Wrocław University of Science and Technology

Doctoral Dissertation

Random-Access Channel in Time-Critical Ad Hoc Network Synchronization

Patryk Stopyra

Supervisor: prof. dr hab. *Mirosław Kutyłowski*

Discipline: Information and Communication Technology

December, 2024

Rozprawę dedykuję rodzicom, Lidii i Piotrowi,
żonie Eli i synkowi Xaweremu,
w podziękowaniu za wiarę we mnie
i wsparcie w podążaniu za swoją pasją

To my parents, Lidia & Piotr,
wife Ela, and son Xawery,
in gratitude for your unwavering trust
and support in following my passion

Acknowledgments

I wish to express my gratitude to all my teachers. First and foremost, to my parents Lidia and Piotr. You have given me the first key lessons, teaching me I should be as steadfast as granite; your support has allowed me to do what makes me happy in life. Through the outstanding teachers encountered on my path, up to the mentors of my adult life.

Among these, I would like to mention two people in particular here. I would like to thank my supervisor, Prof. Dr. Mirosław Kutyłowski, for your patience, the great knowledge from which I constantly draw, his analytical approach to the problems we solve together, his attention, and the directness of his criticism, which always facilitates our communication (even when the *bit-rate* of my statements causes transmission collisions and disrupts the message!). I would also like to express my appreciation for the commitment of Prof. Dr. Jacek Cichoń, who is my mentor with an incomprehensibly perceptive mind and ingenuity. Thank you for caring, for both of you teaching me to tread on the stones of facts, avoiding getting bogged down in the mud of the wishful thinking paths.

Thank you to my loved ones, the teachers of everyday life. To my wife Ela, for teaching me such a beautiful definition of the word *love*; for believing and supporting me whenever I need you. You are my sanctuary. Wroclaw University of Science and Technology evokes the best associations in me, because I associate it primarily with you. As well, I associate the dissertation, with your presence next to me for the last nights of writing and with your dedication to find and help fix editing errors. Thank you Xawery, my son – you are teaching me the role of a teacher. I appreciate your unusual, for a boy of this age, patience and understanding during the doctoral studies carried out by your parents. Your curiosity, resoluteness, intelligence fill me with faith in the future. Dream cosmic dreams and pursue bold projects in your life!

I would like to thank my supporting family, teaching me kindness and care. Specially: parents Julia and Stefan, and grandparents Barbara and Bogusław.

My doctorate coincided with the *transitional times*: reform of education, which (as reforms tend to do) left many things in an unreadable situation. I would like to express my gratitude to the staff of the Wroclaw University of Science and Technology, the administration of the Department, the Council of Scientific Discipline, and the Faculty – who helped me to find my feet in these circumstances and to survive. In particular, I would like here to thank Dr. Maciej Gębala who is the glue and the spirit of our Department. Not only for your arch-interesting lectures on the Theoretical Foundations of Computer Science and Theory of Computation, but also for your commitment and support during my doctoral studies. I thank Prof. Dr. Małgorzata Kotulska, head of the doctoral program, for her faith and patience; not everything depends on us, not every challenge succeeds, but what is important is who stands with us and what we leave behind.

Special thanks to my co-workers and the management of UBS - for their flexibility enabling me to pursue my Ph.D. while participating in one of the most interesting projects in the area of security and distributed systems being developed in Europe today. Michał Jarmołkowicz and Reinhard Kritzer, thank you for your trust many years ago.

I would like to express my gratitude to all the friends and well-wishers who believed that I would be able to solve the problems set before me, complete my research, and get to where I am today. In particular, I would like to thank Filip, Artur, Asia & Arek, Monika & Jarek, Patient, Rafał, Wojciech, Norbert, Krista, Paweł, Paweł, the groups of *Rock Clan* and *Young Wolves*; Patient, I know it is not easy, but I believe in you – you are almost done, you can complete your Ph.D. thesis too! I am grateful for the expressions of support, as well for the doubts expressed about my stubbornness – both attitudes I appreciate as a manifestation of commitment. Thank you all for your believe, even if you could not understand my motivations!

The parable of the master, the apprentice, and the clay

– Shifu, I have come to learn... What are you contemplating at the hearth?
– Welcome Tudi, I am firing bricks. Yet, every brick I fire is imperfect...
– Indeed. But after all, there are no perfect bricks, Master!
– And it is impractical...
– Yes... A single brick is quite impractical. In any case, no more practical than a stone that can simply be lifted from the ground.
– And it is dangerous...
– Right! A single brick, thrown can easily do a lot of evil, rather little good... So why do you toil so hard Master?
– How else will we build a solid house? When the bricks stand in an even array, together they will support the ceiling and resist the forces of nature to give warm shelter to the next generations. Although individual bricks don't matter much, solid bricks together will support the edifice and form a great work that will last me and you. Do you realize how great a responsibility they face...?
– The responsibility of... bricks, Master...?
– Tudi, all the time we've been talking about you...

Podziękowania

Chciałbym podziękować wszystkim moim nauczycielom. Poczynając od rodziców, Lidii i Piotra, pierwszych pośród nauczycieli. To Wy przekazaliście mi kluczowe nauki, wskazując jak powinienem być wytrwały – niczym granit; Wasze wsparcie pozwoliło mi robić w życiu to co sprawia mi radość. Poprzez wybitnych nauczycieli napotykanych na mojej drodze, aż po mentorów życia dorosłego.

Wśród tych, w szczególności pragnę tu wymienić dwie osoby. Dziękuję promotorowi, prof. dr hab. Mirosławowi Kutyłowskiemu za cierpliwość, wiedzę z której nieustannie czerpię, analityczne podejście do problemów jakie wspólnie rozwiązujemy, poświęconą uwagę oraz bezpośredniość krytyki, która ułatwia nam komunikację (nawet gdy *bit-rate* moich wypowiedzi powoduje kolizje transmisji i zakłóca przekaz!). Dziękuję prof. dr hab. Jackowi Cichoniowi, będącemu dla mnie mentorem o nieodgadnionej przenikliwości umysłu i pomysłowości. Dziękuję za opiekę, za to że obaj nauczyliście mnie stąpać po kamieniach faktów, unikając grzęźnięcia w błocie którym pokryte są drogi myślenia życzeniowego.

Dziękuję moim najbliższym, nauczycielom dnia codziennego. Żonie Eli, za nauczenie mnie tak pięknej definicji słowa *miłość*; za to, że wierzysz i jesteś przy mnie zawsze gdy Cię potrzebuję. Jesteś moją ostoją. Politechnika Wrocławska budzi we mnie najlepsze skojarzenia, bo nierozerwalnie kojarzy mi się z Tobą. Rozprawa zaś z tym, jak przez ostatnie noce byłaś obok i z zaangażowaniem wyszukiwałaś i pomagałaś naprawiać błędy edytorskie. Synkowi Xaweremu, uczysz mnie roli nauczyciela wykazując przy tym niezwykłą, jak na chłopca w tym wieku, cierpliwość i zrozumienie podczas doktoratów realizowanych przez Twoich rodziców. Twoja ciekawość, rezolutność, inteligencja napawają mnie wiarą w przyszłość. Synu, śnij kosmiczne sny i realizuj w życiu śmiałe projekty!

Dziękuję bliskim, którzy mnie wspierają, uczą życzliwości i troski. W szczególności: rodzicom Julii i Stefanowi oraz dziadkom Barbarze i Bogusławowi.

Mój doktorat przypadł na *czasy przejściowe*: reformę edukacji, która (jak to zwykły robić reformy) pozostawiła wiele rzeczy w nieczytelnej sytuacji. Pragnę wyrazić wdzięczność wobec pracowników Politechniki Wrocławskiej, administracji Katedry, Rady Dyscypliny Naukowej i Wydziału, którzy pomogli mi odnaleźć się w tych okolicznościach i przetrwać. W szczególności chciałbym tu podziękować dr Maciejowi Gębali który jest spoiwem i duszą naszej Katedry. Nie tylko za arcyciekawe wykłady z Teoretycznych Podstaw Informatyki i z Teorii Obliczeń, ale za zaangażowanie i wsparcie podczas realizowanego doktoratu. Dziękuję pani prof. dr hab. inż. Małgorzacie Kotulskiej, kierownikowi studiów doktoranckich, za wiarę i cierpliwość; nie wszystko zależy od nas, nie każde wyzwanie się udaje, ważne kto był przy nas i co po sobie pozostawiamy.

Dziękuję współpracownikom i kierownictwu firmy UBS - za elastyczność i umożliwienie realizacji mojego doktoratu przy jednoczesnym współtworzeniu jednego z najciekawszych projektów w obszarze bezpieczeństwa i systemów rozproszonych jakie powstają obecnie w Europie. Michał Jarmołkowicz i Reinhard Kritzer, dziękuję za Wasze zaufanie wiele lat temu.

Pragnę wyrazić moją wdzięczność przyjaciołom i życzliwym, którzy wierzyli, że dam radę rozwiązać problemy przede mną postawione, dokończyć badania, dotrzeć do punktu w którym się znajduję. W szczególności chcę tu wymienić Filipa, Artura, Asię i Arka, Monikę i Jarka, Patient, Rafała, Wojciecha, Norberta, Kristę, Pawła, Pawła, *Skalne Bracto* oraz *Młodych Wilków*. Patient, wiem, że nie jest łatwo, ale wierzę w Ciebie – też dokończysz swoją rozprawę! Jestem wdzięczny za słowa wsparcia tudzież za wątpliwości wyrażane wobec mego uporu - obie postawy doceniam jako wyraz zaangażowania. Dziękuję, że we mnie wierzyliście, nawet jeśli nie rozumieliście moich motywacji!

Przypowieść o mistrzu, uczniu i glinie

– Shifu, przyszedłem na nauki... Nad czym dumasz przy tym palenisku?
– Witaj Tudi, wypalam cegły. Lecz każda cegła jaką wypalam jest niedoskonała...
– Rzeczywiście. Ale przecież nie ma doskonałych cegieł Mistrzu!
– I niepraktyczna...
– Tak... Pojedyncza cegła jest dosyć niepraktyczna. W każdym razie nie bardziej praktyczna od kamienia, który można po prostu podnieść z ziemii.
– I niebezpieczna...
– Racja! Pojedyncza cegła, rzucona może z łatwością wyrządzić wiele zła, raczej niewiele dobra... Więc czemu się tak trudzisz Mistrzu?
– A jak inaczej zbudujemy solidny dom? Kiedy cegły staną w równym szyku, wspólnie podeprą strop i oprą się siłom natury aby dać ciepłe schronienie kolejnym pokoleniom. Choć pojedyncze nie mają dużego znaczenia, solidne cegły wspólnie wspierające gmach utworzą wielkie dzieło które przetrwa i mnie i Ciebie. Zdajesz sobie sprawę jak wielka stoi przed nimi odpowiedzialność...?
– Odpowiedzialność... cegieł, Mistrzu...?
– Przecież od początku mówimy o tobie Uczniu...

Abstract

In this doctoral dissertation, I examine two strategies for slotted channel random access concerning the time-critical ad hoc network synchronization problem. The synchronization is understood by collecting a *control message* from every network agent. The agents act independently; not knowing each other a priori, they form an ad hoc network by accessing the same communication channel.

The challenging part is that all agents' synchronization is triggered simultaneously and must occur in a constrained time. The standard contention resolution techniques, such as backoff, are not really useful in this situation. Instead, each agent shall access multiple channel slots aligning to some strategy, hoping for at least one non-collision transmission. The prevailing application use cases are traffic coordination systems, Unmanned Aerial Vehicles (UAV) swarms, and intermittent sensor networks; their mission strictly depends on network synchronization within a constrained timeframe.

This study focuses primarily on the characteristics of two proposed channel random access strategies. The abstract communication model is defined where the time-constraint requirement is formalized by the finite-length communication round consisting of n slots. I assume that if two agents transmit in the same slot, there is a collision, and no message comes through. The goal is that every agent transmits successfully (without collision) in at least one slot.

