images captured by cameras [6, 12], at time steps $1, 2, \ldots, t$. All observations are available at the BS, which runs the predictive models and takes actions upon a predicted blockage. While our method is general, we showcase the potential of the approach by predicting at time t the probability that either *any* or *all* of the slots in a time window $[t + \xi, t + \xi + \tau]$ for some time lag $\xi \geq 0$ are blocked. We refer to τ as the *prediction interval* and ξ as the *prediction delay*. The values τ and ξ can emerge from timing constraints within the system: For example, if a blockage is predicted, then a controller can switch to a more robust control strategy or a semi-autonomous opera-

tion. We shall denote the collection of prediction statistics for the K devices by $\mathbf{z}_t = (z_{1,t}, \ldots, z_{K,t})$.

The ideal *any*-predictor is given by the posterior probability

$$f_k^{\cup}(\mathbf{o}_{1:t}) = \Pr\left(\{\gamma_{k,t+\xi+1} \leq \gamma_0\} \cup \cdots \cup \{\gamma_{k,t+\xi+\tau} \leq \gamma_0\} \mid \mathbf{o}_{1:t}, \pi\right) \tag{C.2}$$

$$= \Pr\left(z_{k,t}^{\cup} = 1 \mid \mathbf{o}_{1:t}, \pi\right), \tag{C.3}$$

for $k = 1, \ldots, K$ where, for brevity, we introduced the outage variable $z_{k,t}^{\cup} = \mathbb{1}[\{\gamma_{k,t+\xi+1} \leq \gamma_0\} \cup \cdots \cup \{\gamma_{k,t+\xi+\tau} \leq \gamma_0\}]$.

The ideal *all*-predictor is similar, but with the intersection of events instead of the union

$$f_k^{\cap}(\mathbf{o}_{1:t}) = \Pr\left(\{\gamma_{k,t+\xi+1} \leq \gamma_0\} \cap \cdots \cap \{\gamma_{k,t+\xi+\tau} \leq \gamma_0\} \mid \mathbf{o}_{1:t}, \pi\right) \tag{C.4}$$

$$= \Pr\left(z_{k,t}^{\cap} = 1 \mid \mathbf{o}_{1:t}, \pi\right), \tag{C.5}$$

where the outage variable $z_{k,t}^{\cap}$ is similarly defined. Compared to the *any*-predictor, the *all*-predictor is related to the survival time of the system and is useful for systems that are sensitive to consecutive errors. To simplify the notation, we will omit the union and intersection symbols from the notation when the results are valid for both cases.

The observations $\mathbf{o}_{1:t}$ and the prediction statistics $\mathbf{z}_{1:t}$ are random variables drawn from an unknown conditional distribution $p(\mathbf{z}_{1:t}, \mathbf{o}_{1:t}|\pi)$. To enable learning of the unknown predictors (C.3) and (C.5), we choose as part of the inductive bias a model class of parametric functions, such as neural networks, and we assume the availability of a dataset of historical observations and target variables from several factory cells (i.e., realizations of π). Furthermore, to exploit the dependency between different cells, as defined by the unknown distribution $p(\pi)$, we frame the setting as a *meta-learning* problem. To be aligned with the meta-learning literature, we refer to the set of historical observations from a single cell as a *task*.

We assume that a *meta-training dataset* $\mathcal{D} = \{\mathcal{D}_1, \ldots, \mathcal{D}_N\}$ is available comprising sequences of length T_{train} from N meta-training tasks $\mathcal{D}_n = (\mathbf{o}_{1:T_{\text{train}}}^{(n)}, \mathbf{z}_{1:T_{\text{train}}}^{(n)})$. Each meta-training task corresponds to a different factory cell. The meta-training dataset is assumed to be available offline to optimize a training procedure that enables effective refinement based on short training sequences. Following the meta-learning literature, we refer to the new task as *meta-test* task, and denote the training set of a meta-test task n' by $\mathcal{D}_{n'} = (\mathbf{o}_{1:T_{\text{test}}}^{(n')}, \mathbf{z}_{1:T_{\text{test}}}^{(n')})$.

3.3 Meta-Learning Predictor

To capture the memory in the fading process, we approximate the ideal posterior distribution $f_k(\mathbf{o}_{1:t})$ in Eqs. (C.3) and (C.5) via RNNs. We specifically consider RNNs composed of a number of input layers followed by recurrent layers and a number output layers. In general, the parameterized RNN based predictive functions can be written as

$$f_{\varphi_k}(\mathbf{o}_{1:t}) = \sigma\left(g_{\varphi_k}^{(\text{out})}\left(g_{\varphi_k}^{(\text{rec})}\left(g_{\varphi_k}^{(\text{in})}\left(\mathbf{o}_t\right), \mathbf{s}_{k,t-1}\right)\right)\right), \tag{C.6}$$

$$\mathbf{s}_{k,t} = \eta_{\varphi_k}\left(g_{\varphi_k}^{(\text{in})}\left(\mathbf{o}_t\right), \mathbf{s}_{k,t-1}\right), \tag{C.7}$$

where $g_{\varphi_k}^{(\text{in})}(\cdot), g_{\varphi_k}^{(\text{rec})}(\cdot), g_{\varphi_k}^{(\text{out})}(\cdot)$ are the the input, recurrent and output layers, respectively; $\sigma(x) = 1/(1+e^{-x})$ is the sigmoid function; and $\mathbf{s}_{k,t} \in \mathbb{R}^d$ represents the internal state of the recurrent layer. Common RNN models the Long Short-Term Memory (LSTM) and the Gated Recurrent Unit (GRU) [17].

As a measure of prediction accuracy, we consider the *weighted binary cross entropy* (BCE) loss function

$$\ell(z, x) = -wz\log(x) - (1-z)\log(1-x), \tag{C.8}$$

where positive examples are multiplied by the constant $w > 0$ to compensate for a potentially imbalanced number of positives and negatives [18]. The population loss for a given task characterized by variables π is then

$$L(\boldsymbol{\varphi}) = E_{p(\mathbf{z}_{1:t}, \mathbf{o}_{1:t}|\pi)}\left[\frac{1}{K}\sum_{k=1}^{K}\ell(z_{k,t}, f_{\varphi_k}(\mathbf{o}_{1:t}))\right], \tag{C.9}$$

where the expectation is over the unknown distribution $p(\mathbf{z}_{1:t}, \mathbf{o}_{1:t}|\pi)$ and $\boldsymbol{\varphi} = (\varphi_1, \ldots, \varphi_K)$ are the model parameters.

The population loss $L(\boldsymbol{\varphi})$ is unknown and is in practice replaced by an empirical estimate obtained from data. Because the function parameters φ_k are learned per-device, let us define $\tilde{\mathcal{D}} = \{\tilde{\mathcal{D}}_{1,1}, \ldots, \tilde{\mathcal{D}}_{1,K}, \tilde{\mathcal{D}}_{2,1}, \ldots, \tilde{\mathcal{D}}_{N,K}\}$ as the dataset of observation and target sequences used for each individual device, i.e., $\tilde{\mathcal{D}}_{n,k} = (\mathbf{o}_{1:T_{\text{train}}}^{(n)}, z_{k,1:T_{\text{train}}}^{(n)})$ comprises all observations in task n (e.g., SNRs of all devices) but the target variable only for device k. The empirical loss function for device k in task n can then be written

$$\mathcal{L}_{\mathcal{D}_{n,k}}^{(t)}(\varphi) = \sum_{i=1}^{t} L(z_{k,i}^{(n)}, f_{\varphi}(\mathbf{o}_{1:i}^{(n)})). \tag{C.10}$$

The quality of a model trained by minimizing the training loss (C.10) depends on the number of data points, t, observed so far. In order to reduce the data requirements and improve the prediction at earlier time steps t,

we apply Model-Agnostic Meta-Learning (MAML) [19]. MAML finds a set of model parameters θ, from which the device-specific model parameters φ can be obtained by taking one or more gradient steps computed using the t available training samples in $\mathcal{D}_{n',k}$

$$\varphi \leftarrow \theta + \beta \nabla_\theta \mathcal{L}^{(t)}_{\mathcal{D}_{n,k}}(\theta), \tag{C.11}$$

where β is the learning rate. The initialization θ is optimized via stochastic gradient descent as illustrated in Algorithm 1.

Algorithm 1 Model-Agnostic Meta-Learning (MAML).

Input: Datasets $\tilde{\mathcal{D}}$, meta batch size B, step sizes α, β.
Randomly initialize θ
While convergence criterion not met **do**
  Draw meta-batch $\tilde{\mathcal{D}}'$ of B datasets from $\tilde{\mathcal{D}}$
  For each dataset $\tilde{\mathcal{D}}_{n,k}$ in $\tilde{\mathcal{D}}'$ **do**
    Split $\tilde{\mathcal{D}}_{n,k}$ into two datasets $\tilde{\mathcal{D}}^{\text{tr}}_{n,k}$ and $\tilde{\mathcal{D}}^{\text{te}}_{n,k}$
    Compute $\varphi_{n,k} \leftarrow \theta - \alpha \nabla_\theta \tilde{\mathcal{L}}_{\tilde{\mathcal{D}}^{\text{tr}}_{n,k}}(\theta)$
  Compute $\theta \leftarrow \theta - \beta \nabla_\theta \sum_{n,k \in \tilde{\mathcal{D}}'} \mathcal{L}_{\tilde{\mathcal{D}}^{\text{te}}_{n,k}}(\varphi_{n,k})$

3.4 Evaluation Methodology

3.4.1 Blockage Generation

We model the channel coefficients as Rician fading channels with magnitudes $|h_{k,t}|$ distributed according to

$$p_{|h_{k,t}|}(x) = \frac{x}{\sigma_k^2} \exp\left(-\frac{x^2 + A_{k,t}^2}{2\sigma_k^2}\right) I_0\left(\frac{A_{k,t} x}{\sigma_k^2}\right), \tag{C.12}$$

where $I_0(\cdot)$ is the modified Bessel function of the first kind and order zero, $A_{k,t}$ is the signal amplitude of the dominant path and σ_k is the standard deviation of the resultant amplitude from the remaining paths. To incorporate blockage effects, we express the dominant path amplitude as $A_{k,t} = \zeta_{k,t} \bar{A}_k$, where $\zeta_{k,t}$ models the attenuation caused by blockages and $\bar{A}_k$ is the average SNR of device k when the signal is not blocked. The expected SNR at time t is then

$$\gamma_{k,t} = \zeta_{k,t}^2 \bar{A}_k^2 + 2\sigma_k^2. \tag{C.13}$$

As seen in Fig. C.2, we assume the K devices are static and uniformly distributed within a rectangular area $[-1, 1] \times [-0.5, 0.5]$, and that the BS is

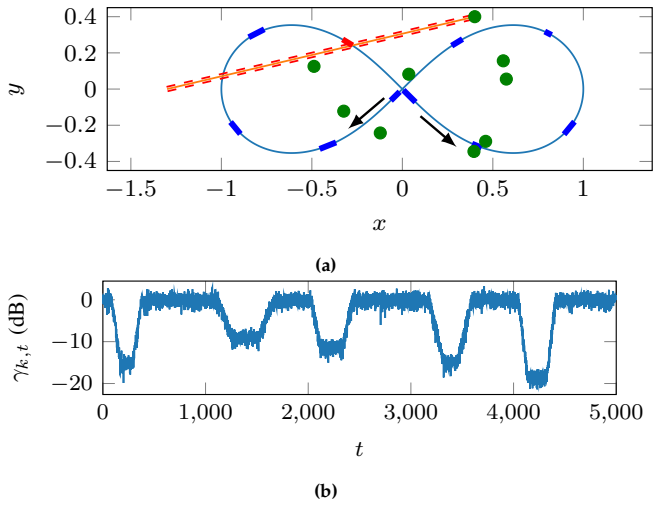


Fig. C.2: a Illustration of the blockage generation process with $K = 10$ devices (dots), $M = 10$ blockage objects (bold segments on the lemniscate), and BS at $(-1.3, 0.0)$. The beamwidth is indicated by the dashed lines. b The SNR sequence generated for the device at $(0.4, 0.4)$.

located at $(-1.3, 0)$. The line-of-sight path is assumed to be dominant and is modelled as a rectangular pencil beam with beamwidth ϑ. This assumption is reasonable when the objects are concentrated in a small area at some distance from the BS, where the beamwidth is approximately constant. The blockage attenuation coefficients $\zeta_{k,t}$ are generated as follows. A (random) number M of blockage objects are moving on the slope of an ∞-shaped Bernoulli lemniscate centred at the origin and with unit half-width. Each blockage object has a random length (but zero width) and moves with a random (but constant) speed. When one of the blockage objects intersects with a line-of-sight path, the signal amplitude of the dominant path is reduced depending on the attenuation factor of the blockage object. Specifically, we adopt a linear diffraction model inspired by measurements in the mmWave spectrum [16], and model the attenuation coefficients $\zeta_{k,t}$ as the percentage of the beam that is blocked, i.e.,

$$\zeta_{k,t} = \prod_{m=1}^{M} (\delta_m)^{p_{k,m,t}}, \tag{C.14}$$

where δ_m is the attenuation factor of object m and $p_{k,m,t}$ is the percentage of the beam cross section that the object blocks at time t. Note that we model the attenuation as a function of the blockage region, and not of the angle between the receiver and the blockage object as in [16]. For simplicity, we ignore the fact that multiple objects may block the same part of the beam, and simply aggregate the contributions from the individual blockage objects. Further-

more, we neglect effects such as reflections. The location of the devices, the number of blockage objects and their lengths and speeds and attenuation factors define the random variable π. Although the setting is clearly simplistic, its main purpose is to model the correlation between device SNRs across time and devices. A snapshot of a sample environment with $K = 10$ devices and $M = 10$ blockage objects is illustrated in Fig. C.2a, and Fig. C.2b shows the SNR sequence generated for the device at $(0.4, 0.4)$. Note the impact of the different object lengths and attenuation factors on the SNR during blockages.

3.4.2 Observation Model

While the proposed algorithm can be used with any observation vector, we evaluate its performance using only SNR observations from all K devices within a factory cell. These are available in most existing wireless systems, and capture the temporal and spatial correlation among blockages. We assume that the SNR of a device can only be obtained if its SNR is higher than γ_0, and indicate the availability of an SNR measurement by a binary feature in the observation vector. Thus, the observations at each time step are represented as the $2K$-dimensional vector $\mathbf{o}_t \in \mathbb{R}^{2K}$ obtained by concatenating the K $(\mathbb{1}[\gamma_{k,t} \leq 0], \gamma_{k,t})$-tuples for $k = 1, \ldots, K$. To impose structure that can be beneficial in meta-learning, we put the tuple of the device that we aim to predict at the top of the observation vector. The order of the remaining $K - 1$ tuples is arbitrary but fixed.

3.4.3 Baseline Models

We consider three reference baseline models. The first is *naïve forecasting*, where the predictor outputs the most recent observed value, that is, $f_k^{\text{naive}}(\mathbf{o}_{k,1:t}) = \mathbb{1}[\gamma_{k,t} \leq 0]$. The second baseline is *joint learning*, whereby the RNN is trained on the meta-training dataset in a task-agnostic manner (i.e., without MAML) as in typical machine learning applications. Finally, we also consider *random initialization*, where the network weights are randomly initialized prior to the evaluation, i.e., without using the meta-training dataset.

3.5 Results

We evaluate the model on tasks with either $K = 20$ or $K = 50$ devices generated with the parameters listed in Table C.1. The specific carrier frequency is irrelevant as it is abstracted in the other parameters. The predictors $f_{\varphi_k}(\mathbf{o}_{1:t})$ are RNNs comprised of a fully connected input layer $g_{\varphi_k}^{(\text{in})}(\cdot)$ with 128 rectified linear units (ReLUs); $g_{\varphi_k}^{(\text{rec})}(\cdot)$ and $\eta_{\varphi_k}(\cdot)$ are given by an LSTM layer

Table C.1: Evaluation Parameters

Parameter	Value
Average unblocked SNR, $\bar{A}_k^2 + 2\sigma_k^2$	0 dB
K-factor, $\bar{A}_k^2/(2\sigma_k^2)$	15 dB
Beamwidth, ϑ	0.025
Blockage SNR threshold, γ_0	-20 dB
Number of blockage objects	$\mathcal{U}\{2,5\}$
Lengths of blockage objects	0.05
Speed of blockage objects (loops/sec.)	$\mathcal{U}(0.005, 0.01)$
Attenuation factors, δ_m	$\mathcal{U}(-30,-10)$ dB
Transmission interval (ms)	50

with 128 hidden units (see e.g., [17, Eqs. (10.40)–(10.44)] for details); and finally $g_{\varphi_k}^{(\text{out})}$ is a connected layer with 128 rectified linear units followed by a linear output layer with a single sigmoid-activated output. To compensate for the fact that only around 3% of the target values in the dataset are positive, we weight the positive examples by $w = 9$ in Eq. (C.8) (higher values of w did not yield better results).

We train the model on data from $N = 100$ tasks, i.e. realizations of π, each containing 20 or 50 devices so that the per-device dataset, $\tilde{\mathcal{D}}$, contains a total of 2000 or 5000 sequences. Each sequence contains $T_{\text{train}} = 10000$ samples, which for MAML are evenly split into a meta-training and meta-testing sub-sequence. In both joint learning and MAML, we use truncated backpropagation through time with a truncation length of 128 samples. Furthermore, to reduce the training time of MAML we split the samples from each device, $\tilde{\mathcal{D}}_{n,k}^{\text{tr}}$ and $\tilde{\mathcal{D}}_{n,k}^{\text{te}}$, into batches of 512 samples for which the loss can be computed in parallel (in lines 7 and 8 of Algorithm 1). The consequence is that the LSTM state is not maintained for more than 512 samples, thus it is only done during training and not in validation and testing. We use a separate dataset of the same size for testing, and use sequences of length $T_{\text{test}} = 10000$ for model adaptation and 10000 for evaluation. We note that the amount of training data generally influences the performance of the predictors, as the inductive bias learned by MAML can lead to potential degradation when sufficient data are available [15].

Fig. C.3(a) shows the CDF of the prediction time for the *any*-predictor with a prediction interval of $\tau = 25$ and prediction delay $\xi = 0$, where a blockage occurs at time $t = 50$. Thus, the predictor is expected to predict the blockage from $t = 25$. As can be seen, the MAML predictor generally predicts the blockage earlier than the predictors with randomly and jointly learned initialization, indicating that the MAML procedure learns an inductive bias

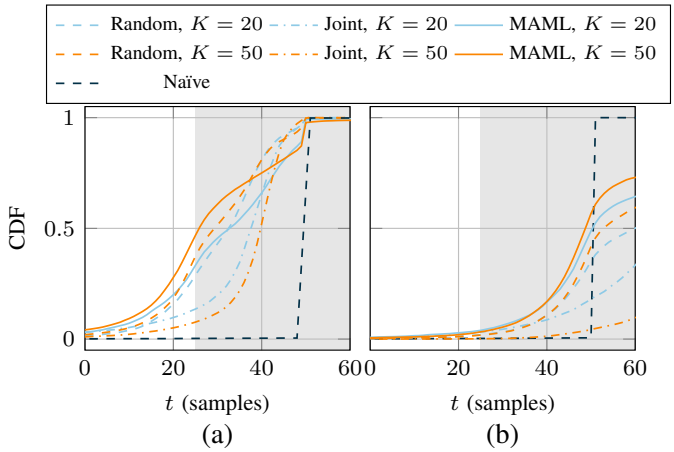


Fig. C.3: CDF of prediction time for the (a) *any*-predictor ($\xi = 0$, $\tau = 25$) and (b) *all*-predictor ($\xi = 25$, $\tau = 3$) with $N = 1000$. The detection threshold is $f_k(\mathbf{o}_{k,1:t}) > 0.5$ and the blockage starts at $t = 50$ so the predictors should output the blockage from $t = 25$.

that is beneficial for efficient learning. The randomly initialized predictor performs better than the jointly learned initialization for both $K = 20$ and $K = 50$, but not as good as MAML. The joint predictor performs the worst of the neural network based models, which indicates that it is unlikely to find a universal criterion that predicts a blockage. Contrary to the MAML and random predictors, the joint predictor performs worse for $K = 50$ than for $K = 20$.

Under the same conditions, the prediction time CDF for the *all*-predictor with $\tau = 3$ and $\xi = 25$ is shown in Fig. C.3(b). Compared to the *any*-predictor, the predictors are generally slower at predicting the blockage, which is likely due to the fact that the overall probability of experiencing $\tau = 3$ consecutive blockages is less than the probability of experiencing any blockage in a window of 25 slots, as was the case with the *any*-predictor. Nevertheless, the MAML predictor again generally predicts the blockage earlier than the random and joint predictors.

Fig. C.4 shows the impact of the length of the adaptation sequence T_{test} for the *any*-predictor. MAML outperforms random initialization especially when T_{test} is small, while the latter approach may be advantageous when T_{test} is large. In contrast, the jointly trained initialization does not adapt well to the new task. This illustrates the key property that joint learning aims to minimize the average loss across all tasks, whereas MAML tries to minimize the loss of each individual task.

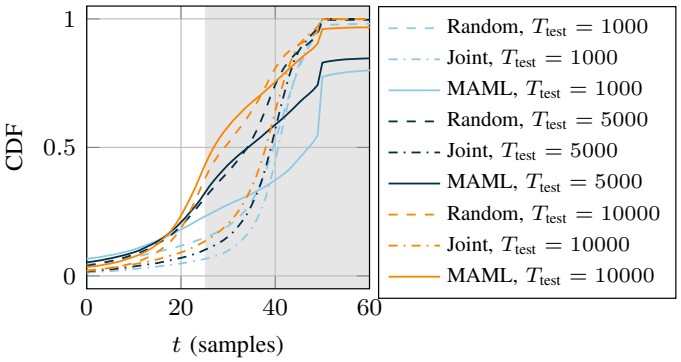


Fig. C.4: CDF of prediction time for the *any*-predictor for a range of adaptation sequence lengths T_{test} with $K = 50$, $\xi = 0$, and $\tau = 25$.

3.6 Conclusions

In this paper, we have studied the use of meta-learning to train a recurrent neural network as a blockage predictor for mmWave and THz systems using few samples. While our method is general, we trained the predictors to exploit correlation in SNRs from multiple devices, and considered both an *any*-predictor, which predicts whether there will be at least one blocked transmission within a time window, and an *all*-predictor, which predicts whether the entire window will be blocked. We have shown that meta-learning is beneficial as compared to standard training methods in that it leads to predictors that predict blockages earlier, based on fewer observed data samples.

References

[1] M. S. Elbamby, C. Perfecto, M. Bennis, and K. Doppler, "Toward low-latency and ultra-reliable virtual reality," *IEEE Network*, vol. 32, no. 2, pp. 78–84, 2018.

[2] J. Sachs, K. Wallstedt, F. Alriksson, and G. Eneroth, "Boosting smart manufacturing with 5G wireless connectivity," *Ericsson Tech. Rev.*, 2019.

[3] 3GPP, "Technical Specification Group Services and System Aspects; Service requirements for cyber-physical control applications in vertical domains," 3rd Gener. Partnership Project (3GPP), Technical Specification (TS) 22.104, sep. 2020, version 17.4.0.

[4] C. She *et al.*, "A tutorial on ultrareliable and low-latency communications in 6g: Integrating domain knowledge into deep learning," *Proc. IEEE*, vol. 109, no. 3, pp. 204–246, 2021.

[5] D. Burghal, R. Wang, and A. F. Molisch, "Deep learning and Gaussian process based band assignment in dual band systems," *arXiv preprint arXiv:1902.10890*, 2019.

[6] M. Alrabeiah, A. Hredzak, and A. Alkhateeb, "Millimeter wave base stations with cameras: Vision-aided beam and blockage prediction," in *Proc. IEEE Veh. Technol. Conf. (VTC2020-Spring)*. IEEE, 2020, pp. 1–5.

[7] Z. Ali, A. Duel-Hallen, and H. Hallen, "Early warning of mmWave signal blockage and aoa transition using sub-6 GHz observations," *IEEE Commun. Letters*, vol. 24, no. 1, pp. 207–211, 2019.

[8] A. Alkhateeb, I. Beltagy, and S. Alex, "Machine learning for reliable mmWave systems: Blockage prediction and proactive handoff," in *IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*. IEEE, 2018, pp. 1055–1059.

[9] S. H. A. Shah, M. Sharma, and S. Rangan, "LSTM-based multi-link prediction for mmWave and sub-THz wireless systems," in *IEEE Int. Conf. Commun. (ICC)*. IEEE, 2020, pp. 1–6.

[10] M. Hussain, M. Scalabrin, M. Rossi, and N. Michelusi, "Mobility and blockage-aware communications in millimeter-wave vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13 072–13 086, 2020.

[11] X. Liu *et al.*, "Learning to predict the mobility of users in mobile mmWave networks," *IEEE Wireless Commun.*, vol. 27, no. 1, pp. 124–131, 2020.

[12] G. Charan, M. Alrabeiah, and A. Alkhateeb, "Vision-aided 6G wireless communications: Blockage prediction and proactive handoff," *arXiv preprint arXiv:2102.09527*, 2021.

[13] S. wu, M. Alrabeiah, A. Hredzak, C. Chakrabarti, and A. Alkhateeb, "Deep learning for moving blockage prediction using real millimeter wave measurements," *arXiv preprint arXiv:2101.06886*, 2021.

[14] S. Samarakoon, M. Bennis, W. Saad, and M. Debbah, "Predictive ultra-reliable communication: A survival analysis perspective," *IEEE Commun. Letters*, pp. 1–1, 2020.

[15] O. Simeone, S. Park, and J. Kang, "From learning to meta-learning: Reduced training overhead and complexity for communication systems," in *2020 2nd 6G Wireless Summit (6G SUMMIT)*. IEEE, 2020, pp. 1–5.

[16] T. S. Rappaport, G. R. MacCartney, S. Sun, H. Yan, and S. Deng, "Small-scale, local area, and transitional millimeter wave propagation for 5G communications," *IEEE Trans. Antennas Propag.*, vol. 65, no. 12, pp. 6474–6490, 2017.

[17] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[18] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 9268–9277.

[19] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*. JMLR.org, 2017, p. 1126–1135.

Paper D

Random Access Schemes in Wireless Systems with Correlated User Activity

Anders E. Kalør, Osama A. Hanna, and Petar Popovski

Published in
Proceedings of IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), pp. 1–5, 2018,
doi: 10.1109/SPAWC.2018.8445866.

Abstract

Traditional random access schemes are designed based on the aggregate process of user activation, which is created on the basis of independent activations of the users. However, in Machine-Type Communications (MTC), some users are likely to exhibit a high degree of correlation, e.g. because they observe the same physical phenomenon. This paves the way to devise access schemes that combine scheduling and random access, which is the topic of this work. The underlying idea is to schedule highly correlated users in such a way that their transmissions are less likely to result in a collision. To this end, we propose two greedy allocation algorithms. Both attempt to maximize the throughput using only pairwise correlations, but they rely on different assumptions about the higher-order dependencies. We show that both algorithms achieve higher throughput compared to the traditional random access schemes, suggesting that user correlation can be utilized effectively in access protocols for MTC.

4.1 Introduction

Machine-Type Communication (MTC) represents an important pillar of 5G wireless systems. It will come in two flavors, massive Machine Type Communications (mMTC) and Ultra-Reliable and Low Latency Communications (URLLC) services. In contrast to traditional uses of communication systems dominated by bandwidth intensive, human-initiated activity, machine-type traffic is characterized by a very large number of devices, small packet sizes and possibly strict latency and reliability requirements [1]. Furthermore, machines are likely to produce more correlated and predictable traffic patterns, e.g. if the traffic is generated based on observations of some common physical phenomenon [2].

However, current access protocols do not exploit this correlation between users, and are usually designed based on the aggregate activation process under the assumption that users are independent. Under these conditions, the access protocols are derivatives of slotted ALOHA [3] which achieves maximum throughput per slot of $1/e \approx 0.37$ when the average number of transmissions per slot is 1. The recent class of random access protocols that rely on successive interference cancellation (SIC), such as coded slotted ALOHA [4, 5], can achieve high throughputs at the expense multiple packet replicas sent by the users and complex processing at the receiver. Regardless of the receiver model, none of these protocols considers correlation of the activity among the transmitting devices (users).

In this paper, we investigate how knowledge about the user activity correlation can be used to improve the throughput of the access protocols, and consequently reduce the latency and increase the reliability of the systems. In this first work on the topic, we limit ourselves to the collision model, but the

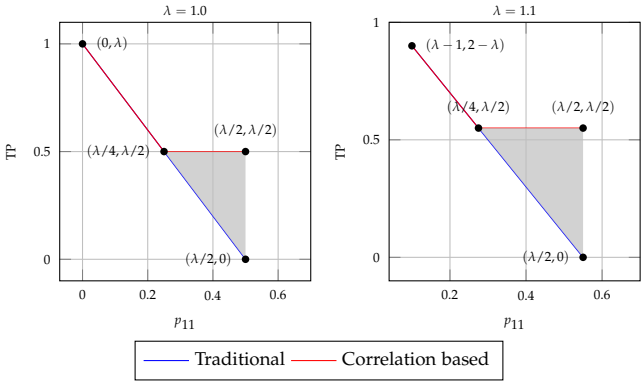


Fig. D.1: Achievable throughputs with two correlated users and different arrival rates λ. The gray area represents the throughput achievable by exploiting correlation statistics.

ideas can be extended to the receivers based on SIC. To illustrate the main idea, consider an example of a system with $N = 2$ users and $K \in \{1, 2\}$ slots. Let X_1 and X_2 denote the random binary events that user 1 and user 2 transmits in a given frame, and let $p_{ij} = \Pr(X_1 = i, X_2 = j)$ with $i, j \in \{0, 1\}$. For simplicity, assume that $p_{01} = p_{10} \triangleq p$, such that the expected number of transmissions is $\lambda = 2(p + p_{11})$. If $K = 1$ the two users contend for the slot, and a transmission will succeed only in the case where a single user transmits. On the other hand, if $K = 2$ the user transmissions will succeed unconditionally. The throughput is given by

$$\mathrm{TP} = \frac{2p + 2p_{11}\mathbb{1}(K > 1)}{K}$$

where $\mathbb{1}(K > 1)$ equals 1 if $K > 1$ and 0 otherwise. Under the traditional assumption of independent users $p_{11} = (\lambda/2)^2$. Since by definition $p = \lambda/2 - p_{11}$ the optimal policy is to allocate two slots only when $\lambda > 1$. However, if the users are correlated, i.e. $p_{11} \neq (\lambda/2)^2$, this is a suboptimal strategy that leads to reduced utilization if $p_{11} < (\lambda/2)^2$ or increased collision probability if $p_{11} > (\lambda/2)^2$. If we instead make use of p_{11}, the optimal strategy is to allocate 2 slots when $\lambda/4 < p_{11}$. This is illustrated in Fig. D.1 for $\lambda = 1.0$ and $\lambda = 1.1$, where the gray region indicates the throughput gain that can be achieved by considering the user correlation. Notice that the two schemes perform equivalently in the region where p_{11} is small, since allocating a single slot is optimal in both schemes.

In Section 4.2 we generalize the allocation problem to N users and K slots and define the system model. Since this turns out to be non-convex and hard to solve even approximately, we reformulate the problem and propose two heuristic allocation algorithms in Section 4.3. In Section 4.4.1 we define a

traffic model that we use in Section 4.4.2 to evaluate the algorithms. Finally, we discuss practical aspects and future work in Section 4.5 and conclude the paper in Section 4.6.

4.2 System Model and Problem Definition

We now generalize the scenario considered in the previous section and consider N users that transmit to a common receiver in frames consisting of K slots. We consider a collision model, such that when there are two or more transmissions in a slot, the receiver observes a failure (erasure). In each frame, each user can have at most one transmission, as in the original framed ALOHA [3].

The user activation is defined through the joint distribution $\Pr(\mathbf{x}) = \Pr(x_1, x_2, \ldots, x_N)$ where $x_n = 1$ if user n transmits in a given slot, and $x_n = 0$ otherwise. It is assumed that this joint distribution is independently sampled in each new $K-$slot frame. As seen in the example with $N = 2$ users, the correlation can be used to decide which users should be assigned to the same slot. Thus, in practical scenarios where the number of slots is much smaller than the number of users, the overall objective is to assign the users to the slots so as to maximize the throughput. We define the allocation matrix $\mathbf{A} \in \mathbb{R}^{N\times K}$ where A_{ij} is the probability that user i transmits in slot j conditioned on activation, and $\sum_j A_{ij} = 1$. Let e_{ij} denote the event that user i selects slot j, i.e. $\Pr(e_{ij}) = \mathbb{E}[e_{ij}] = A_{ij}$. The throughput is given by the expected number of slots in which exactly one user transmits:

$$\mathrm{TP}(\mathbf{A}) = \sum_{k=1}^{K}\sum_{n=1}^{N} T_n^{(k)} \tag{D.1}$$

where

$$\begin{aligned} T_n^{(k)} &= E_{\mathbf{x}}\left[x_n\,\mathbb{E}[e_{nk}]\prod_{m=1}^{N}(1-x_m\,\mathbb{E}[e_{mk}])^{\mathbb{1}(n\neq m)}\right] \\ &= E_{\mathbf{x}}\left[x_n A_{nk}\prod_{m=1}^{N}(1-x_m A_{mk})^{\mathbb{1}(n\neq m)}\right]. \end{aligned}$$

Here, we used the fact that x_n and e_{ij} are independent. $\mathrm{TP}(\mathbf{A})$ is non-convex and finding the slot assignments A_{ij} that maximizes this throughput is hard. Furthermore, computing the expectations requires the full activity distribution or at least estimates of the product expectations of any subset of the users, which are in general unknown and need to be learned. The specification of the joint distribution $\Pr(x_1, \ldots, x_N)$ requires, in general, specification of $2^N - 1$ values. Hence, it can only be specified (estimated) when N is low.

On the other hand, the number of pairwise correlations scales as N^2 and can be considered feasible for estimation. We use the knowledge of pairwise correlations to resort to alternative formulations or heuristics that approximately maximize the throughput.

Without loss of generality, we consider the throughput contribution by user 1 in slot k, $T_1^{(k)}$. As x_n are binary random variables, the expectation equals the probability of the event, i.e.

$$\mathbb{E}[x_1 x_2 \dots x_N] = \Pr(x_1, x_2, \dots, x_N).$$

Using this, we may express $T_1^{(k)}$ as the probability that only user 1 transmits in slot k. It follows from the inclusion-exclusion principle [6] that the terms that include expectations of higher order than two compensate for intersecting events as illustrated graphically in Fig. D.2 for $T_1^{(k)}$ in the case of 4 users. Estimating the higher order expectations poses a challenge, and a reasonable objective is to only estimate the first and second order expectations, and put assumptions on the higher order expectations. Since the exact throughput for a given allocation cannot be determined without the higher order expectations, we may instead use the inclusion-exclusion principle to bound the throughput as

$$T_1^{(k)} \leq A_{1k}\,\mathbb{E}[x_1] - \max_{m=2,\dots,N} A_{1k}A_{mk}\,\mathbb{E}[x_1 x_m] \tag{D.2}$$

$$T_1^{(k)} \geq A_{1k}\,\mathbb{E}[x_1] - \sum_{m=2}^{N} A_{1k}A_{mk}\,\mathbb{E}[x_1 x_m]. \tag{D.3}$$

The lower bound is valid with equality if the higher order expectations of the users that are allocated in slot k are zero, i.e. no more than two transmissions occur in the same slot. Similarly, the upper bound has equality either when the users never transmit in the same slot, or when all users $2, \dots, N$ always transmit jointly with user 1. Both bounds may be used to derive allocations that approximate the optimal throughput. However, the lower bound provides an accurate estimation if the users are unlikely to transmit together, while the upper bound is more accurate when the users are highly correlated.

4.3 Heuristic Algorithms

Here we present two heuristic slot assignment algorithms that attempt to maximize the throughput using only pairwise expectations based on the throughput upper/lower bounds.

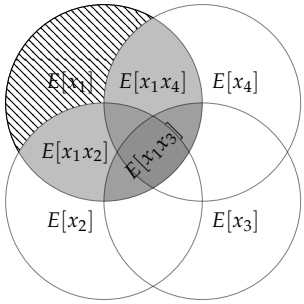


Fig. D.2: Illustration of the events that contribute to the calculation of the expected throughput of user 1 in the case of four users. The hatched region indicates the expected throughput conditioned on a successful transmission.

4.3.1 Min-Max Pairwise Correlation

We first attempt to maximize the throughput using the upper bound (Eq. (D.2)). Since we maximize the upper bound, we optimize for the best (in terms of probability of joint transmission) of the set of generating distributions, namely when all transmissions involving more than a one user happen in the same frame. Hence, we expect it to perform well if the users are strongly correlated in the higher order expectations, and poorly when they are anticorrelated. Considering the constraint $\sum_j A_{ij} = 1$ we equivalently minimize:

$$\mathrm{MMPC}(\mathbf{A}) = \sum_{k=1}^{K} \max_{\{n,m\}\in[1,N]^2} A_{nk}A_{mk}\,\mathbb{E}\left[x_n x_m\right] \tag{D.4}$$

where $[1,N]^2 = \{\{a,b\} : a,b \in 1,2,\ldots,N, a \neq b\}$. While Eq. (D.4) simplifies Eq. (D.1) it still constitutes a non-convex quadratically constrained linear program. While methods to approximate such problems have been proposed in the literature, including semidefinite relaxation [7, 8] and successive convex approximation [9, 10], they require a considerable amount of fine-tuning, or they work by lifting the variables to a higher dimension which is inappropriate for the problem at hand where the number of variables is already large.

Evaluating the performance of various optimization algorithms is beyond the scope of this paper. Instead, we focus on the case where $A_{ij} \in \{0,1\}$, i.e. users are assigned a single slot in which they deterministically transmit if they are active, and propose a simple greedy algorithm that consecutively assigns users that are less likely to transmit together jointly to the same slots. The algorithm, outlined in Algorithm 2, takes a symmetric matrix $\mathbf{C}$ where element ij is $\mathbb{E}[x_i x_j]$ if $i \neq j$ and ∞ if $i = j$, and outputs the matrix $\mathbf{A}$. Here $\mathbf{C}$ can be seen as an adjacency matrix for a fully connected graph where the ver-

tices are slots and the edge weights are the product expectation of two slots. Initially, each user is assigned to its own slot. As the algorithm proceeds, $\mathbf{C}$ is reduced by *merging* two vertices until it contains only K vertices. A merge between vertex i and j is performed by updating the edge weights of vertex j: $C_{jn} = C_{nj} = \textsc{max}\{C_{ni}, C_{nj}\}\ \forall n$, so that the new edge weight represents the maximum joint transmission probability between two users that are assigned to different slots. Then, the i-th column and row of $\mathbf{C}$ are removed (denoted by $\mathbf{C} = \mathbf{C}_{-i}$). A vector $\mathbf{S}$ that maps the users to each of the K slots is maintained in order to construct the (binary) $\mathbf{A}$ matrix in the last step. In lines 9–12 the user-slot mapping is updated as a slot has been removed due to a merge. The initial size of $\mathbf{C}$ is $N \times N$, and hence the outer loop has $\mathcal{O}(N)$ iterations. The complexity within the loop is dominated by the $\textsc{min}(\mathbf{C})$ operation with $\mathcal{O}(N \log N)$, and hence the total complexity is $\mathcal{O}(N^2 \log N)$.

Algorithm 2 MMPC allocation

Input: $\mathbf{C} \in \mathbb{R}^{N \times N}, K$
Output: $\mathbf{A} \in \mathbb{R}^{N \times K}$
1: $\mathbf{S} = [1, 2, \ldots, N], \mathbf{A} = \mathbf{0}$
2: **while** $\textsc{size}(C) > K \times K$ **do**
3: $(i, j) = \textsc{min}(C)$
4: **for** $n = 1, \ldots, \textsc{rows}(C)$ **do**
5: $C_{jn} = \textsc{max}\{C_{ni}, C_{nj}\}$
6: $C_{nj} = C_{jn}$
7: $\mathbf{C} = \mathbf{C}_{-i}$
8: $S_i = \textsc{min}\{S_i, S_j\}$
9: **for** $n = i, \ldots, N$ **do**
10: $S_n = S_n - 1$
11: **for** $n = i, \ldots, N$ **do**
12: $A_{nS_n} = 1$
return S

The active users transmit unconditionally in their assigned slots, which may result in collisions if the users are active simultaneously. To avoid this situation, we can scale the resulting allocation to maximize the probability of singular transmissions. However, as we do not know the joint activity distribution, we instead set the expected number N_i of transmissions in the slot assigned to user i conditioned on activity of user i to one. Let S_i denote the slot of user i and A_i the probability that user i will transmit conditioned on being active. We then have

$$\mathbb{E}[N_i] = A_i + \sum_{n \in \mathcal{N}_{S_i}} A_n \, \mathbb{E}[x_n | x_i] = A_i + \frac{\sum_{n \in \mathcal{N}_{S_i}} A_n \, \mathbb{E}[x_n x_i]}{\mathbb{E}[x_i]},$$

where $\mathcal{N}_{S_i}$ is the set of users assigned to slot S_i. For each slot j we determine the new values $0 \leq A_i \leq 1$ that minimizes the least-squares $\sum_{i \in \mathcal{N}_j} (\mathbb{E}[N_i] - 1)^2$. Although the performance the heuristic depends on the higher-order correlation of the users, results presented in Section 4.4.2 suggest that it works as intended.

4.3.2 Min-Sum Pairwise Correlations

We now consider the lower bound (Eq. (D.3)) and minimize the sum of product expectations in each slot:

$$\mathrm{MPC}(\mathbf{A}) = \sum_{k=1}^{K} \sum_{n=1}^{N} \sum_{m=1}^{N} \mathbf{1}(n \neq m) A_{nk} A_{mk} \, \mathbb{E}\left[x_n x_m\right]. \tag{D.5}$$

Compared to the Min-Max Pairwise Correlation, this function maximizes the throughput under the assumption that no more than two users transmit in the same slot (the higher order expectations are zero). Similar to Eq. (D.4), this also poses a non-convex quadratic program and we use a similar greedy algorithm to approximate a solution. Since we are aiming at minimizing the sum, the weight updating step in Algorithm 2 (line 5) can simply be replaced by the sum $C_{jn} = C_{ij} + C_{ni} + C_{nj}$, while the remaining procedure remains unchanged.

4.3.3 Illustrative Allocation Examples

To illustrate how the two algorithms are different, we consider two example correlations; one for which the Min-Max results in a higher throughput than the Min-Sum algorithm, and one for which the Min-Sum performs better. We consider a scenario with four users and two slots. The first instance, for which the Min-Max algorithm is best, is illustrated on the left in Fig. D.3, where the activity pattern in the top is repeated indefinitely. The graph is defined from the input matrix $\mathbf{C}$. The Min-Max algorithm allocates users 1, 2 and 3 to slot 1, and user 4 to slot 2, resulting in a throughput of $7/4$. On the other hand, the Min-Sum algorithm allocates users 1 and 2 to slot 1, and user 3 and 4 to slot 2, yielding a throughput of only 1. Hence, the Max-Sum algorithm achieves 75% higher throughput. In the right graph in Fig. D.3, the opposite is the case, and the Max-Min allocation results in a throughput of $7/6$ while the Max-Sum allocation achieves $9/6$. The reason for the difference in the performance is the higher order correlations. Specifically, in the left case the higher order correlations are more significant, and hence the upper bound approximation is more accurate than the lower bound. Similarly, the lower bound approximation is more accurate in the case where the Min-Sum algorithm performs best.

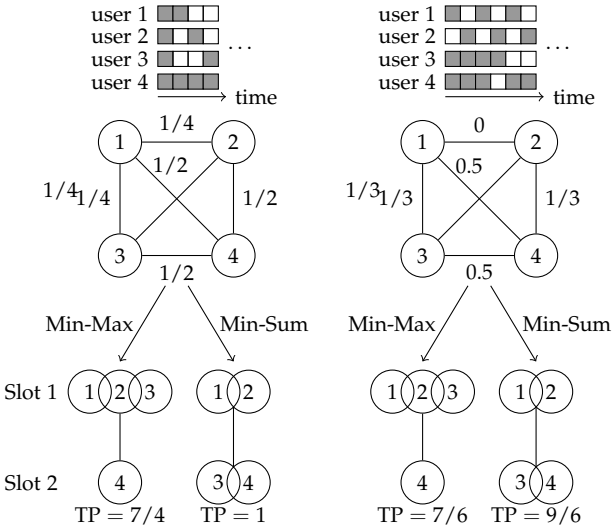


Fig. D.3: Examples where the Min-Max and Min-Sum algorithms result in allocations with different throughputs. A filled square in the top indicates that the user transmits in the given frame.

4.4 Evaluation

4.4.1 Traffic Model for Correlated Activity

We first present the traffic model that will be used to evaluate the algorithms. The existing traffic models are not suitable for this study. The 3GPP model [2] uses a Beta$(3, 4)$ distribution to model the arrival process at the base station. However, since this model describes the aggregate arrival process, it cannot be used for scheduling of the individual users. The activity of individual users is explicitly modeled in [11] through a Coupled Markov Modulated Poisson Processes, where each user is modeled as a Markov chain with transition probabilities defined as a convex combination of a user-local and a central (shared) process. While this exhibits some correlation, it acts at a macro-level and the individual users are still approximately independent at small time scales.

In the model we consider, the users are deployed uniformly in a square area in which events are generated according to a spatio-temporal Poisson point process. The users transmit then if an event occurs within a certain radius. Let $\mathcal{R} = [0, L]^2$ denote the square region of size $L \times L$ and let $\mathbf{x}_i \sim \text{Uniform}(\mathcal{R})$ denote the location of user i ($i = 1, \ldots, N$). We model random spatio-temporal events by a homogeneous Poisson point process on $\mathcal{R}$ with rate λ so that the number of events within a region $\mathcal{D} \subseteq \mathcal{R}$ follows a Poisson

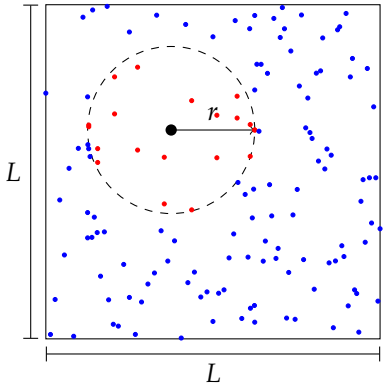


Fig. D.4: Illustration of the spatio-temporal traffic model where the black circle indicates an event, and the red dots are active devices.

process with rate $\lambda A(\mathcal{D})$ where $A(\mathcal{D})$ is the area of $\mathcal{D}$.

The user activity is defined in such a way that all users within radius of r from an event transmit synchronously in the following frame, as illustrated in Fig. D.4. The correlation between two users is defined by the intersecting area between the disks with radius r centered at the users. To overcome edge effects caused by regions outside $\mathcal{R}$, we apply the border method in generation of user locations and regenerate locations that are closer than r to the edge of $\mathcal{R}$. The area of the disk intersection is $\mathcal{D}_{ij} = 2r^2 \cos^{-1}\left(\frac{d_{ij}}{2r}\right) - \frac{d_{ij}}{2}\sqrt{4r^2 - d_{ij}^2}$, where $d_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$, see [12]. Two users transmit in the same frame only if an event occurs within this area, or if one or more events occur within each user's radius. We assume that the frame duration is 1 and that a user transmits at most once per frame. It follows that $\mathbb{E}[x_i] = 1 - e^{-\lambda\pi r^2}$ and for $i \neq j$

$$\mathbb{E}[x_i x_j] = \begin{cases} \left(1 - e^{-\lambda\pi r^2}\right)^2 & d_{ij} \geq r \\ 1 - e^{-\lambda \mathcal{D}_{ij}} + \left(1 - e^{-\lambda(\pi r^2 - \mathcal{D}_{ij})}\right)^2 & d_{ij} < r. \end{cases}$$

4.4.2 Numerical Results

We evaluate the algorithms in a system with a fixed number of $N = 1000$ users, and compare the throughput to the traditional random access scenario where users transmit in a slot drawn from a uniform distribution. Figure D.5 shows the throughput within a square area with side lengths $L = 100$, correlation radius $r = 15$ and $K = 150$ slots. The global event rate λ is varied from $\lambda = 0.1/(L^2)$ to $\lambda = 30/(L^2)$. The schemes perform equivalently when

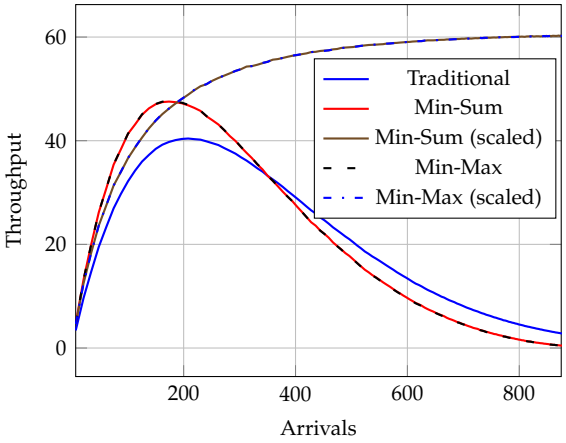


Fig. D.5: Throughput in the spatio-temporal scenario with 150 slots and varying average arrivals.

the number of arrivals is small since collisions are unlikely. However, as the number of arrivals increases, the correlation based schemes perform significantly better than the traditional case, and achieves maximum throughput when the average arrivals is close to the number of slots. When the number of arrivals increases beyond the number of slots, collisions are unavoidable and the throughput decreases significantly. However, with the scaling heuristic only a (random) subset of the users will transmit, and the high throughput is maintained. Although the actual performance of the scaling heuristic depends on the user correlation, the results suggest that it works as intended when the system is under high load.

4.5 Practical Aspects and Future Work

Throughout the paper we have assumed that all product expectations are known, and we have ignored the aspect of control overhead involved in scheduling the users. In most practical systems, the expectations need to be learned and it may be desired to limit the control overhead. The expectations can be learned using maximum likelihood estimation or Bayesian methods, although these may be challenged by the curse of dimensionality imposed by the high number of users. For a high number of users, it may be more suitable to apply methods from data mining, such as frequent itemset mining, where items that frequently occur together are tracked in an online manner [13]. Since frequent itemset mining only keeps track of the users that transmit most frequently together, it will not provide correlation estimates for all pairs of users. However, as the pairwise correlations are likely to be very sparse, this provides a way of compressing the estimates, assuming the

remaining infrequent correlations to be zero.

The proposed random access scheme induces a certain overhead to schedule the users. If the users are assigned single slots, it requires at least $N \log_2(K)$ bits assuming that the exact number of users and slots are known to the users, while in the general case where the users are assigned slot transmission probabilities at least $N(K-1)P$ bits where P is the number of bits used to encode a slot transmission probability and $K-1$ reflects the degrees of freedom in the slot assignment. However, in practical systems where the correlation is sparse, it is possible to reduce the amount of signaling by only scheduling some of the users. Suppose as an example that $M \ll N$ users are scheduled, then only $M \log_2(NK)$ bits are needed in the single slot case and $M(K-1)P \log_2(N)$ bits in the probabilistic case. Furthermore, if the activity changes over time, the information needs to be signaled more often. In this case, the users need to be rescheduled regularly. However, estimating new correlations when the users are already scheduled is challenging since the system faces the classical trade-off between exploration and exploitation. To this end, one may use reinforcement learning and the multi-armed bandit framework.

4.6 Conclusion

This paper has studied how correlated users affect random access protocols, and how information about the correlation can be exploited in the design of random access protocols. We present two algorithms that attempt to maximize the throughput using pairwise correlation information, and evaluate them in a scenario with correlated user activity. We show that the presented algorithms achieve considerably high throughput compared to traditional random access schemes. This suggests that taking correlation information into account in the random access protocols is promising in scenarios where users are likely to be correlated, such as in MTC.

Acknowledgment

This work has been supported in part by WILLOW (ERC Consolidator Grant no. 648382) and TACTILENet (Grant no. 690893), within the Horizon 2020 Program.

References

[1] F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski, "Five disruptive technology directions for 5G," *IEEE Communications Magazine*, vol. 52, no. 2, pp. 74–80, February 2014.

[2] 3GPP, "Study on RAN Improvements for Machine-type Communications," 3rd Generation Partnership Project (3GPP), Technical Report (TR) TR 37.868, October 2014, v.0.8.1.

[3] L. G. Roberts, "Aloha packet system with and without slots and capture," *SIGCOMM Comput. Commun. Rev.*, vol. 5, no. 2, pp. 28–42, Apr. 1975.

[4] E. Paolini, G. Liva, and M. Chiani, "Coded slotted aloha: A graph-based method for uncoordinated multiple access," *IEEE Transactions on Information Theory*, vol. 61, no. 12, pp. 6815–6832, Dec. 2015.

[5] C. Stefanovic and P. Popovski, "Aloha random access that operates as a rateless code," *IEEE Transactions on Communications*, vol. 61, no. 11, pp. 4653–4662, Nov. 2013.

[6] P. Flajolet and R. Sedgewick, *Analytic combinatorics*. cambridge University press, 2009.

[7] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Processing Magazine*, vol. 27, no. 3, pp. 20–34, May 2010.

[8] S. Poljak, F. Rendl, and H. Wolkowicz, "A recipe for semidefinite relaxation for (0,1)-quadratic programming," *Journal of Global Optimization*, vol. 7, no. 1, pp. 51–73, Jul. 1995.

[9] A. Beck, A. Ben-Tal, and L. Tetruashvili, "A sequential parametric convex approximation method with applications to nonconvex truss topology design problems," *Journal of Global Optimization*, vol. 47, no. 1, pp. 29–51, May 2010.

[10] G. Scutari, F. Facchinei, and L. Lampariello, "Parallel and distributed methods for constrained nonconvex optimization—part i: Theory," *IEEE Transactions on Signal Processing*, vol. 65, no. 8, pp. 1929–1944, Apr. 2017.

[11] M. Laner, P. Svoboda, N. Nikaein, and M. Rupp, "Traffic models for machine type communications," in *Wireless Communication Systems (ISWCS 2013), Proceedings of the Tenth International Symposium on*. VDE, 2013, pp. 1–5.

[12] E. W. Weisstein, "Circle-circle intersection. From MathWorld—A Wolfram Web Resource," visited 12/2/2018. [Online]. Available: http://mathworld.wolfram.com/Circle-CircleIntersection.html

[13] C. Giannella, J. Han, J. Pei, X. Yan, and P. S. Yu, "Mining frequent patterns in data streams at multiple time granularities," *Next generation data mining*, vol. 212, pp. 191–212, 2003.

Paper E

Massive Random Access with Common Alarm Messages (extended version)

Kristoffer Stern, Anders E. Kalør, Beatriz Soret, and Petar Popovski

Extended version (including supplementary material) of paper published in *Proceedings of 2019 IEEE International Symposium on Information Theory (ISIT)*, pp. 1–5, 2019, doi: 10.1109/ISIT.2019.8849678.

The layout has been revised.

Abstract

The established view on massive IoT access is that the IoT devices are activated randomly and independently. This is a basic premise also in the recent information-theoretic treatment of massive access by Polyanskiy [1]. In a number of practical scenarios, the information from IoT devices in a given geographical area is inherently correlated due to a commonly observed physical phenomenon. We introduce a model for massive access that accounts for correlation both in device activation and in the message content. To this end, we introduce common alarm messages for all devices. A physical phenomenon can trigger an alarm causing a subset of devices to transmit the same message at the same time. We develop a new error probability model that includes false positive errors, resulting from decoding a non-transmitted codeword. The results show that the correlation allows for high reliability at the expense of spectral efficiency. This reflects the intuitive trade-off: an access from a massive number can be ultra-reliable only if the information across the devices is correlated.

5.1 Introduction

The interconnection of billions of devices within the Internet of Things (IoT) paradigm is one of the main challenges for future networks. Accordingly, the service structure of 5G, fully aligned with the ITU-R vision for IMT-2020, includes the massive Machine Type-Communication (mMTC) as one of the three core connectivity types. mMTC is typically defined through a scenario in which a massive number of IoT devices are connected to a Base Station (BS). The activation of the IoT devices is intermittent, such that at a given time, the IoT devices that are active and have a message to send constitute a random subset from the total set of devices [2]. A main use case for IoT is a distributed sensor network that intelligently monitors and manages a large number of devices [3]. The traffic in such systems can be (quasi-)periodic or event-driven [4]. In addition, source information and time correlations occur when many devices are sensing a common physical phenomenon.

The conventional multiple access channel (MAC) has been well characterized [5–7]. The main results here are derived using the fact that the probability of successful joint decoding goes asymptotically to one with increasing blocklength. However, in the context of mMTC the devices have small data payloads. Even though a small subset of the devices are active simultaneously, the large total number of devices (up to $3.000\,00 \times 10^5$ in a single cell [8]) means that the number of active devices can still be comparable to the blocklength. This results in finite blocklength (FBL) effects. A number of works have addressed the problem of massive access [8, 9]. However, in terms of theoretical rigor and fundamental results two works stand out, both of them assuming independent traffic. The first one is on the many-access

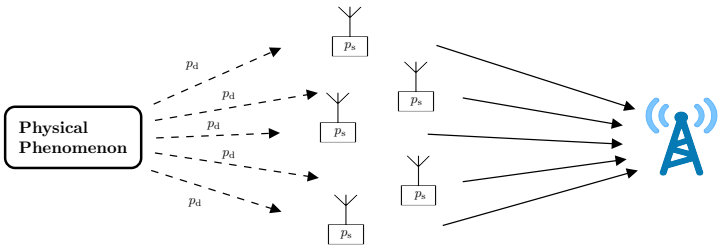


Fig. E.1: System model with common alarm and standard messages. p_d denotes the probability of detecting an alarm, and p_s is the probability of sending a standard message.

channel by X. Chen et al. [10]. This paper shows the scaling of the number of users with the blocklength. On the other hand, Y. Polyanskiy provides a model in [1] that is closer to the way massive access is commonly understood. Key elements of the model are devices employing the same codebook which precludes the identification of users and the error measure is done on a per-device basis. This has also been called unsourced random access [11].

In this we build upon the model in [1] with an important extension: we bring in the correlation of activation and message content across different devices. This is different from the mainstream view on massive random access, where the device activation and message content is independent across the devices. An exemplary case is as follows: IoT devices can send standard messages or alarm messages, the latter with critical reliability requirement and triggered by a commonly observed phenomenon. In normal operation, standard uncorrelated messages are sent. Upon the alarm activation, a number of IoT devices will prioritize it and send the *same* message. This reflects the extreme all-or-nothing correlation where devices are either mutually independent, or they are completely correlated both in source information and in time. Our model intends to capture the following intuitive observation. If the number of devices that transmit the same alarm message increases, then the reliability of the alarm message increases at the expense of the decrease of the total amount of information that comes from the total population of connected IoT devices. The model can be seen as having an (alarm) event that needs to be communicated through a random subset of devices, see Fig. E.1. By removing the alarm event the model boils down to the model in [1].

Differently from previous works, the per-device probability of error is not meaningful for devices transmitting the alarm event in our model. Instead, the common alarm itself can be seen as a "ghost" device, which communicates through the actual IoT devices (see Fig. E.1) and we calculate the error probability with respect to this ghost device. In addition, the fact that we consider two message types (standard and alarm messages) necessitates the introduction of false positive errors, namely decoding a codeword that was

Notation	
N:	Total number of devices
K:	Number of active devices
K_a:	Number of devices sending an alarm message
p_{a}:	Alarm probability
p_{s}:	Standard message probability
p_{d}:	Alarm detection probability
$\mathcal{M}_{\mathrm{s}}$:	Set of standard messages
$\mathcal{M}_{\mathrm{a}}$:	Set of alarm messages
n:	Blocklength
P':	Average transmission power
P:	Maximal transmission power
ϵ_{s}:	Target probability of error for standard messages without an alarm
A:	Alarm event
ϵ_{a}:	Target probability of error for alarm messages
ϵ_{sa}:	Target probability of error for standard messages in alarm event
ϵ_{fp}:	Target probability of false positive
S:	Spectral efficiency
H:	Entropy
W_j	Message transmitted by the j-th device
a_i^j (lower case):	$(a_i, \ldots, a_j)$, $i \leq j$ for scalars/vectors
X_i^j (upper case):	$(X_i, \ldots, X_j)$, $i \leq j$ for random variables/vectors
a_i^{i-1}:	Empty tuple
$\sum_{i=j}^{j-1} a_j$:	0
$[\mathcal{S}]^k$	Set of all k-subsets of the set $\mathcal{S}$
$\mathcal{X}$:	$\in \mathbb{R}^n$ Input alphabet
$\mathcal{Y}$:	$\in \mathbb{R}^n$ Output alphabet

Table E.1: Notation used throughout this paper.

not transmitted. In the system model in Fig. E.1, decoding an alarm message when no alarm has occurred is critical. This type of error is, typically, not considered in a common communication-theoretic setting, where an error is defined as the event in which a decoder is not decoding a codeword correctly.

The rest of the paper is organized as follows. Section 5.2 introduces the system model including the source information and time correlations. In Section 5.3 the entropy and the spectral efficiency of the correlated devices is derived. Section 5.4 defines the alarm random access code based on the novel error model, and the error bound is derived in Section 5.5. Finally, numerical evaluations are presented in Section 5.6, and concluding remarks are given in Section 5.7. Table E.1 lists the notation used in this paper.

5.2 Correlation Model

We consider the uplink in a random access channel in which each access opportunity is a block of n channel uses. In each block, K out of N devices

transmit a message from one of the two disjoint message sets $\mathcal{M}_\mathrm{s}$ and $\mathcal{M}_\mathrm{a}$, consisting of $M_\mathrm{s} = |\mathcal{M}_\mathrm{s}|$ standard messages and $M_\mathrm{a} = |\mathcal{M}_\mathrm{a}|$ alarm messages, respectively. A typical case is having a stringent reliability requirement for the alarm messages, and a high throughput and massive access requirement for the rest. As also done in [1], we assume that the number of active devices, K, is known by the receiver.

Let $P_{\boldsymbol{Y}|\boldsymbol{X}_1^K} : [\mathcal{X}^n]^K \to \mathcal{Y}^n$ be a memoryless multiple access channel (MAC) satisfying permutation invariance where $\mathcal{X}, \mathcal{Y}$ are the input and output alphabets. That is, the distribution $P_{\boldsymbol{Y}|\boldsymbol{X}_1^K}(\cdot|\boldsymbol{x}_1^K)$ coincides with $P_{\boldsymbol{Y}|\boldsymbol{X}_1^K}(\cdot|\boldsymbol{x}_{\pi(1)}, \ldots, \boldsymbol{x}_{\pi(K)})$ for any $\boldsymbol{x}_1^K \in [\mathcal{X}^n]^K$ and any permutation π. This assumption relates to the fact that no user identification is done at the receiver, i.e. unsourced random access [11]. Therefore, all devices use the same encoder $f : \mathcal{M}_\mathrm{s} \cup \mathcal{M}_\mathrm{a} \to \mathcal{X}^n$ and the receiver decodes according to the possibly randomized map $g : \mathcal{Y}^n \to [\mathcal{M}_\mathrm{s} \cup \mathcal{M}_\mathrm{a}]^{K-K_\mathrm{a}+1}$, where K_a is the random number of devices that send alarm messages and $[\mathcal{S}]^k$ denotes the set of all k-subsets of the set $\mathcal{S}$.

We denote the message transmitted by the j-th device as W_j. The transmitted messages are chosen according to the following model: An alarm event, A, occurs with probability p_a, and there is no alarm with probability $1 - p_\mathrm{a}$. If no alarm occurs then the system acts as in [1], i.e. each device transmits a message uniformly chosen from $\mathcal{M}_\mathrm{s}$ with probability p_s, and it is silent with probability $1 - p_\mathrm{s}$. If an alarm occurs, with probability p_d a device will detect it and transmit an alarm message. Contrary to the standard messages, all devices detecting the alarm send the *same* message chosen uniformly from $\mathcal{M}_\mathrm{a}$. With probability $1 - p_\mathrm{d}$ the device will act as if no alarm has occurred. It follows that $\mathbb{P}[W_j \in \mathcal{M}_\mathrm{a}] = p_\mathrm{a} p_\mathrm{d}$ and $\mathbb{P}[W_j \in \mathcal{M}_\mathrm{s}] = p_\mathrm{s} - p_\mathrm{a} p_\mathrm{s} p_\mathrm{d}$. Notice that the probability p_d in our model is the joint event of detecting an alarm and deciding to transmit a corresponding alarm message. The latter can be seen as a system design parameter and its impact to the system performance, particularly in the tradeoff between reliability and spectral efficiency, is discussed in next section.

In contrast to practical random access scenarios, we assume that the number of active devices, K, is known by the receiver. This assumption can be justified by noting that K could be estimated using the same procedure as in [12]. Specifically, the base station can decode the received packets under the assumption that $K = 0, 1, \ldots, N$, and then re-generate the resulting packets and subtract them from the received signal. K can then be determined based on the residual, which will equal the noise $\mathbf{Z}$, see (E.12) if the correct value of K has been determined. Furthermore, since the number of alarm messages, K_a, is assumed unknown in the model, an incorrectly estimated K will mainly affect the decoding of the non-critical standard messages.

5.3 Spectral efficiency

In this section, we study how the presence of common alarm messages affects the information transmitted in the system. We consider the system spectral efficiency defined as $S = \frac{H(W_1^K)}{n}$, where K is the number of devices transmitting messages $W_1, ...W_K$, H is the joint entropy function and n is the blocklength.

The total number of devices, N, in the network affects the system spectral efficiency. To see this, consider the case with a high alarm detection probability p_d, a low p_s, alarm probability $p_\mathrm{a} = 0.5$, and suppose we receive 10 messages, i.e. $K = 10$. If also $N = 10$, then there is a high probability that an alarm has occurred since we know that all devices transmitted and that p_d is high. Moreover, in this case all devices have most likely transmitted the same message, resulting in a low spectral efficiency. On the other hand, with $N = 10000$ devices in the network the probability that an alarm has occurred is low, being unlikely that 9990 devices do not detect an alarm when p_d is high. In this case, the messages are likely to be distinct, resulting in a high spectral efficiency.

The exact expression for the system spectral efficiency for this model is stated in Theorem 1.

Theorem 1. *For K out of N received messages and correlated devices as describe in Section 5.2 the system spectral efficiency, S, is*

$$S = \frac{1}{n} \sum_{k=1}^{K} H(W_k | W_1^{k-1}), \tag{E.1}$$

where $H(W_k | W_1^{k-1})$ *is given by*

$$\begin{aligned} H(W_k | W_1^{k-1}) = \; & (B_0 + B_1) \sum_{i=1}^{k-1} \binom{k-1}{i} p_\mathrm{a} p_\mathrm{d}^i ((1-p_\mathrm{d}) p_\mathrm{s})^{k-1-i} N_0 \\ & - B_2 \left(B_3 \log_2 \frac{B_3}{M_\mathrm{a}} + (1 - B_3) \log_2 \frac{1 - B_3}{M_\mathrm{s}} \right), \end{aligned} \tag{E.2}$$

and

$$N_0 = \frac{(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K-(k-1)}(1-p_\mathrm{d})^{N-K}}{p_\mathrm{a}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K}(1-p_\mathrm{d})^{N-K} + (1-p_\mathrm{a})p_\mathrm{s}^K}, \tag{E.3}$$

$$B_0 = -\frac{p_\mathrm{d}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \log_2 \left(\frac{p_\mathrm{d}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \right), \tag{E.4}$$

$$B_1 = \frac{(1-p_\mathrm{d})p_\mathrm{s}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \left(\log_2 M_\mathrm{s} - \log_2 \left(\frac{(1-p_\mathrm{d})p_\mathrm{s}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \right) \right), \tag{E.5}$$

$$B_2 = \frac{p_\mathrm{a}(1-p_\mathrm{d})^{N-K+(k-1)}p_\mathrm{s}^{k-1}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K-(k-1)} + (1-p_\mathrm{a})p_\mathrm{s}^K}{p_\mathrm{a}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K}(1-p_\mathrm{d})^{N-K} + (1-p_\mathrm{a})p_\mathrm{s}^K}, \tag{E.6}$$

$$B_3 = \frac{p_\mathrm{a}p_\mathrm{d}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K-k}(1-p_\mathrm{d})^{N-K+k-1}p_\mathrm{s}^{k-1}}{p_\mathrm{a}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K-k+1}(1-p_\mathrm{d})^{N-K+k-1}p_\mathrm{s}^{k-1} + (1-p_\mathrm{a})p_\mathrm{s}^K}. \tag{E.7}$$

Proof of Theorem 1 can be found in Appendix A.

For $p_\mathrm{a} = 0$ or $p_\mathrm{d} = 0$ (i.e. no correlation) the system spectral efficiency is the well-known $\frac{K}{n}\log_2 M_\mathrm{s}$ as in [1].

5.4 Alarm Random Access Codes

We now define a random access code that allows for reliability diversity for standard and alarm messages. This entails having different error events for the two message types. Specifically, in order to capture the characteristics of alarm messages, we introduce reliability constraints that relates to the certainty of decoding alarm messages in the event of an alarm, but also to the certainty of *not* decoding alarm messages when no alarms has occurred (false positives). This is different from usual analysis since we need not only consider the event of incorrectly decoding a message, but also the type of message that is decoded instead.

The error events are listed in Table E.2, where we have included the "No error" column to emphasize the opposite characteristics of alarm messages and standard messages. We define error events for standard messages as in [1], i.e. errors are considered per-device and the event that more than one device sends the same standard message results in an error. In contrast, no error occurs if multiple devices transmit the same alarm message. Similarly, decoding distinct alarm messages also results in an error since only one alarm is assumed to be active at a time, while decoding distinct standard messages is naturally not an error. Formally, we define the following error events: $E_j \triangleq \{W_j \notin g(\boldsymbol{Y})\} \cup \{W_j = W_i \text{ for some } i \neq j\}$ is the event of not decoding the message from the j-th device, $E_\mathrm{a} \triangleq \{W_0 \notin g(\boldsymbol{Y})\} \cup \{|g(\boldsymbol{Y}) \cap \mathcal{M}_\mathrm{a}| > 1\}$ for $W_0 \in \mathcal{M}_\mathrm{a}$ is the event of not decoding an alarm message or decoding more

	Classification of events	
	Error	**No error**
No alarm	- A standard message is not decoded: $\{\mathcal{M}_\mathrm{s} \ni W_j \notin g(\boldsymbol{Y})\}$ - More than one device sends the same message: $\{W_j = W_i \text{ for some } i \neq j\}$ - At least one alarm message is decoded: $\{g(\boldsymbol{Y}) \cap \mathcal{M}_\mathrm{a} \neq \varnothing\}$ (false positive)	- A standard message is decoded: $\{\mathcal{M}_\mathrm{s} \ni W_j \in g(\boldsymbol{Y})\}$ - Different messages are sent: $\{W_j \neq W_i \; \forall \; i \neq j\}$ - No alarm message is decoded: $\{g(\boldsymbol{Y}) \cap \mathcal{M}_\mathrm{a} = \varnothing\}$ (true negative)
Alarm	- The alarm message is not decoded: $\{W_0 \notin g(\boldsymbol{Y})\}$ (false negative) - More than one alarm message is decoded: $\{\lvert g(\boldsymbol{Y}) \cap \mathcal{M}_\mathrm{a}\rvert > 1\}$ - A standard message is not decoded: $\{\mathcal{M}_\mathrm{s} \ni W_j \notin g(\boldsymbol{Y})\}$ - Two or more device sends the same standard message: $\{W_i = W_j \in \mathcal{M}_\mathrm{s} \text{ for some } i \neq j\}$	- The alarm message is decoded: $\{W_0 \in g(\boldsymbol{Y})\}$ (true positive) - More than one device sends the same alarm message: $\{W_j = W_i = W_0 \in \mathcal{M}_\mathrm{a} \text{ for some } i \neq j\}$

Table E.2: Error events in the considered system. In the alarm event we denote the alarm message by W_0.

than one, and $E_\mathrm{fp} \triangleq \{g(\boldsymbol{Y}) \cap \mathcal{M}_\mathrm{a} \neq \varnothing\}$ is the event of decoding any alarm message (which is an error when no alarm has occurred). This leads to the following definition of a K-user alarm random access (ARA) code.

Definition 1. *An $(M_\mathrm{s}, M_\mathrm{a}, n, \epsilon_\mathrm{a}, \epsilon_\mathrm{s}, \epsilon_\mathrm{sa}, \epsilon_\mathrm{fp})$ alarm random access (ARA) code for the K-user channel $P_{\boldsymbol{Y}|\boldsymbol{X}_1^K}$ is a pair of (possibly randomized) maps, the encoder $f : \mathcal{M}_\mathrm{s} \cup \mathcal{M}_\mathrm{a} \to \mathcal{X}^n$, and the decoder $g : \mathcal{Y}^n \to [\mathcal{M}_\mathrm{s} \cup \mathcal{M}_\mathrm{a}]^{K-K_\mathrm{a}+1}$ satisfying*

$$\mathbb{P}\left[E_\mathrm{a}|A\right] \leq \epsilon_\mathrm{a}, \tag{E.8}$$

$$\frac{1}{K}\sum_{j=1}^{K} \mathbb{P}\left[E_j|\neg A\right] \leq \epsilon_\mathrm{s}, \tag{E.9}$$

$$\mathbb{E}[K_\mathrm{a}]\frac{1}{K-K_\mathrm{a}}\sum_{j=1}^{K-K_\mathrm{a}} \mathbb{P}\left[E_j|A\right] \leq \epsilon_\mathrm{sa}, \tag{E.10}$$

$$\mathbb{P}\left[E_\mathrm{fp}|\neg A\right] \leq \epsilon_\mathrm{fp}, \tag{E.11}$$

where $\boldsymbol{X}_j = f(W_j)$, $W_1, \ldots, W_K \in \mathcal{M}_\mathrm{s}$ when there is no alarm and $W_1, \ldots, W_{K-K_\mathrm{a}} \in \mathcal{M}_\mathrm{s}$, $W_{K-K_\mathrm{a}+1} = \ldots = W_K = W_0 \in \mathcal{M}_\mathrm{a}$ in the alarm event for a random number, K_a, alarm messages.

The left hand side of (E.8) is the probability of not decoding or resolving the alarm message in the alarm event. The left hand side of (E.9) is the average

per-device error probability when there is no alarm, and (E.10) refers to the case when there is an alarm. Lastly left hand side of (E.11) is the probability of false positives. In a practical scenario the entities ϵ_{a}, ϵ_{s}, ϵ_{sa} and ϵ_{fp} can be treated as reliability requirements, in which case the achievability of an ARA code is of interest.

In the remainder of the paper we limit the analysis to the Gaussian MAC (GMAC) given by

$$\boldsymbol{Y} = \sum_{m=1}^{K} \boldsymbol{X}_m + \boldsymbol{Z}. \tag{E.12}$$

where $\boldsymbol{Z} \in \mathbb{R}^n$ is a standard Gaussian noise vector. Additionally a maximal average transmission power, P, is included. That is we require $\left\|f(W_j)\right\|_2^2 \leq nP$. This model is based on the assumption that the blocklength is short enough to be within the coherence time of the channel. This allows for the devices to do channel inversion and precode their signals so that they add up coherently at the receiver. This gives the possibility of a very high reliability for alarm messages.

5.5 Random Coding Error Bound

The achievability conditions for an ARA code are presented in Theorem 2, which provides bounds for the error probabilities ϵ_{a}, ϵ_{s}, ϵ_{sa} and ϵ_{fp} for a given blocklenght n, message set sizes M_{a} and M_{s}, average transmission power P', and maximal transmission power P.

Theorem 2. *Fix $P' < P$. There exists an $(M_{\mathrm{a}}, M_{\mathrm{s}}, n, \epsilon_{\mathrm{a}}, \epsilon_{\mathrm{s}}, \epsilon_{\mathrm{sa}}, \epsilon_{\mathrm{fp}})$ alarm random access code for the K-user GMAC satisfying power-constraint P and*

$$\epsilon_{\mathrm{a}} \leq \sum_{K_{\mathrm{a}}=0}^{K} p_{K_{\mathrm{a}}}(K_{\mathrm{a}}) a(K, K_{\mathrm{a}}) + p_0, \tag{E.13}$$

$$\epsilon_{\mathrm{s}} \leq b(K) + c(K) - b(K)c(K), \tag{E.14}$$

$$\epsilon_{\mathrm{sa}} \leq \sum_{K_{\mathrm{a}}=0}^{K} p_{K_{\mathrm{a}}}(K_{\mathrm{a}}) \left(1 - d\left(K, K_{\mathrm{a}}\right)\left(1 - c\left(K - K_{\mathrm{a}}\right)\right)\right) \tag{E.15}$$

$$\epsilon_{\mathrm{fp}} \leq b(K). \tag{E.16}$$

Defining $\phi(k, \alpha) = \frac{1}{2}\ln(1 + 2kP'\alpha)$ and $\Phi(k, \alpha) = \frac{\alpha}{1+2kP'\alpha}$, then related to (E.13)*:*

$$p_{K_{\mathrm{a}}}(k) = \binom{K}{k} \frac{p_{\mathrm{d}}^k \left(\left(1 - p_{\mathrm{d}}\right) p_{\mathrm{s}}\right)^{K-k}}{\left(p_{\mathrm{d}} + (1 - p_{\mathrm{d}}) p_{\mathrm{s}}\right)^K}, \tag{E.17}$$

$$a(K, K_{\mathrm{a}}) = \min\left(\sum_{K'_{\mathrm{a}}=0}^{K} e^{-nE_{\mathrm{a}}},\, 1\right), \tag{E.18}$$

$$p_0 = \mathbb{P}\left[\frac{1}{n}\sum_{i=1}^{n} Z_i^2 > \frac{P}{P'}\right], \tag{E.19}$$

$$E_{\mathrm{a}} = \max_{0\le\rho\le 1, 0<\lambda_{\mathrm{a}}} -\frac{\rho}{n}\ln(M_{\mathrm{a}}-1) + \xi_{\mathrm{a}}, \tag{E.20}$$

$$\xi_{\mathrm{a}} = \rho\phi(K_{\mathrm{a}}'^2, \lambda_{\mathrm{a}}) + \phi(K_{\mathrm{a}}^2, \rho\beta_{\mathrm{a}}) + \phi(K-K_{\mathrm{a}}, \gamma_{\mathrm{a}}) + \phi(1/P', \psi_{\mathrm{a}}), \tag{E.21}$$

$$\psi_{\mathrm{a}} = \Phi(K-K_{\mathrm{a}}, \gamma_{\mathrm{a}}),\ \ \gamma_{\mathrm{a}} = \Phi(K_{\mathrm{a}}^2, \rho\beta_{\mathrm{a}}) - \rho\lambda_{\mathrm{a}},\ \ \beta_{\mathrm{a}} = \Phi(K_{\mathrm{a}}'^2, \lambda_{\mathrm{a}}). \tag{E.22}$$

Related to (E.16)*:*

$$b(K) = \min\left(\sum_{K_{\mathrm{a}}'=1}^{K} e^{-nE_{\mathrm{fp}}},\ 1\right), \tag{E.23}$$

$$E_{\mathrm{fp}} = \max_{0\le\rho\le 1,\ 0<\lambda_{\mathrm{fp}}} -\frac{\rho}{n}\ln(M_{\mathrm{a}}) + \xi_{\mathrm{fp}}, \tag{E.24}$$

$$\xi_{\mathrm{fp}} = \rho\phi(K_{\mathrm{a}}'^2, \lambda_{\mathrm{fp}}) + \phi(K, \rho\beta_{\mathrm{fp}}) + \phi(1/P', \gamma_{\mathrm{fp}}), \tag{E.25}$$

$$\gamma_{\mathrm{fp}} = \Phi(K, \rho\beta_{\mathrm{fp}}),\ \ \beta_{\mathrm{fp}} = \Phi(K_{\mathrm{a}}'^2, \lambda_{\mathrm{fp}}) - \lambda_{\mathrm{fp}}, \tag{E.26}$$

Related to (E.14)

$$c(K) = \sum_{t=1}^{K} \frac{t}{K}\min(p_t, q_t) + \frac{\binom{K}{2}}{M_{\mathrm{s}}} + Kp_0, \tag{E.27}$$

$$p_t = e^{-nE_t}, \tag{E.28}$$

$$E_t = \max_{0\le\rho,\rho_1\le 1} -\rho\rho_1 tR_1 - \rho_1 R_2 + E_0(\rho,\rho_1), \tag{E.29}$$

$$E_0(\rho,\rho_1) = \rho\rho_1\phi(t,\lambda_{\mathrm{s}}) + \rho_1\phi(t,\mu) + \frac{1}{2}\ln(1-2b\rho_1), \tag{E.30}$$

$$b = \rho\lambda_{\mathrm{s}} - \Phi(t,\mu),\ \ \mu = \rho\Phi(t,\lambda_{\mathrm{s}}),\ \ \lambda_{\mathrm{s}} = \frac{P't-1+\sqrt{D}}{4(1+\rho_1\rho)P't}, \tag{E.31}$$

$$D = (P't-1)^2 + 4P't\frac{1+\rho\rho_1}{1+\rho}, \tag{E.32}$$

$$R_1 = \frac{1}{n}\ln(M_s) - \frac{1}{nt}\ln(t!),\ \ R_2 = \frac{1}{n}\ln\binom{K}{t}, \tag{E.33}$$

$$q_t = \inf_{\gamma_{\mathrm{s}}} \mathbb{P}\left[I_t \le \gamma_{\mathrm{s}}\right] + e^{n(tR_1+R_2)-\gamma_{\mathrm{s}}}, \tag{E.34}$$

$$I_t = \min_{S_0\in[\mathcal{M}_{\mathrm{s}}]^t} i_t\left(\sum_{W\in S_0}\boldsymbol{c}_W; \boldsymbol{Y} \middle| \sum_{W\in S_0^c}\boldsymbol{c}_W\right), \tag{E.35}$$

$$i_t(\boldsymbol{a};\boldsymbol{y}|\boldsymbol{b}) = nC_t + \frac{\ln e}{2}\left(\frac{\|\boldsymbol{y}-\boldsymbol{b}\|_2^2}{1+P't} - \|\boldsymbol{y}-\boldsymbol{a}-\boldsymbol{b}\|_2^2\right), \tag{E.36}$$

where $C_t = \phi(1/2, t)$*,* $S_0 \in [\mathcal{M}_{\mathrm{s}}]^t$ *is a t-subset of true standard messages and*

$\boldsymbol{c}_W \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_n P')$ *is codeword corresponding to message* W. *Related to* (E.15)*:*

$$d(K, K_\mathrm{a}) = (1 - (a(K, K_\mathrm{a}) + p_0))(1 - e(K, K_\mathrm{a}) + p_0), \tag{E.37}$$

$$e(K, K_\mathrm{a}) = \min \left(\sum_{\substack{K'_\mathrm{a}=0 \\ K'_\mathrm{a} \neq K_\mathrm{a}}}^{K} e^{-nE_\mathrm{sa}},\ 1 \right), \tag{E.38}$$

$$E_\mathrm{sa} = \max_{0<\lambda_\mathrm{sa}} \phi((K_\mathrm{a} - K'_\mathrm{a})^2, \lambda_\mathrm{sa}) + \phi(K - K_\mathrm{a}, \beta_\mathrm{sa}) + \phi(1/P', \gamma_\mathrm{sa}), \tag{E.39}$$

$$\gamma_\mathrm{sa} = \Phi\left(K - K_\mathrm{a}, \beta_\mathrm{sa}\right), \quad \beta_\mathrm{sa} = \Phi\left((K_\mathrm{a} - K'_\mathrm{a})^2, \lambda_\mathrm{sa}\right) - \lambda_\mathrm{sa}. \tag{E.40}$$

Proof of Theorem 2 can be found in Appendix B.