The first strategy, Bernoulli trials (BT), is a default approach where every agent transmits in each slot with a predefined probability $p \in [0, 1]$. Using the alternative strategy, t-Slots, every agent chooses at random exactly t out of n available slots and transmits in these slots; $t \in \{1, 2, \ldots, n\}$ is a strategy parameter.

The research hypothesis states that t-Slots behaves better than BT. The hypothesis is confirmed in two phases.

First, I analyze the restrictive case where each agent has to communicate successfully, having at least one collision-less transmission. The second case is when up to f agents may fail to communicate due to collisions. In both cases, I show that t-Slots is superior to BT in the sense of a higher probability of success and, at the same time, fewer messages sent over the shared communication channels. For both cases, I present the optimal choice of the parameters p and t that maximizes the probability of success. These values can be provided based on analytical formulas derived for both strategies. In the (complex) case of t-Slots, it was necessary to use analytical combinatorics tools.

The formulas derived for BT and t-Slots allowed me to provide a comprehensive comparative analysis. The examples analyzed combine the calculated results and the statistical tests I performed; the statistical results perfectly align with the calculated synchronization probabilities. Separately, for all-agents and fault-tolerant synchronization, I compare the strategies in terms of synchronization achievability, suboptimal configuration tolerance, properties of their algorithms, energy consumption, and scalability. The gathered results confirm the research hypothesis.

The implementations of the BT and t-Slots success probability formulas are delivered as a Python library, along with the raw statistical test data.

Keywords— mobile ad hoc networks, ad hoc initialization, network synchronization, coordination systems, traffic control, drone swarms, multiple access, random-access channels, slotted communication channels, real-time systems

Streszczenie

W niniejszej dysertacji doktorskiej badam dwie strategie losowego dostępu do kanału komunikacyjnego podzielonego na sloty (ang. *slotted random-access channel*). Strategie te dotyczą problemu ograniczonej czasowo synchronizacji sieci ad hoc. Synchronizacja ta jest rozumiana jako odebranie wiadomości kontrolnych od każdego z agentów w sieci. Agenci działają niezależnie; nie wiedząc o sobie nic z wyprzedzeniem tworzą dynamicznie sieć ad hoc poprzez dostęp do wspólnego kanału komunikacyjnego.

Wyzwanie stanowi fakt, że synchronizacja agentów wyzwolona jest jednocześnie dla całej sieci oraz musi nastąpić w ograniczonym czasie. Standardowe techniki rozwiązywania kolizji, takie jak wycofanie (ang. *backoff*), nie są w tym przypadku użyteczne. Przeciwnie, każdy agent powinien podejmować wielokrotną próbę transmisji swojej wiadomości w różnych dostępnych slotach z nadzieją na co najmniej jedną bezkolizyjną transmisję. Najważniejsze zastosowania jakim odpowiada omawiany schemat komunikacji dotyczą koordynacji ruchu (adaptacyjne światła drogowe), roju dronów oraz rozproszonej sieci sensorów (zwłaszcza sensorów o przerywanym trybie pracy). Powodzenie zadań realizowanych przez powyższe systemy ściśle zależy od sukcesu synchronizacji sieci, następującego w zadanym czasie.

Badania bazują przede wszystkim na analizie charakterystyki dwóch zaproponowanych strategii losowego dostępu do kanału komunikacyjnego. W pracy definiuję abstrakcyjny model komunikacji, w którym ograniczenie czasowe zrealizowane jest poprzez rundę komunikacji złożoną z ustalonej, skończonej liczby n slotów. Zakładam przy tym, że jeżeli dwóch agentów transmituje w tym samym slocie, następuje kolizja i żadna z wysłanych wiadomości nie może być odczytana. Nadrzędnym celem komunikacji jest to aby każdy agent sieci co najmniej raz nadał swoją wiadomość bezkolizyjnie.

Pierwszą zaproponowaną strategią są Próby Bernoullego (BT). To domyślne podejście w którym każdy z agentów transmituje w każdym z dostępnych slotów rundy komunikacji ze zdefiniowanym wcześniej prawdopodobieństwem $p \in [0, 1]$. Drugą zaproponowaną strategią jest t-SLOTS, w której każdy z agentów wybiera losowo t pośród n dostępnych slotów rundy komunikacji i to w nich podejmuje transmisję swojej wiadomości. Przy tym $t \in \{1, 2, \ldots, n\}$ jest parametrem ustalonym z wyprzedzeniem.

Postawiłem następującą hipotezę badawczą: strategia t-SLOTS ma lepsze właściwości niż strategia BT. Hipoteza ta jest dowiedziona w dwóch fazach.

W pierwszej fazie analizuję rygorystyczne zagadnienie w którym każdy agent musi mieć co najmniej jedną bezkolizyjną transmisję w rundzie komunikacji. W drugiej fazie badam uogólniony przypadek w którym nie więcej niż f agentom nie udaje się nadać wiadomości ze względu na kolizje z innymi nadającymi agentami. Wykazuję przewagę t-Slots nad BT objawiającą się wyższym prawdopodobieństwem poprawnej synchronizacji oraz mniejszą oczekiwaną liczbą podejmowanych transmisji w optymalnym przypadku. Dla obu faz prezentuję optymalne ustawienia parametrów obu strategii, maksymalizujące prawdopodobieństwo poprawnej synchronizacji. Wartoście te mogłem wywnioskować bazując na analitycznych formułach na prawdopodobieństwo synchronizacji wyprowadzonych dla obu strategii. W złożonym przypadku, dopuszczającym występowanie ograniczonej liczby agentów którym nie udało się nadać wiadomości, konieczne było zastosowanie przeze mnie narzędzi kombinatoryki analitycznej.

Formuły wyprowadzone dla strategii BT oraz t-Slots pozwoliły mi na kompleksowe porównanie obu analizowanych strategii. Przypadki które analizowałem łączyły rezultaty matematyczne oraz wyniki przeprowadzonych przeze mnie testów statystycznych. Przy tym wyniki obliczeniowe dla prawdopodobieństwa udanej synchronizacji są całkowicie zgodne z wynikami testów statystycznych.

Osobno dla rygorystycznego przypadku całkowitej synchronizacji oraz dla uogólnionego przypadku częściowej synchronizacji porównuję obie strategie pod kątem prawdopodobieństwa osiągnięcia synchronizacji, odporności na nieoptymalną konfigurację p oraz t, właściwości ich algorytmów, zużycia energii oraz skalowalności. Zebrane wyniki potwierdzają hipotezę badawczą.

Implementacje wyprowadzonych formuł prawdopodobieństwa synchronizacji strategii BT oraz t-Slots opublikowane są jako biblioteka języka Python, wraz z danymi przeprowadzonych eksperymentów oraz narzędziem do symulacji i automatyzacji testów statystycznych dla wybranego zagadnienia.

Keywords— mobilne sieci ad hoc, inicjalizacja sieci ad hoc, synchronizacja sieci, systemy koordynacji, kontrola ruchu, roje dronów, wspóldzielony dostęp do kanału, kanały dostępu losowego, kanały komunikacyjne z podziałem na sloty, systemy czasu rzeczywistego

Contents

Chapter 1

Introduction

1.1 Motivation

The motivation behind this work is an emerging demand for the ad hoc, time-constrained synchronization of distributed agents. The challenge behind this is simultaneous channel access of the ad hoc network agents. Wireless connection systems use random-access channels to address contention during such spontaneous connections; the synchronization most likely leads to channel congestion.

The stated problem is usually addressed in the developed conceptual systems with either the pre-configured TDMA/FDMA multiple access network [5, 81] or standard contention-resolution methods deriving from ALOHA [19, 71] and CSMA [14, 45]. The first solution is proper for prototyping or high-integrity military applications, but not realistic in scalable systems where it is impossible to coherently pre-configure all unrelated agents. The second approach is more realistic but suffers from the nature of the applied protocols, designed to resolve contention with a promise of the eventual consistency of all agents, not a consistency achieved in the constrained timeframe. These protocols often rely on the acknowledgment of the message delivery [22, 64], which cannot always be granted in the rapid time-constrained synchronization, which limits their usage. Even when the chance of time-critical synchronization is estimated in these systems, the results has a high variation due to the applied contention resolution mechanisms, such as a randomized or exponential backoff [88]. Thus, neither approach is viable for mission-critical applications, where the probability of time-constrained synchronization must be well-estimated and predictable.

The most promising of the emerging applications are, indeed, related to mission-critical systems. The synchronization is understood as a collection of *control messages* from all involved agents (either by a central orchestrator or by other agents). Control messages are instantiated specifically for the use case. In *Adaptive Traffic Control Systems* (ATCS), control messages are vehicle presence announcements [10, 41, 94] . In *Autonomous Vehicles* (AV) coordination systems, control messages may contain coordinates or mission objectives

[16, 54, 80]. In distributed sensor networks and *Internet of Things* (IoT) devices, the control message can be a beacon signal or the device output [50, 81, 98].

To provide a more sustainable solution to the problem sketched, this dissertation thoroughly analyzes two random-channel access strategies designed purposely for the time-criticality objective: BT and t-SLOTS. The strategies were proposed in the author's previous work [23], where they were also overviewed under the restrictive all-agents synchronization requirement. Synchronization must take place within a single communication round consisting of a finite number of independent slots (n) randomly accessed by agents transmitting their messages; this is an abstraction for the slotted random-access channel under the synchronization time constraint. When an agent transmits their message in the slot alone, it succeeds to communicate; when more than single agents transmit in the slot, collision occurs and no message comes through. BT strategy assumes that every agent transmits its message with the predefined probability p in every slot, ending with the np expected number of transmissions; it is a Bernoulli trials process, hence the name of strategy. t-SLOTS strategy defines that every agent selects exactly t out of n available slots, where t is predefined.

There is only a subtle difference in slot selection method, but t-SLOTS results in a lower variability of channel load. The analysis for the absolute synchronization and the generalized fault-tolerant synchronization shall lead to the derivation of synchronization probability formulas and the optimal configuration of the strategy parameters. Consequently, not only the arbitrary time-critical synchronization could be modeled for BT and t-SLOTS, but also the strategies could be comprehensively analyzed to prove the universal advantage of the proposed t-SLOTS strategy.

1.2 Outline and contribution

The key contribution of this work is a derivation of formulas for the synchronization probability for BT and t-SLOTS strategies, provided in two versions: for absolute synchronization of all agents and fault-tolerant synchronization. They lead to the proposal of the optimal configurations of strategies and their comprehensive comparison, which confirms the stated research hypothesis.

Along with the mathematical formulas, their implemented version is provided as an open-source Python library [85]. The statistical tests confirmed the results; the underlying raw test data and source code of the Python tool implemented for testing are available as an open source [86]. Sample author's implementations of BT and t-SLOTS agents in C language are available [84].

This dissertation is structured into five chapters. The first chapter, an *Introduction*, highlights the work motivation, explains the structure of the work, offers disambiguation of the key used terms, clarifies the mathematical notation, and accumulates the key mathematical facts referred to in the subsequent reasoning.

The second chapter, *The Synchronization problem*, introduces the background of Multiple Access (MA) and time-critical ad hoc network synchronization problems. Examples of practical applications of the research findings are briefly presented, emphasizing the importance of the topic. The following sections of this chapter lay the study's theoretical foundation, specifying the analyzed model. They are dedicated to formally defining the problem, strategies, and the research hypothesis.

The third chapter, *All-agents synchronization*, is devoted to the absolute synchronization scenario in which the successful communication of all agents is studied. This chapter refers to the author's previous research [23], citing and extending therein findings. The agent's algorithms, synchronization sample spaces, formulas for the probability of synchronization, optimal configuration settings, and example-based properties discussions are presented for both strategies. The chapter concludes with the comparison of BT and t-SLOTS strategies against factors such as probability of synchronization, immunity to misconfiguration, algorithm properties, energy efficiency, and scalability.

The fourth chapter, *Fault-tolerant synchronization*, contains the study of fault-tolerant synchronization, where some limited number of agents are allowed to face communication failure. This part is an original research that provides novel results to the generalized problem. Adapted formulas for synchronization probabilities, optimal configuration settings, and example-based impact of the fault tolerance are presented for both strategies. Again, the chapter concludes with a juxtaposition of synchronization probability, misconfiguration immunity, energy efficiency, and scalability of BT and t-SLOTS strategies. This time, the comparison focuses on the impact of the fault-tolerance level.

Finally, the last chapter, *Summary*, collates previous conclusions to confirm the research hypothesis and identify future research directions.

1.3 Definitions

Below, a disambiguation of the key terms used in the dissertation may be found. Definitions are purely context-oriented and focused on the analyzed problem. Terms are listed in the logical dependency order.

BTS base transceiver station, central network unit facilitates (and often coordinates) wireless communication.

Message is a unit of data exchanged in the network and contains some coordination information; may be an *introduction*, *presence notification*, or a *sensing results* communicate.

Agent is a communication network node, able to send (receive) *messages*; an abstract representation of the remote sensor, user equipment, antenna-equipped device, or vehicle.

Communication channel is medium of communication established between *agents* to transmit their *messages*.

Agent's transmission is an *agent*'s attempt to broadcast its *message* over the *communication channel*.

Communication slot is a single slot of the *communication channel*, where the *agent* is capable to transmit their whole *message*. Often referred to as just a *slot*.

Communication round consists of a finite number of the *communication slots* and is the entire analyzed phase of communication, when *agents* may exchange their information.

Strategy is the algorithm followed by all *agents*, that specifies whether to *transmit* or not in the given *communication slot*; in this work, it always pertains to the random-access channel strategy.

Slot's occupation happens when any *agent* transmits their message in the slot. The communication slot may not be occupied at all or may be occupied by one or more agents.

Transmission set is the set of *communication slots* in which the agent attempts to *transmit* their message.

Collision occurs when two or more *agents transmit* in the same *slot*, effectively jamming each other. No message can be broadcast in the slot where the collision occurred.

Successful transmission takes place when the *agent* is the only one *transmitting* in the *slot*, thus it successfully broadcasts its message.

Communication success occurs when an *agent transmits* their message without *collision* in at least one of available *communication slots*.

Communication failure is the opposite of the *communication success*, occurs when the *agent* does not *transmit* their *message collision*-free at least once.

Synchronization (of the network) is an event of agents exchanging their information through the individual *successful transmissions* in the *communication round*. Often appears along with the *strategy* name, indicating which channel access *strategy* all network agents implement in order to synchronize.

All-agent Synchronization is a *synchronization*, which is told to be achieved if and only if all the participating agents had a *communication success*.

Fault-tolerant Synchronization is a *synchronization*, which is told to be achieved if the number of agents having a *communication failure* does not exceed a declared threshold.

Failure tolerance is the tolerated number of *agents* that have a *failed communication*, while the network still succeeds to achieve the *fault-tolerant synchronization*.

1.4 Notation

Consistent mathematic notation is used in this work, as stated below.

Across the work, tuples are noted with brackets "()," while vectors are noted with angle brackets "⟨⟩." The square brackets have a meaning established by the context. They serve as a parenthesis of the probability function Pr[...]; when they classically denote the interval on $\mathbb{R}$ they are always accompanied by the "∈" symbol, ex. $p \in [0, 1]$; otherwise, in Section 3.3 and Section 4.3, they denote a condition of *extraction* (Fact 8), such as $[{x_1}^{n_1}{x_2}^{n_2}]$.

Vector $\overline{x}$'s L1 norm is denoted as $\|\overline{x}\|$; given $\overline{x} = \langle x_1, x_2, \ldots, x_n\rangle$

$$\|\overline{x}\| = \sum_{i=1}^{n} |x_i| \ . \tag{1.1}$$

For $\overline{x} = \langle x_1, x_2, \ldots, x_n\rangle$, $\overline{y} = \langle y_1, y_2, \ldots, y_n\rangle$, the shortcut notation is applied for consecutive scalar exponentiations ${x_1}^{y_1}{x_2}^{y_2}\ldots{x_n}^{y_n} = \overline{x}^{\overline{y}}$. Additionally, the element-wise vector-to-scalar comparison "$\preceq$", "$\succeq$" operations are introduced

$$\preceq, \succeq : \mathbb{R}^n \times \mathbb{R} \to \{true, false\} \ , \tag{1.2}$$

given the vector $\overline{x} = \langle x_1, x_2, \ldots, x_n\rangle$ and value s

$$\overline{x} \preceq s \iff (\forall x_i \in \overline{x}) \ \ x_i \leqslant s \ . \tag{1.3}$$

where "$\succeq$" is the opposite

$$\overline{x} \succeq s \iff (\forall x_i \in \overline{x}) \ \ x_i \geqslant s \ . \tag{1.4}$$

Derivatives in the text are always expressed with a Leibniz notation as $\frac{d}{dx}f(x)$. Sometimes the formulas exceed the available space; they are then substituted with a bold capital letter, like **A**, **B**, and analyzed independently. When analyzing functions, the terms of *concavity* and *convexity* are referred to with their standard meaning. Given the function f and $\lambda \in [0, 1]$

$$\text{function } f \text{ is concave} \iff f(\lambda x_1 + (1-\lambda)x_2) \geqslant \lambda f(x_1) + (1-\lambda) f(x_2) \ , \tag{1.5}$$

and

$$\text{function } f \text{ is convex} \iff f(\lambda x_1 + (1-\lambda)x_2) \leqslant \lambda f(x_1) + (1-\lambda) f(x_2) \ . \tag{1.6}$$

When referring to the events, especially the synchronization events, like the $\mathsf{Success}^{\mathsf{BT}}_{n,k,p}$, in the superscript, the name of the strategy is highlighted, and in the subscript, the communication network setup parameters are given. When some parameter is not part of the network setup but still defines the event, it is provided in the following brackets, for instance, $\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}(f)$.

The big-O notation is used when analyzing algorithms

$$f(n) = O(g(n)) \iff (\exists c > 0)(\exists n_0)(\forall n \geqslant n_0) \ \ f(n)| \leqslant c|g(n)| \ . \tag{1.7}$$

1.5 Mathematical preliminaries

The common mathematical facts (formulas, theorems, schemes) used in the reasoning of this dissertation are listed in this section with references.

Fact 1 (Binomial Theorem). *[82] For any $n \in \mathbb{N}$ and $x, y \in \mathbb{R}$*

$$(x+y)^n = \sum_{r=0}^{n} \binom{n}{r} x^r y^{n-r} \tag{1.8}$$

where $\binom{n}{r}$ is a binomial coefficient defined by

$$\binom{n}{r} = \frac{n!}{r!(n-r)!} \ . \tag{1.9}$$

Fact 2 (Multinomial Theorem). *[82] For any $n \in \mathbb{N}$ and $x_1, \ldots, x_r \in \mathbb{R}$*

$$(x_1 + ... + x_r)^n = \sum_{\substack{\nu_1+...+\nu_r=n \\ \nu_1,...,\nu_r \in \mathbb{N}}} \binom{n}{\nu_1, \nu_2, \ldots, \nu_k} {x_1}^{\nu_1} {x_2}^{\nu_2} \ldots {x_k}^{\nu_k} = \sum_{\substack{\|\overline{\nu}\|=n \\ \overline{\nu} \succeq 0}} \binom{n}{\overline{\nu}} \overline{x}^{\overline{\nu}} \tag{1.10}$$

where $\binom{n}{\nu_1,...,\nu_k}$ is a multinomial coefficient

$$\binom{n}{\nu_1, \nu_2, ..., \nu_r} = \frac{n!}{\nu_1!, \nu_2!, \ldots, \nu_r!} \ . \tag{1.11}$$

Fact 3 (Probability of success in Binomial Distribution). *[68] Given the binomial distribution with n experiments and p probability of success in each experiment, the probability of the overall m successes equals*

$$\Pr\left[\mathsf{Successes}_{B(n,p)}(m)\right] = \binom{n}{m} p^m (1-p)^{n-m} \ . \tag{1.12}$$

Fact 4 (Power Set). *The Power Set $\mathcal{P}(X)$ is a set of all the subsets of X. The cardinality of Power Set $|\mathcal{P}(X)| = 2^{|X|}$.*

Fact 5 (Dominant Family of Sets). *[4] Family of sets $\mathbf{F}$ is* dominant*, if no set is contained in the union of the remaining sets*

$$(\forall S \in \mathbf{F}) \left(S \nsubseteq \bigcup_{S_i \in \mathbf{F} \setminus \{S\}} S_i \right) \ . \tag{1.13}$$

Fact 6 (Inclusion-Exclusion Principle). *[39] Given the finite sets $A_1, A_2, \ldots, A_n$*

$$\left| \bigcup_{i=1}^{n} A_i \right| = \sum_{\emptyset \neq J \subseteq \{1,2,...,n\}} (-1)^{|J|+1} \left| \bigcap_{j \in J} A_j \right| \ . \tag{1.14}$$

Fact 7 (Margin of Error, MOE). *[53] Given the sample size n, standard deviation of the sample σ, the Margin of Error is*

$$MOE_\gamma = Z_\gamma \left(\frac{\sigma}{\sqrt{n}}\right) \tag{1.15}$$

where Z_γ is standard normal deviate (Z-score), $Z_{0.999} \approx 3.090$.

Fact 8 (Extraction operation). *[27] Given C that does not include any variable ${x_i}^{n_i}$, a multivariable polynomial $C{x_1}^{m_1}\dots{x_k}^{m_k}$, and $\overline{x}^{\overline{n}} = {x_1}^{n_1}\dots{x_k}^{n_k}$. $[x_1^{n_1}\dots x_k^{n_k}]$ denotes an* extraction operation *for multivariable polynomials.*

$$[x_1^{n_1}\dots x_k^{n_k}]Cx_1^{m_1}\dots x_k^{m_k} = \begin{cases} C & \text{if } m_i = n_i \text{ for } i = 1,\dots,k \\ 0 & \text{otherwise.} \end{cases} \tag{1.16}$$

Additionally, for $\prod$ (and $\sum$)

$$[{x_1}^{n_1}\dots{x_k}^{n_k}]\prod_i X_i = \prod_i [{x_1}^{n_1}\dots{x_k}^{n_k}]X_i\ . \tag{1.17}$$

The additional condition may appear in the square brackets operation to further specify the extraction operation. For example, to extract polynomials containing x_i of non-zero exponents $[{x_1}^{n_1}\dots{x_k}^{n_k}, n_1,\dots,n_k \geqslant 1] = [\overline{x}^{\overline{n}}, \overline{n} \succeq 1]$.