5.6 Numerical evaluation

The bounds in Theorem 2 are given for a fixed number of active devices, K, but the probability of a given value of K depends on whether an alarm has happened or not. Therefore, we consider the average bound over the distribution of K conditioned on the alarm state and the total number of devices, N. The distribution of K given an alarm is

$$p_K(k|A) = \binom{N}{k} (p_\mathrm{d} + (1 - p_\mathrm{d})p_\mathrm{s})^k (1 - p_\mathrm{d})^{N-k} (1 - p_\mathrm{s})^{N-k}, \tag{E.41}$$

and the distribution of K given no alarm is

$$p_K(k|\neg A) = \binom{N}{k} p_\mathrm{s}^k (1 - p_\mathrm{s})^{N-k}. \tag{E.42}$$

We first study the trade-off between the probability of error for alarm messages and the per-device spectral efficiency S, during the event of an alarm. We consider a setting with $N = 1.000 \times 10^3$ devices and a blocklength of $n = 3.0000 \times 10^4$. The alarm and standard messages are 3 and 1.00×10^2 bits, respectively. The probability of activation when there is no alarm is $p_\mathrm{s} = 0.01$, and the transmission power is chosen such that the target average error bound for standard messages is $\epsilon_\mathrm{s} = 10^{-1}$, and the probability of false positive alarms is below $\epsilon_\mathrm{fp} = 10^{-5}$. Having only a few bits for alarm messages is a realistic setting, e.g. in a sensor network the alarm could be that a sensed value is too high or too low resulting in only one bit needed for the alarm message.

In Fig. E.2 it can be seen that the probability of error increases for increasing spectral efficiency (decreasing p_d). Notice that the maximum spectral efficiency is achieved when the error probability is one (or equivalently, $p_\mathrm{d} = 0$),

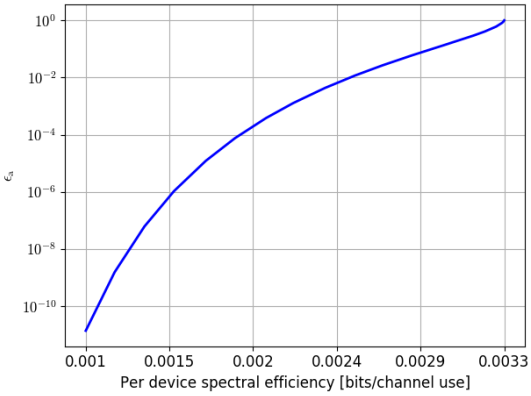


Fig. E.2: Trade-off between probability of error for alarm messages and the spectral efficiency. Blocklength $n = 3.0000 \times 10^4$, $N = 1.000 \times 10^3$, target error probabilities $\epsilon_s = 10^{-1}$, $\epsilon_{fp} = 10^{-5}$, set sizes $M_s = 2^{100}$, $M_a = 2^3$, $p_s = 0.01$ and $p_a = 1$.

i.e. no alarm messages are detected. This is expected since a higher number of devices transmitting alarm messages reduces the per-device spectral efficiency, but increases the received signal-to-noise ratio of alarm messages. Furthermore, very high reliability is achievable. This trade-off between spectral efficiency and probability of error is not surprising since this is also the case when the blocklength or message set size are changed. The novelty is in the fact that it is the correlation between devices that causes the trade-off.

We now consider the minimal average transmission power, P', required to satisfy some target error probabilities. We assume no power restriction. That is $p_0 = 0$ in Theorem 2. Let all parameters be fixed except P' and p_d. This is an optimization problem on the form

$$
\begin{aligned}
&\underset{\substack{0\leq P' \\ 0\leq p_d\leq 1}}{\text{minimize}} && P' \\
&\text{subject to} \\
&&\sum_{K=0}^{N} p_K(K|\neg A)(b(K) + c(K) - b(K)c(K)) \leq \epsilon_s \\
&&\sum_{K=0}^{N} p_K(K|\neg A)b(K) \leq \epsilon_{fp} \\
&&\sum_{K=0}^{N} p_K(K|A) \sum_{K_a=0}^{K} p_{K_a}(K_a)a(K, K_a) \leq \epsilon_a \\
&&\sum_{K=0}^{N} p_K(K|A) \sum_{K_a=0}^{K} p_{K_a}(K_a)(1 - d(K, K_a)(1 - c(K - K_a))) \leq \epsilon_{sa}
\end{aligned}
\tag{E.43}
$$

The constraint functions are strictly increasing for decreasing values of P' and p_{d}, and the first two constraints do not depend on p_{d}. Therefore, the problem can be efficiently solved using bisection by first minimizing P' subject to the first two constraints, and then determine p_{d} from the last two constraints using the P' from the previous step. This provides a feasible solution for the values of N, ϵ_{a} and ϵ_{sa} of interest in this paper. However, if no feasible $p_{\mathrm{d}} \in [0, 1]$ can be obtained from the second step, then a solution can be found by setting $p_{\mathrm{d}} = 1$ and minimizing P' subject to the last two constraints.

We use the same system parameters as in the previous scenario, except that we now fix $\epsilon_{\mathrm{a}} = \epsilon_{\mathrm{fp}} = 10^{-5}$ and $\epsilon_{\mathrm{s}} = \epsilon_{\mathrm{sa}} = 10^{-1}$. Based on the optimal p_{d} and the values of p_{s}, p_{a}, we evaluate the minimal average energy-per-bit $\mathbb{E}[p_K] \frac{E_0}{N_0} = \frac{nP'}{2\,\mathbb{E}[p_K] H(W_1^K)/K}$.

In Fig. E.3 the solid blue line shows the energy-per-bit as a function of total number devices, N, for this setup. Notice that optimization is done for each N. Additionally, the achievable energy-per-bit for the uncorrelated case ($p_{\mathrm{d}} = 0$) is included for reference, and is obtained as described in [1] but without the transmission power restriction. It can be seen that almost the same energy-per-bit is achievable for correlated and uncorrelated devices up to approximately 13000 devices, where the energy-per-bit required in the correlated case starts to increase significantly. This is due to the fact that the bound for false positives starts to dominate the choice of P'. Thus, due to high multi-access interference, the probability of decoding a false positive is higher than the probability of failing to decode a standard message. The increase in power needed to accommodate the false positive target probability ϵ_{fp} causes the error probability of standard messages to go well below their target error probability ϵ_{s}. In fact, with more than 17500 devices the error bound for standard messages is also approximately 10^{-5}. This is similar to the behavior in the uncorrelated case where the finite blocklength penalty is the dominating constraint when N is small, while multi-access interference dominates for large N [1]. This is seen in the increase in the slope at around 15000 devices in the uncorrelated case.

The effect of increasing alarm probability, p_{a}, can be seen as the dashed curves in Fig. E.3. The energy-per-bit is higher for larger p_{a} due to the increased rate of alarm events where spectral efficiency is lower. The energy-per-bit in alarm events corresponds to the curve for $p_a = 1$. Notice that the energy requirement P' and the probability p_{d} are not altered by varying p_{a} since the error probabilities for ARA codes are conditioned on the occurrence of an alarm. The high energy-per-bit for small N and high p_{a} is due to the large number of devices (relative to N) that must devote their resources to a single alarm message in order to accommodate the target alarm reliability. In general, the curves corresponding to different values of p_{a} are approaching each other for increasing N. This is caused by the almost constant number of

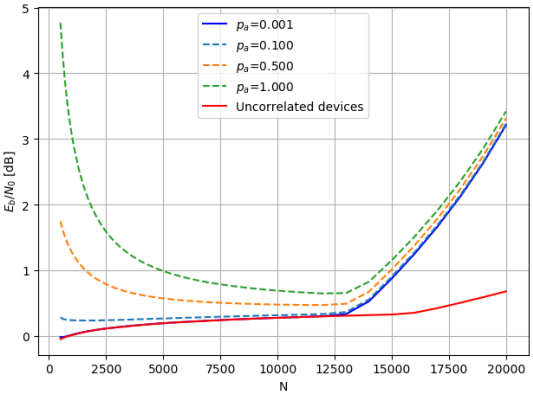


Fig. E.3: Trade-off between $\frac{E_b}{N_0}$ and the number of devices, N, for different values of alarm probability p_a and for uncorrelated devices. Blocklength $n = 3.0000 \times 10^4$, target error probabilities $\epsilon_\mathrm{a} = \epsilon_\mathrm{fp} = 10^{-5}$, $\epsilon_\mathrm{s} = \epsilon_\mathrm{sa} = 10^{-1}$, set sizes $M_\mathrm{s} = 2^{100}$, $M_\mathrm{a} = 2^3$ and $p_\mathrm{s} = 0.01$.

alarm messages required to achieve the alarm target reliability. For increasing N this ratio of alarm messages to standard messages grows and the traffic will be mostly standard messages.

5.7 Conclusions

We have studied the trade-off between reliability and spectral efficiency in a massive random access scenario where the devices can send standard messages or alarm messages. The alarm messages are triggered by a common physical phenomenon and introduce correlation in both the transmitted messages and the activation of devices. We derive the system spectral efficiency and propose an achievability bound for alarm random access codes. We show that very reliable transmissions of alarm messages can be achieved, but that the correlation causes a trade-off in spectral efficiency. In particular, when the multi-access interference is moderate, the cost of providing high reliability of alarm messages is small in terms of the average energy-per-bit. However, when multi-access interference is high, the probability of decoding a false positive alarm message dominates the error probabilities, and the cost of providing high reliability is significant.

A Proof of Theorem 1

To explicitly show the dependency on the number of messages define

$$T_K^N = \{W_1 \in \mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}\} \cup \cdots \\ \cdots \cup \{W_K \in \mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}\} \cup \{W_{K+1} \in \varnothing\} \cup \cdots \cup \{W_N \in \varnothing\} \quad \text{(E.44)}$$

as the event that the first K out of N devices transmit and the rest are silent. Due to symmetry in the devices, and without loss of generality, we assume the K first devices that are transmitting. By the law of total probability this event has probability

$$p(T_K^N) = p_\mathrm{a}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^K(1-p_\mathrm{d})^{N-K}(1-p_\mathrm{s})^{N-K} \\ + (1-p_\mathrm{a})p_\mathrm{s}^K(1-p_\mathrm{s})^{N-K}, \quad \text{(E.45)}$$

System spectral efficiency, S, is defined as $S = H(W_1^K)/n$ where the joint entropy of all K messages can be expressed using the chain rule for entropy [13, Theo. 2.5.1] as

$$H(W_1^K) = \sum_{k=1}^{K} H(W_k|W_1^{k-1}). \quad \text{(E.46)}$$

Thus we need to express the conditional entropy $H(W_k|W_1^{k-1})$ given by

$$H(W_k|W_1^{k-1}) = \sum_{w_1 \in \mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}} \cdots \sum_{w_{k-1} \in \mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}} p(w_1^{k-1}|T_K^N) H(W_k|W_1^{k-1} = w_1^{k-1}), \quad \text{(E.47)}$$

where

$$H(W_k|W_1^{k-1} = w_1^{k-1}) = -\sum_{w_k \in \mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}} p(w_k|w_1^{k-1}, T_K^N) \log_2(p(w_k|w_1^{k-1}, T_K^N)), \quad \text{(E.48)}$$

for $k \leq K$.

Observe that $\mathcal{M}_\mathrm{s}$ and $\mathcal{M}_\mathrm{a}$ are disjoint so that we can split each sum in (E.47) into two sums over $w_i \in \mathcal{M}_\mathrm{a}$ and $w_i \in \mathcal{M}_\mathrm{s}$. For convenience, we define the set $A^k = \{w_1^k \mid w_1^k \in [\mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}]^k, \exists\ 0 \leq i \leq k : w_i \in \mathcal{M}_\mathrm{a}\}$ as the set of k-subsets of $\mathcal{M}_\mathrm{a} \cup \mathcal{M}_\mathrm{s}$ that contain at least one alarm message and rewrite (E.47) as

$$H(W_k|W_1^{k-1}) = \sum_{w_1^{k-1} \in A^{k-1}} p_\mathrm{A}(w_1^{k-1}|T_K^N) H_\mathrm{A}(W_k|W_1^{k-1} = w_1^{k-1}) + \\ \sum_{w_1^{k-1} \in [\mathcal{M}_\mathrm{s}]^{k-1}} p_\mathrm{S}(w_1^{k-1}|T_K^N) H_\mathrm{S}(W_k|W_1^{k-1} = w_1^{k-1}). \quad \text{(E.49)}$$

We first derive an expression for $H_{\mathrm{A}}(W_k|W_1^{k-1}=w_1^{k-1})$ using the fact that at least one of $w_1,\ldots,w_{k-1}$ belongs to $\mathcal{M}_{\mathrm{a}}$. We additionally split the sum in (E.48) into two sums; one over $w_k\in\mathcal{M}_{\mathrm{a}}$ and one over $w_k\in\mathcal{M}_{\mathrm{s}}$:

$$\begin{aligned}H_{\mathrm{A}}(W_k|W_1^{k-1}=w_1^{k-1})&\\ =-\sum_{w_k\in\mathcal{M}_{\mathrm{a}}}&p(w_k|w_1^{k-1}\in A^{k-1},T_K^N)\log_2(p(w_k|w_1^{k-1}\in A^{k-1},T_K^N))\\ -\sum_{w_k\in\mathcal{M}_{\mathrm{s}}}&p(w_k|w_1^{k-1}\in A^{k-1},T_K^N)\log_2(p(w_k|w_1^{k-1}\in A^{k-1},T_K^N)).\end{aligned}\tag{E.50}$$

Using Bayes' theorem we obtain

$$\begin{aligned}&p(W_k\in\mathcal{M}_{\mathrm{a}}|w_1^{k-1}\in A^{k-1},T_K^N)\\ &\quad=\frac{p(T_K^N|W_k\in\mathcal{M}_{\mathrm{a}},w_1^{k-1}\in A^{k-1})p(W_k\in\mathcal{M}_{\mathrm{a}}|w_1^{k-1}\in A^{k-1})}{p(T_K^N|w_1^{k-1}\in A^{k-1})}\\ &\quad=\frac{(p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}})^{K-k}(1-p_{\mathrm{d}})^{N-k}(1-p_{\mathrm{s}})^{N-k}p_{\mathrm{d}}}{(p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}})^{K-(k-1)}(1-p_{\mathrm{s}})^{N-k}(1-p_{\mathrm{s}})^{N-k}}\\ &\quad=\frac{p_{\mathrm{d}}}{p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}}}.\end{aligned}\tag{E.51}$$

Since all devices that detect the alarm transmit the same message, the first term in (E.50) is

$$\begin{aligned}-\sum_{w_k\in\mathcal{M}_{\mathrm{a}}}&p(w_k|w_1^{k-1}\in A^{k-1},T_K^N)\log_2(p(w_k|w_1^{k-1}\in A^{k-1},T_K^N))\\ &=-\frac{p_{\mathrm{d}}}{p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}}}\log_2\left(\frac{p_{\mathrm{d}}}{p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}}}\right)\\ &\triangleq B_0.\end{aligned}\tag{E.52}$$

Similarly, for the summation over $w_k\in\mathcal{M}_{\mathrm{s}}$ in (E.50) we obtain

$$\begin{aligned}&p(W_k\in\mathcal{M}_{\mathrm{s}}|w_1^{k-1}\in A^{k-1},T_K^N)\\ &\qquad=\frac{(1-p_{\mathrm{d}})p_{\mathrm{s}}}{p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}}}=1-p(W_k\in\mathcal{M}_{\mathrm{a}}|w_1^{k-1}\in A^{k-1},T_K^N).\end{aligned}\tag{E.53}$$

Since the standard messages are not mutually exclusive, and equally likely, it

follows that the second term in (E.50) becomes

$$\begin{aligned}
&- \sum_{w_k \in \mathcal{M}_\mathrm{s}} p(w_k|w_1^{k-1}, T_K^N) \log_2(p(w_k|w_1^{k-1}, T_K^N)) \\
&\quad = - \sum_{w_k \in \mathcal{M}_\mathrm{s}} \frac{1}{M_\mathrm{s}} \frac{(1-p_\mathrm{d})p_\mathrm{s}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \log_2 \left(\frac{1}{M_\mathrm{s}} \frac{(1-p_\mathrm{d})p_\mathrm{s}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \right) \\
&\quad = \frac{(1-p_\mathrm{d})p_\mathrm{s}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \left(\log_2 M_\mathrm{s} - \log_2 \left(\frac{(1-p_\mathrm{d})p_\mathrm{s}}{p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s}} \right) \right) \\
&\quad \triangleq B_1, \qquad \text{(E.54)}
\end{aligned}$$

Substituting (E.52) and (E.54) into (E.50) yields $H_\mathrm{A}(W_k|W_1^{k-1} = w_1^{k-1}) = B_0 + B_1$.

We now derive an expression for $p_\mathrm{A}(w_1^{k-1}|T_K^N)$ in (E.49). Let $1 \leq i \leq k-1$ denote the (random) number of alarm messages in w_1^{k-1} and, without loss of generality, assume that the alarm messages occupy the first i positions of W_1^{k-1}, i.e. $w_1, \ldots, w_i \in [\mathcal{M}_\mathrm{a}]^i$ and $w_{i+1}, \ldots, w_{k-1} \in [\mathcal{M}_\mathrm{s}]^{k-i+1}$. For a fixed i, the probability $p_\mathrm{A}(w_1^{k-1}|T_K^N)$ is obtained using Bayes' theorem as

$$\begin{aligned}
&p_\mathrm{A}(W_1^i = w_1^i \in [\mathcal{M}_\mathrm{a}]^i, W_{i+1}^{k-1} = w_{i+1}^{k-1} \in [\mathcal{M}_\mathrm{s}]^{k-(i+1)}|T_K^N) \\
&\quad = \frac{1}{M_\mathrm{a} M_\mathrm{s}^{k-(i+1)}} \\
&\qquad \times \frac{(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^{K-(k-1)}(1-p_\mathrm{d})^{N-K} p_\mathrm{a} p_\mathrm{d}^i (1-p_\mathrm{d})^{k-(i+1)} p_\mathrm{s}^{k-(i+1)}}{p_\mathrm{a}(p_\mathrm{d} + (1-p_\mathrm{d})p_\mathrm{s})^K (1-p_\mathrm{d})^{N-K} + (1-p_\mathrm{a}) p_\mathrm{s}^K} \\
&\quad = \frac{p_\mathrm{a} p_\mathrm{d}^i ((1-p_\mathrm{d})p_\mathrm{s})^{k-1-i}}{M_\mathrm{a} M_\mathrm{s}^{k-1-i}} N_0 \qquad \text{(E.55)}
\end{aligned}$$

where N_0 is given as in (E.3). Notice that as before only one alarm message is used at a given time so M_a is not raised to the power of i. Since there are exactly $\binom{k-1}{i} M_\mathrm{a} M_\mathrm{s}^{k-1-i}$ equiprobable and disjoint message sets w_1^{k-1} consisting of i alarm messages and $k-1-i$ standard messages, the first term of (E.49) can be expressed as

$$\begin{aligned}
&\sum_{w_1^{k-1} \in A^{k-1}} p_\mathrm{A}(w_1^{k-1}|T_K^N) H_\mathrm{A}(W_k|W_1^{k-1} = w_1^{k-1}) \\
&\quad = \sum_{i=1}^{k-1} \binom{k-1}{i} \sum_{\substack{w_1^i \in [\mathcal{M}_\mathrm{a}]^i \\ w_{i+1}^{k-1} \in [\mathcal{M}_\mathrm{s}]^{k-1-i}}} \frac{p_\mathrm{a} p_\mathrm{d}^i ((1-p_\mathrm{d})p_\mathrm{s})^{k-1-i}}{M_\mathrm{a} M_\mathrm{s}^{k-1-i}} N_0 (B_0 + B_1) \\
&\quad = (B_0 + B_1) \sum_{i=1}^{k-1} \binom{k-1}{i} p_\mathrm{a} p_\mathrm{d}^i ((1-p_\mathrm{d})p_\mathrm{s})^{k-1-i} N_0. \qquad \text{(E.56)}
\end{aligned}$$

We now consider the second term in (E.49). Here the conditional messages in H_{S} and messages in p_{S} are all standard messages. In contrast to the previous case, this can happen both when there is no alarm, and when there is alarm but none of the devices detect it. As before, we rewrite (E.48) as

$$\begin{aligned}
&H_{\mathrm{S}}(W_k|W_1^{k-1}=w_1^{k-1}) \\
&\quad= -\sum_{w_k\in\mathcal{M}_{\mathrm{a}}} p(w_k|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N)\log_2(p(w_k|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N)) \\
&\qquad-\sum_{w_k\in\mathcal{M}_{\mathrm{s}}} p(w_k|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N)\log_2(p(w_k|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N)).
\end{aligned} \tag{E.57}$$

Since each alarm message is equally likely, applying Bayes' theorem and the law of total probability repeatedly yields

$$\begin{aligned}
&p(w_k\in\mathcal{M}_{\mathrm{a}}|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N) \\
&\quad=\frac{1}{M_{\mathrm{a}}}\frac{p_{\mathrm{a}}p_{\mathrm{d}}(p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}})^{K-k}(1-p_{\mathrm{d}})^{N-K+k-1}p_{\mathrm{s}}^{k-1}}{p_{\mathrm{a}}(p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}})^{K-k+1}(1-p_{\mathrm{d}})^{N-K+k-1}p_{\mathrm{s}}^{k-1}+(1-p_{\mathrm{a}})p_{\mathrm{s}}^{K}} \\
&\quad\triangleq\frac{1}{M_{\mathrm{a}}}B_3.
\end{aligned} \tag{E.58}$$

Similarly, for $p(w_k\in\mathcal{M}_{\mathrm{s}}|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N)$ we obtain

$$p(w_k\in\mathcal{M}_{\mathrm{s}}|w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1},T_K^N)=\frac{1}{M_{\mathrm{s}}}(1-B_3). \tag{E.59}$$

Therefore we get

$$\begin{aligned}
H_{\mathrm{S}}(W_k|W_1^{k-1}=w_1^{k-1}) &= -\sum_{w_k\in\mathcal{M}_{\mathrm{a}}}\frac{B_3}{M_{\mathrm{a}}}\log_2\frac{B_3}{M_{\mathrm{a}}}-\sum_{w_k\in\mathcal{M}_{\mathrm{s}}}\frac{1-B_3}{M_{\mathrm{s}}}\log_2\frac{1-B_3}{M_{\mathrm{s}}} \\
&= -B_3\log_2\frac{B_3}{M_{\mathrm{a}}}-(1-B_3)\log_2\frac{1-B_3}{M_{\mathrm{s}}}.
\end{aligned} \tag{E.60}$$

Finally, $p_{\mathrm{S}}(w_1^{k-1}|T_K^N)$ is given by

$$\begin{aligned}
&p(w_1^{k-1}\in[\mathcal{M}_{\mathrm{s}}]^{k-1}|T_K^N) \\
&\quad=\frac{1}{M_{\mathrm{s}}^{k-1}}\frac{p_{\mathrm{a}}(1-p_{\mathrm{d}})^{N-K+(k-1)}p_{\mathrm{s}}^{k-1}(p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}})^{K-(k-1)}+(1-p_{\mathrm{a}})p_{\mathrm{s}}^{K}}{p_{\mathrm{a}}(p_{\mathrm{d}}+(1-p_{\mathrm{d}})p_{\mathrm{s}})^{K}(1-p_{\mathrm{d}})^{N-K}+(1-p_{\mathrm{a}})p_{\mathrm{s}}^{K}} \\
&\quad\triangleq\frac{1}{M_{\mathrm{s}}^{k-1}}B_2.
\end{aligned} \tag{E.61}$$

Using (E.60) and (E.61), the last term in (E.49) can be expressed as

$$\begin{aligned}
&\sum_{w_1^{k-1}\in[\mathcal{M}_\mathrm{s}]^{k-1}} p_\mathrm{S}(w_1^{k-1}|T_K^N) H_\mathrm{S}(W_k|W_1^{k-1}=w_1^{k-1}) \\
&\quad= \sum_{w_1^{k-1}\in[\mathcal{M}_\mathrm{s}]^{k-1}} \frac{B_2}{M_\mathrm{s}^{k-1}} \left(-B_3 \log_2 \frac{B_3}{M_\mathrm{a}} - (1-B_3)\log_2 \frac{1-B_3}{M_\mathrm{s}}\right) \\
&\quad= -B_2 \left(B_3 \log_2 \frac{B_3}{M_\mathrm{a}} + (1-B_3)\log_2 \frac{1-B_3}{M_\mathrm{s}}\right) \qquad \text{(E.62)}
\end{aligned}$$

Inserting (E.56) and (E.62) into (E.47) yields the final expression:

$$\begin{aligned}
H(W_k|W_1^{k-1}) = (B_0+B_1)\sum_{i=1}^{k-1} \binom{k-1}{i} p_\mathrm{a} p_\mathrm{d}^i ((1-p_\mathrm{d})p_\mathrm{s})^{k-1-i} N_0 \\
- B_2 \left(B_3 \log_2 \frac{B_3}{M_\mathrm{a}} + (1-B_3)\log_2 \frac{1-B_3}{M_\mathrm{s}}\right) \qquad \text{(E.63)}
\end{aligned}$$

□

B Proof of Theorem 2

Generate the $M_\mathrm{a}+M_\mathrm{s}=M$ codewords $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_M \overset{i.i.d.}{\sim} \mathcal{N}(\boldsymbol{0}, P'\boldsymbol{I}_n)$. We assume that the first M_a codewords are alarm messages and the last M_s codewords are standards messages. Recall that W_j is the codeword selected by the j-th device. If $\|\boldsymbol{c}_{W_j}\|_2^2 > nP$ then device j transmits $\boldsymbol{0}$ instead, i.e. $\boldsymbol{X}_j=\boldsymbol{0}$, otherwise $\boldsymbol{X}_j=\boldsymbol{c}_{W_j}$, $1\le j\le K$. Decoding is done in two steps. The transmitted alarm message (if any) is decoded first and canceled from the received signal, and then the decoder proceeds to decoding the standard messages. Notice that the order of decoding reflects the fact that alarm messages are expected to have a higher reliability requirement than standard messages.

In the first step, the decoder estimates the transmitted alarm message $\widehat{W}$ and the number of devices transmitting the alarm message $\widehat{K}_\mathrm{a}$:

$$\widehat{W}, \widehat{K}_\mathrm{a} = \underset{\substack{W\in\mathcal{M}_\mathrm{a}\\ 0\le K_\mathrm{a}\le K}}{\arg\min} \|K_\mathrm{a}\boldsymbol{c}_W - \boldsymbol{Y}\|_2^2. \qquad \text{(E.64)}$$

If $\widehat{K}_\mathrm{a}\ge 1$ the decoder outputs $\widehat{W}$ as the alarm message and zero otherwise. The estimated interference from the alarm messages is subtracted from the received signal in a successive interference cancellation fashion as $\boldsymbol{Y}_\mathrm{SIC} = \boldsymbol{Y} - \widehat{K}_\mathrm{a}\boldsymbol{c}_{\widehat{W}}$. Next the decoder outputs the set of standard messages $\hat{\mathcal{S}} \in [\mathcal{M}_\mathrm{s}]^{K-\widehat{K}_\mathrm{a}}$

$$\hat{\mathcal{S}} = \underset{\mathcal{S}\in[\mathcal{M}_\mathrm{s}]^{K-\widehat{K}_\mathrm{a}}}{\arg\min} \left\| \sum_{W\in\mathcal{S}} \boldsymbol{c}_W - \boldsymbol{Y}_\mathrm{SIC} \right\|_2^2. \qquad \text{(E.65)}$$

We consider Gallager type bounds [14]. Initially we ignore the power constraint and assume that the transmitted codewords are $\boldsymbol{X}_j = \boldsymbol{c}_{W_j}$ instead of $\boldsymbol{X}_j = \boldsymbol{c}_{W_j} \mathbb{1}\{\left\|\boldsymbol{c}_{W_j}\right\|_2^2 < nP\}$. Furthermore, we assume that W_j are drawn without replacement from $\mathcal{M}_\mathrm{s}$. The contributions from these assumptions will be taken into account later. We consider each of the four bounds in the theorem separately.

B.1 Alarm decoding error

We start with the bound in (E.13), i.e. the probability of not decoding an alarm message in the alarm event, denoted by $\mathbb{P}\left[E_\mathrm{a}|A\right]$. From symmetry we assume that devices $1, \ldots, K_\mathrm{a}$ are transmitting alarm message $1 = W_1 = W_2 = \cdots = W_{K_\mathrm{a}}$. We want to bound $\mathbb{P}\left[\widehat{W} \neq 1\right]$. Define the interference as $\boldsymbol{S} = \sum_{m=K_\mathrm{a}+1}^{K} \boldsymbol{X}_m$, where we note the dependency on K_a. We then have $\boldsymbol{Y} = K_\mathrm{a}\boldsymbol{X}_1 + \boldsymbol{S} + \boldsymbol{Z}$. Let W' be a random index in $\mathcal{M}_\mathrm{a} \setminus 1$ and let $0 \leq K'_\mathrm{a} \leq K$ be some integer. Then by definition of the decoder (E.64) an error occurs if

$$\|K'_\mathrm{a}\boldsymbol{c}_{W'} - (K_\mathrm{a}\boldsymbol{X}_1 + \boldsymbol{S} + \boldsymbol{Z})\|_2^2 < \|K_\mathrm{a}\boldsymbol{X}_1 - (K_\mathrm{a}\boldsymbol{X}_1 + \boldsymbol{S} + \boldsymbol{Z})\|_2^2, \tag{E.66}$$

i.e. if the distance, in L_2-norm, from a multiple of a wrong codeword $\boldsymbol{c}_{W'}$ to the received signal $\boldsymbol{Y}$ is smaller than from the K_a true alarm transmissions $\boldsymbol{X}_1$. We therefore define the error event

$$F_\mathrm{a}(W', K'_\mathrm{a}) = \{\|K_\mathrm{a}\boldsymbol{X}_1 - K'_\mathrm{a}\boldsymbol{c}_{W'} + \boldsymbol{S} + \boldsymbol{Z}\|_2^2 < \|\boldsymbol{S} + \boldsymbol{Z}\|_2^2\} \tag{E.67}$$

We want to bound the probability of this event for all possible combinations of W' and K'_a, thus we define the collection of events

$$F_\mathrm{a}(K'_\mathrm{a}) = \bigcup_{W' \in \mathcal{M}_\mathrm{a} \setminus 1} F_\mathrm{a}(W', K'_\mathrm{a}) \tag{E.68}$$

and

$$F_\mathrm{a} = \bigcup_{0 \leq K'_\mathrm{a} \leq K} F_\mathrm{a}(K'_\mathrm{a}). \tag{E.69}$$

Clearly, $\mathbb{P}\left[F_\mathrm{a}\right] = \mathbb{P}\left[\widehat{W} \neq 1\right] = \mathbb{P}\left[E_\mathrm{a}|A\right]$ since the decoder is designed to only output one alarm message, thereby eliminating the possibility of collision of alarm messages at the decoder.

We first use the fact that $\boldsymbol{S}$ is a sum of Gaussian random vectors and hence

is also Gaussian, and obtain the bound

$$\mathbb{P}\left[F_{\mathrm{a}}(W', K'_{\mathrm{a}})|\boldsymbol{X}_1, K_{\mathrm{a}}, \boldsymbol{S}, \boldsymbol{Z}\right] \leq e^{\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}\,\mathbb{E}[\boldsymbol{c}_{W'}]e^{-\lambda_{\mathrm{a}}\|K_{\mathrm{a}}\boldsymbol{X}_1 - K'_{\mathrm{a}}\boldsymbol{c}_{W'}+\boldsymbol{S}+\boldsymbol{Z}\|_2^2} \tag{E.70}$$

$$= e^{\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}\frac{e^{-\frac{\lambda_{\mathrm{a}}\|K_{\mathrm{a}}\boldsymbol{X}_1+\boldsymbol{S}+\boldsymbol{Z}\|_2^2}{1+2K'^2_{\mathrm{a}}P'\lambda_{\mathrm{a}}}}}{(1+2K'^2_{\mathrm{a}}P'\lambda_{\mathrm{a}})^{n/2}} \tag{E.71}$$

$$= e^{\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}e^{-\frac{\lambda_{\mathrm{a}}\|K_{\mathrm{a}}\boldsymbol{X}_1+\boldsymbol{S}+\boldsymbol{Z}\|_2^2}{1+2K'^2_{\mathrm{a}}P'\lambda_{\mathrm{a}}}}e^{-\frac{n}{2}\ln(1+2K'^2_{a}P'\lambda_{\mathrm{a}})} \tag{E.72}$$

$$= e^{\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2-\beta_{\mathrm{a}}\|K_a\boldsymbol{X}_1+\boldsymbol{S}+\boldsymbol{Z}\|_2^2-n\phi(K'^2_a,\lambda_{\mathrm{a}})}, \tag{E.73}$$

where $\lambda_{\mathrm{a}} \in \mathbb{R}_+$, $\phi(k,\alpha) = \frac{1}{2}\ln(1+2kP'\alpha)$ and $\beta_{\mathrm{a}} = \Phi(K'^2_a, \lambda_{\mathrm{a}})$ where $\Phi(k,\alpha) \triangleq \frac{\alpha}{1+2kP'\alpha}$. The bound in (E.70) follows from the Chernoff bound [15] and (E.71) uses the identity [1]

$$\mathbb{E}\, e^{-\gamma\|\sqrt{\alpha}\boldsymbol{Q}+\boldsymbol{u}\|_2^2} = \frac{e^{-\frac{\gamma\|\boldsymbol{u}\|_2^2}{1+2\alpha\gamma}}}{(1+2\alpha\gamma)^{n/2}}, \tag{E.74}$$

where $\boldsymbol{u} \in \mathbb{R}^n$, $\alpha \in \mathbb{R}_+$, $\gamma \geq -\frac{1}{2\alpha}$ and $\boldsymbol{Q} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$.

Next we use Gallager's ρ-trick [14] to bound $\mathbb{P}\left[F_{\mathrm{a}}(K'_{\mathrm{a}})\right]$. For events $A_1, A_2, \ldots$ and $\rho \in [0,1]$ we have $\mathbb{P}\left[\cup_j A_j\right] \leq \left(\sum_j \mathbb{P}\left[A_j\right]\right)^{\rho}$. We get

$$\mathbb{P}\left[F_{\mathrm{a}}(K'_{\mathrm{a}})|\boldsymbol{X}_1, K_{\mathrm{a}}, \boldsymbol{S}, \boldsymbol{Z}\right] \leq (M_{\mathrm{a}}-1)^{\rho}\, e^{\rho\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2-\rho\beta_{\mathrm{a}}\|K_{\mathrm{a}}\boldsymbol{X}_1+\boldsymbol{S}+\boldsymbol{Z}\|_2^2-\rho n\phi(K'^2_{\mathrm{a}},\lambda_{\mathrm{a}})}. \tag{E.75}$$

Taking expectation over $\boldsymbol{X}_1$ and using (E.74) yields

$$\mathbb{P}\left[F_{\mathrm{a}}(K'_{\mathrm{a}})|K_{\mathrm{a}}, \boldsymbol{S}, \boldsymbol{Z}\right] \leq (M_{\mathrm{a}}-1)^{\rho}e^{\rho\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}\,\mathbb{E}[\boldsymbol{X}_1]e^{-\rho\beta_{\mathrm{a}}\|K_{\mathrm{a}}\boldsymbol{X}_1+\boldsymbol{S}+\boldsymbol{Z}\|_2^2}e^{-\rho n\phi(K'^2_{\mathrm{a}},\lambda_{\mathrm{a}})} \tag{E.76}$$

$$= (M_{\mathrm{a}}-1)^{\rho}e^{\rho\lambda_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}\frac{e^{-\frac{\rho\beta_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}{1+2K_a^2P'\rho\beta_{\mathrm{a}}}}}{(1+2K_a^2P'\rho\beta_{\mathrm{a}})^{n/2}}e^{-\rho n\phi(K'^2_a,\lambda_{\mathrm{a}})} \tag{E.77}$$

$$= (M_{\mathrm{a}}-1)^{\rho}e^{-\gamma_{\mathrm{a}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2-n\tau}, \tag{E.78}$$

where $\tau = \rho\phi(K'^2_{\mathrm{a}}, \lambda_{\mathrm{a}}) + \phi(K^2_{\mathrm{a}}, \rho\beta_{\mathrm{a}})$ and $\gamma_{\mathrm{a}} = \Phi(K^2_{\mathrm{a}}, \rho\beta_{\mathrm{a}}) - \rho\lambda_{\mathrm{a}}$. Now in the same manner as in (E.76)-(E.78) expectation is taken over $\boldsymbol{S}$ and $\boldsymbol{Z}$ where (E.74) is used for both. We get

$$\mathbb{P}\left[F_{\mathrm{a}}(K'_{\mathrm{a}})|K_{\mathrm{a}}\right] \leq e^{\rho\ln(M_{\mathrm{a}}-1)-n\xi_{\mathrm{a}}}, \tag{E.79}$$

where $\xi_{\mathrm{a}} = \tau + \phi(K-K_{\mathrm{a}}, \gamma_{\mathrm{a}}) + \phi(1/P', \psi_{\mathrm{a}})$ and $\psi_{\mathrm{a}} = \Phi(K-K_{\mathrm{a}}, \gamma_{\mathrm{a}})$. Introducing $E_{\mathrm{a}} = \max_{0\leq\rho\leq1, 0<\lambda_{\mathrm{a}}} -\frac{\rho}{n}\ln(M_{\mathrm{a}}-1) + \xi_{\mathrm{a}}$ and applying the union

bound gives

$$\mathbb{P}\left[F_{\mathrm{a}}|K_{\mathrm{a}}\right] = \min\left(\sum_{K'_{\mathrm{a}}=0}^{K} e^{-nE_{\mathrm{a}}},\ 1\right) \tag{E.80}$$

$$\triangleq a(K, K_{\mathrm{a}}). \tag{E.81}$$

Finally, we take the expectation over K_{a}. The distribution of K_{a} is a binomial distribution given by

$$p_{K_{\mathrm{a}}}(k) = \binom{K}{k} \frac{p_{\mathrm{d}}^k \left((1-p_{\mathrm{d}})\, p_{\mathrm{s}}\right)^{K-k}}{\left(p_{\mathrm{d}} + (1-p_{\mathrm{d}})p_{\mathrm{s}}\right)^K}, \tag{E.82}$$

where the normalization coefficient arises because of the certainty that K devices were active. It follows that

$$\mathbb{P}\left[F_{\mathrm{a}}\right] \leq \sum_{K_{\mathrm{a}}=0}^{K} p_{K_{\mathrm{a}}}(K_{\mathrm{a}}) a(K, K_{\mathrm{a}}). \tag{E.83}$$

We now consider the impact of the power constraint. Since the standard messages are treated as interference in this bound, we ignore the power constraint for the standard messages as the bound is still valid. For the alarm messages only one is active at a given time, so we add the following term to the error probability:

$$\mathbb{P}\left[\|\boldsymbol{c}_j\|_2^2 > nP\right] = \mathbb{P}\left[\frac{1}{n}\sum_{i=1}^{n} Z_i^2 > \frac{P}{P'}\right] \tag{E.84}$$

$$\triangleq p_0, \tag{E.85}$$

where $\boldsymbol{Z} = [Z_1, \ldots, Z_n]^T \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_n)$. This gives the bound in (E.13).

B.2 False positive alarms

We now turn to the bound in (E.16), i.e. the bound for the probability of false positive alarms, $\mathbb{P}\left[E_{\mathrm{fp}}|\neg A\right]$. In this case the true $K_{\mathrm{a}} = 0$, and a false positive occurs if the decoder outputs $\widehat{K}_{\mathrm{a}} > 0$. Let $W' \in \mathcal{M}_{\mathrm{a}}$ and $0 < K'_{\mathrm{a}} \leq K$. We define the error event

$$F_{\mathrm{fp}}(W', K'_{\mathrm{a}}) = \{\left\|\boldsymbol{S} - K'_{\mathrm{a}}\boldsymbol{c}_{W'} + \boldsymbol{Z}\right\|_2^2 < \|\boldsymbol{S} + \boldsymbol{Z}\|_2^2\}. \tag{E.86}$$

The only difference between the error event $F_{\mathrm{fp}}(W', K'_{\mathrm{a}})$ and $F_{\mathrm{a}}(W', K'_{\mathrm{a}})$ is the absence of the true alarm messages. Therefore we define

$$F_{\mathrm{fp}}(K'_{\mathrm{a}}) = \bigcup_{W' \in \mathcal{M}_{\mathrm{a}}} F_{\mathrm{fp}}(W', K'_{\mathrm{a}}) \tag{E.87}$$

and

$$F_{\text{fp}} = \bigcup_{0<K'_{\text{a}}\leq K} F_{\text{fp}}(K'_{\text{a}}). \tag{E.88}$$

We have that $\mathbb{P}\left[F_{\text{fp}}\right] = \mathbb{P}\left[E_{\text{fp}}|\neg A\right]$. As in Section B.1 we use the Chernoff bound and the identity (E.74), and take the expectation over $\boldsymbol{c}_{W'}$ to get the bound

$$\begin{aligned}\mathbb{P}\left[F_{\text{fp}}(W', K'_{\text{a}})|\boldsymbol{S}, \boldsymbol{Z}\right] &\leq e^{\lambda_{\text{fp}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2} e^{\frac{-\lambda_{\text{fp}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}{1+2K'^2_{\text{a}}P'\lambda_{\text{fp}}} - n\phi(K'^2_{\text{a}},\lambda_{\text{fp}})} \\ &= e^{-\beta_{\text{fp}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2 - n\phi(K'^2_{\text{a}},\lambda_{\text{fp}})},\end{aligned} \tag{E.89}$$

where $\lambda_{\text{fp}} \in \mathbb{R}_+$ and $\beta_{\text{fp}} = \Phi(K'^2_{\text{a}}, \lambda_{\text{fp}}) - \lambda_{\text{fp}}$. Using Gallager's ρ-trick we obtain

$$\mathbb{P}\left[F_{\text{fp}}(K'_{\text{a}})|\boldsymbol{S}, \boldsymbol{Z}\right] \leq e^{\rho\ln(M_{\text{a}}) - \rho\beta_{\text{fp}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2 - \rho n\phi(K'^2_{\text{a}},\lambda_{\text{fp}})}. \tag{E.90}$$

Taking the expectation over $\boldsymbol{S}$ and $\boldsymbol{Z}$, and applying the union bound over K'_{a}, we get the bound

$$\mathbb{P}\left[F_{\text{fp}}\right] \leq \min\left(\sum_{K'_{\text{a}}=1}^{K} e^{-nE_{\text{fp}}},\ 1\right) \tag{E.91}$$

$$\triangleq b(K), \tag{E.92}$$

where $E_{\text{fp}} = \max_{0\leq\rho\leq t, 0<\lambda_{\text{fp}}} -\frac{\rho}{n}\ln M_{\text{a}} + \xi_{\text{fp}}$ for $\xi_{\text{fp}} = \rho\phi(K'^2_{\text{a}}, \lambda_{\text{fp}}) + \phi(K, \rho\beta_{\text{fp}}) + \phi(1/P', \gamma_{\text{fp}})$ and $\gamma_{\text{fp}} = \Phi(K, \rho\beta_{\text{fp}})$.

Similar to the previous case, constraining the transmission power results in less interference and hence lower error probability. Therefore, the bound given by (E.92) is still an upper bound on the error probability in the power constrained case, and no additional term is needed.

B.3 Standard message error with no alarm

We now consider the bound in (E.14), i.e. the bound for the per-device probability of error for standard messages when no alarm has occurred, $\frac{1}{K}\sum_{j=1}^{K}\mathbb{P}\left[E_j|\neg A\right]$. Since the standard message decoder relies on canceling the interference caused by the alarm messages, we assume that correct decoding of standard messages can only occur if the decoder does not output a false positive alarm. If there is no false positive, the scenario coincides with the one derived in [1] given by $c(K)$ in (E.27). Since the probability of a false positive alarm is bounded by $b(K)$, the probability of error is bounded as

$$\frac{1}{K}\sum_{j=1}^{K}\mathbb{P}\left[E_j|\neg A\right] \leq 1 - (1-b(K))(1-c(K)) \tag{E.93}$$

$$= b(K) + c(K) - b(K)c(K)) \tag{E.94}$$

The bound above ignores the impact of power constraint for the interfering standard messages. However, as in the previous section, the bound is still valid in the case of a false positive alarm. When no false positive alarm is decoded, the power constraint and collision error of standard messages are accounted for in $c(K)$ through the last two terms in (E.27) as derived in [1].

B.4 Standard message error with alarm

Finally, we turn to the bound in (E.15), i.e. the average per-device probability of error for the standard messages in the alarm event, $\mathbb{E}[K_\mathrm{a}]\frac{1}{K-K_\mathrm{a}}\sum_{j=1}^{K-K_\mathrm{a}}\mathbb{P}\left[E_j|A\right]$. As in Section B.3, we assume that there is automatically an error if the alarm is incorrectly decoded. Assume that the first K_a devices are transmitting the alarm message $1 = W_1 = \cdots = W_{K_\mathrm{a}}$. The probability of error for standard messages is bounded by the probability that the alarm is incorrectly decoded or the standard messages are incorrectly decoded after correctly canceling the interference from alarm messages. Since the probability for the latter event is bounded by $c(K-K_\mathrm{a})$ we obtain

$$\begin{aligned}
&\frac{1}{K-K_\mathrm{a}}\sum_{j=1}^{K-K_\mathrm{a}}\mathbb{P}\left[E_j|A\right] \\
&\quad\leq 1-\mathbb{P}\left[\widehat{W}=1,\widehat{K}_\mathrm{a}=K_\mathrm{a}\right](1-c(K-K_\mathrm{a})) \\
&\quad= 1-\mathbb{P}\left[\widehat{K}_\mathrm{a}=K_\mathrm{a}|\widehat{W}=1\right]\mathbb{P}\left[\widehat{W}=1\right](1-c(K-K_\mathrm{a})) \\
&\quad\leq 1-\mathbb{P}\left[\widehat{K}_\mathrm{a}=K_\mathrm{a}|\widehat{W}=1\right](1-a(K,K_\mathrm{a}))(1-c(K-K_\mathrm{a})),
\end{aligned} \tag{E.95}$$

where $a(K,K_\mathrm{a})$ is given by (E.81).

To derive a bound on $\mathbb{P}\left[\widehat{K}_\mathrm{a}=K_\mathrm{a}|\widehat{W}=1\right]$ we consider the complementary event $\mathbb{P}\left[\widehat{K}_\mathrm{a}\neq K_\mathrm{a}|\widehat{W}=1\right]$. Let $0\leq K'_\mathrm{a}\leq K$ and $K'_\mathrm{a}\neq K_\mathrm{a}$, and define error event

$$F_\mathrm{sa}(K'_\mathrm{a}) = \{\left\|(K_\mathrm{a}-K'_\mathrm{a})\boldsymbol{X}_1+\boldsymbol{S}+\boldsymbol{Z}\right\|_2^2 < \|\boldsymbol{S}+\boldsymbol{Z}\|_2^2\}. \tag{E.96}$$

This event is similar to the error event $F_\mathrm{a}(W',K'_\mathrm{a})$ with the exception that here the alarm message is known. We define

$$F_\mathrm{sa} = \bigcup_{\substack{0\leq K'_\mathrm{a}\leq K\\ K'_a\neq K_\mathrm{a}}} F_\mathrm{sa}(K'_\mathrm{a}), \tag{E.97}$$

and, as in Section B.1, use the Chernoff bound and the identity (E.74) with

the expectation over $\boldsymbol{X}_1$ to get the bound

$$\mathbb{P}\left[F_{\text{sa}}(K'_{\text{a}})|\boldsymbol{S},\boldsymbol{Z},K_{\text{a}}\right] \leq e^{\lambda_{\text{sa}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2} e^{\frac{-\lambda_{\text{sa}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2}{1+2(K_{\text{a}}-K'_{\text{a}})^2P'\lambda_{\text{sa}}}} - e^{-\frac{n}{2}\ln(1+2(K_{\text{a}}-K'_{\text{a}})^2P'\lambda_{\text{sa}}} \tag{E.98}$$

$$= -e^{-\beta_{\text{sa}}\|\boldsymbol{S}+\boldsymbol{Z}\|_2^2 - n\phi((K_{\text{a}}-K'_{\text{a}})^2,\lambda_{\text{sa}})}, \tag{E.99}$$

where $\beta_{\text{sa}} = \Phi\left((K_{\text{a}}-K'_{\text{a}})^2,\lambda_{\text{sa}}\right) - \lambda_{\text{sa}}$. Taking the expectation over $\boldsymbol{S}$ and $\boldsymbol{Z}$ using (E.74) we have

$$\mathbb{P}\left[F_{\text{sa}}(K'_{\text{a}})|K_{\text{a}}\right] \leq e^{-nE_{\text{sa}}}, \tag{E.100}$$

where $E_{\text{sa}} = \max_{0<\lambda_{\text{sa}}} \phi\left((K_{\text{a}}-K'_{\text{a}})^2,\lambda_{\text{sa}}\right) + \phi(K-K_{\text{a}},\beta_{\text{sa}}) + \phi(1/P',\gamma_{\text{sa}})$ and $\gamma_{\text{sa}} = \Phi(K-K_{\text{a}},\beta_{\text{sa}})$. Finally, the union bound over K'_{a} is used to get

$$\mathbb{P}\left[F_{\text{sa}}|K_{\text{a}}\right] \leq \min\left(\sum_{\substack{K'_{\text{a}}=0\\K'_{\text{a}}\neq K_{\text{a}}}}^{K} e^{-nE_{\text{sa}}},\ 1\right) \tag{E.101}$$

$$\triangleq e(K,K_{\text{a}}). \tag{E.102}$$

It follows that $\mathbb{P}\left[\widehat{K}_{\text{a}} = K_{\text{a}}|\widehat{W}=1\right] \geq 1-e(K,K_{\text{a}})$. Substituting this into (E.95) and taking expectation over K_{a} as in Section B.1 gives

$$\begin{aligned}&\mathbb{E}[K_{\text{a}}]\frac{1}{K-K_{\text{a}}}\sum_{j=1}^{K-K_{\text{a}}}\mathbb{P}\left[E_j|A\right]\\ &\quad\leq \sum_{K_{\text{a}}=0}^{K} p_{K_{\text{a}}}(K_{\text{a}})\left(1-(1-a(K,K_{\text{a}}))(1-e(K,K_{\text{a}}))\left(1-c\left(K-K_{\text{a}}\right)\right)\right).\end{aligned} \tag{E.103}$$

The bound above already includes collisions and the power constraint in the decoding of standard messages through $c\left(K-K_{\text{a}}\right)$. However, we still need to include the power constraint of the alarm messages. As in Section B.1, this is done by adding p_0 to each of the two error event bounds $e(K,K_{\text{a}})$ and $a(K,K_{\text{a}})$. By defining $d(K,K_{\text{a}}) \triangleq (1-(a(K,K_{\text{a}})+p_0))(1-(e(K,K_{\text{a}})+p_0))$ the final bound becomes

$$\mathbb{E}[K_{\text{a}}]\frac{1}{K-K_{\text{a}}}\sum_{j=1}^{K-K_{\text{a}}}\mathbb{P}\left[E_j|A\right] \leq \sum_{K_{\text{a}}=0}^{K} p_{K_{\text{a}}}(K_{\text{a}})\left(1-d\left(K,K_{\text{a}}\right)\left(1-c\left(K-K_{\text{a}}\right)\right)\right). \tag{E.104}$$

□

Acknowledgment

This work has been in part supported the European Research Council (ERC) under the European Union Horizon 2020 research and innovation program (ERC Consolidator Grant Nr. 648382 WILLOW) and Danish Council for Independent Research (Grant Nr. 8022-00284B SEMIOTIC).

References

[1] Y. Polyanskiy, "A perspective on massive random-access," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 2523–2527.

[2] 3GPP, "Service requirements for machine-type communications," TS 22.368 V14.0.1, June 2017.

[3] IEC, "White paper: Internet of things: Wireless sensor networks," Inernational Electrotechnical Commission, Tech. Rep., November 2014.

[4] N. Nikaein, M. Laner, K. Zhou, P. Svoboda, D. Drajic, M. Popovic, and S. Krco, "Simple traffic modeling framework for machine type communication," in *ISWCS 2013; The Tenth International Symposium on Wireless Communication Systems*, Aug 2013.

[5] E. Plotnik and A. Satt, "Decoding rule and error exponent for the random multiple-access channel," in *Proceedings. 1991 IEEE International Symposium on Information Theory*, June 1991, pp. 216–216.

[6] R. Ahlswede, "Multi-way communication channels," in *Second International Symposium on Information Theory: Tsahkadsor, Armenia, USSR, Sept. 2-8, 1971*, 1973.

[7] R. Gallager, "A perspective on multiaccess channels," *IEEE Transactions on Information Theory*, vol. 31, no. 2, pp. 124–142, March 1985.

[8] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5g: physical and mac-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, September 2016.

[9] Y. Huang and P. Moulin, "Finite blocklength coding for multiple access channels," in *2012 IEEE International Symposium on Information Theory Proceedings*, July 2012, pp. 831–835.

[10] X. Chen, T. Chen, and D. Guo, "Capacity of gaussian many-access channels," *IEEE Transactions on Information Theory*, vol. 63, no. 6, pp. 3516–3539, June 2017.

[11] A. Vem, K. R. Narayanan, J. Cheng, and J. Chamberland, "A user-independent serial interference cancellation based coding scheme for the unsourced random access gaussian channel," in *2017 IEEE Information Theory Workshop (ITW)*, Nov 2017, pp. 121–125.

[12] O. Ordentlich and Y. Polyanskiy, "Low complexity schemes for the random access gaussian channel (extended version)." [Online]. Available: http://people.lids.mit.edu/yp/homepage/data/isit17_maclattice_full.pdf

[13] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley-Interscience, 2006.

[14] R. Gallager, "A simple derivation of the coding theorem and some applications," *IEEE Transactions on Information Theory*, vol. 11, no. 1, pp. 3–18, January 1965.

[15] M. Goemans, "Lecture notes: Chernoff bounds, and some applications," MIT Mathematics, February 2015. [Online]. Available: http://math.mit.edu/~goemans/18310S15/chernoff-notes.pdf

Paper F

Common Message Acknowledgments: Massive ARQ Protocols for Wireless Access

Anders E. Kalør, Radosław Kotaba, and Petar Popovski

Submitted to
IEEE Transactions on Communications, 2022.
Preprint: `https://arxiv.org/abs/2201.03907`.

The layout has been revised.

Abstract

Massive random access plays a central role in supporting the Internet of Things (IoT), where a subset of a large population of users simultaneously transmit small packets to a central base station. While there has been much research on the design of protocols for massive access in the uplink, the problem of providing message acknowledgments back to the users has been somewhat neglected. Reliable communication needs to rely on two-way communication for acknowledgement and retransmission. Nevertheless, because of the many possible subsets of active users, providing acknowledgments requires a significant amount of bits. Motivated by this, we define the problem of massive ARQ (Automatic Retransmission reQuest) protocol and introduce efficient methods for joint encoding of multiple acknowledgements in the downlink. The key idea towards reducing the number of bits used for massive acknowledgement is to allow for a small fraction of false positive acknowledgments. We analyze the implications of this approach and the impact of acknowledgment errors in scenarios with massive random access. Finally, we show that these savings can lead to a significant increase in the reliability when retransmissions are allowed since it allows the acknowledgment message to be transmitted more reliably using a much lower rate.

6.1 Introduction

A fundamental challenge in supporting the Internet of Things (IoT) is to enable grant-free, or uncoordinated, transmissions from a very large number of users [1]. Furthermore, as the user activation is often triggered by physical phenomena, such as an event that generates sensory data, the traffic patterns are sporadic. Thus, at any instant, the resulting subset of active user that have something to transmit is random. This has initiated a large amount of research devoted to the design of random access schemes that can decode messages from a small random subset of users, often based on techniques derived from ALOHA [2, 3] or compressed sensing [4, 5].

However, despite the great interest in transmission schemes for massive access, the problem of efficiently providing packet reception acknowledgments to a large number of users has been somewhat neglected. Yet, a message acknowledgment is often an useful signal for the application layer, and is necessary in order to implement retransmission schemes, which can greatly increase the transmission reliability. Furthermore, several transmission schemes directly rely on such feedback in order to achieve high performance, e.g., by using rateless codes [2, 6]. Although these schemes require only a single bit of common feedback, it can be beneficial in practice to provide early feedback as soon any individual user is decoded in order to minimize the interference from imperfect SIC. In essence, our work treats the problem of massive ARQ (Automatic Retransmission reQuest) and thus

expands the problem space of the area of massive wireless access.

Compared to grant-based access scenarios, where the BS can send an acknowledgment to a user using a single bit (ACK/NACK), acknowledging a set of users decoded from a grant-free access scenario requires the BS to encode the user identities or some other information that can be used to identify the users that it wants to acknowledge. Encoding the user identifiers requires a significant number of bits when the number of users is large. A naïve attempt to encode acknowledgments to K users out of a total of N users could be to simply concatenate the identifiers of the K users and transmit an acknowledgment packet of $K \log_2(N)$ bits. However, this approach has two significant drawbacks. First, it requires a variable-length packet, which may not be desired in many protocols that rely on time-division multiplexing. Second, as we will show, it is possible to significantly reduce the number of bits required to encode the acknowledgments by applying source coding techniques to jointly encode the acknowledgments for all K users. A similar idea was exploited in [7] to design feedback for collision-free scheduling of K out of N users succeeding a massive random access scenario. However, they assumed that the uplink was error-free, which makes the use of acknowledgments obsolete in the first place.

In order to achieve substantial reductions in the acknowledgment message length, our key proposal is to allow for a small but non-negligible fraction of *false positive* acknowledgments, i.e., that a transmitting user erroneously determines that its message is among the acknowledged messages. Such errors are atypical in existing systems, which are often designed to suppress false positives using error detection mechanisms such as cyclic redundancy checks (CRCs), or by encoding the feedback message such that false positives are very rare at the cost of a larger *false negative* probability [8]. The reason for this is that false positive acknowledgments remain undetected after a transmission round and thus can be hard to resolve and lead to unreliable communication. This is in contrast to false negative errors, which may for instance occur if there are errors in the CRC but the message is intact, and for which the cost is merely an unnecessary retransmission. In this sense, a false positive acknowledgment can be "fatal" as it leads to the situation where the user believes that its message was successfully received by the BS when it in fact was lost.

The practical consequences of false positive acknowledgments depend on the application. Applications that require high reliability are likely to be severely impacted by even a small fraction of false positive acknowledgments. On the other hand, in exemplary IoT applications, such as sensing or monitoring, false positive acknowledgments may result only in missed sensor readings because failed measurement transmissions will not be retransmitted, which is unlikely to have big consequences. However, if such events cannot be tolerated, false positive acknowledgments can be detected

and subsequently resolved using mechanisms at higher layers such as packet numbering at the cost of a detection latency.

The impact of false positives and false negatives in feedback has been studied thoroughly for automatic repeat requests (ARQ) and hybrid automatic repeat requests (HARQ) in the single-user setting, where only a single-bit acknowledgment message is needed. The general conclusion from these studies is that the probability of false positive acknowledgments needs to be significantly smaller than the uplink error probability, since they, contrary to the uplink and a false negatives, cannot be repaired by a retransmission [9, 10]. The same result holds in the finite blocklength regime, where the downlink message should be designed to achieve low false positive probability, but the false negative probability should be held constant and relatively large independently of the total reliability requirement [8]. Although the reliability of the feedback is generally less important when the maximum number of transmissions is small since the uplink reliability plays a more significant role in determining the total reliability, these results hold even with as few as two transmission rounds [10]. Nevertheless, because of the large feedback message required in massive access regime and the fact that the feedback acknowledges multiple users, these results cannot be directly transferred to the scenario that we consider.

The paper has three main contributions. *First,* it is the core idea of allowing false positives. We show that by allowing a small fraction of false positive acknowledgments, the number of bits required for the feedback message can be significantly reduced, while the introduction of false negative acknowledgment does not yield comparable savings. Furthermore, we present various practical methods for efficient encoding of acknowledgments with false positives. *Second,* we study how the distribution of the number of active users impacts the feedback message, and derive closed-form bounds on the false positive probability based on the first and second moments of the distribution. *Third,* we quantify the impact of false positive acknowledgments on the overall reliability by studying transmission schemes with multiple transmission rounds. In this context, we show that the message length reduction that results from introducing false positives allows the feedback to be transmitted with a much lower rate, which in turn results in a significant increase in the overall reliability.

We note that, in both grant-free and grant-based settings and irrespectively of the feedback encoding, feedback can be designed either in an adaptive or non-adaptive manner. Adaptive feedback schemes are intrinsically non-trivial due to the half-duplex structure of most wireless systems, which requires the feedback instants to be either fully pre-planned or controlled by the transmitting user. In this paper, the focus is on the feedback message and assume that the feedback moments are known.

The remainder of the paper is organized as follows. Section 6.2 introduces

the overall system model. Information-theoretic bounds for a fixed number of decoded users are presented in Section 6.3, and Section 6.4 introduces and analyzes a number of practical encoding schemes for this setting. Section 6.5 analyzes the case in which the number of decoded users is random, and the case with multiple transmission rounds is analyzed in Section 6.6. Finally, numerical results are presented in Section 6.7 and the paper is concluded in Section 6.8.

6.2 System Model

We consider a typical massive access scenario comprising a single base station (BS) that serves a massive set of potentially active users $[N] = \{1, 2, \ldots, N\}$ (typically N is in the order of thousands). As is often the case in practical systems, we assume that each of the N users has a unique identifier known to both the users and the BS. If the BS requires an initial handshake procedure for users to join the network, then N corresponds to the number of users associated with the BS, and N will be in the order of thousands (for instance in NB-IoT, the Cell Radio Network Temporary Identifier (C-RNTI) can identify up to $N = 65523$ users [11]). On the other hand, if no such procedure exists, then each user can have a globally unique identifier, such as a MAC address, and N will be in the order of 2^{32} to 2^{64}.

We assume a general frame structure in which the air interface is divided into a number of recurring random access opportunities in which a random subset $\mathcal{A} \subseteq [N]$ of users are active and transmit their messages in the uplink. The uplink transmission is followed by a downlink feedback message, multicasted by the BS, that provides acknowledgments to the users that the BS decoded in the uplink. Users that receive an acknowledgment have completed their transmission, while users that do not receive an acknowledgment are allowed to retransmit up to $L - 1$ times. We assume that each uplink message contains the transmitter's identifier such that the BS is able to determine the identity of the sender upon decoding of the packet[1]. In general, the number of active users is random and typically will be much smaller than N. The set of active users (including its cardinality) is unknown to the BS, which tries to recover it from the received signals. We denote the set of recovered users by $\mathcal{S} = \{s_1, s_2, \ldots, s_K\}$, where $s_k \in [N]$ and assume that, conditioned on K, $\mathcal{S}$ is drawn uniformly from the set of all K-element subsets of $[N]$, denoted $[N]_K = \{\mathcal{K} \subseteq [N] \mid |\mathcal{K}| = K\}$. Due to decoding errors, $\mathcal{S}$ may be different from the actual set of active users $\mathcal{A}$. We denote by $\epsilon_{\text{ul},n}$ the probability that a transmitting user n is not decoded. This probability typically depends on

[1]We make this assumption for clarity of presentation, but the analysis holds even if there is no identity (e.g., as in unsourced random access [12]) by treating the messages as temporary identities. In that case N corresponds to the number of distinct messages.

the random access mechanisms as well as the value of K, the signal-to-noise ratios (SNRs) of the transmitting users, etc.

To make the transmitters aware of potential errors and to ensure reliable transmission, the BS transmits a common B-bit feedback message after the random access opportunity that allows the users to determine whether their own identifier is a member of $\mathcal{S}$. The message is transmitted through a packet erasure channel so that the packet is received by user n with probability $1 - \epsilon_{\mathrm{dl},n}$. The erasure probability depends on the SNR of the individual users, the channel, and the transmission rate of the feedback[2]. Formally, such a feedback scheme is defined by an encoder, the downlink channel, and a set of decoders, one for each user. We define the encoder as

$$f : [N]_K \to \{0,1\}^B, \tag{F.1}$$

and the erasure channel as

$$\Pr(Y_n = X \mid X \in \{0,1\}^B) = 1 - \epsilon_{\mathrm{dl},n}, \tag{F.2}$$

$$\Pr(Y_n = \mathrm{e} \mid X \in \{0,1\}^B) = \epsilon_{\mathrm{dl},n}, \tag{F.3}$$

where X is the packet transmitted by the BS, Y_n is the packet received by user n, and e denotes an erasure. Finally the individual decoders are defined as

$$g_n : \{0,1\}^B \cup \mathrm{e} \to \{0,1\}, \qquad n = 1, \ldots, N \tag{F.4}$$

which output 1 if user n is believed to be a member of $\mathcal{S}$ and 0 otherwise (throughout the paper we will assume that the decoder outputs 0 if it observes an erasure). Both the encoder and the decoders may depend on K (which is random), but this dependency can be circumvented by encoding K separately in the feedback message at an average cost of approximately $H(K)$ bits where $H(\cdot)$ is the entropy function[3]. As we will see, this overhead is minimal when compared to the number of bits required to encode the acknowledgments in most settings of practical interest. We will refer to B as the message length of a scheme.

To characterize the performance of a feedback scheme, we define the false positive (FP) probability, denoted ϵ_{fp}, as the probability that a user whose uplink message was not decoded $n \in \mathcal{A} \setminus \mathcal{S}$ erroneously concludes that it belongs to $\mathcal{S}$

$$\epsilon_{\mathrm{fp}} = \mathbb{E}\left[\Pr\left(g_n\left(f\left(\mathcal{S}\right)\right) = 1 \mid n \in \mathcal{A} \setminus \mathcal{S}\right) \mid K\right], \tag{F.5}$$

[2]In practice, an erasure channel represents the case where the decoder can detect if a packet is decoded incorrectly, e.g., through an error-detecting code. The size of such a code with negligible false positive probability is small compared to the size of the feedback message, and thus we will ignore the overhead it introduces.

[3]A pragmatic alternative when the activation distribution is unknown would be to assume that at most K' users can be decoded simultaneously and then dedicate a fixed number of $\log_2(K')$ bits to describe K.

where the expectation is taken over n and the distribution $p(\mathcal{S}|K)$ (but not the channel, which we will treat independently). Similarly, we define the false negative (FN) probability ϵ_{fn} as the probability that a decoded user $n \in \mathcal{S}$ incorrectly concludes that it does not belong to the set

$$\epsilon_{\text{fn}} = \mathbb{E}\left[\Pr\left(g_n\left(f\left(\mathcal{S}\right)\right) = 0 \mid n \in \mathcal{S}\right) \mid K\right]. \tag{F.6}$$

Note that these definitions ignore the channel, and thus allow us to treat ϵ_{fp} and ϵ_{fn} independently of the event of an erasure. Note also that both ϵ_{tp} and ϵ_{fp} are conditioned on K. We discuss the case when K is random further in Section 6.5.

Using these definitions, we denote by B^* the minimum message length B required for a scheme with K active users out of N that achieves false positive and false negative probabilities at most ϵ_{fp} and ϵ_{fn}, respectively.

6.3 Information Theoretic Bounds

We first consider the source coding part of the problem, namely the functions f and g_n defined previously, while for clarity ignoring the erasure channel between the BS and the users. Specifically, in this section we derive information theoretic bounds on the minimum message length B required for the feedback message. To start with, we treat K as constant, and thus neglect the bits required to encode K in the message, which would be the same for all schemes.

6.3.1 Error-Free Coding

We first consider error-free schemes, i.e., schemes that have $\epsilon_{\text{fp}} = \epsilon_{\text{fn}} = 0$. A naïve construction of the feedback message is to concatenate the K identifiers in $\mathcal{S}$ to produce a message of $B = K\log_2(N)$ bits. However, such a construction is sub-optimal because there are only $\binom{N}{K}$ subsets of K users, and $\log_2\binom{N}{K}$ bits are sufficient to distinguish each subset. This leads to the feedback message length

$$B^*_{\text{error-free}} = \left\lceil \log_2 \binom{N}{K} \right\rceil \tag{F.7}$$

$$\geq \lceil K\log_2(N/K) \rceil, \tag{F.8}$$

where the inequality follows from $\binom{N}{K} \geq (N/K)^K$.

A message length of $B^*_{\text{error-free}}$ can be achieved using e.g., enumerative source coding [13]. However, the decoding is impractical for large sets because each user needs to check each of the $\binom{N-1}{K-1}$ subsets that it can belong to.

6.3.2 Encoding with Bounded Errors

The required feedback message length for the error-free encoding scales with the logarithm of N, which can be significant when N is in the order of 2^{32} or 2^{64}. One way to reduce the impact of N is to allow for non-zero false positive and false negative probabilities. To do so, a feedback message must acknowledge at most $K + \epsilon_{\text{fp}} N$ users, out of which at least $(1 - \epsilon_{\text{fn}})K$ must be in $\mathcal{S}$. For $\epsilon_{\text{fp}} < 0.5$ (which is typically the region of interest), it can be shown using combinatorial arguments that [14] (see Appendix A for details)

$$B^*_{\text{fp,fn}} \geq \log_2 \binom{N}{K} - \log_2 \left(K \binom{\lfloor \epsilon_{\text{fp}} N \rfloor + K}{\lceil (1-\epsilon_{\text{fn}})K \rceil} \binom{N}{\lfloor \epsilon_{\text{fn}} K \rfloor} \right) \tag{F.9}$$

$$\begin{aligned} &\geq K \log_2 \left(\frac{1}{\epsilon_{\text{fp}} + \frac{K}{N}} \right) - K \log_2 \left(\frac{e}{1-\epsilon_{\text{fn}}} \right) \\ &\quad - \epsilon_{\text{fn}} K \log_2 \left(\frac{1-\epsilon_{\text{fn}}}{\epsilon_{\text{fn}} \left(\epsilon_{\text{fp}} + \frac{K}{N} \right)} \right) - \log_2(K), \end{aligned} \tag{F.10}$$

where (F.10) follows from the observation that rounding cannot decrease the message length and the inequality $\left(\frac{N}{K}\right)^K \leq \binom{N}{K} \leq (eN/K)^K$. Note that if K is held constant, the bound is independent of N as $N \to \infty$.

The introduction of false positives has the potential to offer significantly greater gains than false negatives. In particular, when ϵ_{fn} is small as is typically desired, the required message length is only negligibly smaller than the one required if no false negatives were allowed. The reason for this is that the set of potential false negatives, $\mathcal{S}$, is much smaller than the set of potential false positives $[N] \setminus \mathcal{S}$. When $\epsilon_{\text{fn}} = 0$, the bound can be tightened further as [15, 16]

$$B^*_{\text{fp}} \geq K \log_2 \left(1/\epsilon_{\text{fp}} \right) - \frac{\log_2(e)(1-\epsilon_{\text{fp}})K^2}{\epsilon_{\text{fp}} N + (1-\epsilon_{\text{fp}})K}, \tag{F.11}$$

where the last term vanishes as $N \to \infty$.

A (non-constructive) achievability bound for the case with $\epsilon_{\text{fn}} = 0$ and $K \leq N\epsilon_{\text{fn}}$ was provided in [15]. The overall idea is to sequentially generate all $\lfloor N\epsilon_{\text{fp}} \rfloor$-element subsets of $[N]$, and then transmit the index of the first subset that includes all K elements in $\mathcal{S}$. Using a set-cover theorem by Erdős and Spencer [17, Theorem 13.4], they show that the number of $\lfloor N\epsilon_{\text{fp}} \rfloor$-element subsets required to cover all K-element subsets of $[N]$, denoted $M(N, \lfloor N\epsilon_{\text{fp}} \rfloor, K)$, is upper bounded by

$$M(N, \lfloor N\epsilon_{\text{fp}} \rfloor, K) \leq \left(1 + \ln \binom{\lfloor N\epsilon_{\text{fp}} \rfloor}{K} \right) \frac{\binom{N}{K}}{\binom{\lfloor N\epsilon_{\text{fp}} \rfloor}{K}}. \tag{F.12}$$

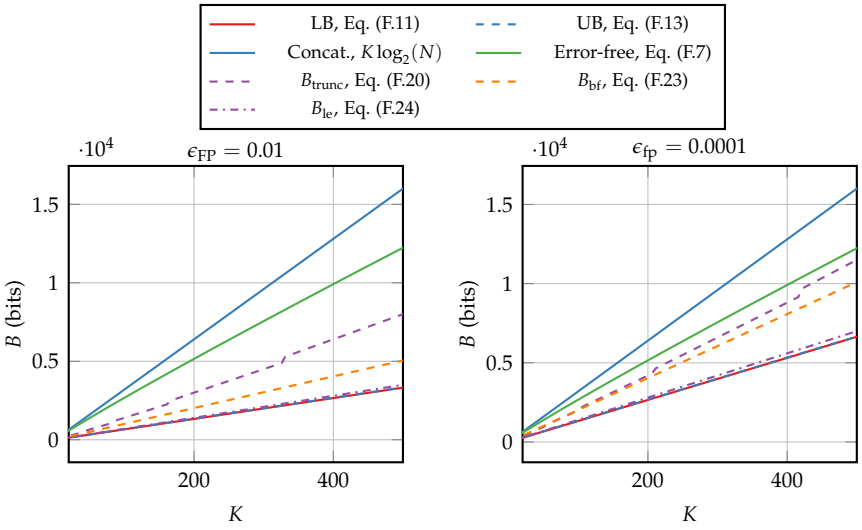


Fig. F.1: Message length, B, required to provide acknowledgment feedback for $N = 2^{32}$ and $\epsilon_{\text{fn}} = 0$ with $\epsilon_{\text{fp}} = 0.01$ and $\epsilon_{\text{fp}} = 0.0001$.

Taking the logarithm and bounding the binomial coefficients gives the following upper bound on the required feedback message length

$$B^*_{\text{fp}} \leq \log_2 \binom{N}{K} - \log_2 \binom{\lfloor N\epsilon_{\text{fp}} \rfloor}{K} + \log_2 \left(1 + \ln \binom{\lfloor N\epsilon_{\text{fp}} \rfloor}{K}\right) \tag{F.13}$$

$$\leq K \log_2 \left(e/\epsilon_{\text{fp}}\right) + \log_2 \left(1 + K \ln \left(\frac{N\epsilon_{\text{fp}}}{K}\right)\right). \tag{F.14}$$

Note that this also serves as an upper bound for the case with $\epsilon_{\text{fn}} > 0$. By comparing (F.14) to the lower bound in (F.11), it can be seen that the bounds are tight within an additive term $O(\log N)$ as $N \to \infty$, i.e., for sufficiently large N,

$$B^*_{\text{fp}} = K \log_2(1/\epsilon_{\text{fp}}) \pm O(\log N), \tag{F.15}$$

which is lower than the error-free scheme in Eq. (F.8) when $\epsilon_{\text{fp}} \geq K/N$. To illustrate the potential gain of introducing a small fraction of false positives, suppose $N = 2^{32}$ and $K = 100$. Encoding the acknowledgment in an error-free manner requires approximately $B = \log_2 \binom{2^{32}}{100} \approx 2675$ bits, while only $B = 100 \log_2(100) \approx 664$ bits are required if we can tolerate $\epsilon_{\text{fp}} = 0.01$, and $B = 100 \log_2(10000) \approx 1329$ bits for $\epsilon_{\text{fp}} = 0.0001$. The required feedback message lengths for these cases are shown in Fig. F.1 and compared to a number of practically realizable schemes presented next. As expected, the upper (UB) and lower (LB) bounds are very tight (within 14 bits in the considered range).

6.4 Practical Schemes

In this section, we present a number of practical designs of f and g_n, and compare them to the bounds derived in the previous section. Motivated by the fact that false negatives provide little reduction in the feedback message length, we will restrict ourselves to schemes with $\epsilon_{\text{fn}} = 0$. Furthermore, we will again assume that the number of decoded users K is fixed and defer the discussion of random activations to Section 6.5.