Fact 9 (Expansion of n-powered $(1+x_i)$ product). *[23]*
Given $\overline{x}^{\overline{\alpha}} = {x_1}^{\alpha_1}{x_2}^{\alpha_2}\dots{x_k}^{\alpha_k}$

$$\left(\prod_{i=1}^{k}(1+x_i)\right)^n = \sum_{0\preceq\overline{\alpha}\preceq n} \overline{x}^{\overline{\alpha}} \prod_{i=1}^{k}\binom{n}{\alpha_i} \tag{1.18}$$

Proof.

$$\left(\prod_{i=1}^{k}(1+x_i)\right)^n = \prod_{i=1}^{k}(1+x_i)^n = \prod_{i=1}^{k}\sum_{j=0}^{n}\binom{n}{j}{x_i}^j = \sum_{0\preceq\overline{\alpha}\preceq n} \overline{x}^{\overline{\alpha}} \prod_{i=1}^{k}\binom{n}{\alpha_i}$$

□

Fact 10 (Pigeonhole Principle). *Also known as* Dirichlet's drawer principle. *If n items are distributed among m containers and $n > m$, then at least one container must contain more than one item.*

Fact 11 (Möbius Inversion Formula). *[52] Given $f, g : P(\{1,2,\dots,k\}) \to \mathbb{R}$ and $g(A) = \sum_{B\subseteq A} f(B)$, then*

$$f(B) = \sum_{C\subseteq B} g(C)(-1)^{|B|-|C|}\ . \tag{1.19}$$

Chapter 2

The Synchronization problem

2.1 Multiple access techniques

Since the early days of computer networks, contention has been a vital challenge. Regardless of the communication medium (cable or radio), the collision of transmissions most likely destroys the transferred data, turning the signal into an unprocessable noise. Permanent allocation of a medium access time or a communication spectrum to specific agents is beyond the capability of most networks due to the unacceptable waste of resources and rigidity of such a solution. Instead, a medium must be utilized appropriately and accessible for the dynamic set of agents. These requirements are addressed by a contention medium that allows multiple agents to access the same channel on demand at the cost of potential transmission collisions. A primary example of the contention medium is the random-access channel, where devices (network agents) access the channel whenever they need to transmit their messages. Concepts of how to share the medium to avoid or manage collisions to minimize their impact are commonly referred to as a *Multiple Access* (MA). The Multiple Access protocols are classified in the *Medium Access Control* (MAC), a sublayer of the Data Link layer in the OSI Model [59].

The protocols for dealing with these collisions and ensuring proper channel utilization were developed for decades. The leading examples are the *Time Division Multiple Access* (TDMA) [33, 50] and the *Frequency Division Multiple Access* (FDMA) [33, 60]. TDMA splits the communication timeline, and FDMA splits the bandwidth into smaller, individually assigned slots, previously cutting the transmission timeline into constant-length frames. In decentralized networks, nodes may compete for randomly selected transmission slots until the acknowledgment of non-interfered transmission, marking a slot's assignment for the following frames [72]. A substantially different approach is taken in a *Code Division Multiple Access* (CDMA) [33, 77]. CDMA protocols accept signal interference while exerting orthogonal encoding (such as Hadamard-Walsh codes [3]), noise filtering, and power management techniques in signal processing to elicit individual messages from agents [46]. This subtle technique comes with its disadvantages and diverse efforts to remediate them – lengthy signal encoding that affects energy consumption [30], synchronous nature [57], duty to assign individual channel key assignment [34, 91, 42], and near-far agents problem [61].

The above techniques pursue a better utilization of the contention medium but require some orchestration or pre-allocation. More challenging is addressing the initial transmission in the random-access channel, especially in the wireless networks. Random access protocol must consider issues like a problem of hidden and exposed agents [55], distant nodes signal fading and delays [15], Doppler Effect [2, 95], signal interference related to the internal transmission collisions and the external noise [58]. Collisions in such channels are unavoidable, even when the carrier sense [49, 32] is available, there is no guarantee that the other device does not start transmission at the same time or that there is no colliding signal not strong enough to be detected. If the transmission fails due to the collision, it is repeated. However, the channel throughput is always constrained. Therefore, high congestion of agents could lead to channel saturation; various techniques are used to prevent it, for example, the backoff mechanism [88].

The pioneering wireless random-access channel MA protocol was the *ALOHA Random Access* (ALOHA RA), implemented for ALOHAnet [1] and established in 1968. The solution, developed at the University of Hawai'i, allowed to connect Hawai'i Islands with a wireless computer network. ALOHA RA allows network nodes to compete for a transmission channel by accessing it on demand with random delay retrials until receiving an acknowledgment, announced by a single base station over a separate coordination channel. Further analysis led soon to the development of S-ALOHA [76], which uses communication time slots to discretize the channel and significantly improve efficiency. Although intermittency of throughput and latency encouraged researchers to look for other solutions, the improved concepts, based on ALOHA were later proposed [15, 20, 18, 71] and, highly modified, protocol itself still finds the applications, especially in networks built of energy-concerned units, such as sensors or satellites [11, 18, 56, 92].

The *Carrier Sense Multiple Access* (CSMA) [49, 75, 32] was developed in the 1970s as an alternative to the ALOHA RA. It minimizes the network's transmission data loss thanks to simultaneous channel listening. CSMA is employed for collision avoidance (CA; used in Wi-Fi, IEEE 802.11) [70, 79], detection (CD; Ethernet networks) [67], or resolution (CR; special application, usually LAN-based) [19]. The drawbacks of CSMA are considerable energy demand, especially in the saturated channels [32, 9], reduced efficiency during congestion, and helplessness when dealing with hidden, exposed [55] or distant [15] network agents.

The schemes mentioned above have a common modus operandi: segment the available transmission spectrum into a finite number of slots randomly accessed by communicating agents. To design more versatile, higher throughput, reduced latency random-access protocols, the practical implementations often combine the above strategies in hybrid solutions [63, 75, 32, 79].

2.2 Time-constrained ad hoc synchronization

Ad hoc networks are initialized on demand, usually locally, among a moderate number of devices (agents) to provide monitoring, coordination, search-and-rescue, and other services [64]. These networks are predominantly wireless, decentralized, and infrastructure-less [64] as their primary purpose is to build an ephemeral communication structure, exchange information, and disassemble. Often, they are built of energy-concerned devices [44].

Ad hoc networks use mostly the random-access channel for communication, and most commonly, it is a slotted channel due to its throughput advantages [14, 71]. They usually rely on the distributed multiple access techniques evolved from ALOHA or CSMA, involving the channel sense [14, 45] and decentralized acknowledgments [22, 64]. As stated in the previous section, these techniques are prone to high overhead and exposed/hidden agent problems.

The synchronization, i.e., data collection from all network agents, is a standard operation executed in ad hoc networks. In practice, many use cases (presented in the next section) benefit from or rely on the additional requirement: a time constraint [13, 81]. When the synchronization is time-critical, the paradigm for synchronization alternates from the promise of all agents' eventual channel access to the maximization of collected agents' data within a given timeframe. The caveat is the contention resolution method applied.

Popular contention resolution methods do not address the time-constrained requirement [13], basing their synchronization promise on the low latency that deteriorates on congestion [78] – which is the probable condition when synchronization is triggered for all network agents. This is one reason why some use cases mentioned in Section 2.3 are not developing rapidly. The present implementations with time-criticality in mind still use low latency protocols that offer an eventual access promise [16, 54, 80] ; this works well during low congestion but is unpredictable in more challenging environments and is thus unsuitable for mission-critical systems. The real-time distributed systems (e.g., military or avionics systems) rather operate on preconfigured networks [5, 81]; they are not ad hoc solutions and do not fit the dynamic network synchronization problem. The time-constrained promise is often considered in the low latency scheduling [21] (promising packet delivery in the time declared based on the network traffic), which does not address the problem in question.

Previous author's work [23] proposes strategies to address the requirement of time-critical synchronization. However, the surprising findings of that research require further elaboration to assess the proposed approach, especially in the valid conditions of accepted limited fault tolerance.

2.3 Practical applications

This section illustrates the use cases of time-constrained synchronization. These are a potential target of practical application for the channel access strategies discussed in this work (Section 2.5).

2.3.1 Traffic coordination system

The primary application of time-critical ad hoc synchronization is the vehicle coordination system, sometimes referred to as a *Vehicle-to-Infrastructure* (V2I) network . The generic concept behind it is a central *Control Tower* (CT) that harvests the positioning information of vehicles in the area and manages the traffic according to collected data [5, 21, 41] . The network agent set is dynamic; most communicate with the tower for the first time, requiring the CT's response as soon as possible.

An example of such a system is the *Adaptive Traffic Control System* (ATCS), which collects the traffic density of the incoming vehicles to adapt the traffic lights cycle [10, 41, 94] . This application is perfect for early adaptation of time-constrained synchronization procedures, as the synchronization failures do not entail dramatic costs, just non-optimal traffic lights cycle (which is standard for non-adaptive systems).

Another instance is the automated control tower coordinating traffic of any *Autonomous Aehicle* (AV), from drones [47] to ships [43]. This system would collect data from many independent sources, like cameras, LIDARs, and radars, to achieve the desired *Quality of Service* (QoS). Radio synchronization to collect position information or the desired route data directly from AV could be one of them [90]. Time-criticality in this case is highly exhibited, as the moving vehicle may not wait to access the communication channel with the control tower. In this mission-critical case, even milliseconds of signal delay may be consequential.

CT may act as a network orchestrator; depending on the traffic load, it may transmit the synchronization parameters to all vehicles via its dedicated channel. Beneficial in this application is the fact that the traffic size may usually be estimated, so the approximated synchronization parameters can be determined ahead of time.

2.3.2 Vehicle to Vehicle (V2V) communication

This application differs as there is no central control tower, and communication is established solely between moving vehicles. The network is decentralized, and the synchronization target would be an exchange of data between all synchronizing vehicles. Tasks realized by this network are versatile - from active cruise control through the propagation of hazardous conditions warnings to vehicle collision avoidance [10, 51].

The set of agents is constantly changing as the vehicles are moving. Thus, the ad hoc network must be constantly recreated. The synchronization strategies studied in this work could reduce the latency of information exchange while conserving energy.

2.3.3 Autonomous drone swarms

Unmanned Aerial Vehicles (UAV) may form coordinated or uncoordinated swarms to execute some collective target [16, 54]. They shall establish a communication channel, which could suffer a high latency condition due to the congestion [80]. Remediation would be time-constrained synchronization proposed in this work; individual drones may exchange their control messages to adjust trajectories, avoid crashes, and coordinate the shared mission. Military operations exposed that the maintained remote connections are fragile points of attack [5, 81]. Therefore, a periodical ad hoc synchronization could be more resilient to jamming or signal interception on the battlefield.

Again, if the swarm is purposely assembled for some operation, the number of ad hoc network agents is known; thus, the synchronization parameters may be precisely adjusted for the most reliable communication.

2.3.4 Satelitte constellations

Although currently emerging with *Starlink*, *Kuiper*, and *OneWeb* satellite deployments [35], this use case is more challenging than other presented. As deployed in orbit, satellites are distant from each other, which incurs signal delays and distortions [95]. The channel bandwidth must be adequately wide, and communication must consider the sufficient guard intervals [36].

Sufficiently numerous satellite constellations must employ some level of autonomy. They also must synchronize with neighbors to coordinate the trajectory and warn each other about the space debris. As continuous communication, as well as random channel access, are energetically costly between satellites, the ad hoc synchronization discussed in this work could be beneficial. Noteworthily, an abstraction of the communication channel used in the analysis (Definition 2.3) allows representing the communication slots by different subcarriers [37] of the channel. With this interpretation, signal delays become less important as all agents transmit in parallel; they collide only when more than one neighbor satellite selects the same subcarrier

As the satellites are strictly supervised, and their mesh density is planned – their synchronization parameters may be precisely adjusted maximizing the probabilities of synchronizations.

2.3.5 Ad hoc sensor networks

Distributed sensor networks are usually built with low-energy battery devices or even intermittent energy devices [89, 74]. Devices casually remain hibernated; upon some trigger (environmental event affecting all sensors), they wake up, sense the data, attempt to communicate it, and return to the hibernated state [98]. Communication in these networks is primarily decentralized and strictly time-critical; if the collected data is not exchanged on time, it is lost as agents hibernate until the next cycle.

Proposed synchronization strategies, in particular t-SLOTS (Section 2.5), may improve the data transmitted over the random-access channel and conserve energy.

2.3.6 IoT devices

The *Internet of Things* (IoT) device network use case is similar to the sensor network in terms of the low-energy profile, decentralization, and no continuous network connection [50]. Communication network changes are more dynamic (especially in the context of the wearable IoT) and heterogeneous (as devices of various purposes communicate together) [36] . Additionally, IoT devices are usually expected to work immediately after activation. That makes them an interesting target for the implementation of the presented synchronization strategies.

2.3.7 Massive connectivity network access points

5G network offers its own standard for time-critical services, called *Ultra Reliable Low Latency Communication* (URLLC) and *Time Sensitive Networking* (TSN) [7]. These could address some of the use cases above differently.