6.4.1 Identifier Truncation

We start by considering a simple truncation scheme in which the feedback message is constructed by first truncating the identifiers of each of the K decoded users to $b < \log_2(N)$ bits (say, the b least significant bits), and then concatenating them to construct a feedback message of Kb bits. To check whether a user with identifier n is among the K decoded users, one simply checks whether the b least significant bits of n is contained in the message. Clearly, while this cannot cause false negatives, it can lead to false positives if a user that is not decoded in the uplink shares the same b least significant bits with a decoded user. Assuming that the identifiers are uniformly distributed, the false positive probability is

$$\epsilon_{\text{fp}} = 1 - \left(1 - \frac{1}{2^b}\right)^K. \tag{F.16}$$

By rearranging and ceiling to ensure b is integer we obtain $b = \left\lceil -\log_2\left(1 - (1-\epsilon_{\text{fp}})^{1/K}\right)\right\rceil$. The feedback message length is then bounded by

$$\begin{aligned} B_{\text{trunc}} &= K\left\lceil -\log_2\left(1 - (1-\epsilon_{\text{fp}})^{\frac{1}{K}}\right)\right\rceil && \text{(F.17)}\\ &\geq K\left\lceil -\log_2\left(1 - e^{-\frac{\epsilon_{\text{fp}}}{K(1-\epsilon_{\text{fp}})}}\right)\right\rceil && \text{(F.18)}\\ &\geq K\left\lceil -\log_2\left(\frac{\epsilon_{\text{fp}}}{K(1-\epsilon_{\text{fp}})}\right)\right\rceil && \text{(F.19)}\\ &= K\left\lceil \log_2\left(1/\epsilon_{\text{fp}}\right) + \log_2\left(K(1-\epsilon_{\text{fp}})\right)\right\rceil, && \text{(F.20)} \end{aligned}$$

where the first inequality follows from $1 - x \geq e^{-\frac{x}{1-x}}$ for $0 \leq x < 1$ and that $-\log_2(1-x)$ is monotonically increasing for $x < 1$, and the second inequality is due to $1 - e^{-x} \leq x$ for $x \geq 0$ and that $-\log_2(x)$ is monotonically decreasing. The last term in Eq. (F.20) is strictly positive when $K > \frac{1}{1-\epsilon_{\text{fp}}}$, which

is the case for the values of K and ϵ_{fp} that we are interested in. Thus, the scheme requires approximately $K \log_2(K(1-\epsilon_{\text{fp}}))$ bits more than the lower bound in Eq. (F.15).

6.4.2 Universal Hashing

The downside of the identifier truncation scheme presented earlier is that it requires the identifiers to have high entropy, which may be difficult to guarantee in practice. To circumvent this, we can *hash* the identifier instead of using the identifier directly. To illustrate, we consider a scheme based on universal hashing, which can be implemented efficiently in practice. Formally, an (n, v)-family of universal hash functions is a family of functions $h : [n] \to [v]$ such that for a hash function h chosen uniformly at random and for any two distinct values $x, y \in [n]$, $\Pr(h(x) = h(y)) \leq 1/v$. The event that $h(x) = h(y)$ is typically referred to as a *collision*. Using this assumption, we can concatenate the hash of each of the K users to construct a message of $K2^v$ bits. The probability that the hash of an arbitrary user that is not among the K decoded users collides with any of the decoded users is

$$\epsilon_{\text{fp}} = 1 - \left(1 - \frac{1}{v}\right)^K, \tag{F.21}$$

which is exactly the same as in the previous section, but does not require that the user identities are uniformly distributed, and thus is a practically appealing alternative to identifier truncation. However, the required number of bits is still quite far from the lower bound.

6.4.3 Bloom Filter

A Bloom filter [18] uses T independent universal hash functions $h_i : [N] \to [B]$ for $i = 1, \ldots, T$, and is constructed by setting the message bits at positions $\{h_i(s_k) \mid s_k \in \mathcal{S}, i = 1, \ldots, T\}$ equal to '1' and the remaining bits equal to '0'. In order to decode the message and check whether an identifier n belongs to the set, the decoder simply checks if the bits at positions $\{h_i(n) \mid i = 1, \ldots, T\}$ are equal to '1'. Clearly, the decoder can only observe false positives and not false negatives.

It can be shown that the minimum false positive probability is obtained when the probability that a given bit is '1' is exactly $1/2$, and that T should be chosen as $T = (B/K)\ln(2)$ to achieve this [19] (in practice, one needs to round to the nearest integer). The resulting false positive probability is non-trivial, but can be approximated as [19]

$$\epsilon_{\text{fp}} \approx 2^{-\lceil (B/K)\ln(2) + 0.5 \rceil}. \tag{F.22}$$

By assuming equality in the approximation we obtain

$$B_{\text{bf}} = K \log_2(e) \log_2(1/\epsilon_{\text{fp}}), \tag{F.23}$$

revealing that Bloom filter is approximately within a factor $\log_2(e) \approx 1.44$ of the asymptotic lower bound in Eq. (F.15). Nevertheless, it is better than the previous two schemes approximately when $K > \frac{\epsilon_{\text{fp}}^{1-\log_2(e)}}{1-\epsilon_{\text{fp}}}$.

6.4.4 Linear Equations

An alternative family of constructions is based on solving a set of linear equations in a Galois field, first proposed in [16, 20]. To simplify the analysis, we will assume that we have access to a fully random hash function. An (n, b)-family of fully random hash functions is a family of functions $h : [n] \to [b]$ such that for each value $x \in [n]$, it outputs a value chosen uniformly at random from $[b]$. While such hash functions have desirable properties, they are not practical as they require an exponential number of bits to store. Nevertheless, in many practical problems the fully random hash function can be replaced by a simpler hash function with a negligible penalty, especially when the input is randomized [21].

Returning to the encoding scheme, suppose we have a fully random hash function $h_1 : [N] \to \text{GF}(2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil})^K$, i.e., mapping from $[N]$ to K-element vectors in $\text{GF}\left(2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil}\right)$, and a universal hash function $h_2 : [N] \to 2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil}$. Then, we can construct the equation $h_1(s_k) \cdot z = h_2(s_k)$ in $\text{GF}(2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil})$, where $\cdot$ is the inner product. By constructing an equation for each $s_k \in \mathcal{S}$, we obtain the set of K equations with K variables $H_1 z = h_2$, where $H_1 \in \text{GF}(2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil})^{K \times K}$ is the matrix of rows $h_1(s_1), \ldots, h_1(s_K)$ and $h_2 \in \text{GF}(2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil})^K$ is the vector with elements $h_2(s_1), \ldots, h_2(s_K)$. In order for this system to have a solution, we require H_1 to be full rank. It can be shown that this happens with probability at least $1 - \frac{1}{2^{\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil - 1}}$ [20], which is large for the values of ϵ_{fp} that we consider (e.g., greater than 0.99 for $\epsilon_{\text{fp}} = 0.01$ and greater than 0.9999 for $\epsilon_{\text{fp}} = 0.0001$). By repeating the procedure with new hash functions $h_1', h_1'', \ldots$, the probability of generating a matrix with full rank can be made arbitrarily large at the cost of a message length penalty required to store the number of trials. In practice, this penalty is negligible compared to the total size of the message. For instance, with $\epsilon_{\text{fp}} = 0.01$ and up to 16 trials, requiring only four additional bits, the failure probability is in the order of 10^{-34}.

Provided that the resulting matrix H_1 has full rank, we can obtain the solution z to the set of equations. A decoder can then check whether an identifier n is contained in the set by simply checking if $h_1(s_k) \cdot z = h_2(s_k)$.

Thus, neglecting the potential overhead caused by repeating the hashing procedure, only the vector z needs to be communicated, which contains K entries of $\lceil \log_2(1/\epsilon_{\text{fp}}) \rceil$ bits each. Combining these observations, we obtain the feedback message length

$$B_{\text{le}} = K \lceil \log_2(1/\epsilon_{\text{fp}}) \rceil, \tag{F.24}$$

which, disregarding the rounding, matches the asymptotic information theoretic bound in Eq. (F.15). As we will see in Section 6.7, the practical performance matches closely with the bound.

It is worth noting that finding z uses Gaussian elimination, which requires $O(K^3)$ operations. This makes the method infeasible for large K. However, the operation can be performed fast as long as K is at most in the order of hundreds, which is the main interest in this paper. When K is larger, the construction can be improved by introducing sparsity in H_1 at the cost of a small overhead, see e.g., [16, 20, 22].

6.5 Random User Activity

So far, we have assumed that K is fixed and optimized the feedback for a specific value of K. In practice, the number of active devices is random and unknown to both the BS and the devices, and thus the number of messages produced by the random access decoder at the BS, K, is in general also random. Furthermore, K may even be correlated over time and depend on the feedback itself. However, to simplify the analysis, we will here assume that K is independent across frames and drawn from the distribution $p(K)$.

The optimal designs of the feedback schemes depend on K and in many of the schemes the recipient must know K to be able to decode the message. Hence, it is reasonable to include the value of K in the feedback message, which incurs only a very small overhead. To illustrate, suppose the random access mechanism is designed to support at most $K' = 1024$ simultaneously active users, then a 10-bit overhead is required to encode K. On the other hand, if the desired false positive probability is $\epsilon_{\text{fp}} = 0.001$, then approximately $\log_2(0.001) \approx 10$ bits are required per user in the feedback message, so the overhead introduced by encoding K merely corresponds to encoding one additional user. When more than K' users are active, we accept that the error probability can be arbitrarily high. In that case, we may decide to pick a random subset comprising K' of the $K > K'$ decoded users at the cost of $K - K'$ false negatives.

If we allow the feedback message to have a variable length, then we can achieve the desired ϵ_{fp} (and ϵ_{fn}) as long as $K \leq K'$ without significant overprovisioning when $K < K'$ by adjusting the message length to K. However, the random user activity has a more significant impact on the performance

when the length of the feedback message needs to remain fixed for any value of K, e.g., due to protocol constraints, as the error probabilities depends on the instantaneous value of K. It seems reasonable in this case to optimize message length either based on the average false positive/negative probabilities or by the probability that the these probabilities exceed some thresholds $\tilde{\epsilon}_{\text{fp}}$ and $\tilde{\epsilon}_{\text{fn}}$. Assuming for clarity that $\epsilon_{\text{fn}} = 0$, we can formalize the first case by defining the message length selection rule

$$B = \inf \left\{ B' \geq 0 : \mathbb{E}_{K \sim p(K)}[\epsilon_{\text{fp}}(K, B')] \leq \tilde{\epsilon}_{\text{fp}} \right\}, \tag{F.25}$$

where $\epsilon_{\text{fp}}(K, B')$ is the false positive probability achieved with K users and a message length of B' bits, and $\tilde{\epsilon}_{\text{fp}}$ is the specified false positive probability target. Similarly, for the second case we have

$$B = \inf \left\{ B' \geq 0 : \Pr(\epsilon_{\text{fp}}(K, B') > \tilde{\epsilon}_{\text{fp}}) \leq \delta \right\}, \tag{F.26}$$

where δ specifies the maximum allowed probability that the false positive probability exceeds $\tilde{\epsilon}_{\text{fp}}$.

Computing these feedback message lengths requires complete knowledge of the distribution of K, which is often not available. Instead, we proceed by deriving bounds based on the first moments of $p(K)$ using the asymptotic expression for the feedback message length given in Eq. (F.15) for $\epsilon_{\text{fn}} = 0$, which is accurate for large N. We first present an upper bound on the expected false positive $\bar{\epsilon}_{\text{fp}} = \mathbb{E}_{K \sim p(K)}[2^{-B/K}]$.

Proposition 1. *Let the number of decoded users K be random with mean $\bar{K} = \mathbb{E}[K]$ and variance $\text{Var}[K]$. Then for a given message length B the expected false positive probability $\bar{\epsilon}_{\text{fp}}$ is upper bounded as*

$$\bar{\epsilon}_{\text{fp}} < 2^{-B/\bar{K}} + \frac{2.66\text{Var}[K]}{B^2}. \tag{F.27}$$

Proof. By rearranging the expression in Eq. (F.15) we obtain $\epsilon_{\text{fp}}(K, B) \approx 2^{-B/K}$. To bound $\bar{\epsilon}_{\text{fp}}$, we consider the first-order Taylor expansion of $2^{-B/K}$ around $\bar{K} = \mathbb{E}[K]$ given as

$$2^{-B/K} = 2^{-B/\bar{K}} + \frac{2^{-B/\bar{K}}}{\bar{K}^2}(K - \bar{K}) + \frac{2^{-B/Z} B \ln(2) \left(B \ln(2) - 2Z\right)}{Z^4} \frac{(K - \bar{K})^2}{2}, \tag{F.28}$$

for some Z between $\bar{K}$ and K. By analyzing its derivatives it can be shown that the term $\frac{2^{-B/Z} B \ln(2)(B \ln(2) - 2Z)}{Z^4}$ is bounded and attains its maximum at

$Z = \frac{\ln(2)(3-\sqrt{3})}{6} B$. From this we obtain the bound

$$\frac{2^{-B/Z}\left(B\ln(2) - 2Z\right)}{Z^4} \leq \frac{e^{-\frac{6}{3-\sqrt{3}}}\left(\frac{6}{3-\sqrt{3}} - 2\right)}{\left(\frac{3-\sqrt{3}}{6}\right)^3 \ln^2(2) B^2} \tag{F.29}$$

$$= \frac{\zeta}{B^2}, \tag{F.30}$$

where $\zeta = e^{-\frac{6}{3-\sqrt{3}}}\left(\frac{6}{3-\sqrt{3}} - 2\right)\left(\frac{6}{3-\sqrt{3}}\right)^3 \ln^{-2}(2)$. By inserting into Eq. (F.28), taking expectation and rearranging we obtain

$$\bar{\epsilon}_{\text{fp}} \leq 2^{-B/\bar{K}} + \frac{\zeta \text{Var}[K]}{2B^2}. \tag{F.31}$$

□

The proof is completed by noting that $\zeta/2 < 2.66$.

The result in Proposition 1 can be used to select the feedback message length according to the rule in Eq. (F.25). We now derive a similar general bound on the probability that ϵ_{fp} exceeds $\tilde{\epsilon}_{\text{fp}}$ that can be used for the alternative feedback message length selection rule in Eq. (F.26).

Proposition 2. *Let the number of decoded users K be random with mean $\bar{K} = \mathbb{E}[K]$ and variance $\text{Var}[K]$. For a given message length B and $\tilde{\epsilon}_{\text{fp}} \geq 2^{-B/\bar{K}}$, the probability that the false positive probability ϵ_{fp} exceeds $\tilde{\epsilon}_{\text{fp}}$ can be bounded as*

$$\Pr\left(\epsilon_{\text{fp}} > \tilde{\epsilon}_{\text{fp}}\right) \leq \frac{\text{Var}[K]}{\left(\frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})} - \bar{K}\right)^2} \tag{F.32}$$

Proof. Note first that

$$\Pr\left(\epsilon_{\text{fp}} > \tilde{\epsilon}_{\text{fp}}\right) = \Pr\left(K > \frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})}\right) \tag{F.33}$$

$$\leq \Pr\left(K \geq \frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})}\right). \tag{F.34}$$

Applying Chebyshev's inequality yields

$$\Pr\left(K \geq \frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})}\right) \leq \Pr\left(|K - \bar{K}| \geq \frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})} - \bar{K}\right) \tag{F.35}$$

$$\leq \frac{\text{Var}[K]}{\left(\frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})} - \bar{K}\right)^2}. \tag{F.36}$$

□

Because the bound in Proposition 2 does not assume much about the distribution of K, it is in general not very tight. If we further assume that the users activate independently (but not necessarily identically distributed), we can tighten the bound as follows.

Proposition 3. *Let the number of decoded users $K = \sum_{i=1}^{N} k_i$ where $k_i \in \{0,1\}$ are independently Bernoulli random variables with $\Pr(k_i = 1) = p_i$, and let $\bar{K} = \mathbb{E}[K] = \sum_{i=1}^{N} p_i$. For a given feedback message length B and $\tilde{\epsilon}_{\text{fp}} \geq 2^{-B/\bar{K}}$, the probability that the false positive probability ϵ_{fp} exceeds $\tilde{\epsilon}_{\text{fp}}$ can be bounded as*

$$\Pr\left(\epsilon_{\text{fp}} > \tilde{\epsilon}_{\text{fp}}\right) < \Pr\left(\frac{e^{\left(\eta_{\tilde{\epsilon}_{\text{fp}}}-1\right)}}{\left(\eta_{\tilde{\epsilon}_{\text{fp}}}\right)^{\eta_{\tilde{\epsilon}_{\text{fp}}}}}\right)^{\bar{K}}, \tag{F.37}$$

where $\eta_{\tilde{\epsilon}_{\text{fp}}} = B/\left(\bar{K}\log_2(1/\tilde{\epsilon}_{\text{fp}})\right)$.

Proof. As in Proposition 2 we have

$$\Pr\left(\epsilon_{\text{fp}} > \tilde{\epsilon}_{\text{fp}}\right) \leq \Pr\left(K \geq \frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})}\right). \tag{F.38}$$

Defining $\eta_{\tilde{\epsilon}_{\text{fp}}} = B/\left(\bar{K}\log_2(1/\tilde{\epsilon}_{\text{fp}})\right)$ and applying the Chernoff bound for Poisson trials (see e.g., Theorem 4.4 in [23]), we obtain

$$\Pr\left(K \geq \frac{B}{\log_2(1/\tilde{\epsilon}_{\text{fp}})}\right) = \Pr\left(K \geq \eta_{\tilde{\epsilon}_{\text{fp}}}\bar{K}\right) \tag{F.39}$$

$$< \Pr\left(\frac{e^{\left(\eta_{\tilde{\epsilon}_{\text{fp}}}-1\right)}}{\left(\eta_{\tilde{\epsilon}_{\text{fp}}}\right)^{\eta_{\tilde{\epsilon}_{\text{fp}}}}}\right)^{\bar{K}}, \tag{F.40}$$

which completes the proof. □

We remark that although these bounds are based on the asymptotic bound from Eq. (F.15), similar bounds can be obtained for the practical schemes by first bounding the rounding error. For instance, for the scheme based on linear equations, we have $B_{\text{le}} = K\lceil\log_2(1/\epsilon_{\text{fp}})\rceil \leq K(\log_2(1/\epsilon_{\text{fp}}) + 1)$, which is straightforward to bound using the same methodology as in Propositions 1 to 3.

6.6 Random Access with Feedback and Retransmissions

In this section, we analyze the impact of feedback in a scenario with L transmission rounds, each comprising an uplink and a downlink phase. We first

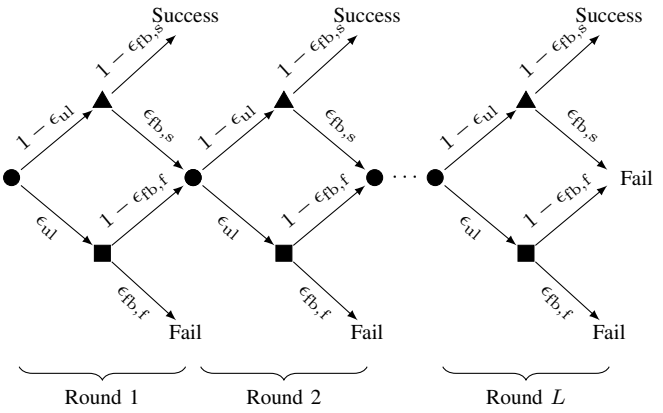


Fig. F.2: Events in the case with L transmission rounds. The circles represent the start of a round, triangles represent the case when a packet is successfully decoded by the BS, and the squares are when the packets are not decoded by the BS. A success occurs when a packet is both decoded by the BS and the user decodes the acknowledgment.

analyze the problem with packet erasure channels in both uplink and downlink, and then extend the analysis to a richer channel model that allows us to characterize the trade-off between false positives/negatives in the feedback message and the transmission rate. We assume that the transmission rounds are independent, and that the channel erasure probabilities are the same in all rounds.

6.6.1 Packet Erasure Channels

We consider the transmission scenario from the perspective of a single user and assume an uplink erasure probability ϵ_{ul}, downlink erasure probability ϵ_{dl}, false positive probability ϵ_{fp}, and false negative probability ϵ_{fn}, which are the same in all transmission rounds. Under these conditions, the transmission process is illustrated in Fig. F.2, where $\epsilon_{\text{fb,s}} = 1-(1-\epsilon_{\text{dl}})(1-\epsilon_{\text{fn}})$ and $\epsilon_{\text{fb,f}} = (1-\epsilon_{\text{dl}})\epsilon_{\text{fp}}$ are the probabilities that the user makes a wrong decision based on the feedback, conditioned on success or failure in the uplink, respectively. We assume that the user succeeds only if it received an acknowledgment for the packet transmitted in the same round, i.e., a false positive acknowledgment is a failure even if the uplink was successful in a previous round but the downlink in that round was unsuccessful. Furthermore, the user retransmits if it is unable to decode the feedback. The failure probability is then given as

$$\Pr(\text{fail}) = 1 - \sum_{l=1}^{L} \ell^{l-1}(1-\epsilon_{\text{ul}})(1-\epsilon_{\text{fb,s}}), \tag{F.41}$$

where $\ell = \epsilon_{\text{ul}}(1-\epsilon_{\text{fb,f}}) + (1-\epsilon_{\text{ul}})\epsilon_{\text{fb,s}}$ is the probability that the user proceeds from one transmission round to the next. To gain some insight into the behavior, suppose first that $L = 1$, in which case the expression reduces to

$$\Pr(\text{fail}) = 1 - (1-\epsilon_{\text{ul}})(1-\epsilon_{\text{dl}})(1-\epsilon_{\text{fn}}), \tag{F.42}$$

suggesting that ϵ_{ul}, ϵ_{dl} and ϵ_{fn} have equal importance in minimizing the failure probability. Furthermore, because false positives can only occur when the uplink fails, in which case the entire transmission fails since $L = 1$, the failure probability is independent of the false positive probability ϵ_{fp}. Similarly, suppose now that we allow an infinite number of retransmissions. By taking the limit $L \to \infty$ in Eq. (F.41) we obtain

$$\Pr(\text{fail}) = 1 - \frac{(1-\epsilon_{\text{ul}})(1-\epsilon_{\text{fn}})}{1-\epsilon_{\text{fn}}-\epsilon_{\text{ul}}(1-\epsilon_{\text{fn}}-\epsilon_{\text{fp}})}. \tag{F.43}$$

Note that this expression depends on the uplink erasure probability and the false positive/negative probabilities, but not on the downlink erasure probability ϵ_{dl}, which is because an erasure in the downlink always will result in a retransmission. When $\epsilon_{\text{fp}} = 0$ the expression reduces to $\Pr(\text{fail}) = 0$ indicating that even with false negatives (but no false positives), an arbitrarily high reliability can be achieved by increasing the transmission rounds. On the other hand, when $\epsilon_{\text{fn}} = 0$ the expression in Eq. (F.43) reduces to

$$\Pr(\text{fail}) = 1 - \frac{1-\epsilon_{\text{ul}}}{1-\epsilon_{\text{ul}}(1-\epsilon_{\text{fp}})}, \tag{F.44}$$

suggesting that when $\epsilon_{\text{fp}} > 0$ retransmissions cannot fully compensate for an unreliable uplink channel. The intuition behind this is that an unreliable uplink increases the probability of receiving a false positive, which in turn increases the failure probability.

Note that the analysis above holds even when the number of users K is random if the length of the feedback message and the transmission rate (channel coding rate) of the feedback message are adapted based on the instantaneous K to match ϵ_{dl}, ϵ_{fp}, and ϵ_{fn}. However, if the length of the feedback message remains fixed, ϵ_{fp} and ϵ_{fn} depends on the instantaneous K. In this case, a reasonable strategy is to use Eq. (F.41) to determine an appropriate ϵ_{fp}, and then apply either Proposition 1 or Proposition 2 to select the feedback message length such that the target false positive probability is satisfied with the desired probability.

6.6.2 Source/Channel Coding Trade-off

In practice, the erasure probability of the downlink transmission, ϵ_{dl}, is a function of the transmission rate and depends on the SNR at the receiver.

Furthermore, for a given number of symbols transmitted over the channel, the rate depends on the length of the feedback message B, which directly impacts the false positive/negative probabilities. Consequently, there is an inherent trade-off between ϵ_{dl}, ϵ_{fp} and ϵ_{fn}, which determine the overall reliability of the system. To illustrate the trade-off, suppose we can construct a feedback message with $\epsilon_{\text{fn}} = 0$ and false positive probability ϵ_{fp} using $K \log_2(1/\epsilon_{\text{fp}})$ bits, and that we aim to transmit it over a quasi-static fading channel with additive noise and instantaneous SNR given by γ. For a given number of symbols c, the transmission rate is given as $K \log_2(1/\epsilon_{\text{fp}})/c$ and the probability of decoding error is thus

$$\epsilon_{\text{dl}} = \Pr\left(\log_2(1+\gamma) < \frac{K \log_2(1/\epsilon_{\text{fp}})}{c}\right) \tag{F.45}$$

$$= \Pr\left(\gamma < \epsilon_{\text{fp}}^{-K/c} - 1\right), \tag{F.46}$$

illustrating, as expected, that decreasing ϵ_{fp} causes ϵ_{dl} to increase since a higher transmission rate is required.

If the number of symbols for the feedback message and the number of retransmissions L are fixed, the transmission rate and the false positive/negative probabilities need to be jointly optimized to minimize the failure probability in Eq. (F.41). This is in general a non-convex optimization problem that requires numerical evaluation of Eq. (F.41) over a range of ϵ_{fp}. The asymptotic expression in Eq. (F.43) suggests that when L is large, we should aim at minimizing ϵ_{fp} and ϵ_{fn} since ϵ_{dl} has no impact on the failure probability. In particular, in this case the introduction of false positives will lead to a worse performance compared to identifier concatenation, as there is no gain in reducing the length of the acknowledgment packet. However, when L is small, the downlink erasure probability has an increasing impact since a successful downlink transmission is required to succeed. In particular, to minimize the failure probability for $L = 1$ the downlink probability and the false negative probability should be equal, while ϵ_{fp} has no impact, as can be seen in Eq. (F.42).

6.7 Numerical Results

In this section, we evaluate the feedback schemes in a typical massive random access setting. We first present results that illustrate the impact of false positives in the setting with multiple transmission rounds and fixed K over a simple erasure channel. We then investigate the case with random K, and finally we exemplify the trade-off between allocating channel symbols for the uplink and the feedback under a random access channel in the uplink and

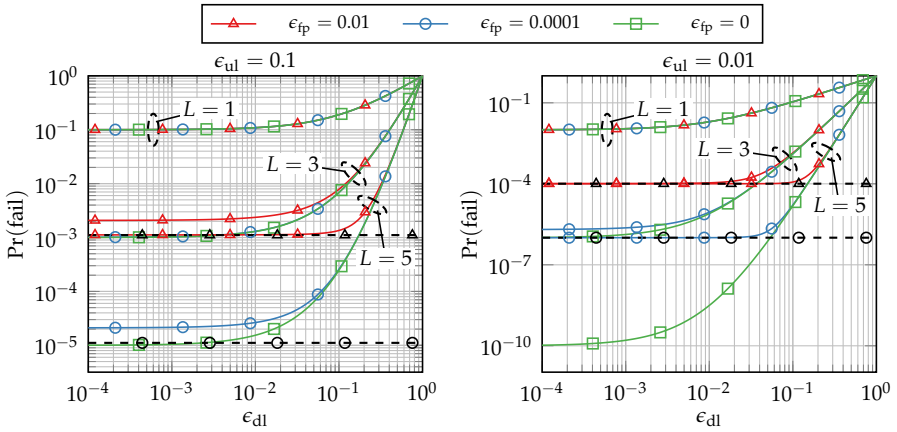


Fig. F.3: The probability that a transmission fails with L rounds obtained using Eq. (F.41). The dashed lines show the asymptotic results for $L \to \infty$.

a Rayleigh fading channel in the downlink. Except for the cases where it is explicitly mentioned, we will assume that $\epsilon_{\text{fn}} = 0$.

6.7.1 Fixed K and L Transmission Rounds

When K is fixed, the false positive probability ϵ_{fp} is constant and can be picked arbitrarily by choosing an appropriate feedback message length B. The probability that a transmission fails in a setting with L retransmissions is shown in Fig. F.3 along with the asymptotic results for $L \to \infty$, obtained using Eqs. (F.41) and (F.44), respectively. Although the downlink erasure probability ϵ_{dl} has no impact as $L \to \infty$, it has a significant impact when L is small. In particular, for finite L the failure probability is at least $(\epsilon_{\text{dl}})^L$, as a successful downlink transmission is required in order for the user to succeed. Similarly, we can observe an error floor as ϵ_{dl} approaches zero caused by both the false positive probability and the uplink erasure probability. When ϵ_{fp} is small the floor is approximately at $(\epsilon_{\text{ul}})^L$. On the other hand, when ϵ_{ul} is small, the error floor is dominated by ϵ_{fp}.

For low ϵ_{dl}, the failure probabilities are rather close to the asymptotic failure probabilities despite L being as low as 3 or 5 (where the solid and dashed lines coincide). In this regime, the failure probability is limited only by ϵ_{fp} and ϵ_{ul}, and increasing the downlink reliability or the number of transmission rounds will not lead to a reduced failure probability.

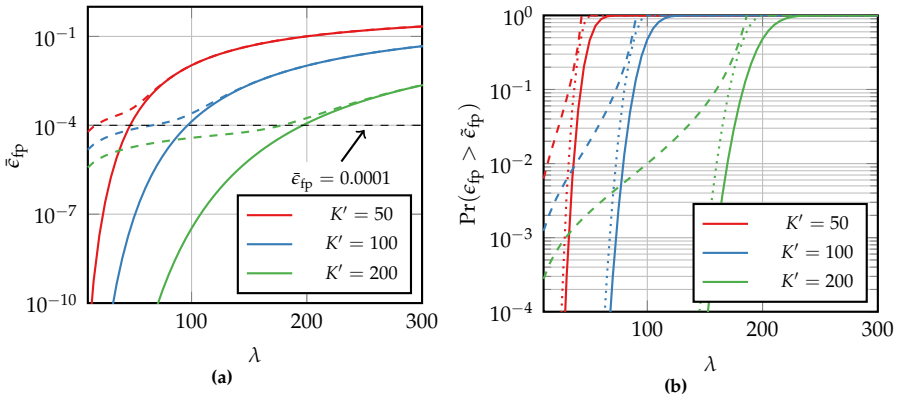


Fig. F.4: Illustrations of the bounds for (a) the expected false positive probability $\bar{\epsilon}_{\text{fp}}$, and (b) the probability that ϵ_{fp} exceeds $\tilde{\epsilon}_{\text{fp}} = 0.0001$ when K is Poisson distributed with mean λ. The feedback message length B is optimized to guarantee a false positive probability of $\tilde{\epsilon}_{\text{fp}} = 0.0001$ when $K = K'$. In (a) the dashed line shows Proposition 1; in (b) the dashed and dotted lines show Proposition 2 and Proposition 3, respectively.

6.7.2 Random K and L Transmission Rounds

We now study the case when K is random, and start by assessing the accuracy of the bounds derived in Propositions 1 to 3 and investigating how the false positive probability depends on the distribution of K. The impact of the distribution of K is illustrated in Fig. F.4, where K follows a Poisson distribution with mean λ. Figure F.4a shows the expected false positive probability $\bar{\epsilon}_{\text{fp}}$, computed numerically, and the bound from Proposition 1 when the feedback message length B is optimized to provide a false positive probability $\tilde{\epsilon}_{\text{fp}} = 0.0001$ when $K = K'$. As can be seen, the expected false positive probability is larger than the target false positive probability of $\tilde{\epsilon}_{\text{fp}} = 0.0001$ when $\lambda = K'$, suggesting that optimizing based on only the expected number of active users is insufficient. However, the bound, which also takes into account the variance of K, is accurate when λ is close to and greater than K', and can be used to pick a feedback message length that satisfies the target false positive probability when $\lambda = K'$ at the cost of only a minor message length penalty.

The probability that ϵ_{fp} exceeds $\tilde{\epsilon}_{\text{fp}} = 0.0001$ is shown in Fig. F.4b along with the bounds from Propositions 2 and 3. While the bound from Proposition 2 is reasonable when λ is close to K', it is generally quite weak due to the strong concentration of the Poisson distribution around its mean. However, by assuming that the users activate independently as in Proposition 3 the bound can be significantly tightened especially for low λ.

We now turn our attention to the case with L transmission rounds, and assume that the length of the feedback message B is selected using Proposi-

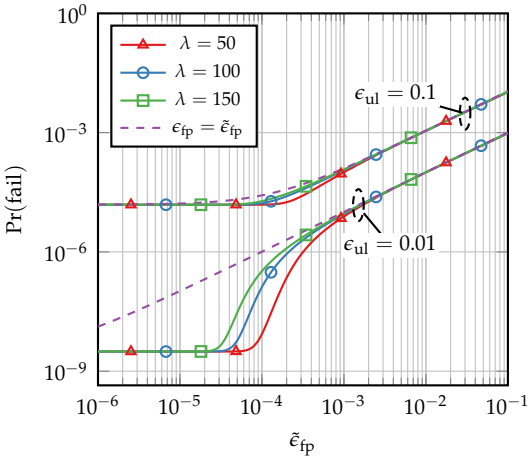


Fig. F.5: Failure probability vs. target average false positive probability $\tilde{\epsilon}_{\text{fp}}$ for Poisson arrivals with mean λ, for $L = 5$ and $\epsilon_{\text{dl}} = 0.01$. The message lengths are selected using the bound in Proposition 1, and the dashed lines show the failure probability when K is deterministic.

tion 1 to satisfy a given false positive requirement $\tilde{\epsilon}_{\text{fp}}$ on average. The failure probability when K is Poisson distributed is shown in Fig. F.5 for $L = 5$ and $\epsilon_{\text{dl}} = 0.01$. Because the message length is selected using the bound from Proposition 1, the failure probability for random K is lower than the one with deterministic K, indicated by the dashed lines. The gap between the failure probability for deterministic K and random K decreases as λ increases, which is due to the bound becoming more tight in this regime. This confirms that the bound can be used as a useful tool to select the message length.

6.7.3 Source/Channel Coding Trade-off

We finish the section by studying the trade-off between the number of bits used to encode the acknowledgments and the transmission rate. We assume that K is Poisson distributed with arrival rate λ and the uplink reliability is $\epsilon_{\text{ul}} = 0.1$. For the downlink, we pick ϵ_{dl} using Eq. (F.46) for the case in which the BS has 64 antennas, there are $L = 5$ transmission rounds, and $c = 2048$ channel symbols are available for the feedback, such that the transmission rate is $B/2048$ bits/symbol. Assuming a quasi-static flat-fading Rayleigh channel with average SNR $\overline{\text{SNR}}$, the instantaneous SNR at the user, γ, is Gamma distributed with shape and scale parameters equal to 64 and $\overline{\text{SNR}}/64$, respectively. We consider four encoding methods, namely identifier concatenation, the error-free (EF) method from Eq. (F.7), the scheme based on linear equations (LE) presented in Section 6.4.4, and the asymptotically optimal scheme from Eq. (F.15). For each value of λ and each encoding scheme,

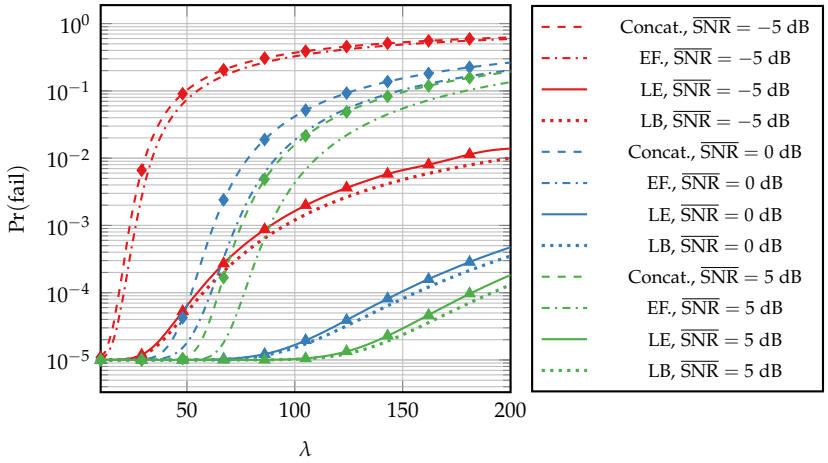


Fig. F.6: Failure probability for the concatenation based encoding compared to the error-free (EF) bound (Eq. (F.7)), the linear equations (LE) scheme (Eq. (F.24)), and the lower bound (LB, Eq. (F.15)) when the acknowledgment is transmitted over a Rayleigh fading channel with $c = 2048$ symbols and 64 transmitter antennas. K is Poisson distributed with mean λ, $\epsilon_{\text{ul}} = 0.1$ and $L = 5$. Diamond and triangle markers are obtained by simulation of the concatenation and LE schemes, respectively.

we optimize B so that $\Pr(\text{fail})$ is minimized when averaged over the instantaneous arrivals given λ. Thus, the transmission rate remains fixed for a given λ. In the concatenation and error-free schemes, we assume that each identifier requires 32 bits, and when the length of the feedback message is less than $32K$ (if the instantaneous K is large compared to the message length), we encode a random subset comprising $\lceil B/32 \rceil$ identifiers, which results in a false negative probability of $\epsilon_{\text{fn}} = 1 - \lceil B/32 \rceil / K$ (but no false positives). Therefore, while these representations are error-free when the number of bits B is adapted to K, they are *not* error free here where K is random and the number of bits is optimized to minimize the failure probability.

The results are shown in Fig. F.6 for $\overline{\text{SNR}} \in \{-5, 0, 5\}$ dB. The figure shows that, despite introducing false positives, the failure probability can be substantially decreased when the scheme based on linear equations is used compared to both the straightforward concatenation scheme and the bound given by the EF scheme. This is because admitting false positives allows the message length to be significantly reduced (and thus, the transmission rate), which in turn leads to much higher reliability of the downlink feedback. Furthermore, as expected the scheme based on linear equations performs close to the asymptotically optimal bound, with a gap caused only by the rounding in Eq. (F.24). Finally, we see that simulations, indicated by the markers, agree with the theoretical analysis, manifesting that the gains can be attained in practice. We note that these results have been obtained under

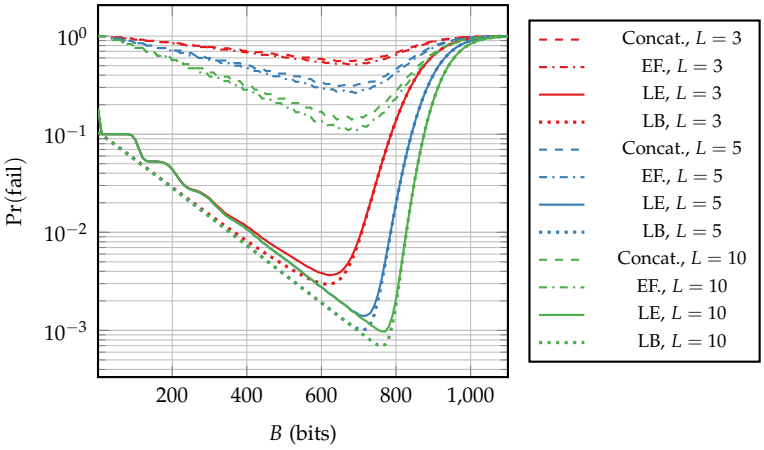


Fig. F.7: Failure probability for various acknowledgment message lengths B using the concatenation scheme, the error-free (EF) bound, the linear equations (LE) scheme, and the asymptotic lower bound (LB) with false positives over a Rayleigh fading channel with $c = 2048$ symbols, 64 transmitter antennas. K is Poisson distributed with mean $\lambda = 100$, $\epsilon_{\text{ul}} = 0.1$ and $\overline{\text{SNR}} = -5$ dB.

the assumption that the BS does not have channel state information (CSI) of the decoded users, although many massive random access schemes obtain this as part of the decoding procedure [4, 5]. When CSI is available, the BS can increase the SNR at the devices that were successful in the uplink, while the unsucessful users will experience a lower SNR. This effectively suppresses the false positive probability, leading to an even smaller failure probability.

Fig. F.7 shows the failure probability vs. feedback message length B for $\lambda = 100$, $\overline{\text{SNR}} = -5$ dB, and various number of transmission rounds L. As also suggested by the analysis in Section 6.6.2, for the asymptotic lower bound and the scheme based on linear equations, the feedback message length B that minimizes the failure probability increases as the number of transmission rounds increases, causing ϵ_{dl} to decrease. On the other hand, for the concatenation scheme, the feedback message length that minimizes the failure probability is the same independently of L. This is because ϵ_{fn}, which increases with B, and ϵ_{dl}, which decreases with B, both lead to the same event, namely a retransmission. This point has a high false negative probability of $\epsilon_{\text{fn}} \approx 0.78$, while the outage probability is low ($\epsilon_{\text{dl}} \approx 0.05$). On the other hand, the schemes that allows for false positives has ϵ_{fp} in the range 0.005 to 0.021, while ϵ_{dl} is ranges from approximately 0.01 to 0.30. This illustrates, in line with existing literature, the fact that the false positive probability should generally be kept smaller then the false negative probability. Despite this, the resulting failure probability is significantly smaller when false positives are allowed, compared to the case where they are not.

6.8 Conclusion

In this work, we have studied the use of message acknowledgments in a massive random access setting. We have shown that because of the large number of users that are active at any given time, encoding the feedback message requires a significant number of bits. To reduce this amount. we propose to allow for a small fraction of false positive acknowledgments, which results in a significant reduction in the length of the acknowledgment message. We have presented and analyzed a number of practical schemes of various complexity that can be used to realize these reductions, and shown that their performance is close to the information-theoretic optimum. With the basis of these schemes, we have studied their performance when the number of decoded users is random, and derived bounds on the false positive probability in this setting. Finally, we have studied how the schemes perform in a scenario with retransmissions, and shown, through numerical results, the extent to which reducing the feedback message length can improve the overall reliability of the random access scenarios.

A Derivation of Eq. (F.9)

For completeness, we derive here the lower bound in Eq. (F.9) presented as Proposition 4 in [14].

Suppose we construct a feedback message that acknowledges a set of users $\mathcal{W} \subset [N]$. We are interested in finding the number of sets $\mathcal{S}$ of size K that such a message can acknowledge while the requirements in terms of false positives and false negatives are satisfied. Clearly, in order to meet the false positive requirement, we must have $|\mathcal{W}| \leq K + \lfloor \epsilon_{\text{fp}} N \rfloor$.

Consider first the sets $\mathcal{S}$ for which the message $\mathcal{W}$ has exactly i false negatives and thus $K - i$ true positives. In order for $\mathcal{W}$ to be a valid message for such a set, at least $K - i$ users of $\mathcal{S}$ must belong to $\mathcal{W}$, while the remaining i users can be any of the $N - |\mathcal{W}|$ users that are not acknowledged by $\mathcal{W}$. For a given message $\mathcal{W}$, the number of such sets is $\binom{|\mathcal{W}|}{K-i}\binom{N-|\mathcal{W}|}{i} \leq \binom{K+\lfloor \epsilon_{\text{fp}} N \rfloor}{K-i}\binom{N}{i}$. Thus, the number of sets with up to $\lfloor \epsilon_{\text{fn}} K \rfloor$ false negatives is at most

$$\sum_{i=0}^{\lfloor \epsilon_{\text{fn}} K \rfloor} \binom{K + \lfloor \epsilon_{\text{fp}} N \rfloor}{K - i} \binom{N}{i} \leq K \binom{K + \lfloor \epsilon_{\text{fp}} N \rfloor}{K - \lfloor \epsilon_{\text{fn}} K \rfloor} \binom{N}{\lfloor \epsilon_{\text{fn}} K \rfloor}. \tag{F.47}$$

The total number of bits to represent all $\binom{N}{K}$ possible sets $\mathcal{S}$ is therefore at

most

$$B^*_{\text{fp,fn}} \geq \log_2 \left(\frac{\binom{N}{K}}{K\binom{K+\lfloor \epsilon_{\text{fp}} N \rfloor}{K-\lfloor \epsilon_{\text{fn}} K \rfloor}\binom{N}{\lfloor \epsilon_{\text{fn}} K \rfloor}} \right) \tag{F.48}$$

$$= \log_2 \binom{N}{K} - \log_2 \left(K \binom{K + \lfloor \epsilon_{\text{fp}} N \rfloor}{\lceil (1-\epsilon_{\text{fn}}) K \rceil} \binom{N}{\lfloor \epsilon_{\text{fn}} K \rfloor} \right). \tag{F.49}$$

Note that this bound is valid only when $\epsilon_{\text{fp}} < 1/2$, as it otherwise might be beneficial to encode the users that should *not* be acknowledged instead of the users that should.

References

[1] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5G: physical and mac-layer solutions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 59–65, 2016.

[2] C. Stefanovic and P. Popovski, "ALOHA random access that operates as a rateless code," *IEEE Trans. Commun.*, vol. 61, no. 11, pp. 4653–4662, 2013.

[3] E. Paolini, G. Liva, and M. Chiani, "Coded slotted ALOHA: A graph-based method for uncoordinated multiple access," *IEEE Trans. Inf. Theory*, vol. 61, no. 12, pp. 6815–6832, 2015.

[4] L. Liu, E. G. Larsson, W. Yu, P. Popovski, C. Stefanovic, and E. de Carvalho, "Sparse signal processing for grant-free massive connectivity: A future paradigm for random access protocols in the internet of things," *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 88–99, 2018.

[5] A. Fengler, P. Jung, and G. Caire, "Pilot-based unsourced random access with a massive mimo receiver in the quasi-static fading regime," in *2021 IEEE 22nd Int. Workshop Signal Process. Advances Wireless Commun. (SPAWC)*, 2021, pp. 356–360.

[6] M. Shirvanimoghaddam, M. Dohler, and S. J. Johnson, "Massive multiple access based on superposition raptor codes for cellular M2M communications," *IEEE Trans. Wireless Commun.*, vol. 16, no. 1, pp. 307–319, 2017.

[7] J. Kang and W. Yu, "Minimum feedback for collision-free scheduling in massive random access," *IEEE Trans. Inf. Theory*, pp. 1–1, 2021.

[8] J. Östman, R. Devassy, G. Durisi, and E. G. Ström, "Short-packet transmission via variable-length codes in the presence of noisy stop feedback," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 214–227, 2021.

[9] S. C. Draper and A. Sahai, "Variable-length channel coding with noisy feedback," *Eur. Trans. Telecommun.*, vol. 19, no. 4, pp. 355–370, 2008.

[10] P. Wu and N. Jindal, "Coding versus ARQ in fading channels: How reliable should the PHY be?" *IEEE Trans. Commun.*, vol. 59, no. 12, pp. 3363–3374, 2011.

[11] 3GPP, "Medium access control (MAC) protocol specification," 3rd Generation Partnership Project (3GPP), TS 36.321, Sep. 2021, v16.6.0.

[12] Y. Polyanskiy, "A perspective on massive random-access," in *2017 IEEE Int. Symp. Inf. Theory (ISIT)*, 2017, pp. 2523–2527.

[13] T. Cover, "Enumerative source encoding," *IEEE Trans. Inf. Theory*, vol. 19, no. 1, pp. 73–77, 1973.

[14] R. Pagh and F. F. Rodler, "Lossy dictionaries," in *Eur. Symp. Algorithms*. Springer, 2001, pp. 300–311.

[15] L. Carter, R. Floyd, J. Gill, G. Markowsky, and M. Wegman, "Exact and approximate membership testers," in *Proc. tenth annu. ACM Symp. Theory Comp. (STOC)*. ACM Press, 1978.

[16] M. Dietzfelbinger and R. Pagh, "Succinct data structures for retrieval and approximate membership," in *Int. Colloq. Automata, Languages, and Program.* Springer, 2008, pp. 385–396.

[17] P. Erdős and J. Spencer, *Probabilistic methods in combinatorics*. Academic Press New York, 1974.

[18] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, p. 422–426, Jul 1970.

[19] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Math.*, vol. 1, no. 4, p. 485–509, Jan 2004.

[20] E. Porat, "An optimal bloom filter replacement based on matrix solving," in *Int. Comput. Sci. Symp. Russia*. Springer, 2009, pp. 263–273.

[21] K.-M. Chung, M. Mitzenmacher, and S. Vadhan, "When simple hash functions suffice," in *Beyond the Worst-Case Analysis of Algorithms*, T. Roughgarden, Ed. Cambridge University Press, 2021, p. 567–585.

[22] P. C. Dillinger, L. Hübschle-Schneider, P. Sanders, , and S. Walzer, "Fast succinct retrieval and approximate membership using ribbon," *CoRR*, vol. abs/2106.12270, 2021.

[23] M. Mitzenmacher and E. Upfal, *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.

Paper G

Minimizing the Age of Information From Sensors With Common Observations

Anders E. Kalør and Petar Popovski

Published in
IEEE Wireless Communications Letters, vol. 8, no. 5, pp. 1390–1393, 2019,
doi: 10.1109/LWC.2019.2919528.

The layout has been revised.

Abstract

We study the average Age of Information (AoI) in a system where physical sources produce independent discrete-time updates that are each observed by several sensors. We devise a model that is simple, but still capable to capture the main tradeoffs. Two sensor scheduling policies are proposed to minimize the AoI of the sources; one in which the system parameters are assumed known, and one in which they are learned. Both policies are able to exploit the common sensor information to reduce the AoI, resulting in large reductions in AoI compared to common schedules.

7.1 Introduction

The rising number of use cases for Internet-of-Things (IoT) and Machine-Type Communication (MTC) has given rise to a metric termed Age of Information (AoI). It describes the timeliness of information at the receiver, see e.g. [1, 2]. Common to most AoI scenarios is that the information should be delivered in a timely manner, and usually only the most recent information is of interest. In this letter, we consider the scenario depicted in Fig. G.1, where K sources generate independent status updates at random time instants. Updates from source k are observed by sensor n according to a Bernoulli process with parameter p_{nk}. Each sensor stores the most recently observed update from each source. In each time slot, the Base Station (BS) can schedule one sensor to transmit its most recent update from a specific source. If the update is more recent than the most recent update known to the BS, the AoI of the source is reduced. Since each source update may be observed by multiple sensors, the updates stored by the sensors are correlated. For example, with two sources and two sensors, the most recent update from Source 1 may be known to both Sensor 1 and 2, while the most recent update from Source 2 may be known only to Sensor 2. This is a simplification of a scenario that is applicable in many IoT systems, such as sensors with overlapping observation areas that detect objects from a video feed, or redundant sensors for reliable monitoring the same physical phenomenon, such as oil spill. Unlike the independent updates, the fact that source updates are observed by multiple sensors introduces redundancy in the network, which can be used to reduce the AoI.

The AoI metric [1] has been used to study timely update policies in various multi-user contexts. In [2] the sources communicate to a receiver through a shared queue, [3] considers random access system, and scheduled access is considered in [4] and [5]. Temporally correlated updates from a single source are considered in [6], and in [7] the authors consider a wireless camera network in which the cameras have overlapping fields of view. Cameras with overlapping scenes are connected to the same destination nodes, which delay the processing until frames from all cameras that capture a specific

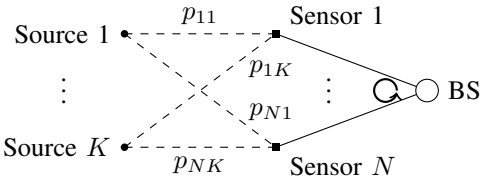


Fig. G.1: System model with K sources and N sensors.

scene have been received. Scenarios where sensors observe spatially correlated processes are presented in [8] and [9]. In both cases the sources sample and transmit updates independently at fixed (but possibly different) rates, and the authors study the estimation error resulting from various update strategies. The scenarios differ from ours as we treat centralized scheduling with arbitrary scheduling intervals, as well as discrete updates rather than a continuous process.

7.2 System Model

We consider a system with K sources, N sensors and one base station as depicted in Fig. G.1. The sources generate updates according to independent Poisson processes with rates $\lambda_1, \ldots, \lambda_K$. An update generated by source k is observed by sensor n with probability p_{nk}. Denote by $\Phi_{nk} = \{\phi_1^{(nk)}, \phi_2^{(nk)}, \ldots\}$ the set of time instances at which sensor n observes an update from source k. We then define the AoI of source k at sensor n at time t as $\Delta_{nk}(t) = t - \max_i \phi_i^{(nk)}$.

In each time slot $t = 1, 2, \ldots$, the BS can schedule a sensor to transmit its most recent update from a specific source. The transmission is error-free and instantaneous. Let $u_{nk}(t) = 1$ if sensor-source pair (n, k) is requested at time t, and $u_{nk}(t) = 0$ otherwise. We constrain the transmissions to be orthogonal by setting $\sum_n \sum_k u_{nk}(t) \leq 1$. The AoI of source k at the BS is then given by the following process:

$$\Delta^{(k)}(t) = \min \left(\sum_n u_{nk}(t) \Delta_{nk}(t), \Delta^{(k)}(t-1) + 1 \right). \tag{G.1}$$

To simplify the notation we let $\Delta^{(k)}(0) = 0$ for all k, and define the average AoI at the BS at time t as

$$\Delta'(t) = \frac{1}{K} \sum_{k=1}^{K} \Delta^{(k)}(t). \tag{G.2}$$

A realization of the process in Eq. (G.1) with a single source and two sensors is illustrated in Fig. G.2, where AoI at each sensor $\Delta_{nk}(t)$ is illustrated

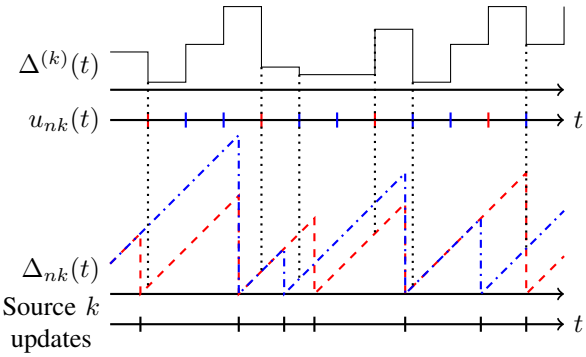


Fig. G.2: The AoI for a source k observed by two sensors, red (dashed) and blue (dash-dotted). Each sensor is scheduled according to an (arbitrary) policy indicated by $u_{nk}(t)$.

by red and blue lines. When a sensor is scheduled, indicated in the $u_{nk}(t)$ axis, the AoI at the BS $\Delta^{(k)}(t)$ is updated according to Eq. (G.1).

We seek policies π that characterize the scheduling, $u_{nk}(t)$, and minimize the long-term average of $\Delta'(t)$. Mathematically, we aim at solving the problem

$$\underset{\pi}{\text{minimize}} \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[\Delta'(t) \mid \Lambda(t-1)\right], \tag{G.3}$$

where $\Lambda(t)$ denotes the system state at time t containing the AoI at the BS and the *sampling histories*, $\mathcal{S}_{nk}(t)$, i.e. $\Lambda(t) = \Big(\mathcal{S}_{11}(t), \ldots, \mathcal{S}_{NK}(t), \Delta^{(1)}(t), \ldots, \Delta^{(K)}(t)\Big)$. $\mathcal{S}_{nk}(t)$ will be defined later.

We first consider the optimal policy under the assumption that the BS knows the parameters λ_k and p_{nk}. We then relax this requirement and consider a model-free policy inspired by contextual bandits that learns the parameters by exploring the system while keeping the AoI low.

7.3 Policies with Known Parameters

We first consider the set of policies for which the BS is assumed to know the system parameters λ_k and p_{nk}. It is straightforward to show that $\mathbb{E}\left[\Delta'(t) \mid \Lambda(t-1)\right]$ can be written as

$$\begin{aligned}\mathbb{E}\left[\Delta'(t) \mid \Lambda(t-1)\right] &= \Delta'(t-1) + 1 \\ &\quad - \frac{1}{K} \sum_{k=1}^{K} \sum_{n=1}^{N} u_{nk}(t) \mathbb{E}\left[\left(\Delta^{(k)}(t-1) + 1 - \Delta_{nk}(t)\right)^{+} \,\middle|\, \Lambda(t-1)\right],\end{aligned} \tag{G.4}$$

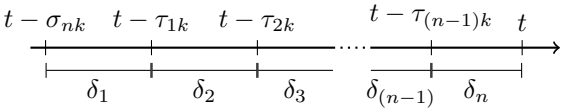


Fig. G.3: Notation used to derive the optimal policy.

where $(x)^+ = \max(x, 0)$. Since $\Delta'(t-1)+1$ is constant, the optimal policy for Eq. (G.3) is to maximize the expected reduction in $\Delta'(t)$:

$$(n(t),k(t)) \;=\; \arg\max_{n,k} \mathbb{E}\left[\left(\Delta^{(k)}(t-1)+1-\Delta_{nk}(t)\right)^+ \;\middle|\; \Lambda(t-1)\right], \quad \text{(G.5)}$$

where $(n(t),k(t))$ is sensor-source pair scheduled in time slot t, i.e. $u_{n(t)k(t)} = 1$. For simplicity, in the following we omit the dependency on $t-1$ in the expression for $\Lambda(t-1)$. Denote by Z_{nk} the event $\Delta_{nk}(t) < \Delta^{(k)}(t-1)+1$, i.e. that sensor-source pair (n,k) has a more recent observation. We then have

$$\begin{aligned}&(n(t),k(t))\\ &\quad= \arg\max_{n,k} \Pr(Z_{nk} \mid \Lambda)\Big(\Delta^{(k)}(t-1)+1-\mathbb{E}[\Delta_{nk}(t) \mid Z_{nk},\Lambda]\Big). \quad \text{(G.6)}\end{aligned}$$

To obtain an expression for $\Pr(Z_{nk} \mid \Lambda)$, observe that due to the memoryless property of the Poisson updates, Z_{nk} only depends on the sampling history since the previous reduction in $\Delta^{(k)}(t)$, or the last time sensor n was scheduled if this is more recent. Denoting by τ_{nk} the time since sensor-source pair (n,k) was last scheduled, and letting $\sigma_{nk} = \min(\tau_{nk}, \Delta^{(k)}(t-1)+1)$, we define

$$\mathcal{S}_{nk} = \{\tau_{mk} \mid m = 1,2\ldots N \wedge m \neq n \wedge \tau_{mk} < \sigma_{nk}\}. \quad \text{(G.7)}$$

Since both $\mathcal{S}_{nk}$ and $\Delta^{(k)}$ are independent across sources, we can consider each source independently in the derivation of $\Pr(Z_{nk} \mid \Lambda)$. Without loss of generality, suppose that the BS observes a new update from source k at time t by sampling sensor n after having unsuccessfully sampled sensor-source pairs $(1,k),(2,k),\ldots,(n-1,k)$. Notice that we may sample other sources between the samples of source k, and hence denote the time between sampling sensor-source pairs (j,k) and $(j+1,k)$ by δ_{j+1} as illustrated in Fig. G.3. From the memoryless property of the exponential distribution we have

$$\Pr(Z_{nk} \mid \Lambda) = 1 - \prod_{j=1}^{n}(1 - \Pr(Z_{nk}^{(j)} \mid \Lambda)), \quad \text{(G.8)}$$

where $Z_{nk}^{(j)}$ is the event that sensor n observed an update from source k during δ_j. Since per definition all samples up to t were unsuccessful, sensors

$j, \dots, n-1$ did not observe an update in δ_j. Denoting by L_j the random number of source updates in δ_j and using the fact that the sensor observations are independent conditioned on L_j, it follows that the probability that sensor n also did *not* observe an update in δ_j is given by

$$\begin{aligned} 1 - \Pr(Z_{nk}^{(j)} \mid \Lambda) &= \sum_{l=0}^{\infty} \frac{(\delta_j \lambda_k)^l e^{-\delta_j \lambda_k}}{l!} (1 - p_{nk})^l \\ &= e^{-\delta_j \lambda_k p_{nk}}, \end{aligned}$$

where we used the definition of conditional probability and the fact that L_j is Poisson distributed. Inserting into (G.8) yields

$$\begin{aligned} \Pr(Z_{nk} \mid \Lambda) &= 1 - \prod_{j=1}^{n} e^{-\delta_j \lambda_k p_{nk}} \\ &= 1 - e^{-\lambda_k p_{nk} \sum_{j=1}^{n} \delta_j} \\ &= 1 - e^{-\lambda_k p_{nk} \sigma_{nk}}. \end{aligned} \tag{G.9}$$

A consequence of this expression is that σ_{nk} completely describes the sampling history of sensor-source pair (n, k). In other words, the fact that we have not observed anything from previously scheduled sensors does not give us any information regarding the likelihood that sensor n has an update.

We derive $\mathbb{E}[\Delta_{nk}(t) \mid Z_{nk}, \Lambda]$ in a similar fashion. By independence of the observed sensors and memoryless observations, the time since the last observation $\Delta_{nk}(t)$ follows an exponential distribution truncated to $[0, \sigma_{nk}]$:

$$\begin{aligned} \mathbb{E}[\Delta_{nk}(t) \mid Z_{nk}, \Lambda] &= \frac{\int_0^{\sigma_{nk}} x \lambda_k p_{nk} e^{-\lambda_k p_{nk} x} \mathrm{d}x}{1 - e^{-\lambda_k p_{nk} \sigma_{nk}}} \\ &= \frac{1 - e^{-\lambda_k p_{nk} \sigma_{nk}} (\lambda_k p_{nk} \sigma_{nk} + 1)}{\lambda_k p_{nk} \left(1 - e^{-\lambda_k p_{nk} \sigma_{nk}}\right)}. \end{aligned} \tag{G.10}$$

By substituting (G.9) and (G.10) into Eq. (G.5) and rearranging we find that the optimal policy is

$$\begin{aligned} (n(t), k(t)) = \arg\max_{n,k} \left(\Delta^{(k)}(t-1) + 1 - \frac{1}{\lambda_k p_{nk}} \right) \left(1 - e^{-\lambda_k p_{nk} \sigma_{nk}}\right) \\ + \sigma_{nk} e^{-\lambda_k p_{nk} \sigma_{nk}}, \end{aligned} \tag{G.11}$$

which can be solved by iterating over all (n, k)-pairs with $p_{nk} > 0$.

7.4 Policies with Unknown Parameters

In this section, we consider policies that do not require λ_k and p_{nk} to be known. To this end, we formulate the problem as a contextual bandit problem [10], in which the agent (the BS) observes context information (the state)

prior to selecting an action, and receives an immediate reward (the reduction in AoI) that is independent of previous actions. This is suitable for our problem as the AoI reductions are conditionally independent given the state and action.

Several policies have been proposed for solving contextual bandits, ranging from the simple, sub-optimal ϵ-greedy [10] to the optimal but more complex Upper Confidence Bound (UCB) algorithms [11]. In this work, to demonstrate the applicability of contextual bandits to the problem, we restrict our focus on the simple ϵ-greedy policy, and remark that the performance may be improved by using an algorithm with better performance guarantees.

Because the state space is continuous (and hence infinite), we apply an approximate method that models the expected reward, $\mathbb{E}[R]$, achieved by scheduling a specific sensor-source pair (n,k) as a parametric function. The agent then picks the sensor-source pair that yields the highest $\mathbb{E}[R]$. Inspired by Eq. (G.6) we model the expected reward as

$$\begin{aligned}\mathbb{E}[R \mid \Lambda,(n,k)] &\approx \hat{p}\left(\boldsymbol{\psi}_{(n,k)}^{\mathrm{T}}\boldsymbol{x}_{\Lambda,(n,k)}\right)\left(\Delta^{(k)}(t-1)+1-\boldsymbol{\theta}_{(n,k)}^{\mathrm{T}}\boldsymbol{x}_{\Lambda,(n,k)}\right)\\ &\triangleq \hat{r}\left(\boldsymbol{x}_{\Lambda,(n,k)},\boldsymbol{\theta}_{(n,k)},\boldsymbol{\psi}_{(n,k)}\right),\end{aligned} \tag{G.12}$$

where $\hat{p}(x) = 1/(1+e^{-x})$ is the sigmoid function, $\boldsymbol{x}_{\Lambda,(n,k)}$ is a feature vector that represents the current state and the action of scheduling pair (n,k), and $\boldsymbol{\theta}_{(n,k)}$ and $\boldsymbol{\psi}_{(n,k)}$ are unknown parameter vectors that need to be learned. $\hat{p}(\cdot)$ represents the probability of receiving a non-zero reward, and $\boldsymbol{\theta}_{(n,k)}^{\mathrm{T}}\boldsymbol{x}_{\Lambda,(n,k)}$ is an estimate of the expected new age, conditioned on a non-zero reward, obtained by scheduling sensor-source pair (n,k) in state Λ. While this linear model is unable to represent the actual reward function in Eq. (G.11), the goal is to obtain an approximation that predicts which sensor-source pairs are likely to yield a high reward.

The feature vectors $\boldsymbol{x}_{\Lambda,(n,k)}$ are the input to the reward estimate $\hat{r}(\cdot)$ and represent the action of scheduling pair (n,k) in the current state. Since σ_{nk} is the only state parameter that characterizes the probability and the age of a new observation (see Eqs. (G.9) and (G.10)), we simply define $\boldsymbol{x}_{\Lambda,(n,k)} = [1,\sigma_{nk}]^{\mathrm{T}}$, where the first entry represents the bias element that allows the function to be shifted from the origin. While more complex features could be included, such as non-linear transformations of σ_{nk}, such features have shown not to give better performance in the scenario considered in Section 7.5.

7.4.1 Training Algorithm

In the ϵ-greedy algorithm [10], shown in Algorithm 3, the agent schedules with probability $1-\epsilon$ the sensor-source pair that maximizes the expected

reward, and with probability ϵ a pair selected uniformly at random. Therefore, ϵ controls the exploration/exploitation trade off and is usually set to a low value. After performing an action and observing the resulting reward, R, the agent updates the parameter vectors $\boldsymbol{\theta}_{(n,k)}$ and $\boldsymbol{\psi}_{(n,k)}$ using stochastic gradient descent. We update $\boldsymbol{\psi}_{(n,k)}$ using the cross-entropy loss function with derivative $\frac{\partial}{\partial \psi_i}\ell(y, \boldsymbol{\psi}^{\mathrm{T}}\boldsymbol{x}) = (\hat{p}(\boldsymbol{\psi}^{\mathrm{T}}\boldsymbol{x}) - y)x_i$, where y is 1 if $R > 0$ and 0 otherwise, i.e. $y = \mathbb{I}(R > 0)$. Similarly, $\boldsymbol{\theta}_{(n,k)}$ is updated with the least-squares loss function with derivative $\frac{\partial}{\partial \theta_i}\ell(y, \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{x}) = (\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{x} - y)x_i$ where $y = \Delta^{(k)}(t-1) + 1 - R$. Notice that $\boldsymbol{\theta}$ is only updated if the reward is nonzero.

Algorithm 3 ϵ-greedy algorithm.

Input: $\epsilon \geq 0$, learning rate $\alpha > 0$, initial state Λ
Initialize $\boldsymbol{\theta}_{(n,k)} \leftarrow \mathbf{0}$, $\boldsymbol{\psi}_{(n,k)} \leftarrow \mathbf{0} \quad \forall (n,k)$
For $t = 1, 2, \ldots$ **do**
  With probability ϵ **do**
    Draw (n', k') uniformly at random
  Else
    $(n', k') \leftarrow \arg\max_{(n,k)} \hat{r}\left(\boldsymbol{x}_{\Lambda,(n,k)}, \boldsymbol{\theta}_{(n,k)}, \boldsymbol{\psi}_{(n,k)}\right)$
  End
  $\Delta^{(k)\prime} \leftarrow \Delta^{(k)}(t)$
  Schedule (n', k'), observe reward R and new state Λ'
  If $R > 0$ **then**
    $\hat{\Delta}_{nk} \leftarrow \boldsymbol{\theta}_{(n',k')}^{\mathrm{T}} \boldsymbol{x}_{\Lambda,(n',k')}$
    $\boldsymbol{\theta}_{(n',k')} \leftarrow \boldsymbol{\theta}_{(n',k')} - \alpha(\hat{\Delta}_{nk} - \Delta^{(k)\prime} + R)\boldsymbol{x}_{\Lambda,(n',k')}$
  End if
  $\hat{p} \leftarrow \left(1 + \exp(-\boldsymbol{\psi}_{(n',k')}^{\mathrm{T}} \boldsymbol{x}_{\Lambda,(n',k')})\right)^{-1}$
  $\boldsymbol{\psi}_{(n',k')} \leftarrow \boldsymbol{\psi}_{(n',k')} - \alpha\left(\hat{p} - \mathbb{I}(R > 0)\right)\boldsymbol{x}_{\Lambda,(n',k')}$
  $\Lambda = \Lambda'$
End for

7.5 Numerical Results

We consider the case with $K = 20$ sources and $N = 20$ sensors. To obtain a wide variety of observation probabilities, we let sensor n observe updates from each source with probability $p_{nk} = 1/2^n$ for $k = 1, \ldots, K$. In addition to the two algorithms presented in the previous sections (optimal and ϵ-greedy), we evaluate random sampling, highest σ_{nk} first, which samples the sensor-source pair with highest σ_{nk}, and a genie-aided algorithm that is aware of the

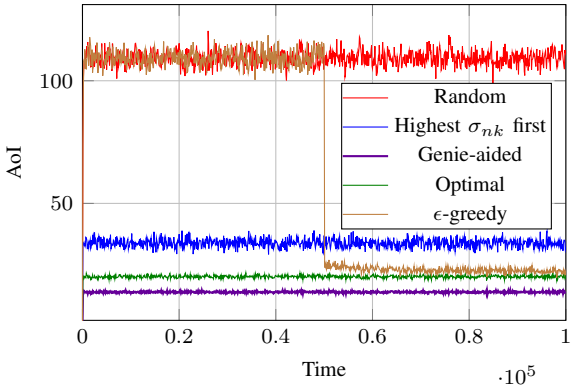


Fig. G.4: Average AoI for $\lambda = 0.5$ and $p_{nk} = 1/2^n$. The initial exploration of the ϵ-greedy policy is reflected in the decrease in AoI after 50000 time slots.

age at each sensor and always schedules the one that results in the largest reduction in age. For ϵ-greedy we use a learning rate of $\alpha = 10^{-5}$. Smaller values result in slow convergence, while larger values cause divergence due to the sporadic rewards. In order to shorten the convergence time, we perform full exploration in the first 50 000 time slots by setting $\epsilon = 1$, and then after 50 000 time slots reduce it to $\epsilon = 0.1$. Each simulation is run for a total duration of 100 000 time slots, but to eliminate the bias of the random exploration in the ϵ-greedy policy, we discard the initial 60 000 time slots from all policies.

We first study the average AoI over time in a scenario where updates are generated with equal rates $\lambda = 0.5$. Figure G.4 shows the average AoI for the different policies averaged over 30 realizations. The optimal policy performs close to the genie-aided policy, and the ϵ-greedy algorithm also achieves a low AoI after the initial random exploration, which suggests that the system parameters can be efficiently learned and exploited despite the simple reward model.

Figure G.5 shows the same scenario for varying values of λ. As before, the optimal strategy performs close to the genie-aided strategy. Furthermore, ϵ-greedy performs close to optimally for $\lambda > 0.03$, while the performance is bad for low values of λ. This is due to overfitting of the prediction model when non-zero rewards occur rarely. It is likely that this can be improved by proper tuning of α and increasing the simulation time to allow for the slower convergence. All policies except the random approach an AoI of $K/2 = 10$ as $\lambda \to \infty$, which is the average age obtained if the sources are scheduled in a round-robin fashion and always provide new updates.

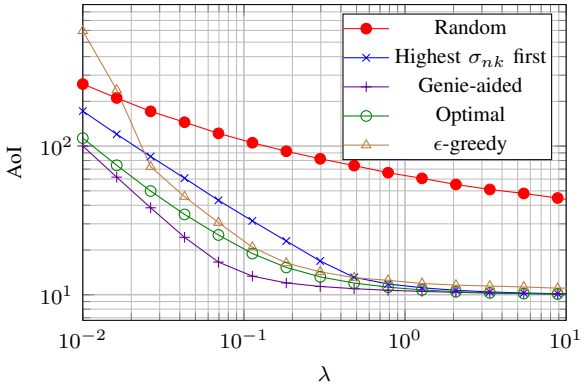


Fig. G.5: Average AoI with $p_{nk} = 1/2^n$ for varying λ.

7.6 Conclusion

We have studied sensors with common observations using the framework of AoI. We have presented two sensor scheduling policies that can exploit the correlation and shown that they can lead to a significant reduction of the AoI compared to policies that do not take correlation into account. The first policy is optimal and performs close to a genie-aided policy, but assumes that the system parameters are known. The second policy, based on contextual bandits, learns the parameters by exploring the system. Although this causes a performance gap, the results indicate that the contextual bandits framework can be used to learn a good policy, but that it is sensitive to the learning rate that needs to be tuned for the specific system characteristics.

References

[1] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Proc. IEEE INFOCOM*, 2012, pp. 2731–2735.

[2] R. D. Yates and S. Kaul, "Real-time status updating: Multiple sources," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2012, pp. 2666–2670.

[3] R. D. Yates and S. K. Kaul, "Status updates over unreliable multiaccess channels," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 331–335.

[4] I. Kadota, A. Sinha, and E. Modiano, "Optimizing age of information in wireless networks with throughput constraints," in *Proc. IEEE INFOCOM*, 2018.

[5] R. Talak, I. Kadota, S. Karaman, and E. Modiano, "Scheduling policies for age minimization in wireless networks with unknown channel state," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, 2018, pp. 2564–2568.

[6] S. Poojary, S. Bhambay, and P. Parag, "Real-time status updates for correlated source," in *Proc. IEEE Information Theory Workshop (ITW)*, 2017, pp. 274–278.

[7] Q. He, G. Dan, and V. Fodor, "Minimizing age of correlated information for wireless camera networks," in *Proc. IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2018, pp. 547–552.

[8] Z. Jiang and S. Zhou, "Status from a Random Field: How Densely Should One Update?" *arXiv e-prints*, p. arXiv:1901.05096, Jan 2019.

[9] J. Hribar, M. Costa, N. Kaminski, and L. A. DaSilva, "Updating strategies in the internet of things by taking advantage of correlated sources," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, Dec 2017, pp. 1–6.

[10] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An introduction*, 2nd ed. The MIT Press, 2018.

[11] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proc. 19th International Conference on World Wide Web (WWW)*. ACM, 2010, pp. 661–670.

Paper H

Timely Monitoring of Dynamic Sources with Observations from Multiple Wireless Sensors

Anders E. Kalør and Petar Popovski

Submitted to
IEEE/ACM Transactions on Networking, 2021.
Preprint: `https://arxiv.org/abs/2012.12179`.

The layout has been revised.

Abstract

Age of Information (AoI) has recently received much attention due to its relevance in IoT sensing and monitoring applications. In this paper, we consider the problem of minimizing the AoI in a system in which a set of sources are observed by multiple sensors in a many-to-many relationship, and the probability that a sensor observes a source depends on the state of the source. This model represents many practical scenarios, such as the ones in which multiple cameras or microphones are deployed to monitor objects moving in certain areas. We formulate the scheduling problem as a Markov Decision Process, and show how the age-optimal scheduling policy can be obtained. We further consider partially observable variants of the problem, and devise approximate policies for large state spaces. Our evaluations show that the approximate policies work well in the considered scenarios, and that the fact that sensors can observe multiple sources is beneficial, especially when there is high uncertainty of the source states.

8.1 Introduction

Sensing and monitoring of the environment is a generic use case for the Internet of Things (IoT) and massive Machine Type Communications (mMTC), representing scenarios in which a very large number of devices are connected wirelessly (up to 300,000 within a single cell [1]). In a typical sensing and monitoring scenario, a large number of sensors are deployed across a large area. The sensors sporadically or periodically sense and transmit their observations to a destination node, e.g., in the cloud, which processes the observations and possibly initiates some actions, such as adjusting control parameters, raising an alarm, etc. One example of such systems is the smart grid [2, 3], in which sensors continuously monitor the state of the power grid and report their observations to a central controller. Other prominent examples include industrial manufacturing systems, where cameras regularly take pictures of the products in order to track the process, and monitoring of the environment, such as temperature and humidity [4].

However, the large number of devices in the IoT imposes a significant constraint on the number of observations that can be communicated from the sensors. As a result, designing intelligent schemes for selecting when and which observations to transmit can provide notable gains in the performance of such systems. Although defining a data relevance or importance measure is inherently an application-specific task, for a large amount of IoT use cases the *age* of the observations is a reasonable criterion. More specifically, the age refers to the time elapsed since the generation of the most recent observation that is known to the destination. Returning to the previous examples, timely updates are indeed critical in smart grid monitoring systems, where some

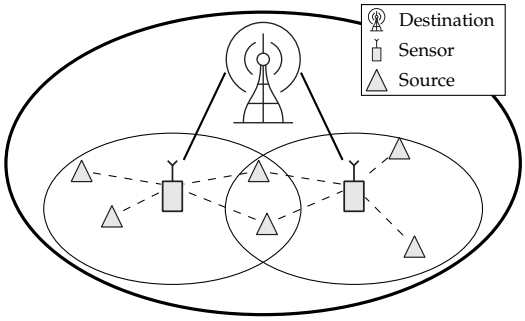


Fig. H.1: The studied scenario comprising a number of sources that are observed by sensors, and a destination node that requests the observations from the sensors. The objective is to schedule the sensors so as to minimize the AoI of the sources.

action should be taken in case of anomalies. Similarly, environmental monitoring also calls for frequent reportings. This wide applicability of using the age of the observations as a relevance measure has led to the notion of *Age of Information* (AoI), which has inspired a large number of works (see e.g., [5] and Section 8.1.2).

Another characteristic that is shared among many IoT use cases is the fact that the sensor observations may be correlated. This may, for instance, arise from sampling some physical phenomenon that is not confined to exist only at the points where the sensors are located but spans an area monitored by several sensors, as illustrated in Fig. H.1. Conversely, a single sensor may be able to observe several phenomena at the same time. Unless this correlation is considered in the design of the communication protocols, it is likely to lead to redundant information being communicated to the destination. In addition, the phenomena may be dynamic, so that they change characteristics over time. For instance, a phenomenon may exhibit mobility such that it enters and leaves the range of each individual sensor. Similarly, a phenomenon may be observable by the sensors only when it is in certain states, e.g., powered on.

In this paper, we study a generalized version of the system depicted in Fig. H.1, in which a destination node monitors a set of dynamic sources with states defined by a Markov chain. The sources are monitored through a set of sensors. A sensor can observe a given source with some probability that depends on the state of the source. At each time slot, the destination node requests one sensor to perform a measurement, and the sensor reports the measurement through an erasure channel to the destination node. For instance, the model can represent an industrial scenario with automatic guided vehicles (AGVs) that are observed by a number of cameras. The location of each AGV is represented by its state and the probability that a camera observes an AGV depends on whether the AGV is within the camera's field of view. We

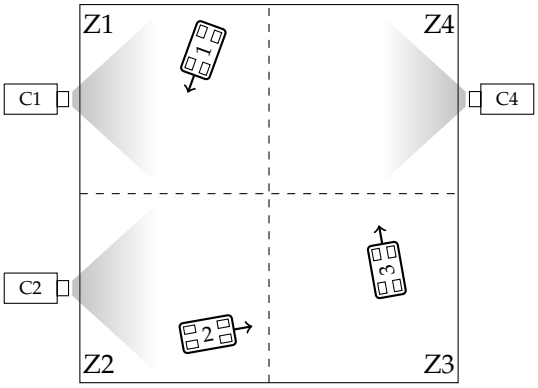


Fig. H.2: Example scenario of a factory with AGVs and three cameras that cover different zones. We consider the problem of scheduling sensors (cameras) to maximize the freshness of their observations of a set of sources (AGVs). Each sensor may observe multiple sources at the same time, e.g., if multiple AGVs are within the same zone.

consider both the case in which the source states are fully observable to the destination node, and the case in which the source states are unobservable and need to be inferred through the measurements. In the case of AGVs, the fully observable case could for instance represent the scenario in which the AGVs are equipped with localization systems so that their locations are known to the scheduler, while the unobservable case could represent the case in which the locations of the AGVs are only observed through the sampled cameras, e.g., using image recognition techniques.

Although this scenario represents a somewhat extreme case of sensing, since sources are either perfectly observed or not observed at all, it allows us to conveniently abstract detailed aspects, such as source coding and distortion. Our overall aim is to design scheduling schemes that minimize the average AoI at the destination node, so that it at any time has a timely overview of the sources. The combination of dynamic phenomena and the fact that sensors can observe multiple sources introduces a choice between getting an update from a sensor that observes a single source for which the current data is with high AoI and likely outdated, or sampling a sensor that observes multiple sources, for which the current data is fresher, i.e., with smaller AoIs. The optimal choice that minimizes the long-term average AoI requires a long-sighted strategy and depends on the dynamics of the source states, as illustrated through the following example.