In the context of this work, the point of interest is elsewhere. 5G network, like every massive connectivity network, incorporates the Random Access Preamble Transmission procedure to connect devices from beyond the network. During this procedure, agents (the User Equipment) select the random preamble and transmission slots to access the random access channel [40, 87, 7].

Methods provided by this research, especially the formulas for the BT and t-Slots fault-tolerant synchronization probabilities, might be helpful in modeling and estimating the performance of the initial phase of network access. It is not a strictly time-critical synchronization scenario, but the obtained results (Section 4.4) suggest that the t-Slots synchronization triggered repetitively has appealing contention resolution properties. Therefore, the fault-tolerant t-Slots synchronization could be applied in the first stage of the massive connectivity network access procedure. This solution should be benchmarked, but it can potentially improve the present state of the art.

2.4 Formal problem statement

In order to formulate the precise research hypothesis, the problem illustrated has to be defined formally first. Since the studied model is generic and widely applicable, the definitions below are aimed at possibly being universal.

Definition 2.1 (Agent). *An* agent *is a communication network member able to transmit/receive messages.*

Definition 2.2 (Drain). *A* drain *is a special agent able to collect messages transmitted by any other agent in the network.*

While the agent is a standard actor of any potential communication, a drain is an abstract formalism introduced to unify the perception of successful transmission. A drain may be interpreted as one of the agents, as a bystander collecting messages transmitted by agents, or as a whole single-hop broadcast communication network (in the last case it is assumed that if the message is received by some agent, it may be received by every agent in the network). As stated in the previous chapter, the process of the actual network communication is a subject of physical limitations and anomalies, like hidden or exposed nodes – they are all disregarded to isolate the studied problem.

Agents transmit their messages over the slotted communication channel, which is commonly applied in wireless networks. Slotted channels are more efficient in collision reduction and energy conservation; also, they increase the overall throughput [76, 56]. The agents' communication is assumed to be constrained, so the synchronization finishes in the known, limited time, which addresses the mission-critical applications mentioned in the previous chapter. This limited window of communication is referred to as a *communication round*.

Definition 2.3 (Communication round)**.** *A* communication round *is a whole space of agents' interaction that consists of a finite number of communication slots such that if only one agent transmits in a given slot, the drain is capable of receiving the message.*

A single communication slot shall be capable of carrying the whole message of the agent. The transmitted message itself may interpreted freely – but in this work, it is considered as the *presence announcement*, *coordination*, or a *synchronization* message. This corresponds with the applications presented in the previous section, where agents sharing no initial data about each other aim to ad hoc synchronize their information.

The communication round is an abstraction agnostic to the medium being used. The wireless communication slots may represent a channel partition resulting from time-domain multiplexing, frequency-spectrum multiplexing, code-division multiplexing, or any combination of them. Notably, due to the finite number of communication slots, the communication round is constrained in all the domains above. For the model, it is only significant how many communication slots appear in the communication round and that the single agent's transmission in a slot leads to the successful delivery of its message to the drain (only the occurrence of a collision may prevent the successful message delivery).

Definition 2.4 (Successful transmission)**.** *If only one agent transmits its message in the communication slot, a drain can receive the message, and the transmission is told to be* successful*.*

Definition 2.5 (Collision)**.** *If two or more agents transmit their messages in the same slot, the* collision *occurs, and a drain can receive neither message.*

In the scope of the whole communication round, not-pre-coordinated agents individually aim to exchange their information by repeating the transmission:

Definition 2.6 (Communication success)**.** *An agent is told to have a successful communication if it has at least one successful transmission throughout the communication round.*

Definition 2.7 (Communication failure). *An agent is told to have a communication failure if it has no successful transmission throughout the communication round.*

It is assumed that a communication does not adopt a carrier sensing [31, 45]. Agents blindly transmit their messages in the slot without any collision detection mechanism; also, they are unaware of whether their transmission was successful. This is typical in ad hoc protocols, designed to be rapid, simple, and have a low energy profile. Any techniques allowing to decode a truly parallel transmission, such as the orthogonal encoding [97], shall be instead considered as an introduction of the extra slots to the communication round[1].

With the above preliminaries, the main problem is stated:

Problem 1 (Time-critical ad hoc synchronization). *There are agents $A_1, \ldots, A_k$. For every $i \in \{1, 2, \ldots, k\}$, agent A_i has to transmit its message M_i. Messages must be sent within slots $S_1, S_2, \ldots, S_n$ of a single communication round (while the whole M_i transmission fits in a single communication slot). Agents have no prior knowledge of each other nor the feedback of a successful transmission in the slot. Each agent A_i has to access the communication channel randomly, choosing on their own from $S_1, S_2, \ldots, S_n$ a subset σ_i of communication slots (transmission set of A_i) and transmit M_i repetitively in these slots.*

The problem is determining agents' slot random-access strategy that maximizes the chance of at least one successful transmission of each M_i (A_i agent successful communication*).*

This study focuses on the ad hoc synchronization[2] of a random set of agents, starting from no awareness of each other, communicating over the slotted random-access channel to exchange their messages in the constrained time. Time constraint is a consequence of bounding the communication in a single *communication round* (as per Definition 2.3).

Given the agent count k and the slot number n, the synchronization chance naturally depends on the k/n ratio. Two additional factors are relevant to achieving the synchronization objective: the slot random-access strategy and the transmission set cardinalities. Agents have to follow a symmetry-breaking

[1] Due to the ad hoc specifics of the analyzed problem, the network employing the Code Division Multiple Access [33, 77] (or the Orthogonal Frequency Division Multiplexing [8, 17]) would have to let the agents randomly select their orthogonal spreading codes (orthogonal subcarriers), effectively expanding each available slot into many, aligned with exact codes (subcarriers). The new construct remains in the model, with more slots available. Agent's transmission strategy for such a network would deal with some subtlety, as one agent would rather not select multiple slots representing different encodings at the same time. Above is, however, an edge case; in this work, slots are considered to be independent

[2] The stated problem of not-pre-coordinated channel random access is often referred to in the available literature [24, 25, 64, 93] in the context of ad hoc network *initialization* (sometimes, more precisely, *neighbor discovery*) rather than *synchronization*. In this work, the *synchronization* term is used to underline the fact that the model application is much broader. The target may be collecting the agent's coordination data by some drain, exchanging ad hoc information in the dynamic population of passing-by actors, negotiating the BTS access, and others.

randomized slot selection strategy, which choice may impact the synchronization outcome. Adjusting the agent's transmission set cardinality requires finding the right compromise between the self-promoting more frequent transmissions and the medium-preserving less frequent transmissions. Although the first increases an individual agent's chance for successful communication, it also pollutes the medium, provoking collisions and hindering the overall network synchronization objective.

The statement relating to *no prior knowledge* about other agents deserves more explanation; despite no information being exchanged between agents before the communication round, some technical aspects of the communication have to be orchestrated ahead: either pre-programmed or announced[3]. These technical aspects are: agents' clocks, common channel definition, communication round slot size and count, agents' random-access strategy with its configurable parameters. The synchronization process may be triggered by some external signal received by all agents (central base station call for synchronization; an environmental event), started according to the predefined timestamp, or initiated ad hoc through sending the first synchronization message by any participating agents. The problem's model is consistent with all of these implementation approaches, remaining widely applicable; nonetheless, the above aspects are not further considered in this work in favor of the agents' random-access strategy impact analysis. The strategy and configuration are assumed to be uniform across all agents.

2.5 Channel random access strategies

As shown in Section 2.3, the studied problem has a wide range of purposeful applications; accordingly, every performance, trustworthiness, or energy preservation enhancement is technologically significant. The apparent subject for improvement is the strategy used by agents to construct their *transmission set* (according to the problem 1). Unfortunately, agents have minimal information in the ad hoc network without collision detection. All they can do is employ any kind of random access strategy in the channel to avoid collisions. This work reviews two intuitive strategies, one being commonly used and one proposed as an improved alternative.

Section 2.1 presents widespread channel random-access methods. Most of the described tactics, like a backoff or a contention window, are designed for eventual consistency, not time-critical synchronization; they also require some collision detection or central orchestration. Nakano and Olariu proposed a dedicated algorithm for the unknown number k of agents and no collision detection in [64]. The solution, however, requires the leader selection first, which contradicts

[3] The problem model may represent a distributed system without a leader or an ad hoc network with a central orchestrator. In the first case, agents must be preconfigured at their deployment and may individually adapt their configuration. In the latter case, an orchestrator (a control tower or a base station) broadcasts protocol coordination messages over its dedicated channel.

the stated time-critical synchronization problem, where the communication is instant, and there is no space for the iterated protocols (ex., when the slots are interpreted as subcarriers in the frequency domain). Vasudevan, Adler, and Goeckel proposed the feedback-based algorithm in [93], which suffers from the same condition.

The default random-access channel synchronization strategy for ad hoc networks derives from the ALOHA protocol. It is evaluated by: Nakano and Olariu in Section 3 of [64]; Cichoń, Kutyłowski, and Zawada in [25]; Vasudevan, Adler, and Goeckel in [93]. The strategy assumes each agent transmits with a probability inversely proportional to the number of yet-unsynchronized agents (initially, the probability is $\frac{1}{k}$) and then remains silent upon the successful transmission. This simple approach is adapted to the restrictive model introduced in the previous section: agents may not rely on collision detection, so they keep sending messages with their initial probability; the initial probability is also not determined as the agent's count may not be known, though it is arbitrarily configured. In effect, the communication turns out to be a set of slot-independent *Bernoulli trials*; hence this name used in the study:

Definition 2.8 (Bernoulli trials (BT) strategy)**.** *Let an ad hoc network of k agents communicate over a finite communication round consisting of n slots and configurable fixed transmission probability p.*

Bernoulli trials strategy *is a synchronization strategy where for $i \in \{1, 2, \ldots, k\}$, $j \in \{1, 2, \ldots, n\}$ agent A_i transmits its message M_i, in a slot S_j with a probability $p \in \mathrm{Uniform}\,(0, 1)$, independently of its decision in other slots [23].*

In the following, the strategy is referenced as BT *strategy* or simply: BT. The strategy addresses Problem 1; the undeniable advantage of BT is its simplicity. It is assumed that the configured value of p is fair and is configured in the same way for all agents.

Since the number of communication round slots is known and preconfigured, the alternative channel random access strategy is proposed:

Definition 2.9 (t-Slots strategy)**.** *Let an ad hoc network of k agents communicate over a finite communication round consisting of n slots and the given fixed transmission set cardinality t.*

t-Slots *strategy is a synchronization strategy where for $i \in \{1, 2, \ldots, k\}$ agent A_i selects its transmission set, choosing randomly t out of n available communication slots, then transmits its message M_i in these t selected slots.*

The strategy will be referenced as t-Slots in this document. Similarly to BT, t-Slots strategy addresses the problem 1. The further analysis assumes that the configured cardinality t is fair (i.e. common for all agents), and that the selection of slots to build the transmission sets follows a uniform distribution.

2.6 Research hypothesis

One last technical aspect must be clarified ahead of the hypothesis statement. Two strategies introduced in the previous section appear asymptotically convergent; therefore, a scale factor is essential in their juxtaposition. The size of the communication round n is flexible: while considered possibly compact and always limited, depending on the application use case, the expected network size, and the medium available, the practical slot count may surge to an arbitrary value. The network size k is more impactful, as this is limited in all practical applications. Estimation of the *pragmatic* ad hoc network size can be based on the reference to the analytic works and the service provider's specification. It turns out that the pragmatic synchronization networks contain from just a few, up to a few thousands nodes. Contemporary protocols can address up to hundreds of agent devices connecting to a single BTS over the random-access channel during congestion, while realistically handling much less during their common operation [62, 73, 96]. Dynamic development of the military ad hoc networks and the civil Internet-of-Things elevates this threshold to the challenging order of thousands of agents synchronizing ad hoc in the local area[4] [5, 6]. With the provided background, the research-relevant network size is specified:

Definition 2.10 (Pragmatic-size ad hoc network)**.** *A synchronizing ad hoc network size is told to be* pragmatic *if n, the number of its agents, does not exceed low thousands.*

Notably, the available sources quoted above mostly focus on the final consistency promise rather than time-critical synchronization. The adequate network scale to address the time-critical use cases described in Section 2.3 varies between tens and hundreds of agents.