8.1.1 Motivating example

Consider a factory with AGVs, where cameras are used to monitor the AGVs in order to ensure that they are operating normally. Suppose for simplic-

Table H.1: Myopic and optimal policies for a single round in the scenario from Fig. H.2 with four zones and three AGVs starting in zones Z1, Z2, and Z3 and with initial AoIs of 1, 1, and 4, respectively. The decisions are written $(\text{AoI } 1, \text{AoI } 2, \text{AoI } 3) \rightarrow$ decision. The Myopic policy achieves a total AoI of 38 while the optimal achieves 36.

Time slot	1	2	3	4	5	6	...
Locations	(Z1, Z2, Z3)	(Z2, Z3, Z4)	(Z3, Z4, Z1)	(Z4, Z1, Z1)	(Z1, Z1, Z2)	(Z1, Z2, Z3)	
Myopic	$(1,1,4) \rightarrow$ C1	$(1,2,5) \rightarrow$ C4	$(2,3,1) \rightarrow$ C4	$(3,1,2) \rightarrow$ C4	$(1,2,3) \rightarrow$ C1	$(1,1,4) \rightarrow$ C1	
Optimal	$(1,1,4) \rightarrow$ C1	$(1,2,5) \rightarrow$ C4	$(2,3,1) \rightarrow$ C4	$(3,1,2) \rightarrow$ C1	$(4,1,1) \rightarrow$ C1	$(1,1,2) \rightarrow$ C1	

ity that the factory is divided into four zones (zones 1–4), as illustrated in Fig. H.2. Zones 1, 2 and 4 are covered by cameras C1, C2 and C4, respectively, while zone 3 is hidden for the cameras. In each time slot, the AGVs move from one zone to the next in a anti-clockwise direction, except in zone 1, where they spend two time slots before moving to the next zone, e.g., due to physical obstacles that prevents the AGVs from moving at their regular speed. Now, suppose further that due to capacity limitations it is only possible to request an image from one camera every time slot, and thus they need to be scheduled. Putting this into the framework presented earlier, the AGVs are sources with states that corresponds to their location, and the cameras are sensors. A naïve scheduling scheme would be to schedule the cameras in a round-robin fashion. Although this may work reasonably well in a small and simplistic scenario like this, it will not perform well in more complex scenarios with many AGVs and zones. An alternative approach would be to exploit the structure of the factory and schedule the cameras only when the AGVs are expected to be inside the cameras fields of view. For instance, a myopic AoI scheduler would schedule the camera with the AGV that has been captured least recently once it enters a camera region. However, as illustrated in Table H.1, it is possible to do even better using a long-sighted policy that exploits the fact that the AGVs move slower in region 1, which is more likely to contain multiple AGVs in the same time slot. By following the optimal policy, the total age can be reduced from 38 time slots in the myopic policy to 36 time slots. This illustrates that finding the optimal scheduling policy is non-trivial, since it requires that both the dynamics of the AGVs and the locations of all AGVs jointly are taken into account.

Note also that we assumed that the movement of the AGVs is fully deterministic, and that the sensor observations are ideal. The scheduling problem becomes more challenging when the movement is random and the observations are non-ideal, since the scheduler may have to start scheduling an AGV several time slots before it is expected to enter the hidden zone. Similarly, it is more difficult to define a good policy when the locations of the AGVs are not fully observable, but revealed only when the AGVs are captured by one the cameras. In that case, the scheduler needs to consider both the AGV dynamics and its belief in the location estimates.

8.1.2 Related Work

The concept of Age of Information has recently received much attention in the context of Internet of Things (IoT), where it has been used to characterize the fundamental trade-off between update rate and timeliness in Poisson update systems a capacity constraint, see e.g., [5–8]. A common characteristic that often appears in these systems is that the AoI is a U-shaped function of the update rate, so that there exists a rate that minimizes the AoI.

The literature of AoI for systems with multiple sources can roughly be divided into the ones that consider systems that are queuing-based, and ones that study scheduling- or sampling-based systems. Notable contributions in the first category include [9] in which updates from each source are independently sensed and processed by multiple servers before being delivered to the destination. [10–13] characterize the AoI when updates from multiple sources are delivered to the destination through a shared queue, and similar scenarios, but with scheduling and packet management, are studied in [14, 15]. Packet management schemes for the case in which updates are queued at the source before being transmitted over a shared medium are studied in [16].

Contributions related to sampling of multiple sources include [17], which proposes various scheduling policies to minimize the AoI of independent sources. [18] considers the problem of scheduling sources to transmit through a shared queue. They define the problem as a Markov Decision Process (MDP) and derives the optimal policy. A similar problem is studied in [19], where each source has its own queue. In [20] sources are scheduled to transmit over erasure channels in which the error probabilities evolve according to a Markov chain. The authors present the optimal policy under the assumption that the scheduler can observe the instantaneous channel states before making a scheduling decision. In contrast to our work, they do not distinguish between sources and sensors.

Correlated sources and joint observations in the context of AoI have previously been studied in [21–26]. In [21], the authors consider remote estimation of a Gaussian Markov random field that is sampled and queued by multiple sources before it is sent to the destination node. A similar situation with spatially correlated observations are studied in [22], where the authors show how the correlation can be exploited to reduce the energy consumption of the sources. For the case with discrete sources, [23] studies a scenario comprising a number of cameras that each captures a specific scene and transmits its images to a destination fog computing node over a shared channel. The same scenes may be captured by multiple cameras, and the authors propose a joint fog node assignment and scheduling scheme so that the average age of each scene at the fog nodes is minimized. We note that our work represents a generalization of that scenario by allowing each camera (sensor) to observe multiple scenes (sources), and also allows the sources to be dynamic. Similar

scenarios have been studied in [24–26]. In [24] we proposed a scheduling mechanism for the case in which each sensor can observe multiple sources, and each source update is observed by a random subset of the sensors. This work differs from [24] by assuming that observations are generated "at will", and by allowing sources to be stateful, so that the probability that a sensor observes a source depends on its state. Furthermore, we rigorously prove the optimality of the proposed policies. A similar scenario is considered in [25], where multiple sensors observe a single source that generates updates and each update is observed by a given sensor with certain probability. The authors focus on the partially observable case where the sensor observations are unknown to the destination node, and they study the performance of the myopic scheduling policy for the problem. Finally, [26] propose a scheduling scheme for the case with two types of correlated sensors, one receiving random updates from the source and one that is able to sample the source. They show that a long-sighted policy provides significantly lower AoI than the myopic policy.

8.1.3 Contributions and Paper Organization

Our aim with this paper is not to provide new fundamental results on AoI, but rather to study how dynamic sensors influence the scheduling decisions when minimizing the AoI. Specifically, the contributions of this paper can be summarized as follows

- We define a general system model that captures many realistic IoT scenarios, such as camera monitoring, in which sources are dynamic and their observation probabilities change according to their state. Furthermore, sources and sensors have a many-to-many relationship in that each source can be observed by multiple sensors, and each sensor can observe multiple sources. We rigorously prove the existence of an optimal deterministic stationary policy for the problem when the source states are fully observable, and we show how it can be found using value iteration.
- We analyze three partially observable variants of the problem, in which the source states cannot be directly observed, but instead (1) the source states are revealed only for the observed sources, (2) the source identities, but not the states, are revealed of the observed sources, or (3) the sensor measurements do not reveal whether they contain a source. Furthermore, we devise heuristic scheduling policies for these problems.
- Using numerical evaluations, we characterize the different policies and study the impact of the source dynamics and the fact that sensors can observe multiple sources on the AoI. We show that the latter can be

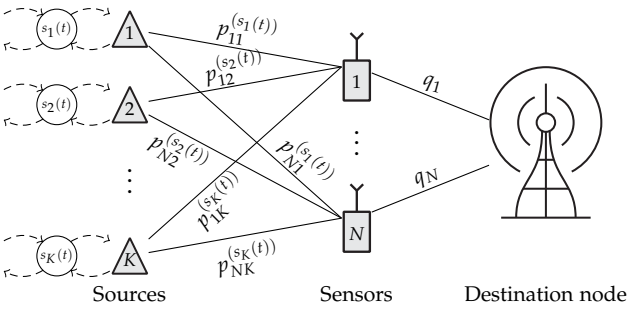


Fig. H.3: A general system with K stateful sources, N sensors and one destination node. In each time slot t, the destination node requests a sensor to perform a measurement. Sensor n observes source k with probability $p_{nk}^{(s_k(t))}$, where $s_k(t)$ denotes the state of source k. The measurement from sensor n is delivered to the base station with probability q_n.

beneficial especially in the partial observable case where there is uncertainty about the source states, provided that the policy is able to take the uncertainty into account.

The remainder of the paper is organized as follows. In Section 8.2 we formally define the system model. We then analyze the random scheduling policy in Section 8.3, followed by the optimal scheduling policies in Section 8.4 for the case where the source states are fully observable. In Section 8.5 we consider two cases: (1) unobservable source states, representing the situation in which the sources can be identified from the observations, and (2) undetectable sources, representing the situation in which the observations do not reveal the detected sources to the scheduler. Section 8.6 presents numerical results, and finally the paper is concluded in Section 8.7.

8.2 System Model and Problem Formulation

We consider a system with N sensors that observe K stateful sources, as illustrated in Fig. H.3. The time is divided into time slots and in each slot a sensor, denoted $a_t \in \{1, 2, \ldots, N\}$, is requested to perform a measurement, which is intended to be delivered to a central destination node. The measurement contains a random subset of the sources, and the probability that a sensor measurement contains a source depends on the state of the source. We denote the state of source k in time slot $t = 1, 2, \ldots$ by $s_k(t) \in \{1, 2, \ldots, S_k\}$, and the probability that a measurement by sensor n contains source k in state s by $p_{nk}^{(s)}$. $s_k(t)$ is modeled as a discrete-time Markov chain with transition matrix $\mathbf{R}_k = [r_{ij}^{(k)}]$, where $r_{ij}^{(k)} = \Pr(s_k(t) = j | s_k(t-1) = i)$. The Markov chain is assumed to be irreducible and aperiodic, and to have the stationary state dis-

tribution $\boldsymbol{\beta}_k$ defined as the vector that satisfies $\boldsymbol{\beta}_k \mathbf{R}_k = \boldsymbol{\beta}_k$. In the remainder of the paper, we will refer to the event that a sensor measurement contains a given source as the sensor *observes* the source. The measurements are delivered from the sensor to the destination node through an erasure channel with error probability $1 - q_n$, which captures both errors in the request and the delivery transmissions.

We remark that the system model contains a number of interesting special cases, including the one where the channel is capacity constrained, i.e., where the sensor can deliver updates from at most $C \leq K$ sources in each time step. This can be modelled by defining, for each sensor, $\binom{K}{C}$ *virtual sensors* that observe the various subsets of C sources.

The overall goal is to minimize the average AoI at the destination node. Let $u_n(t)$ be a binary indicator of the event that sensor n has been scheduled at time t, i.e.,

$$u_n(t) = \mathbb{1}\left[a_t = n\right], \tag{H.1}$$

where $\mathbb{1}[\cdot]$ is the indicator function that equals one if the argument is true and otherwise is zero. The scheduled sensor can carry up to K observations, and thus the AoI of source k at the destination node, denoted by $\Delta_k(t)$, is described by the process

$$\Delta_k(t+1) = \begin{cases} 1 & \text{if } \sum_n \zeta_{nk}(t) u_n(t) = 1, \\ \Delta_k(t) + 1 & \text{otherwise,} \end{cases} \tag{H.2}$$

where $\zeta_{nk}(t)$ is the Bernoulli random variable that indicates successful measurement and transmission of source k, i.e., is one with probability $p_{nk}^{(s_k(t))} q_n$ and zero with probability $1 - p_{nk}^{(s_k(t))} q_n$. In other words, the AoI of source k is reset if the sensor successfully measures source k *and* the measurement is successfully transferred to the destination node. We shall assume that each source has a state in which it can be observed by a sensor, i.e., $\sum_{n=1}^{N} \sum_{s=1}^{S_k} q_n p_{nk}^{(s)} > 0$ for all k. To simplify the subsequent notation we let $\Delta_k(0) = 0$ for all k. Then, we denote the average AoI at the destination at time t as

$$\Delta'(t) = \frac{1}{K} \sum_{k=1}^{K} \Delta_k(t) \tag{H.3}$$

and seek a scheduling policy π that minimizes the long term expected AoI

$$\underset{\pi \in \Pi}{\text{minimize}} \limsup_{T \to \infty} \frac{1}{T} \mathbb{E}\left[\sum_{t=1}^{T} \Delta'(t)\right]. \tag{H.4}$$

Here $\pi \in \Pi$ of a sequence of decision rules that map the entire history of states and observations to the action space, potentially through a random mapping, and the expectation is taken over the decisions, the AoI process,

and the source state transitions. The problem forms an average-cost MDP, in which the actions a_t belong to the finite action space $\{1, 2, \ldots, N\}$. The state space represents the current AoI of each source as well as the states of the sources. To formalize the setting, we denote the system state at the destination node as

$$\mathbf{\Lambda} = (\boldsymbol{S}, \boldsymbol{\Delta}), \tag{H.5}$$

where $\boldsymbol{S} = \{s_1, s_2, \ldots, s_K\}$ and $\boldsymbol{\Delta} = \{\Delta_1, \Delta_2, \ldots, \Delta_K\}$ are the vectors of source states and AoIs, respectively. Furthermore, to be consistent with the MDP literature and to make the dependency on the action explicit, we define the cost of taking action a in state $\mathbf{\Lambda}$ as the expected AoI in the following slot, i.e.,

$$C(\mathbf{\Lambda}, a) = \frac{1}{K}\sum_{k=1}^{K}\Delta_k + 1 - \frac{1}{K}\sum_{n=1}^{N}\sum_{k=1}^{K} u_n p_{nk}^{(s_k)} q_n \Delta_k, \tag{H.6}$$

where we used the definition in Eq. (H.2). Using this definition, we express the problem as

$$\underset{\pi\in\Pi}{\text{minimize}} \limsup_{T\to\infty} \frac{1}{T+1}\,\mathbb{E}\left[\sum_{t=0}^{T} C(\mathbf{\Lambda}_t, a_t)\,\middle|\,\mathbf{\Lambda}_0\right]. \tag{H.7}$$

We remark that we assume the system parameters to be known, although in practice the parameters would need first to be learned, e.g., exploration phase. However, the problem of efficiently learning the parameters is out of the scope for this paper.

8.3 Random sampling policy

We start by studying the random sampling policy, which will serve as a baseline scheme in the numerical evaluation. Contrary to the adaptive policies presented later (i.e., with memory), the memoryless nature of the random sampling policy allows us to derive a closed-form expression of the expected long-term AoI under that policy.

The random sampling policy selects in each slot a sensor drawn uniformly at random from the set of sensors $\{1, 2, \ldots, N\}$, i.e., each sensor is selected with probability $1/N$. However, while the sampling decisions in each slot are independent, the observations are correlated over time as they depend on the source states, which do not change independently from one slot to another. To see this, consider the camera scenario considered in the introduction. Clearly, if a source is in zones 1,3, or 4, it is likely that it will also be observable in the next time slot. Similarly, if on the other hand a source is in zone 2, it is likely to be invisible in the following time slot since zone 2 is

not covered by any camera. In particular, the observations evolve according to a Markov chain which is independent of the scheduling decisions. Using this observation, we can construct the Markov chain of the AoI process and derive the following result.

Theorem 3. *The random sampling policy that schedules in each slot a sensor drawn uniformly at random from the set of sensors achieves an expected long-term average AoI of*

$$E[\Delta_{random}] = \frac{1}{K}\sum_{k=1}^{K} E[\Delta_{random,k}], \tag{H.8}$$

where $E[\Delta_{random,k}]$ *is the expected AoI of source k given as*

$$E[\Delta_{random,k}] = \boldsymbol{\beta}_k \mathbf{R}_k^{(succ)} \left(\mathbf{I} - \mathbf{R}_k^{(fail)}\right)^{-2} \mathbf{1}^T, \tag{H.9}$$

$\mathbf{I}$ *and* $\mathbf{1}$ *are the identity matrix and the row vector of all ones of appropriate dimensions, respectively, and*

$$\mathbf{R}_k^{(succ)} = \operatorname{diag}(\mathbf{p}_k)\,\mathbf{R}_k, \tag{H.10}$$

$$\mathbf{R}_k^{(fail)} = (\mathbf{I} - \operatorname{diag}(\mathbf{p}_k))\,\mathbf{R}_k, \tag{H.11}$$

$$\mathbf{p}_k = \frac{1}{N}\sum_{n=1}^{N} q_n \begin{bmatrix} p_{nk}^{(1)} & p_{nk}^{(2)} & \cdots & p_{nk}^{(S_k)} \end{bmatrix}. \tag{H.12}$$

Proof. See Appendix A. □

It follows immediately from Theorem 3 that the expression for the average AoI can be greatly simplified when the sources have a single state, as presented in the following corollary.

Corollary 1. *When the sources have a single state, i.e.,* $S_k = 1$*, the random sampling policy achieves an expected long-term AoI of*

$$E[\Delta_{random}] = \frac{N}{K}\sum_{k=1}^{K}\left(\frac{1}{\sum_{n=1}^{N} q_n p_{nk}^{(1)}}\right). \tag{H.13}$$

Proof. When the sources have a single state the expressions reduce to scalar equations and

$$E[\Delta_{\text{random},k}] = \left(\frac{1}{N}\sum_{n=1}^{N} q_n p_{nk}^{(1)}\right) \times \left(1 - \left(1 - \frac{1}{N}\sum_{n=1}^{N} q_n p_{nk}^{(1)}\right)\right)^{-2} \tag{H.14}$$

$$= \left(\frac{1}{N}\sum_{n=1}^{N} q_n p_{nk}^{(1)}\right)^{-1}. \tag{H.15}$$

The result is obtained by averaging over the K sources and rearranging. □

8.4 Optimal policy with observable states

We now turn our attention to adaptive policies, i.e., policies that depend on the current system state, and we start by studying the case where the source states are observable to the scheduler. In this case, the system is fully observable and the problem can be analyzed as an average-cost MDP with unbounded costs (see e.g., [27–29]). We first prove that there exists a stationary policy that solves Eq. (H.7), and we then show how such a policy can be found using relative value iteration.

8.4.1 Structure of optimal policy

Policies for MDPs can generally be characterized as randomized/deterministic and history dependent/stationary (Markovian) [28]. Due to their simple structure, deterministic stationary policies are desired from both an analytical and practical perspective. A policy is said to be deterministic stationary if the same deterministic decision rule is used in each time slot, or, more formally, if $\mathbf{\Lambda}_{t_1} = \mathbf{\Lambda}_{t_2}$ implies that $a^*_{t_1} = a^*_{t_2}$, where a^*_t is the optimal action at time t. Contrary to discounted MDPs, for which an optimal policy is guaranteed to be stationary, optimal policies for unbounded average-cost MDPs are in general not guaranteed to be deterministic and stationary, and may be both history dependent and stochastic [27, 28]. One way to guarantee the existence of a deterministic stationary policy through a set of sufficient conditions provided in [29], which we will show are satisfied for the problem in Eq. (H.7). In the following paragraphs, we provide an overview of the conditions and the optimal policy, while we defer the formal proofs to Appendix B.

The main idea behind the conditions is to show that a deterministic stationary policy for the corresponding α-discounted problem exists in the limit as $\alpha \to 1$. To formalize, we define the value function as the total discounted cost under policy π for a given discount factor $0 < \alpha < 1$ and initial state $\mathbf{\Lambda}$ as

$$V_{\alpha,\pi}(\mathbf{\Lambda}) = \lim_{T\to\infty} \mathbb{E}\left[\sum_{t=0}^{T} \alpha^t C(\mathbf{\Lambda}_t, \alpha_t)\middle| \mathbf{\Lambda}_0 = \mathbf{\Lambda}\right], \tag{H.16}$$

and let

$$V_\alpha(\mathbf{\Lambda}) = \inf_\pi V_{\alpha,\pi}(\mathbf{\Lambda}). \tag{H.17}$$

Note that $V_\alpha(\mathbf{\Lambda}) \geq 0$ because all costs are positive. The first condition, presented in Appendix B as Lemma 1, is that there exists a deterministic stationary policy that minimizes $V_\alpha(\mathbf{\Lambda})$. Note that this is generally not guaranteed due to the countable state space. Provided that it exists, the policy is given as

$$a_t = \arg\max_a \left\{C(\mathbf{\Lambda}_t, a) + \alpha\, \mathbb{E}_{\mathbf{\Lambda}_{t+1}}\left[V_\alpha(\mathbf{\Lambda}_{t+1})|\mathbf{\Lambda}_t, a\right]\right\}, \tag{H.18}$$

and the value function satisfies the Bellman equation

$$V_\alpha(\mathbf{\Lambda}) = \min_a \left\{ C(\mathbf{\Lambda}, a) + \alpha \, \mathbb{E}_{\mathbf{\Lambda}'} \left[V_\alpha(\mathbf{\Lambda}') | \mathbf{\Lambda}, a \right] \right\}. \tag{H.19}$$

A consequence of this is that the optimal discounted policy can be found using the iterative value iteration procedure [28]

$$V_\alpha^{n+1}(\mathbf{\Lambda}) = \min_a \left\{ C(\mathbf{\Lambda}, a) + \alpha \, \mathbb{E}_{\mathbf{\Lambda}'} \left[V_\alpha^n(\mathbf{\Lambda}') | \mathbf{\Lambda}, a \right] \right\}, \tag{H.20}$$

where $n = 0, 1, \ldots$ is the iteration number and $V_\alpha^0(\mathbf{\Lambda})$ is an arbitrary initial state. Then, $V_\alpha^{n+1}(\mathbf{\Lambda}) \to V_\alpha(\mathbf{\Lambda})$ as $n \to \infty$ for any $\mathbf{\Lambda}$ and $0 < \alpha < 1$.

Although the discounted problem converges to the optimal value function, we are ultimately interested in the undiscounted value function, i.e., with $\alpha = 1$. However, the value iteration in Eq. (H.20) diverges to ∞ for nonzero cost functions. To avoid this problem, it is useful to instead consider the relative value function $h_\alpha(\mathbf{\Lambda}) = V_\alpha(\mathbf{\Lambda}) - V_\alpha(\mathbf{\Lambda}_0)$, where $\mathbf{\Lambda}_0$ is an arbitrary fixed state. Using the relative value function, we define the relative value iteration procedure as

$$h_\alpha^{n+1}(\mathbf{\Lambda}) = \min_a \left\{ C(\mathbf{\Lambda}, a) + \alpha \, \mathbb{E}_{\mathbf{\Lambda}'} \left[h_\alpha^n(\mathbf{\Lambda}') | \mathbf{\Lambda}, a \right] \right\} - h_\alpha^n(\mathbf{\Lambda}_0), \tag{H.21}$$

which can be interpreted as the total cost of state $\mathbf{\Lambda}$ relative to the cost of state $\mathbf{\Lambda}_0$. Contrary to the value iteration in Eq. (H.20), the relative value iteration procedure does not increase unboundedly for nonzero costs as $\alpha \to 1$ (provided that an optimal deterministic stationary policy exists), and it does not impact the sequence of maximizing actions [28].

The remaining two conditions required for an optimal deterministic stationary policy to exist relate to the boundedness of $h_\alpha(\mathbf{\Lambda})$. The second condition, presented in Lemma 3 of Appendix B, states that $h_\alpha(\mathbf{\Lambda})$ must be uniformly bounded from below in both α and $\mathbf{\Lambda}$. The third condition concerns the upper bound of $h_\alpha(\mathbf{\Lambda})$. However, because the state space is countable and the cost function $C(\mathbf{\Lambda}, a)$ is unbounded from above, $h_\alpha(\mathbf{\Lambda})$ is not uniformly upper bounded in $\mathbf{\Lambda}$. Instead, the third condition requires that $h_\alpha(\mathbf{\Lambda})$ is uniformly bounded from above only in α, i.e., $h_\alpha(\mathbf{\Lambda}) \leq M_{\mathbf{\Lambda}}$ for all $\mathbf{\Lambda}$ and α, *and* that $\mathbb{E}_{\mathbf{\Lambda}_{t+1}}[M_{\mathbf{\Lambda}_{t+1}} | \mathbf{\Lambda}_t, a_t] < \infty$ for all $\mathbf{\Lambda}_t$ and a_t. This condition is stated in Lemma 4 in Appendix B.

The fact that the conditions above are satisfied guarantees the existence of a deterministic stationary optimal policy. The result is summarized in the following theorem.

Theorem 4. *There exists a function $h(\mathbf{\Lambda})$ such that the policy*

$$a_t = \arg\min_a \left\{ C(\mathbf{\Lambda}_t, a) + \mathbb{E}_{\mathbf{\Lambda}_{t+1}} \left[h(\mathbf{\Lambda}_{t+1}) | \mathbf{\Lambda}_t, a \right] \right\} \tag{H.22}$$

is optimal. Furthermore, $h(\mathbf{\Lambda})$ *satisfies*

$$g + h(\mathbf{\Lambda}) = \min_{a} \left\{ C(\mathbf{\Lambda}, a) + \mathbb{E}_{\mathbf{\Lambda}'} \left[h(\mathbf{\Lambda}') | \mathbf{\Lambda}, a \right] \right\}, \tag{H.23}$$

for some constant g.

Proof. See Appendix B. □

8.4.2 Relative value-iteration

As mentioned previously, the regular value iteration in Eq. (H.20) diverges for $\alpha = 1$, and so $h(\mathbf{\Lambda})$ is often computed directly using relative value iteration in Eq. (H.21). However, relative value iteration is not directly applicable to our problem due to the infinite state space. Instead, we resort to an approximate solution and truncate the state space. This may be justified by the fact that some states are very unlikely to be reached, or because the AoI does not impose a higher penalty once it is above a certain threshold. More specifically, we consider the Q-truncated AoI obtained by modifying the dynamics in (H.2) as

$$\Delta_k^Q(t+1) = \begin{cases} 1 & \text{if } \sum_n \zeta_{nk}(t) u_n(t) = 1 \\ \min(Q, \Delta_k(t) + 1) & \text{otherwise.} \end{cases} \tag{H.24}$$

In particular, because the unbounded system is stable and the cost function is nondecreasing in the AoI of each source, Λ_k, (see Lemma 2 in Appendix B), we can achieve an arbitrary good approximation using a sufficiently large Q. However, a large value of Q comes at the cost of an increased computational complexity of relative value iteration. The value function for this problem, $h(\mathbf{\Lambda})$, can be found using relative value iteration [28], provided that the MDP is unichain, i.e., that every deterministic stationary policy has a single recurrent class plus a possibly empty class set of transient states. This is indeed the case for this problem, as can be seen by considering the state $\mathbf{\Lambda} = (\boldsymbol{S}_0, (Q, Q, \ldots, Q))$, which can be reached from all states with nonzero probability for any $\boldsymbol{S}_0$. The relative value iteration is summarized in Algorithm 4.

It can be shown [30] that $h^n(\mathbf{\Lambda})$ satisfies $\underline{\delta}^n \leq \underline{\delta}^{n+1} \leq g \leq \bar{\delta}^{n+1} \leq \bar{\delta}^n$, where g is the constant in Eq. (H.23), and

$$\underline{\delta}^n = \min_{\mathbf{\Lambda}} \{ h^{n+1}(\mathbf{\Lambda}) - h^n(\mathbf{\Lambda}) \}, \tag{H.25}$$

$$\bar{\delta}^n = \max_{\mathbf{\Lambda}} \{ h^{n+1}(\mathbf{\Lambda}) - h^n(\mathbf{\Lambda}) \}. \tag{H.26}$$

Using this, a ϵ-optimal approximation of the value function for some $\epsilon > 0$ can be obtained by using the convergence criterion $\bar{\delta}^n - \underline{\delta}^n < \epsilon$, which is guaranteed to be satisfied within a finite number of iterations.

Algorithm 4 Relative value iteration.

Pick $\mathbf{\Lambda}_0$ arbitrarily
$h^0(\mathbf{\Lambda}) \leftarrow 0, \quad \forall \mathbf{\Lambda}$
For $n = 0, 1, \ldots$ until convergence **do**
  $\lambda^{n+1} \leftarrow \min_a \left\{ C(\mathbf{\Lambda}_0, a) + \mathbb{E}_{\mathbf{\Lambda}'} \left[h^n(\mathbf{\Lambda}') | \mathbf{\Lambda}_0, a \right] \right\}$
  $h^{n+1}(\mathbf{\Lambda}) \leftarrow \min_a \left\{ C(\mathbf{\Lambda}, a) + \mathbb{E}_{\mathbf{\Lambda}'} \left[h^n(\mathbf{\Lambda}') | \mathbf{\Lambda}, a \right] \right\} - \lambda^{n+1}, \quad \forall \mathbf{\Lambda}$
End for

8.4.3 Approximate policies

Each epoch in the value iteration algorithm requires a pass over all states, actions, and possible future states. With a state space of size $S_1 \times S_2 \times \ldots \times S_K \times Q^K$, this is infeasible even for small scenarios. The problem of large dimensionality has been addressed in several ways in the literature, most notably by computing the value function for only a subset of the states, and then use those values to fit an approximate function using e.g., a decision tree or a neural network, or by using reinforcement learning methods to approximate the optimal policy in some high dimensional feature space [30, 31]. While such methods often perform well in practice, they are typically very problem dependent and their convergence properties are not well understood. Another strategy that has frequently been applied to AoI problems is to frame the problem as a restless multi-armed bandit problem (e.g., in [32]), for which good heuristic methods exist, such as the Whittle index policy [33]. To frame the problem as a restless multi-armed, the problem must be separable into independent sub-problems, such as the scheduling of individual sources or sensors. Since the sensors in the system that we consider can observe multiple sources and each source can be observed by multiple sensors, such a separation is not possible.

Instead, as we are mainly concerned with how the common observation model impacts the AoI, we resort to the myopic policy when the state space is too large to allow the value function to be computed. The myopic policy schedules the sensors with the lowest immediate cost, i.e.,

$$a_t = \arg\min_a \left\{ C(\mathbf{\Lambda}_t, a) \right\}. \tag{H.27}$$

The myopic policy is generally sub-optimal, and as we illustrate in Section 8.6 the memory in the AoI process makes the myopic policy sub-optimal even for very simple scenarios, such as one in which the sources only have a single state, i.e., $S_k = 1$ for all k. Nevertheless, as presented in the following proposition, it turns out that the myopic policy is optimal when all of the following two conditions are satisfied: (1) $S_k = 1$ for all k, (2) each sensor observes exactly one source with nonzero probability p, i.e., for all sensors $n = 1, 2, \ldots, N$, $p_{nk}^{(1)} = p$ for some $k = k'_n$ and $p_{nk}^{(1)} = 0$ for $k \neq k'_n$.

Proposition 4. *The myopic policy*

$$a_t = \arg\min_a \{C(\mathbf{\Lambda}_t, a)\} \tag{H.28}$$

is optimal if the sources have a single state, i.e., $S_k = 1$ for all k, each sensor observes exactly one source with probability p and all other sources with probability 0, and the channel error probabilities are equal, i.e., $q_n = q$ for all n.

Proof. This is a special case of [34, Theorem 5] with weight $a = 1$. □

8.5 Policies for unobservable states

In this section, we consider the problem where part of the system states are unobservable and need to be inferred through the measurements. We consider two extreme instances of the problem. First, we consider the case where the sources, but not their states, can be detected from a measurement, as may for example the case in a camera scenario with image recognition. We also discuss how to handle the situation in which the source state is revealed by the measurements, as in the camera scenario when the states represent the physical location of the sources. In the second instance, we assume that the sources cannot be detected based on the sensor measurement, which is for instance the case for a camera without image recognition, or touch sensors, where several sources can press the sensor, but the sensor reading does not reveal the specific source. In contrast to the first instance where only the source states are unobservable, in this case the source AoIs are also unobservable. In both cases, we assume that the states and transitions probabilities are known. Sometimes, this information may not be available and will need to be learned using e.g., reinforcement learning methods. However, such approaches are beyond the scope of this paper.

The unobservable states leads to a partially observable Markov decision process (POMDP), in which the scheduler needs to keep track of the information obtained through the measurements. This information can be represented as a vector that contains all the previous observations and actions, and the time instances at which they were taken/observed. By treating the information vector as part of the state, the POMDP can be treated as a fully observable MDP. However, the dimension of the information vector increases in every time step, which makes the problem difficult to solve. It turns out that a probability distribution over the states, which we refer to as the belief states, $b(\mathbf{\Lambda})$, is a sufficient statistic [35]. By including the belief state in the system state and redefining the expected cost as

$$\bar{C}(\mathbf{\Lambda}, a) = \sum_{\mathbf{\Lambda}} b(\mathbf{\Lambda}) C(\mathbf{\Lambda}, a), \tag{H.29}$$

we can obtain an MDP that can be solved using relative value iteration as outlined in Section 8.4 (provided that a stationary policy exists). Alternatively, the cost function can be incorporated directly into the myopic policy in (H.27) if the size of the state space renders value iteration intractable. In the remainder of this section, we first derive the belief state of the two considered instances, and then conclude the section by discussing approximate solutions to the POMDP, which are usually necessary due to the state space expansion caused by the inclusion of the belief vector.

8.5.1 Detectable sources

For the case when the sources are detectable from the measurements we define the observation as $\boldsymbol{O}_t = (q, \zeta_1, \ldots, \zeta_K)$, where $q = 1$ indicates successful transmission from the source to the destination, otherwise $q = 0$, and $\zeta_k = 1$ if source k was observed otherwise zero. Because the AoIs are known, it suffices to maintain a belief of the source states $\boldsymbol{S}$. From the conditional independence of the source states, we may factor $b_t(\boldsymbol{S})$ into beliefs for each source as

$$b(\boldsymbol{S}) = \prod_{k=1}^{K} b^k(s_k). \tag{H.30}$$

Due to the Markovian structure of the problem, the belief state can be updated in each time step using

$$\begin{aligned} b_t^k(s_k') &= \Pr(s_k' | \boldsymbol{O}, a, b_{t-1}^k) \\ &= \sum_{s_k=1}^{S_k} \Pr(s_k' | s_k) \Pr(s_k | \boldsymbol{O}, a, b_{t-1}^k) \\ &\propto \sum_{s_k=1}^{S_k} \Pr(s_k' | s_k) \Pr(\boldsymbol{O} | s_k, a) b_{t-1}^k(s_k), \end{aligned} \tag{H.31}$$

where $\Pr(s_k' = j | s_k = i) = r_{ij}^{(k)}$ and the observation probabilities can be computed as

$$\Pr(\boldsymbol{O} | s_k, a) = \begin{cases} q_a \left[\zeta_{ak} p_{ak}^{(s_k)} + (1 - \zeta_{ak})(1 - p_{ak}^{(s_k)}) \right] & \text{if } q = 1, \\ 1 - q_a & \text{otherwise.} \end{cases} \tag{H.32}$$

Note that in some applications it may be natural to assume that a source observation reveals the true state of a source. This scenario can be captured by using the alternative update rule $b_t^k(s_k') = \Pr(s_k' | s_k)$ when source k is observed (and the one in Eq. (H.31) when source k is not observed), where s_k is the true state of source k when the measurement was made.

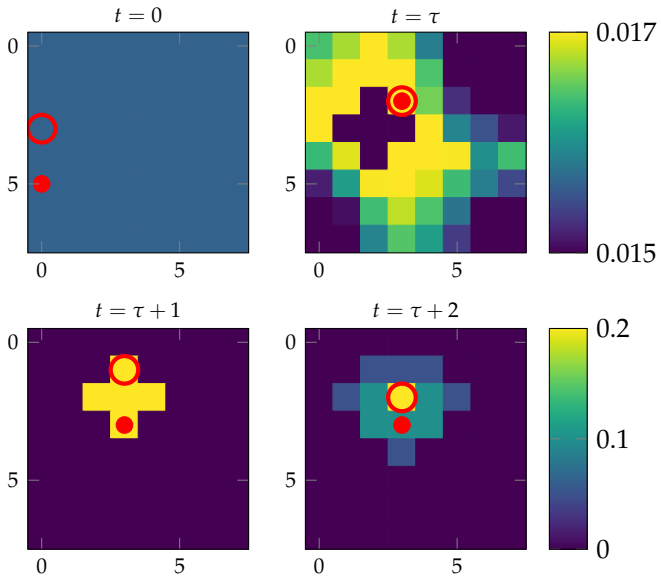


Fig. H.4: Example of how the belief evolves for a single source that moves around in a grid as various sensors (squares) are scheduled. Each sensor observes a single square with probability one. The scheduled sensors is indicated by the empty circle and the source state (its location) is indicated by the solid circle.

An example of how the belief evolves for a single source that moves around in a grid is shown in Fig. H.4. The state of the source (its location) is indicated by the solid red circle, and in each time step one of the sensors, which can each observe a single state with probability one, is scheduled. The scheduled sensor is indicated by the empty circle. Initially, at $t = 0$, the belief is uniform over all source states. As time evolves, the belief changes as indicated at $t = \tau$, where the scheduled sensor observes the source. Then, at $t = \tau + 1$ the belief is updated to reflect the knowledge obtained at time $t = \tau$, and again at $t = \tau + 2$.

8.5.2 Undetectable sources

In some cases, the scheduler may not be able to observe the sources from the sensor readings, e.g., if they are complex to extract or if the sensor data is encrypted. In this case, when sensor measurements reveal nothing about the sources to the scheduler, the source state belief remains constant, but the scheduler needs instead to keep track of its belief over the AoIs. Since the source states are unobservable, we assume they are in stationarity, and denote by $\beta_k^{(s_k)}$ the steady-state probability that source k is in state s_k. In this regime we can assume that each source has a single state with observation

probabilities

$$\hat{p}_{nk} = \sum_{s=1}^{S_k} \beta_k^{(s)} p_{nk}^{(s)}. \tag{H.33}$$

The AoI belief can then be constructed as follows.

$$b(\boldsymbol{\Delta}) = \prod_{k=1}^{K} b^k(\Delta_k), \tag{H.34}$$

with update rule

$$\begin{aligned} b_t^k(\Delta_k') &= \Pr(\Delta_k' | \boldsymbol{O}, a, b_{t-1}^k) \\ &= \begin{cases} b_{t-1}^k(\Delta_k' - 1)(1 - q\hat{p}_{ak}) & \text{for } \Delta_k' > 1, \\ q\hat{p}_{ak} & \text{for } \Delta_k' = 1. \end{cases} \end{aligned} \tag{H.35}$$

In addition to the two cases considered here, there is a third case in which objects are detectable, but they are all identical so only the number of observed sources is revealed. While it is straightforward to derive the belief update rule for this case we omit it for brevity.

8.5.3 Approximate POMDP policies

Because $b(\boldsymbol{\Lambda})$ is constructed from a countable MDP, the state space of the POMDP is also countable. Unfortunately, the conditions used in Section 8.4 to prove the existence of a optimal stationary policy are not straightforward to apply here since there is no natural ordering of the belief states. Furthermore, alternative sufficient conditions such as the ones given in [36, 37] are difficult to prove for the general source state model that we consider. Finally, relative value iteration is challenging due to the state augmentation caused by the belief vector. As a result, one often has to resort to sub-optimal policies based on the value function obtained in Section 8.4 for the observable state space using the two methods outlined below. Note that both methods can be applied in a myopic fashion as well by discarding the expectation term $\mathbb{E}_{\boldsymbol{\Lambda}_{t+1}}[\cdot]$.

Maximum Likelihood (ML) Policy

One way to apply the value functions for the fully observable MDP to the POMDP is to extract the maximum likelihood (ML) state from the belief states, and act as if it was the true state [38]. Thus, the partial observability is hidden from the agent, which simply picks the action

$$a_t = \arg\min_a \left\{ C(\boldsymbol{\Lambda}_t^{\mathrm{ML}}, a) + \mathbb{E}_{\boldsymbol{\Lambda}_{t+1}} \left[h(\boldsymbol{\Lambda}_{t+1}) | \boldsymbol{\Lambda}_t^{\mathrm{ML}}, a \right] \right\}, \tag{H.36}$$

where $\mathbf{\Lambda}_t^{\mathrm{ML}} = \arg\max_{\mathbf{\Lambda}} b_t(\mathbf{\Lambda})$. The policy has two major drawbacks. The first is that it will not take actions to gain information about the states, which in some scenarios can lead to situations in which the belief state will never change, and thus the agent may end up in some sub-optimal state [39]. The second drawback is that the policy discards a large part of the information contained in the belief state, and acts as if there was no uncertainty. As a result, if all the states are almost equally likely, but the action that minimizes the value for the most likely state results in a significant increase in the costs in the other states, following the maximum likelihood policy may result in a very high cost. This situation may arise in the scenario from Fig. H.2 if the most likely state is that all AGVs are in zone Z3, which is not covered by any of the cameras. In this state the costs incurred by the available actions are indifferent, and the agent may decide to schedule zone Z1 even though the AGVs are probably more likely to be in the zones next to Z3, i.e., Z4 or Z2.

Q-MDP Policy

To improve the the second drawback of the maximum likelihood policy, the Q-MDP policy picks the action that minimizes the expected cost [39]. More specifically, the agent selects the action according to the rule

$$a_t = \arg\min_a \left\{ \sum_{\mathbf{\Lambda}_t} b_t(\mathbf{\Lambda}_t) \left(C(\mathbf{\Lambda}_t, a) + \mathbb{E}_{\mathbf{\Lambda}_{t+1}} \left[h(\mathbf{\Lambda}_{t+1}) | \mathbf{\Lambda}_t, a \right] \right) \right\}. \tag{H.37}$$

Although the Q-MDP policy addresses the second drawback of the ML policy, it still does not favor information gaining actions. The advantage of Q-MDP comes at the cost of being computationally more demanding than the ML policy, but it can be efficiently approximated by Monte Carlo estimation by drawing state samples from the belief distribution and averaging their costs.

8.6 Numerical results

In this section, we apply the methods presented in the previous sections to a number of scenarios. We first study the optimal and myopic policies for a number of small toy scenarios, which highlight both the impact of the source dynamics and the possibility for sensors to observe multiple sources. We then consider more complex scenarios and investigate the importance of the amount of state information at the scheduler by comparing the AoI of the different policies. The simulation results in the section represent averages over 10 runs of 100 000 time slots, where the initial 10 000 time slots from each run have been discarded to minimize the impact of the initial state. In all scenarios with relative value iteration, we have truncated the AoI at a value,

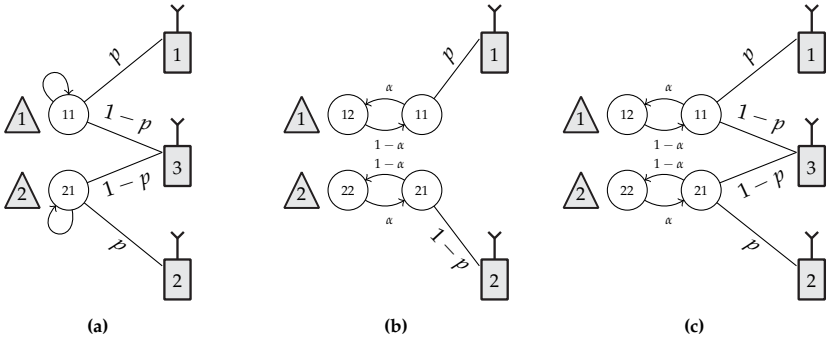


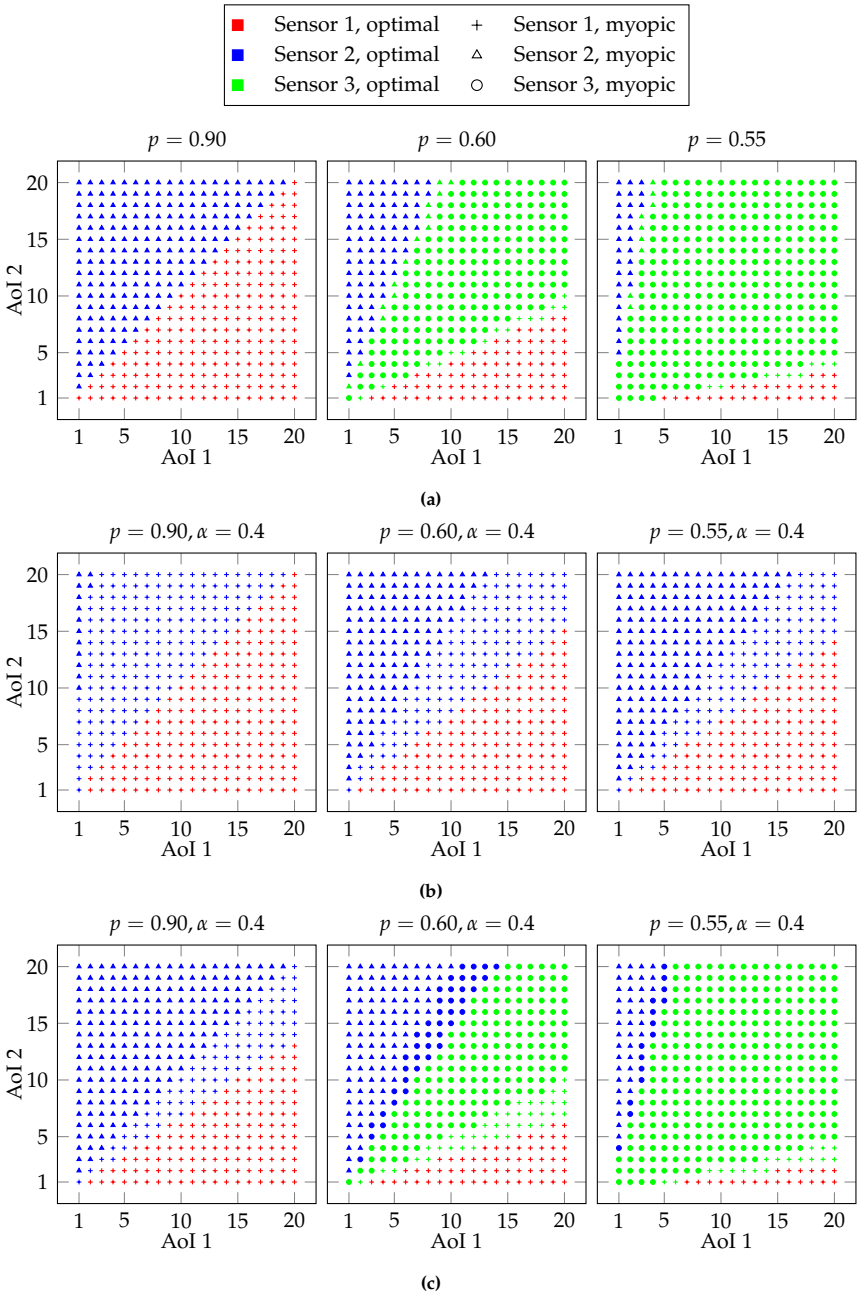
Fig. H.5: Toy scenarios considered in the evaluation.

Q, as high as possible, while keeping the computational time reasonable and ensuring that the probability of having an AoI greater than Q is negligible. Thus, the results are likely to be very close to the ones that would have been achieved with unbounded AoI.

8.6.1 Toy scenarios

We consider the three toy scenarios illustrated in Fig. H.5, and assume that the source states are fully observable. The first scenario, shown in Fig. H.5a, contains two stateless sources, which are observed by three sensors. Sensors 1 and 2 observe only sources 1 and 2, respectively, with probability p, while sensor 3 observes both source 1 and 2 with probability $1 - p$. Thus, the scenario reveals when correlated observations can be beneficial: Intuitively, one would expect that it is beneficial to primarily schedule sensor 3 when p is small, and to schedule sensors 1 and 2 when p is large. The scenario in Fig. H.5b is intended to show the impact of the source states. Each of the two sources has two states, and the sensors can observe the sources only in one of the states. Furthermore, to show how the source observation probabilities and the source state probabilities impact the scheduling decisions, the source transition probabilities are reversed for the two sources, and sensor 1 observes source 1 with probability p, while source 2 is observed by sensor 2 with probability $1 - p$. Finally, the third scenario in Fig. H.5c combines the two first scenarios by including both source states and a sensor that can observe both sources.

For simplicity, we assume a perfect channel between the sensor and the destination, i.e., $q_1 = q_2 = 1$. We limit ourselves to consider the case when source 1 is in state 11 and source 2 is in state 21, which represents the only non-trivial scheduling situation in all of the three scenarios. The value functions used by the optimal policy are obtained with a truncation of the AoI at $Q = 100$. The corresponding optimal policies are shown in Fig. H.6, where



(a)

(b)

(c)

Fig. H.6: Scheduling policies for the toy scenarios in Fig. H.5. The optimal actions obtained using relative value iteration are indicated by the color, while the shapes of the points indicate the myopic actions.

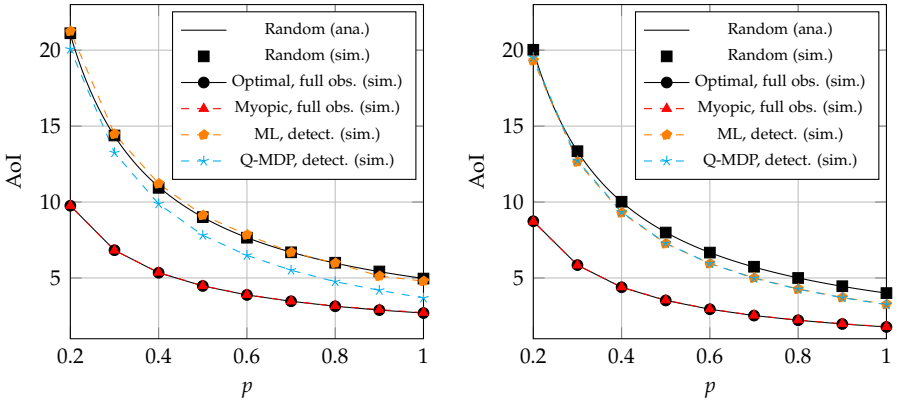
the colors of the points indicate the optimal policy obtained by relative value iteration and the point shapes show the myopic action. The policy for the scenario in Fig. H.5a is shown in Fig. H.6a. It can be seen that the optimal policy is to schedule sensors 1 and 2 when p is high or when the AoI of one source is much higher than the AoI of the other. On the other hand, sensor 3 is advantageous when p is low or the AoI difference is small. Note that the actions by the myopic policy are very close to the optimal ones only with differences close to the decision boundaries. While this subtle difference is unlikely to have a significant impact on the performance, it is due to the memory introduced by the AoI process, causing the scheduling problem is non-trivial. As established by Proposition 4, the difference between the myopic and optimal policies vanishes as p approaches 1, which is also indicated by the figures already for $p = 0.9$ where the myopic and optimal policies coincide in considered range of AoIs.

The difference between the optimal and the myopic policies become more pronounced when the sources have states that impact their observability. In the scenario in Fig. H.5b, source 1 can be observed by a sensor only in a fraction $1 - \alpha$ of the time, while source 2 is in a fraction α of the time. As a result, for $\alpha = 0.4$ shown in Fig. H.6b, the optimal policy is much more likely to schedule sensor 2 than the myopic policy is. The same can be seen in the last scenario in Fig. H.5c, where the optimal policy also favors sensor 2, while the myopic policy is the same as in the first scenario.

8.6.2 Small factory scenario

We now return to the scenario from the introduction where AGVs (sources) move around in four zones (represented by their states), out of which three are covered by cameras (sensors). We study the average AoI achieved by the derived policies and investigate the AoI penalty that is achieved by partial observability (with detectable sources). We assume again that there are three AGVs, and denote by α the probability that they move from one zone to an adjacent one, so that they remain in the same zone with probability $1 - 2\alpha$ (they cannot move diagonally). The probability that a camera fails to observe an AGV that is within its zone, e.g., if the AGV is hidden behind an object, is denoted by $1 - p$, and we assume again no transmission errors, i.e., $q_n = 1$ for all n. The value functions used by all except the random and myopic policies are obtained from relative value iteration truncated at an AoI of $Q = 20$.

The average AoI of five different policies are shown in Fig. H.7a and Fig. H.7b for $\alpha = 0.1$ and $\alpha = 0.4$, respectively. In both cases, the myopic policy performs close to optimally despite the fact that the sources are stateful, indicating that the additional cost of acting myopically is negligible in terms of average AoI. The AoI in the case of $\alpha = 0.1$ is generally higher than for $\alpha = 0.4$, which is because the sources stay longer in the invisible zone.



(a) $\alpha = 0.1$ **(b)** $\alpha = 0.4$

Fig. H.7: Average AoI achieved by the different policies in the small factory scenario.

Regarding the policies with partial observability, it can be seen in Fig. H.7a that the ML policy is only little if at all better than the randomized policy, while the Q-MDP policy is significantly better. This is because for low values of α where the sources are most likely to stay in the same zone between time steps, the ML policy tends to conclude that a given source source is in the hidden zone. Since no sensor can observe this zone, it can do nothing but act like the random policy. On the other hand, the Q-MDP policy is able to make use of the entire belief distribution, and not only the most likely belief, which allows it to take more informed scheduling decisions using the less likely states. However, as α increases, the AoI of the ML policy gets closer to that of the Q-MDP policy (Fig. H.7b), which is because the likelihood of staying in the same zone for a long time decreases, and thus the maximum likelihood belief is more likely to be outside the hidden zone.

8.6.3 Large factory scenario

Finally, we consider a large factory scenario comprising an 8×8 grid of 64 cells, and 10 AGVs. Each of the cells is equipped with a camera that observes the AGVs inside the zone with probability one. However, in addition to the 64 cameras that cover the individual cells, there are sensors that cover larger areas but with less reliable observations. Specifically, in addition to the 64 sensors that cover each cell, there are 16 sensors that cover each of the 2×2 zones, 4 sensors that cover each of the 4×4 zones, and one sensor that covers the entire area. We refer to these groups of sensors as level 1–4 sensors, where level 1 sensors are the sensors that cover a single cell, and the level 4 sensor is the one that covers the entire area. The observation probabilities for the

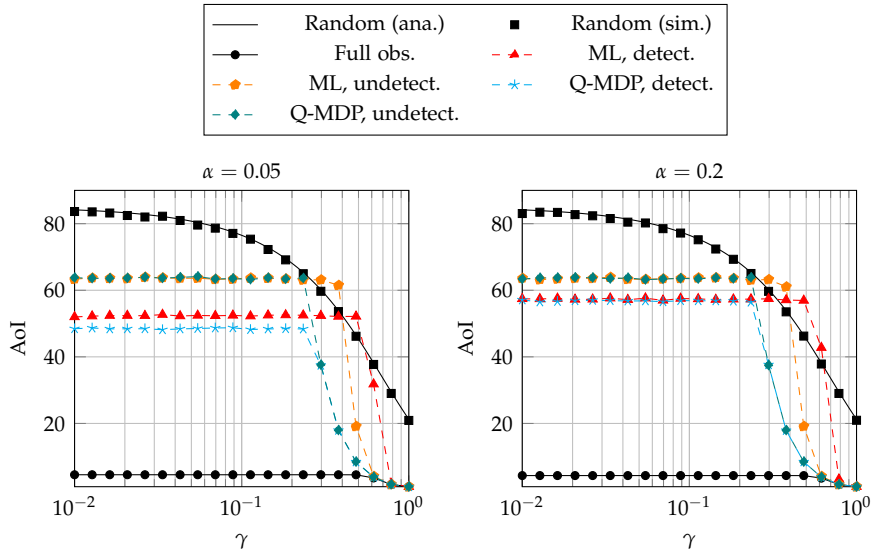


Fig. H.8: Average AoI for the large industrial scenario obtained using myopic policies for $\alpha = 0.05$ and $\alpha = 0.2$.

sensors at a certain level, $l = 1, \ldots, 4$, are defined as

$$p_l = \gamma^{l-1}, \tag{H.38}$$

where γ is a degradation factor that controls how much the observation probability decreases per level. For instance, if $\gamma = 1$, the sensors at each level observe each of the cells within their zones with probability 1, whereas if $\gamma = 1/4$ the observation probability at each level is the reciprocal of the number of cells that the sensors cover. Thus, when γ is high the scheduler is more likely to schedule the higher level sensors than when γ is low. As before α denotes the probability of transitioning into an adjacent cell. Due to the size of the problem, we consider only myopic policies.

Figure H.8 shows the average AoI obtained using various policies for $\alpha = 0.05$ and $\alpha = 0.2$. In general, the average AoI when $\alpha = 0.05$ is lower than for $\alpha = 0.2$, because the AGVs move slower, and thus are more predictable. In both cases, the average AoI is constant for small values of γ (except for the random policy), which reflects that the policy is the same. Specifically, all policies sample only sensors at level one. However, when γ approaches $\gamma = 1/4$, it becomes beneficial to schedule higher level sensors, either when multiple AGVs are likely to be located in the same area, or when the uncertainty is high and scheduling a level one sensor is unlikely to result in an observation. Note that this effect is first reflected for the Q-MDP policies. This is because they, contrary to the ML policies, include the uncertainty in

their decisions. In fact, when $\gamma \geq 1/4$ Q-MDP agents only schedule the sensor at level 4, while the ML agents require a larger value of γ before they start scheduling higher level sensors.

Somewhat surprisingly, the ML policy with undetectable sources leads to a faster AoI decrease compared to the ML policy with detectable sources, and performs better within a range of γ. The reason is that in the detectable case where the AoIs are known to the agent, the policy is likely to be dictated by a few AGVs with high AoI, which are scheduled at the grid level. On the other hand, in the undetectable case where the AoIs are unobservable, the AoI beliefs of the different AGVs are much more likely to be concentrated around the same values, which makes the agent more likely to schedule the sensors at the higher levels.

8.7 Conclusion

Timeliness of information is a relevant performance indicator for many IoT applications in which the AoI serves as an attractive alternative to the traditional end-to-end latency by considering the information from the destination perspective as opposed to the transmitter perspective. In this paper, we have considered the problem of scheduling a set of sensors that observe a set of dynamic sources. Furthermore, we have assumed that each source is observed by multiple sensors with probabilities that depend on the current state of the source. This scenario represents a case that arises naturally in many IoT monitoring applications, such as camera monitoring. We have formulated the scheduling problem as an MDP, and used it to derive optimal and approximate, but more practical, scheduling policies for both the fully observable and the partially observable cases. Through numerical results, we have shown that due to the dynamic sources, partial observability comes at a high cost in terms of AoI compared to the fully observable case. Furthermore, we have shown that the fact that sensors can observe multiple sources can be beneficial especially in the partial observable case where there is uncertainty about the source states. However, this requires that the policy is able to take this into account, which is the case for the approximate Q-MDP policy, but not for the ML policy. On the other hand, the ML policy has a lower complexity, which may be advantageous in some situations.

A Proof of Theorem 3

We first note that the sources are independent, and thus the average AoI of each source can be computed independently, and the total average AoI can be obtained as the arithmetic mean of the average AoIs of the individual

sources.

The AoI of a single source evolves according to a two-dimensional Markov chain in which one dimension represents its AoI and the other represents its internal state. The probability that the source is observed in a given time slot depends on its current state and the sensor that was scheduled. Since exactly one of the N sensors is scheduled, selected with probability $1/N$, the conditional probability that source k is observed given its state can be represented by the vector

$$\mathbf{p}_k = \frac{1}{N} \sum_{n=1}^{N} q_n \begin{bmatrix} p_{nk}^{(1)} & p_{nk}^{(2)} & \cdots & p_{nk}^{(S_k)} \end{bmatrix}, \tag{H.39}$$

where each entry represents one of the S_k source states. Using this, the *conditional* transition matrix given that the source is observed can be written $\mathbf{R}_k^{(\text{succ})} = \text{diag}(\mathbf{p}_k)\mathbf{R}_k$. Similarly, the conditional transition matrix given that the source is *not* observed is $\mathbf{R}_k^{(\text{fail})} = (\mathbf{I} - \text{diag}(\mathbf{p}_k))\,\mathbf{R}_k$. Using this, the truncated joint transition matrix of the source AoI and its state with a maximal AoI of Q, $\widehat{\mathbf{\Psi}}_k \in \mathbb{R}^{S_k Q \times S_k Q}$, is given as

$$\widehat{\mathbf{\Psi}}_k = \begin{bmatrix} \mathbf{R}_k^{(\text{succ})} & \mathbf{R}_k^{(\text{fail})} & 0 & \cdots & 0 \\ \mathbf{R}_k^{(\text{succ})} & 0 & \mathbf{R}_k^{(\text{fail})} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_k^{(\text{succ})} & 0 & 0 & \cdots & \mathbf{R}_k^{(\text{fail})} \\ \mathbf{R}_k^{(\text{succ})} & 0 & 0 & \cdots & \mathbf{R}_k^{(\text{fail})} \end{bmatrix}. \tag{H.40}$$

It is easy to verify that because $\mathbf{R}_k$ defines an irreducible and aperiodic Markov chain, the Markov chain defined by $\widehat{\mathbf{\Psi}}_k$ is irreducible, aperiodic and positive recurrent and thus has a unique stationary distribution given by the vector $\boldsymbol{\phi}_k$ that satisfies

$$\boldsymbol{\phi}_k \widehat{\mathbf{\Psi}}_k = \boldsymbol{\phi}_k. \tag{H.41}$$

Due to the structure of $\widehat{\mathbf{\Psi}}_k$, $\boldsymbol{\phi}_k$ is a (block) vector

$$\boldsymbol{\phi}_k = \begin{bmatrix} \boldsymbol{\phi}_k^{(1)} & \boldsymbol{\phi}_k^{(1)} & \cdots & \boldsymbol{\phi}_k^{(Q)} \end{bmatrix}, \tag{H.42}$$

where $\boldsymbol{\phi}_k^{(q)} = \begin{bmatrix} \Pr(q, s_k = 1) & \cdots & \Pr(q, s_k = K) \end{bmatrix}$ is the S_k-element vector where the j-th entry is the joint probability that source k is in state j and its AoI is q.

By writing out Eq. (H.41) it is straightforward to show that the blocks of $\boldsymbol{\phi}_k$ can be written

$$\boldsymbol{\phi}_k^{(q)} = \boldsymbol{\beta}_k \mathbf{R}_k^{(\text{succ})} \left(\mathbf{R}_k^{(\text{fail})}\right)^{q-1}, \tag{H.43}$$

and the average Q-truncated AoI of source k is thus

$$\bar{\Delta}_k^{(Q)} = \sum_{q=1}^{Q} q \boldsymbol{\phi}_k^{(q)} \mathbf{1}^T \tag{H.44}$$

$$= \boldsymbol{\beta}_k \mathbf{R}_k^{(\text{succ})} \sum_{q=1}^{Q} q \left(\mathbf{R}_k^{(\text{fail})}\right)^{q-1} \mathbf{1}^T \tag{H.45}$$

where $\mathbf{1}$ is the row vector of all ones of appropriate dimension. Provided that the source can be observed by a sensor, i.e., $\sum_{n=1}^{N}\sum_{s=1}^{S_k} q_n p_{nk}^{(s)} > 0$, it follows from the Perron-Frobenius theorem [40] that the eigenvalues of $\mathbf{R}_k^{(\text{fail})}$ are non-negative real and strictly less than one. Thus, the limit $\lim_{Q\to\infty}\sum_{q=1}^{Q} q \left(\mathbf{R}_k^{(\text{fail})}\right)^{q-1}$ exists and the untruncated AoI is

$$E[\Delta_{\text{random},k}] = \lim_{q\to\infty} \bar{\Delta}_k^{(Q)} \tag{H.46}$$

$$= \boldsymbol{\beta}_k \mathbf{R}_k^{(\text{succ})} \sum_{q=1}^{\infty} q \left(\mathbf{R}_k^{(\text{fail})}\right)^{q-1} \mathbf{1}^T \tag{H.47}$$

$$= \boldsymbol{\beta}_k \mathbf{R}_k^{(\text{succ})} \left(\sum_{q=1}^{\infty} \left(\mathbf{R}_k^{(\text{fail})}\right)^{q-1}\right)^2 \mathbf{1}^T \tag{H.48}$$

$$= \boldsymbol{\beta}_k \mathbf{R}_k^{(\text{succ})} \left(\mathbf{I} - \mathbf{R}_k^{(\text{fail})}\right)^{-2} \mathbf{1}^T. \tag{H.49}$$

Averaging over the K sources completes the proof.

B Proof of Theorem 4

We prove Theorem 4 by showing that the sufficient conditions in [29] are fulfilled, by following a similar approach as in [32]. We first state the conditions in a number of lemmas and then prove Theorem 4.

Lemma 1. *There exists a deterministic stationary policy that minimizes the discounted cost $V_\alpha(\boldsymbol{\Lambda})$ for any $0 < \alpha < 1$. Furthermore, $V_\alpha(\boldsymbol{\Lambda})$ is finite for every α and $\boldsymbol{\Lambda}$, and satisfies the Bellman equation*

$$V_\alpha(\boldsymbol{\Lambda}) = \min_a \left\{ C(\boldsymbol{\Lambda}, a) + \alpha \sum_{\boldsymbol{\Lambda}_{t+1}} \Pr(\boldsymbol{\Lambda}_{t+1} | \boldsymbol{\Lambda}_t = \boldsymbol{\Lambda}, a_t = a) V_\alpha(\boldsymbol{\Lambda}_{t+1}) \right\}$$
$$= \min_a \left\{ C(\boldsymbol{\Lambda}, a) + \alpha \, \mathbb{E}_{\boldsymbol{\Lambda}'} \left[V_\alpha(\boldsymbol{\Lambda}') | \boldsymbol{\Lambda}, a \right] \right\}. \tag{H.50}$$

Proof. Let $C_{\boldsymbol{\Lambda},\boldsymbol{\Lambda}_0}$ denote the (random) cost of a first passage from $\boldsymbol{\Lambda}$ to $\boldsymbol{\Lambda}_0$ under some policy. It suffices to show that there exists a stationary policy

that induces an irreducible, aperiodic Markov chain satisfying $\mathbb{E}[C_{\mathbf{\Lambda},\mathbf{\Lambda}_0}] < \infty$ for all $\mathbf{\Lambda}$ [29, Proposition 5]. To show that this condition is satisfied, consider the stationary random policy introduced in Section 8.3 that schedules a random sensor in each time step. Because the induced Markov chains from Appendix A describing the cost of each source are irreducible, aperiodic and positive recurrent, the Markov chain describing the average cost of all K sources (obtained by augmenting the state space) is also irreducible, aperiodic and positive recurrent. Let $\mathbf{\Lambda}_0 = (\boldsymbol{S}_0, \mathbf{\Delta}_0)$, where $\boldsymbol{S}_0$ is an arbitrary set of source states and $\mathbf{\Delta}_0 = \{1, 2, \ldots, K\}$, then, due to the positive recurrence, the expected number of steps required to reach $\mathbf{\Lambda}_0$ from any state $\mathbf{\Lambda}$ is finite. Since $C(\mathbf{\Lambda}, a_t) < \infty$ for any $\mathbf{\Lambda}$ and a_t, we have that $\mathbb{E}[C_{\mathbf{\Lambda},\mathbf{\Lambda}_0}] < \infty$. □

Lemma 2. *$V_\alpha(\mathbf{\Lambda})$ is nondecreasing in the AoI of each source, Δ_k.*

Proof. We prove the result by induction using the value iteration procedure in Eq. (H.20), which is guaranteed to converge to the optimal value function for $0 < \alpha < 1$ from any initial conditions $V_\alpha^0(\mathbf{\Lambda})$. Note first that the cost function $C(\mathbf{\Lambda}, a)$ defined in (H.6) is nondecreasing in Δ_k, and by setting $V_\alpha^0(\mathbf{\Lambda}) = 0$ this also holds for the base case. For the induction step, note that the sum of nondecreasing functions is also nondecreasing, and the minimum operator preserves this property. Therefore, if $V_\alpha^n(\mathbf{\Lambda})$ is nondecreasing in Δ_k, then $V_\alpha^{n+1}(\mathbf{\Lambda})$ is also nondecreasing in Δ_k. It follows that $V_\alpha(\mathbf{\Lambda}) = \lim_{n\to\infty} V_\alpha^n(\mathbf{\Lambda})$ is nondecreasing in Δ_k as well. □

Lemma 3. *There exists a finite, nonnegative N such that $-N \leq h_\alpha(\mathbf{\Lambda})$ for all $\mathbf{\Lambda}$ and α.*

Proof. We prove that the inequality holds by first showing that the number of states $\mathbf{\Lambda}$ with $V_\alpha(\mathbf{\Lambda}) < V_\alpha(\mathbf{\Lambda}_0)$ is finite, and then showing that the value function of these states is bounded from below by a constant that is independent of α. Define $\mathbf{\Lambda}_0 = (\boldsymbol{S}_0, \mathbf{\Delta}_0)$, where $\boldsymbol{S}_0$ is an arbitrary set of source states and $\mathbf{\Delta}_0 = \{1, 1, \ldots, 1\}$. Because the value function $V_\alpha(\mathbf{\Lambda})$ is finite and nondecreasing in Δ_k (as established by Lemmas 1 and 2), any finite $N \geq 0$ satisfies the condition for all states $\mathbf{\Lambda} = (\boldsymbol{S}, \mathbf{\Lambda})$ with $\mathbf{\Lambda} \neq \mathbf{\Lambda}_0$. Furthermore, because the set of source states is finite, there are at most a finite number of states $(\boldsymbol{S}, \mathbf{\Delta}_0)$ for which $V_\alpha((\boldsymbol{S}, \mathbf{\Delta}_0)) < V_\alpha(\mathbf{\Lambda}_0)$. Because $\boldsymbol{S}$ evolves according to an irreducible positive recurrent Markov chain (independent of the policy), any state $(\boldsymbol{S}, \mathbf{\Delta}_0)$ will transition into $(\boldsymbol{S}_0, \mathbf{\Delta})$ for some AoI state $\mathbf{\Delta}$ within finite expected time at a finite expected total cost of $\mathbb{E}[C_{(\boldsymbol{S},\mathbf{\Delta}_0),(\boldsymbol{S}_0,\mathbf{\Delta})}]$. Because the value function is nondecreasing in Δ_k, we have that $V_\alpha((\boldsymbol{S}_0, \mathbf{\Delta})) \geq V_\alpha(\mathbf{\Lambda}_0)$. From the definition of the value function, this implies that $V_\alpha(\mathbf{\Lambda}_0) \leq V_\alpha((\boldsymbol{S}, \mathbf{\Delta}_0)) + \mathbb{E}[C_{(\boldsymbol{S},\mathbf{\Delta}_0),(\boldsymbol{S}_0,\mathbf{\Delta})}]$ and equivalently $-\mathbb{E}[C_{(\boldsymbol{S},\mathbf{\Delta}_0),(\boldsymbol{S}_0,\mathbf{\Delta})}] \leq V_\alpha((\boldsymbol{S}, \mathbf{\Delta}_0)) - V_\alpha(\mathbf{\Lambda}_0)$. Consequently, by setting $N = \max_{\boldsymbol{S}} \mathbb{E}[C_{(\boldsymbol{S},\mathbf{\Delta}_0),(\boldsymbol{S}_0,\mathbf{\Delta}')}]$ where $\mathbf{\Delta}' = \min\{\mathbf{\Delta} | \max_{\boldsymbol{S}} \mathbb{E}[C_{(\boldsymbol{S},\mathbf{\Delta}_0),(\boldsymbol{S}_0,\mathbf{\Delta})}] < \infty\}$ the condition is satisfied. □

Lemma 4. *There exist a nonnegative $M_{\mathbf{\Lambda}}$ such that $h_\alpha(\mathbf{\Lambda}) \leq M_{\mathbf{\Lambda}}$ for all $\mathbf{\Lambda}$ and α. Furthermore, $\mathbb{E}\left[M_{\mathbf{\Lambda}_{t+1}} | \mathbf{\Lambda}_t, a_t\right] < \infty$ for all $\mathbf{\Lambda}_t$ and a_t.*

Proof. It is sufficient to show that for each state $\mathbf{\Lambda}$ and action a, there exists a stationary policy $\pi_{\mathbf{\Lambda}',a}$ that chooses action a in state $\mathbf{\Lambda}'$ and induces an irreducible, aperiodic Markov chain that satisfies $\mathbb{E}[C_{\mathbf{\Lambda},\mathbf{\Lambda}_0}] < \infty$ for all $\mathbf{\Lambda}$ [29, Proposition 5]. To show that this condition is satisfied, let $\pi_{\mathbf{\Lambda}',a}$ be the random scheduling policy presented in Section 8.3 that schedules in each time slot a random sensor, except in state $\mathbf{\Lambda}'$, where it deterministically schedules a specific sensor indicated by the action a. Taking action a in state $\mathbf{\Lambda}'$ comes at a finite cost, and, since the action does not influence the source states but only the AoI, leads to a new state from which all other states can still be reached within finite time. Therefore, the Markov chain induced by $\pi_{\mathbf{\Lambda}',a}$ is still irreducible, aperiodic and positive recurrent, and thus has a unique stationary distribution. Furthermore, it satisfies $C(\mathbf{\Lambda}, a_t) < \infty$ for any $\mathbf{\Lambda}$ and a_t, and by repeating the argument used in the proof of Lemma 1 we conclude that $\mathbb{E}[C_{\mathbf{\Lambda},\mathbf{\Lambda}_0}] < \infty$. □

In accordance with [29, Theorem], the conditions presented in Lemmas 1, 3 and 4 are sufficient for Theorem 4.

References

[1] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, Č. Stefanović, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5G: physical and MAC-layer solutions," *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 59–65, Sep. 2016.

[2] J. J. Nielsen, G. C. Madueño, N. K. Pratas, R. B. Sørensen, Č. Stefanović, and P. Popovski, "What can wireless cellular technologies do about the upcoming smart metering traffic?" *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 41–47, Sep. 2015.

[3] V. C. Gungor, B. Lu, and G. P. Hancke, "Opportunities and challenges of wireless sensor networks in smart grid," *IEEE Trans. Ind. Electron.*, vol. 57, no. 10, pp. 3557–3564, Oct 2010.

[4] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things J.*, vol. 1, no. 1, pp. 22–32, 2014.

[5] A. Kosta, N. Pappas, and V. Angelakis, "Age of information: A new concept, metric, and tool," *Found. Trends Netw.*, vol. 12, no. 3, pp. 162–259, 2017.

[6] M. Costa, M. Codreanu, and A. Ephremides, "On the age of information in status update systems with packet management," *IEEE Trans Inf. Theory*, vol. 62, no. 4, pp. 1897–1910, 2016.

[7] Y. Sun, E. Uysal-Biyikoglu, R. D. Yates, C. E. Koksal, and N. B. Shroff, "Update or wait: How to keep your data fresh," *IEEE Trans. Inf. Theory*, vol. 63, no. 11, pp. 7492–7508, 2017.

[8] E. Najm, E. Telatar, and R. Nasser, "Optimal age over erasure channels," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2019, pp. 335–339.

[9] A. Javani, M. Zorgui, and Z. Wang, "Age of information in multiple sensing," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, 2019, pp. 1–6.

[10] E. Najm and E. Telatar, "Status updates in a multi-stream m/g/1/1 preemptive queue," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM) Workshops*, 2018, pp. 124–129.

[11] S. Farazi, A. G. Klein, and D. Richard Brown, "Average age of information in multi-source self-preemptive status update systems with packet delivery errors," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, 2019, pp. 396–400.

[12] M. Moltafet, M. Leinonen, and M. Codreanu, "On the age of information in multi-source queueing models," *IEEE Trans. Commun.*, vol. 68, no. 8, pp. 5003–5017, 2020.

[13] S. K. Kaul and R. D. Yates, "Timely updates by multiple sources: The M/M/1 queue revisited," in *Proc. Annu. Conf. Inf. Sciences Syst. (CISS)*, 2020, pp. 1–6.

[14] H. B. Beytur and E. Uysal, "Age minimization of multiple flows using reinforcement learning," in *Proc. Int. Conf. Comput., Netw. Commun. (ICNC)*, 2019, pp. 339–343.

[15] M. Moltafet, M. Leinonen, and M. Codreanu, "Average aoi in multi-source systems with source-aware packet management," *IEEE Trans. Commun*, pp. 1–1, 2020.

[16] A. Kosta, N. Pappas, A. Ephremides, and V. Angelakis, "Age of information performance of multiaccess strategies with packet management," *J. Commun. Netw.*, vol. 21, no. 3, pp. 244–255, 2019.

[17] V. Tripathi and S. Moharir, "Age of information in multi-source systems," in *Proc. IEEE Glob. Commun. Conf. (GLOBECOM)*, 2017, pp. 1–6.

[18] A. M. Bedewy, Y. Sun, S. Kompella, and N. B. Shroff, "Optimal sampling and scheduling for timely status updates in multi-source networks," *arXiv preprint arXiv:2001.09863*, 2020.

[19] Z. Jiang, B. Krishnamachari, X. Zheng, S. Zhou, and Z. Niu, "Timely status update in wireless uplinks: Analytical solutions with asymptotic optimality," *IEEE Internet of Things J.*, vol. 6, no. 2, pp. 3885–3898, 2019.

[20] H. Tang, J. Wang, L. Song, and J. Song, "Minimizing age of information with power constraints: Multi-user opportunistic scheduling in multi-state time-varying channels," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 5, pp. 854–868, 2020.

[21] Z. Jiang and S. Zhou, "Status from a random field: How densely should one update?" in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, 2019, pp. 1037–1041.

[22] J. Hribar, M. Costa, N. Kaminski, and L. A. DaSilva, "Using correlated information to extend device lifetime," *IEEE Internet of Things J.*, vol. 6, no. 2, pp. 2439–2448, 2019.

[23] Q. He, G. Dán, and V. Fodor, "Joint assignment and scheduling for minimizing age of correlated information," *IEEE/ACM Trans. Netw.*, vol. 27, no. 5, pp. 1887–1900, 2019.

[24] A. E. Kalør and P. Popovski, "Minimizing the age of information from sensors with common observations," *IEEE Wireless Commun. Lett.*, vol. 8, no. 5, pp. 1390–1393, 2019.

[25] Y. Shao, Q. Cao, and S. C. Liew, "Partially observable minimum-age scheduling: The greedy policy," *arXiv preprint arXiv:2009.13441*, 2020.

[26] B. Zhou and W. Saad, "On the age of information in internet of things systems with correlated devices," *arXiv preprint arXiv:2001.09863*, 2020.

[27] S. Ross, *Introduction to stochastic dynamic programming*. Academic Press, 1983.

[28] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

[29] L. I. Sennott, "Average cost optimal stationary policies in infinite state Markov decision processes with unbounded costs," *Operations Research*, vol. 37, no. 4, pp. 626–633, 1989.

[30] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2005, vol. 2.

[31] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[32] Y. Hsu, E. Modiano, and L. Duan, "Age of information: Design and analysis of optimal scheduling algorithms," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 561–565.

[33] R. R. Weber and G. Weiss, "On an index policy for restless bandits," *Journal of Applied Probability*, vol. 27, no. 3, pp. 637–648, 1990.

[34] I. Kadota, A. Sinha, E. Uysal-Biyikoglu, R. Singh, and E. Modiano, "Scheduling policies for minimizing age of information in broadcast wireless networks," *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2637–2650, 2018.

[35] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Athena Scientific, 2005, vol. 1.

[36] E. Fernández-Gaucherand, A. Arapostathis, and S. I. Marcus, "On the average cost optimality equation and the structure of optimal policies for partially observable Markov decision processes," *Annals Operations Research*, vol. 29, no. 1, pp. 439–469, 1991.

[37] A. Arapostathis, V. S. Borkar, E. Fernández-Gaucherand, M. K. Ghosh, and S. I. Marcus, "Discrete-time controlled Markov processes with average cost criterion: a survey," *SIAM J. Contr. Optimiz.*, vol. 31, no. 2, pp. 282–344, 1993.

[38] A. R. Cassandra, "Exact and approximate algorithms for partially observable Markov decision processes," Ph.D. dissertation, Brown University, USA, 1998.

[39] M. L. Littman, A. R. Cassandra, and L. P. Kaelbling, "Learning policies for partially observable environments: Scaling up," in *Proc. Int. Conf. Mach. Learn.* Elsevier, 1995, pp. 362–370.

[40] S. U. Pillai, T. Suel, and S. Cha, "The Perron-Frobenius theorem: some of its applications," *IEEE Signal Processing Mag.*, vol. 22, no. 2, pp. 62–75, 2005.

ISSN (online): 2446-1628
ISBN (online): 978-87-7573-936-3

AALBORG UNIVERSITY PRESS