Now, referring to the Problem 1, the research hypothesis is stated:

Hypothesis 1. *Given the time-critical ad hoc synchronization problem, the pragmatic-size network can achieve better average synchronization results if every agent randomly selects the constant-size transmission set out of the available slots rather than separately deciding to transmit or not in each slot.*

In other words, the above hypothesis theorizes that in an average case, the t-SLOTS strategy outperforms the BT strategy in the attempt of ad hoc synchronization over the constrained random access channel. It could seem unlikely, as the only difference lies in a method of tossing slots to be included in the transmission set, and both strategies are equally simple, not using any complex computations or smart manipulations. In fact, each slot of the communication round has the same probability of being selected to the transmission set of a particular agent in both t-SLOTS and BT strategies when they are configured adjacently[5]. Although counterintuitive, the hypothesis is to be confirmed in the two following chapters dedicated to all-agent synchronization and to fault-tolerant synchronization.

[4] While the supported total number of connections is aimed for 10^6 per km^2 [69], specifications assume that only a small subset of agents attempt to synchronize with a specific BTS simultaneously.

[5] See Subsection 3.4.1 for details.

Chapter 3

All-agents synchronization

3.1 Objective

Given the formal problem definition (Problem 1), it is fundamental to define the proper measures that help to confirm or refute the stated Hypothesis 1. The *better average synchronization results* in the hypothesis naturally translates to the *higher chance of successful synchronization.* While being the fundamental one, it is not the only important aspect of ad hoc synchronization. Others are: predictability of the network behavior, impact of the misaligned agents' configuration, detectability of the malicious agents, efficient energy management, and channel pollution. In this chapter, the BT and t-SLOTS strategies are assessed and compared concerning the above.

The most accurate quality measure to verify the research hypothesis is the probability of ad hoc synchronization[1], which is individual to the applied slot random access strategy. The above requires a precise statement of what the *synchronization* means. Within this chapter, the ultimate communication success of all participating agents is analyzed, following the definition:

Definition 3.1 (All-agents synchronization)**.** *Given agents $A_1, \ldots, A_k$, the synchronization occurs if for every $i \in \{1, 2, \ldots, k\}$ message of agent A_i is transmitted in at least one collision-free slot of the communication round.*

For both strategies, the objective is to find the all-agents synchronization probability formula, parametrized with the n number of slots, the k count of participating agents, and the strategy-specific configuration. These formulas will allow for the characterization of each strategy and their juxtaposition for the parameters reflecting the pragmatic network sizes. Also, the exact algorithms of BT and t-SLOTS strategy-implementing agent, along with the results of the performed experiments, are presented in this chapter.

[1] A standalone term *synchronization* is used in this chapter to describe the event successful according to the Definition 3.1. Achieve of the *synchronization* is a distinguishing criterion itself; agents may or may not synchronize their information, while a term *successful synchronization* would be a tautology.

All experiments were conducted in the statistical analyzer program developed by the author, available open-source online [86], altogether with the statistical experiment results. The agent algorithms' (Algorithm 1, Algorithm 2) sample C-language implementation [84], as well as the Python library [85] including the implementation of the key formulas derived in this chapter (Theorem 3.1, Theorem 3.4) are open-source, available online.

3.2 Bernoulli trials strategy

Agents' channel random access strategy referenced as Bernoulli trials (BT), analyzed in this section, has been defined formally by Definition 2.8, in Section 2.5. The core analysis of this strategy, comprehending the most important outcomes: BT synchronization probability formula and the optimal configuration p value, has been published in the author's previous work [23]. This study presents and extends the obtained results, offering an improved formalization (incl. the sample space of communication events), detailed proofs, and deeper analysis.

The following subsections present the agent's BT strategy algorithm, communication events sample space, all-agents synchronization probability formula, optimal p configuration, and BT strategy properties supported by statistical experiments and calculation results.

3.2.1 Agent's algorithm

BT strategy sources from the slotted ALOHA protocol [18, 56, 71, 76] , but in the ad-hoc time-critical synchronization attempt (Problem 1) agents have no network feedback, collision detection, or the backoff mechanism implemented. The exact BT strategy agent's algorithm[2] (Algorithm 1) is likely the most straightforward approach to the ad hoc channel random access.

Algorithm 1 Bernoulli trials (BT): Transmission set selection procedure [23]

```
Given generator RNG_m returning m random bits
Given n, p
T ← ∅
for i ← 1 to n do
    r ← RNG_m()
    if r < p then                    ▷ r is interpreted a number in [0, 1]
        T ← T ∪ {i}
    end if
end for
return T                 ▷ T contains indices of the chosen transmission set
```

The required input parameters are: n – the number of slots in the communication round (required for a stop condition), p – the agent's transmission

[2] Although originally featured in [23], the algorithm embraces the general idea present in the literature and the implemented protocols described in Section 2.5.

probability in each communication slot. The returned set T contains indices of slots chosen by an agent to transmit in.

From the high-level view, the algorithm execution takes n iteration steps, performing basic operations for each slot of the communication round. However, this kind of algorithm aimed at embedded systems requires a more detailed understanding. The algorithm's computationally demanding operation is a random number generation. The number m of generated random bits in 5th line varies depending on the required precision and the representation of p. If k is a power of 2, precise drawing with optimal $p = \frac{1}{k}$ (see Theorem 3.2) requires only a few random bits. However, for $k \neq 2^i, i \in \mathbb{N}$, the number of bits rises quickly to guarantee the relevant precision when evaluating condition $r < p$. When the required precision is $\varepsilon \in [0, 1]$, RNG_m must generate $m = \left\lceil \log_2 \left(\frac{1}{\varepsilon}\right) \right\rceil$ bits each step; assuming the complexity of a pseudorandom bit generation to be $O(1)$, the overall time complexity of the BT strategy agent's algorithm is $O\left(n \log_2 \left(\frac{1}{\varepsilon}\right)\right)$. It is noteworthy that precision scarification must be done carefully, as BT is highly sensitive to the changes of p.

If the communication slot can be referenced by a restricted-size index number, the pessimistic space complexity of the presented Algorithm 1 would be $O(n)$ and on average the approximately np slot references are stored in the returned transmission set. As the iteration steps are independent, if communication slots are time-domain separated, the algorithm could be streamlined to be a decision generator, yielding *transmit*/*not-transmit* decision each slot; such an improvement reduces the space complexity to $O(1)$.

3.2.2 Sample space

Before the synchronization event probability formula is presented, the sample space must be established. This space event represents a possible combination of the agent's transmission sets when they implement the BT strategy.

Definition 3.2 (BT communication event). *Given the communication round slots $S_1, \ldots, S_n$, agents $A_1, \ldots, A_k$ and their transmission sets $\sigma_1, \ldots, \sigma_k$, where $(\forall i \in \{1, \ldots, k\})\ \sigma_i \subseteq \{S_1, \ldots, S_n\}$, the* BT *communication event is:*

$$\mathcal{E}^{\mathsf{BT}}_{\sigma_1,\ldots,\sigma_k} = (\sigma_1, \ldots, \sigma_k) \tag{3.1}$$

The sample space of BT communication events contains all the possible combinations of the agent's transmission slots.

Definition 3.3 (BT communication event sample space). *Given the communication round slots $S_1, \ldots, S_n$, agents $A_1, \ldots, A_k$ and their transmission sets $\sigma_1, \ldots, \sigma_k$, where $(\forall i \in \{1, \ldots, k\})\ \sigma_i \subseteq \{S_1, \ldots, S_n\}$, the* BT *communication event sample space is:*

$$\Omega^{\mathsf{BT}}_{n,k} = \{(\sigma_1, \ldots, \sigma_k) : (\forall i \in \{1, \ldots, k\})\, (\sigma_i \in \mathcal{P}\left(\{S_1, \ldots, S_n\}\right))\} \tag{3.2}$$

where $\mathcal{P}\left(\{S_1, \ldots, S_n\}\right)$ is a power set of communication slots.

Lemma 3.1 (Cardinality of the BT communication event sample space)**.**

$$\left|\Omega_{n,k}^{\mathsf{BT}}\right| = 2^{nk} \tag{3.3}$$

Proof. For each $i \in \{1, \ldots, k\}$ transmission set σ_i belongs to the $\mathcal{P}(\{S_1, \ldots, S_n\})$. After Fact 4:

$$|\sigma_i| = |\mathcal{P}(\{S_1, \ldots, S_n\})| = 2^n \tag{3.4}$$

Agents $A_1, \ldots, A_k$ are independent and all their individual transmission set combinations appear in the sample set $\Omega_{n,k}^{\mathsf{BT}}$. Therefore:

$$\left|\Omega_{n,k}^{\mathsf{BT}}\right| = |\sigma_i|^k = 2^{nk} \tag{3.5}$$

□

The naive method of the synchronization probability calculation would be iterating over the BT communication events and counting these representing synchronization, then comparing the number against the sample space cardinality. However, the cardinality of a sample space highlights that the naive method becomes infeasible even for small networks. Given the network of $k = 9$ agents attempting to synchronize in the communication round of $n = 30$ slots, the cardinality of the sample space is on the same order of magnitude as the estimated number of atoms in the universe[3]. This observation highlights the need for the development of a practical formula for the BT agents' synchronization probability.

3.2.3 Synchronization probability

The probability formula to be derived relates to the below event.

Definition 3.4 (BT all-agents synchronization event)**.** $\mathsf{Success}_{n,k,p}^{\mathsf{BT}}$ *denotes a synchronization of k agents implementing* BT *strategy with the transmission probability p, in the communication round of n slots:*

$$\mathcal{E}_{\sigma_1,\ldots,\sigma_k}^{\mathsf{BT}} \in \mathsf{Success}_{n,k,p}^{\mathsf{BT}} \iff (\forall i \in \{1, \ldots, k\}) \left(\sigma_i \nsubseteq \bigcup_{j \neq i} \sigma_j \right) \tag{3.6}$$

The synchronization event, in accordance to the Definition 3.1, is characteristic of at least one successful transmission per agent; this takes place when each agent's transmission set contains at least one unique slot that is not shared by any other transmission set. Families of such sets are named *dominant* (Fact 5).

The BT communication event may be illustrated as a binary matrix of agents' activity, as in Figure 3.1. Each matrix column $x_j = \langle x_{1j}, x_{2j}, \ldots, x_{kj} \rangle$ reflects all agents' activity of the single communication slot. Columns containing only one non-zero value represent slots of the successful transmission; they are marked as e_i, where i is an index of the uniquely transmitting agent ($x_j = e_i \iff x_{ij} = 1 \land \|x_j\| = 1$).

[3] The number of atoms in the universe is estimated to be circa $10^{78} - 10^{82}$ [38]. While the cardinality of BT communication event sample space $|\Omega_{n,k}^{\mathsf{BT}}| = 2^{270} \approx 10^{81}$

k transmitters	**0**	**0**	1	0	1	0	**0**	0	**0**	1	0	**1**
	0	**1**	1	0	1	0	**0**	0	**0**	1	0	**0**
	1	**0**	0	0	1	0	**1**	0	**1**	0	0	**0**

n slots

Figure 3.1: Matrix representation of an exemplary BT communication event $\mathcal{E}^{\mathsf{BT}}_{\sigma_1,\ldots,\sigma_3}$ for $k = 3$ agents and $n = 12$ slots. Columns represent the slots; rows represent the activity of agents. **1**s stand for a transmission, **0**s stands for remaining silent. Depicted event is the all-agent synchronization since every agent avoids collision at least once (A_1 in the last slot, A_2 in the 2^{nd} slot, A_3 in the 1^{st}, 7^{th}, and 9^{th} slot).

Fact 12 (BT synchronization event condition)**.**

$$\mathcal{E}^{\mathsf{BT}}_{\sigma_1,\ldots,\sigma_k} \in \mathsf{Success}^{\mathsf{BT}}_{n,k,p} \iff \big(\forall i \in \{1,2,...,k\}\big)(\exists j \in \{1,2,...,n\})(\overline{x}_j = \overline{e}_i)\big) \tag{3.7}$$

The probability of all-agents synchronization is derived directly from the Fact 12. The proof was originally published in the author's previous work [23]; a more detailed version is presented below.

Theorem 3.1 (BT all-agents synchronization probability)**.** *For any $n, k \in \mathbb{N}_+$, and $p \in [0,1]$ the synchronization probability in Bernoulli trials strategy equals:*

$$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] = \sum_{a=0}^{k} (-1)^a \binom{k}{a} \big(1 - az\big)^n \tag{3.8}$$

where $z = p(1-p)^{k-1}$.

Proof. The $\mathsf{Success}^{\mathsf{BT}}_{n,k,p}$ is the BT synchronization event that occurs for agents transmitting with probability p each slot. Therefore:

$$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] = \Pr\left[(\forall i \in \{1,2,...,k\})(\exists j \in \{1,2,...,n\})(\overline{x}_j = \overline{e}_i)\right] \tag{3.9}$$

Given the opposite event and the Probability Complement Rule, the above equation is equivalent to:

$$\begin{aligned}\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] =& 1 - \Pr\left[\neg(\forall i \in \{1,2,...,k\})(\exists j \in \{1,2,...,n\})(\overline{x}_j = \overline{e}_i)\right] \\ =& 1 - \Pr\left[(\exists i \in \{1,2,...,k\})(\forall j \in \{1,2,...,n\})(\overline{x}_j \neq \overline{e}_i)\right]\end{aligned} \tag{3.10}$$

The condition of the opposite event is met when at least one agent A_i, $i \in \{1, 2, \ldots, k\}$ had no successful communication. However, there may also be more such agents: 2 out of k, 3 out of k, ..., k out of k. To appropriately count the probability of these overlapping events, the probabilistic version of

Inclusion-Exclusion Principle (Fact 6) is applied.

$$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] = 1-\sum_{a=1}^{k}(-1)^{a+1}\sum_{\substack{T\subseteq\{1,\ldots,k\},\\|T|=a}}\Pr\left[\bigwedge_{t\in T}(\forall j\in\{1,2,...,n\})(\overline{x}_j\neq\overline{e}_t)\right] \tag{3.11}$$

Note that for each $j\in\{1,2,\ldots,n\}$ the probability of $\overline{x}_j\neq\overline{e}_t$ is the same, and the events depicting agent's activity in different slots are independent of each other. Also, probabilities of the success transmission of different agents are disjoint; they're also identical since they transmit with the same probability p.

$$\begin{aligned}
\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] =&1-\sum_{a=1}^{k}(-1)^{a+1}\sum_{\substack{T\subseteq\{1,\ldots,k\},\\|T|=a}}\left(\Pr\left[\bigwedge_{t\in T}(\overline{x}_1\neq\overline{e}_t)\right]\right)^n\\
=&1-\sum_{a=1}^{k}(-1)^{a+1}\sum_{\substack{T\subseteq\{1,\ldots,k\},\\|T|=a}}\left(1-\Pr\left[\bigvee_{t\in T}(\overline{x}_1=\overline{e}_t)\right]\right)^n\\
=&1-\sum_{a=1}^{k}(-1)^{a+1}\binom{k}{a}(1-a\cdot\Pr[\overline{x}_1=\overline{e}_1])^n
\end{aligned} \tag{3.12}$$

The particular probability of a success transmission $\Pr[\overline{x}_1=\overline{e}_1]$ is static and equals $p(1-p)^{k-1}$, as it occurs when the single agent transmits and all other does not (single transmission success in Bernoulli trials). From this point, just a few transformations of the summation range lead to the final form of the formula.

$$\begin{aligned}
\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] =&1-\sum_{a=1}^{k}(-1)^{a+1}\binom{k}{a}\left(1-ap(1-p)^{k-1}\right)^n\\
=&1-(-1)\sum_{a=1}^{k}(-1)^{a}\binom{k}{a}\left(1-ap(1-p)^{k-1}\right)^n\\
=&(-1)^0\binom{k}{0}+\sum_{a=1}^{k}(-1)^{a}\binom{k}{a}\left(1-ap(1-p)^{k-1}\right)^n\\
=&\sum_{a=0}^{k}(-1)^{a}\binom{k}{a}\left(1-ap(1-p)^{k-1}\right)^n\\
=&\sum_{a=0}^{k}(-1)^{a}\binom{k}{a}\left(1-az\right)^n,
\end{aligned}$$

where $z=p(1-p)^{k-1}$. □

The formula derived above makes it feasible to calculate the exact probability of BT synchronization event under a given network setup n, k, p, even for high n and k.

3.2.4 Optimal probability choice

The formula of Theorem 3.1 also allows to answer the following question. What is the optimal agents' p value to maximize the synchronization chance?

Driven by the idea that when $p = \frac{1}{k}$ the expected number of agents transmitting in the slot is 1 (which reflects the successful transmission), Theorem 3.2 below proves that this is indeed the optimal value of p. The probability formula and the auxiliary parameter z are analyzed as continuous functions. First, two auxiliary lemmas are proved.

Lemma 3.2. *The function*

$$f(x) = x(1-x)^{k-1}, \tag{3.13}$$

defined for $x \in [0,1]$ has range $\left[0, \frac{1}{k}\right]$ and reaches its maximum for $x = \frac{1}{k}$.

Proof.

$$\frac{d}{dx} f(x) = (1-x)^{k-2}(1-kx) \tag{3.14}$$

Noticeably, $\frac{d}{dx} f(x) > 0$ for $x \in \left[0, \frac{1}{k}\right)$. Also, $\frac{d}{dx} f(x) < 0$ for $x \in \left(\frac{1}{k}, 1\right)$. For $x = \frac{1}{k}$ and $x = 1$, the value of derivative is zero; thus, $f(x)$ reaches its maximum for $x = \frac{1}{k}$, as this is the only point where $\frac{d}{dx} f(x)$ changes its sign from positive to negative on the interval $[0,1]$.

For $x \in [0,1]$, $\min f(x) = 0$ when $x = 0$ and $x = 1$; $\max f(x) = f\left(\frac{1}{k}\right) \leqslant \frac{1}{k}$, because for $k \geqslant 1$:

$$f\left(\frac{1}{k}\right) = \frac{1}{k}\left(1 - \frac{1}{k}\right)^{k-1} \quad \wedge \quad \left(1 - \frac{1}{k}\right)^{k-1} \leqslant 1 \tag{3.15}$$

□

Lemma 3.3. *The function*

$$g_{n,k}(x) = \sum_{a=0}^{k} (-1)^a \binom{k}{a} \left(1 - ax\right)^n, \tag{3.16}$$

is increasing on the interval $(0, \frac{1}{k}]$, for $n, k \in \mathbb{N}_+$.

Proof. This proof follows the mathematical induction over k. For $k = 1$, $g_{n,1}(x) = 1 - (1-x)^n$, and $\frac{d}{dx} g_{n,1}(x) = n(1-x)^{n-1} > 0$ for $x \in [0,1]$.

Then, assume for some k function $g_{n,k}(x)$ increases in the interval $x \in \left(0, \frac{1}{k}\right]$; with this assumption it can be shown that $g_{n,k+1}(z)$ increases in the interval $x \in \left(0, \frac{1}{k+1}\right]$.

Calculate the $\frac{d}{dx}g_{n,k+1}(x)$. The first operation employs the Chain Rule; after a few simplifications, the summation range can be adjusted

$$\begin{aligned}
\frac{d}{dx}g_{n,k+1}(x) &= \sum_{a=0}^{k+1}(-1)^a\binom{k+1}{a}\frac{d}{dx}(1-ax)^n \\
&= \sum_{a=0}^{k+1}(-1)^a\binom{k+1}{a}n(1-ax)^{n-1}(-a) \\
&= n\sum_{a=1}^{k+1}(-1)^{a-1}\binom{k+1}{a}a(1-ax)^{n-1} \\
&= n\sum_{a=0}^{k}(-1)^a\binom{k+1}{a+1}(a+1)\left(1-(a+1)x\right)^{n-1}
\end{aligned} \tag{3.17}$$

Noting that $\binom{k+1}{a+1} = \binom{k}{a}\frac{k+1}{a+1}$, the relation between $\frac{d}{dx}g_{n,k+1}$ and $g_{n,k}$ is exposed

$$\begin{aligned}
\frac{d}{dx}g_{n,k+1}(x) &= n\sum_{a=0}^{k}(-1)^a\binom{k}{a}(k+1)(1-x-ax)^{n-1} \\
&= n(k+1)(1-x)^{n-1}\sum_{a=0}^{k}(-1)^a\binom{k}{a}\left(1-a\frac{x}{1-x}\right)^{n-1} \\
&= n(k+1)(1-x)^{n-1}g_{n,k}\left(\frac{x}{1-x}\right)
\end{aligned} \tag{3.18}$$

Three leading terms of the product are positive for $n,k \in \mathbb{N}_+$ and $x \in (0, \frac{1}{k+1}]$. The sign of the derivate depends ultimately on the last term: $g_{n,k}\left(\frac{x}{1-x}\right)$.

Observing that:

$$x \leqslant \frac{1}{k+1} \iff xk \leqslant 1-x \iff \frac{x}{1-x} \leqslant \frac{1}{k} \tag{3.19}$$

It is guaranteed that for $x \in \left(0, \frac{1}{k+1}\right]$ that $0 < \frac{x}{1-x} \leqslant \frac{1}{k}$. Since the function $g_{n,k}\left(\frac{x}{1-x}\right)$ is increasing on $\left(0, \frac{1}{k}\right]$ by the inductive assumption, and its leftmost value is zero for $n,k \in \mathbb{N}_+$

$$g_{n,k}(0) = \sum_{a=0}^{k}\binom{k}{a}(-1)^a = \sum_{a=0}^{k}\binom{k}{a}1^{k-a}(-1)^a = (1-1)^k = 0 \tag{3.20}$$

The value of $g_{n,k}\left(\frac{x}{1-x}\right)$ is positive; just like the whole $\frac{d}{dx}g_{n,k+1}(x)$. Therefore, the induction step is proven, and the function $g_{n,k}(x)$ increases in the interval $x \in (0, \frac{1}{k}]$, for $n,k \in \mathbb{N}_+$ □

The above lemmas are used to prove the optimal value p of the agent's configuration.

Theorem 3.2. *For any $n \in \mathbb{N}_+$ and $k \geqslant 2$:*

$$\arg\max_{p\in[0,1]} \Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] = \frac{1}{k} \tag{3.21}$$

Proof. Two introduced functions, f and g, represent respectively: parameter z from Theorem 3.1, and $\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right]$. Consequently:

$$\arg\max_{p\in[0,1]} \Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right] \equiv \arg\max_{p\in[0,1]} g(f(p)) \tag{3.22}$$

As shown in Lemma 3.2, for $0 \leqslant p \leqslant 1$ values of function $f(p)$ belong to the range between 0 and $\frac{1}{k}$, reaching the top value for $p = \frac{1}{k}$. After Lemma 3.3, function $g(x)$ is increasing between 0 and $\frac{1}{k}$; therefore, $g(f(p))$ reaches its maximum for $p = \frac{1}{k}$. □

Notably, the optimal configuration is independent of n at all. The agent's transmission probability must be adjusted only to the anticipated number of agents accessing the channel.

3.2.5 Characteristics and statistical experiments

Table 3.1: The optimal BT strategy synchronization probability values in the setup of $n = 30$ slots and $k \in \{4, 6, 8, 10\}$ agents.

p	$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{30,4,p}\right]$	$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{30,6,p}\right]$	$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{30,8,p}\right]$	$\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{30,10,p}\right]$
0.1	0.63776568	0.32612064	0.10069249	**0.01591415**
0.125	0.73372351	0.39156136	**0.11154706**	0.01322441
0.166	0.81784803	**0.42822021**	0.09054602	0.00510975
0.25	**0.86364095**	0.32969834	0.01939766	0.00008824

An open-source Python library developed by author [85] includes the implementation of Theorem 3.1 formula. Statistical experiments were conducted to demonstrate the proven formula empirically [86]. Sample experiment data, accompanied by the calculation results, is presented in Figure 3.2. The figure displays the critical characteristics of $\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right]$.

Firstly, regardless of the communicating agent's number k, the function representing BT all-agents synchronization probability has a single maximum point. In Figure 3.2, it is clearly visible how $\arg\max p$ value varies for different k's, and always equals $\frac{1}{k}$ following the principle from Theorem 3.2. Exact maximum values of BT all-agents synchronization probability are bolded in Table 3.1; the table data corresponds to the setup presented in Figure 3.2.

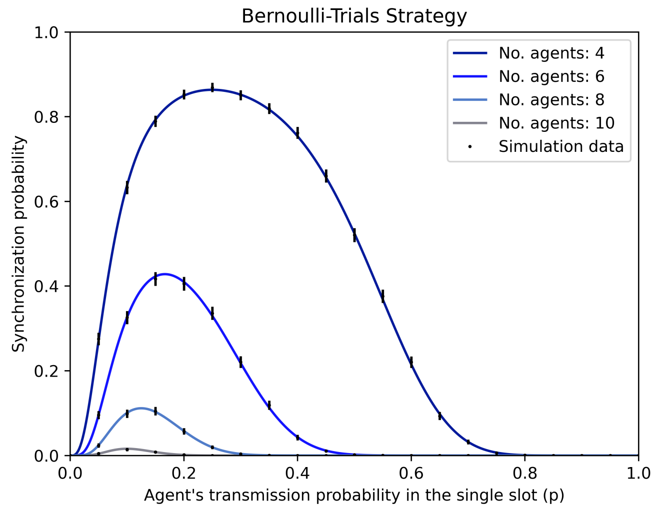


Figure 3.2: The BT strategy synchronization probability (Theorem 3.1) as a function of the single-slot agent transmission probability p. Results for $k \in \{4, 6, 8, 10\}$ agents synchronizing in the communication round of $n = 30$ slots. Vertical ticks represent the statistical test data, with a margin of error (10,000 tests per sample; $MOE_{0.999}$ [53]).

Secondly, the figure shows how quickly the achievable probability of all-agent synchronization plummets when the number of agents k increases. In the 30-slot communication round, four agents implementing the BT strategy have almost a 90% chance of successful synchronization when they are configured optimally. The chance rapidly decreases to less than 43% for six communicating agents, 11% for eight agents, and just below 1.6% for ten agents. Any protocol based on the BT strategy all-agents synchronization in the single communication round remains highly vulnerable to network congestion, quickly losing any practical synchronization chance when the number of communicating agents rises.

Finally, the shape of plots in Figure 3.2 – they are concave and asymmetric, with the left arm sharper than the right one[4]. For any presented k, $\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right]$ values rapidly decay as the p deviates from its optimum value. The slope smooths out as the function values approach zero, finally always converging to zero at both ends of the domain. The asymmetry is also present in the range of near-zero function values. On the left boundary of the domain, the near-zero plateau is observable only for a higher k number of communicating agents. Contrary, the right-hand part of the plot is at least partially flat for any $k > 2$; in the presented setup, when $n = 30$ and $k = 6$, synchronization chance

[4] This holds true for all $k > 2$. However, when $k = 2$, the optimal $p = \frac{1}{2}$, and the $\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{n,k,p}\right]$ function of p remains perfectly symmetric.

is already zero for any $p > \frac{1}{2}$. These observations highlight how particularly fragile against any imperfect configuration of p the BT strategy is.

3.3 tSlots strategy

This section provides a focused analysis of the t-SLOTS channel random access strategy in the all-agent synchronization problem. The strategy itself has been formally stated by Definition 2.9 in Section 2.5. The idea behind this is to eliminate the BT's variance conditions and improve the fairness of channel access. As for the BT strategy, the pivotal results of t-SLOTS performance have been published in the author's previous study [23], and the analysis is extended and more formal.

Analogously to the BT-dedicated section, the following subsections embrace the agent's t-SLOTS strategy algorithm, t-SLOTS communication events sample space, all-agents synchronization probability formula, optimal p configuration, and t-SLOTS strategy characteristics along with results of the calculation and statistical experiments.

3.3.1 Agent's algorithm

Each agent of the t-SLOTS strategy must select a random transmission set of cardinality t independently. The algorithm for selecting random, fixed-size subsets by accumulating the selected elements at the end of the collection with swapping was introduced in [29]. This method is leveraged in the implementation of t-SLOTS agent proposed in the author's previous work [23].

Algorithm 2 t-SLOTS: Basic transmission set selection procedure [23]

1: Given generator RNG_m returning m random bits
2: Given n, t
3: Initialize array $T[1:n]$ with $T[i] = i$
4: **for** $i \leftarrow n$ down to $n - t + 1$ **do**
5: $\quad r \leftarrow \mathsf{RNG}_m()$ ▷ r interpreted as a number $\in [0, 1]$
6: $\quad s \leftarrow \lceil ir \rceil$ ▷ $s \in \{1, \ldots, i\}$
7: $\quad$ Swap $T[s]$ and $T[i]$
8: **end for**
9: **return** $T[n - t + 1 : n]$ ▷ last t elements are the indices of the chosen slots

The required input parameters are: n – the number of slots in the communication round (required for a stop condition), t – the cardinality of the agent's transmission set. The returned collection T contains indices of the transmission set slots. The above applies to all the algorithms presented in this section.

Algorithm 2's time complexity is affected by the n-length table initialization in line 3. When n is significantly larger than t, a lazy algorithm that skips the initialization part may be preferred. The proposed Algorithm 3 offers better time and space complexity, since it only makes t iterations with the basic operations.

Algorithm 3 t-SLOTS: O(t) transmission set selection procedure

1: Given generator RNG_m returning m random bits
2: Given n, t
3: Given empty set T, empty map U ▷ $T \leftarrow \emptyset$
4: **for** $i \leftarrow n$ down to $n - t + 1$ **do**
5: $\quad r \leftarrow \mathsf{RNG}_m()$ ▷ r interpreted as a number $\in [0, 1]$
6: $\quad s \leftarrow \lceil ir \rceil$ ▷ $s \in \{1, \ldots, i\}$
7: $\quad$ **if** $s \in U$ **then**
8: $\qquad T \leftarrow T \cup \{U[s]\}$
9: $\quad$ **else**
10: $\qquad T \leftarrow T \cup \{s\}$
11: $\quad$ **end if**
12: $\quad U[s] \leftarrow i$
13: **end for**
14: **return** T ▷ T contains indices of the chosen slots

Aside from collection operations, the computationally expensive (thus, also energetically expensive) part of each iteration step of both algorithms is the random bits generation. As the algorithms are designed for embedded systems, a deeper look is required into this part of the complexity, which is identical for both procedures. The number of random bits generated in each iteration is limited by $\lceil \log_2 n \rceil$, as they are used to determine at most n distinct numbers in line 6 of both algorithms. The remaining operations of swapping in the array as well as lookup, reading, and writing in the map are all $O(1)$ [26]. Therefore, assuming that the single RNG bit generation operation is also constant-time, in the context of that operation, the Algorithm 2's time complexity is $O(n + t \log_2 n)$, and the Algorithm 3's time complexity is $O(t \log_2 n)$.

One more algorithm, based on a substantially different approach, is worth presenting. Since the anticipated realization of the abstract agent is the sensor, cell device, or autonomous vehicle, the target platform would be a dedicated embedded system with a microcontroller. Such remote device microcontrollers often feature a dedicated security unit [65, 83] providing optimized extra-fast security operations. Algorithm 4 utilizes the Format Preserving Encryption (FPE) [12, 66] along with the key generator.

Despite the extra key K generation in the beginning and the encryption operation during each iteration, Algorithm 4 may be highly efficient if it can take advantage of the built-in FPE module or the controller-optimized security library. The overall time complexity significantly depends on the complexity of the encryption process in line 6. With an assumption that the Enc_K is linear with respect to the number of the output bits, Algorithm 4's time complexity is $O(t \log_2 n)$.

Algorithm 4 also reveals other interesting advantages versus the two pre-

Algorithm 4 t-SLOTS: Transmission set selection procedure using FPE [23]

```
Given FPE encryption procedure Enc, where Enc_K : {1, ..., n} → {1, ..., n}
Given n, t
Generate a random key K
T ← ∅
for i ← 1 to t do
    T ← T ∪ Enc_K(i)        ▷ Enc_K(i) ∈ {1, ..., n} ∧ i ≠ j ⇒ Enc_K(i) ≠ Enc_K(j)
end for
return T                    ▷ T contains indices of the chosen slots
```

ceding ones. The security module can offer a better randomness source for K generation (and tossing in consequence). Usage of the encryption may be an argument for protocol fairness; for instance, when the synchronization aims to select the leader by nominating the agent who made the first successful message transmission – a random K value may legitimate that the agent adheres to the protocol, instead of cheating by choosing early slots. The algorithm does not use key K for any secret encryption; therefore, the key can be revealed and used by the network also to detect a malicious agent's activity (ex., the active jammer pretending to follow the protocol) or even be used by a drain to reconstruct the network activity in an attempt of the post-communication interference cancellation.

The space complexity of the proposed algorithms are respectively: $O(n)$ for Algorithm 2 (to store the n-indices table), $O(t)$ for Algorithm 3, $O(t)$ for Algorithm 4 (assuming here that the encryption does not require any abundant data structures).

3.3.2 Sample space

In order to formalize the sample space, the t-SLOTS communication event is defined.

Definition 3.5 (t-SLOTS communication event)**.** *Given the communication round slots $S_1, \ldots, S_n$, agents $A_1, \ldots, A_k$ with a common configuration $1 \leqslant t \leqslant n$, and their transmission sets $\sigma_1, \ldots, \sigma_k$, where:*

$$(\forall i \in \{1, \ldots, k\})\,(\sigma_i \subseteq \{S_1, \ldots, S_n\} \wedge |\sigma_i| = t) \tag{3.23}$$

the t-SLOTS communication event is:

$$\mathcal{E}^{\mathsf{tSlots}}_{\sigma_1,\ldots,\sigma_k} = (\sigma_1, \ldots, \sigma_k) \tag{3.24}$$

The above event represents any possible arrangements of k agents' t out of total n slots selections. Contrary to the BT, agent's configuration affects the definition of t-SLOTS communication event, effectively cutting off significant number of arrangements. The sample space of communication events is stated for agents adopting the t-SLOTS strategy.

Definition 3.6 (t-Slots communication event sample space)**.** *Given the communication round slots $S_1, \ldots, S_n$, agents $A_1, \ldots, A_k$ with a common configuration $1 \leqslant t \leqslant n$, and their transmission sets $\sigma_1, \ldots, \sigma_k$, where equation 3.23 holds the t-Slots communication event sample space is:*

$$\Omega^{\mathsf{tSlots}}_{n,k,t} = \{(\sigma_1, \ldots, \sigma_k) : (\forall i \in \{1, \ldots, k\}) (\sigma_i \in \mathcal{P}(\{S_1, \ldots, S_n\}) \land |\sigma_i| = t)\} \tag{3.25}$$

where $\mathcal{P}(\{S_1, \ldots, S_n\})$ is a power set of communication slots.

By definition $\Omega^{\mathsf{tSlots}}_{n,k,t}$ is a subset of $\Omega^{\mathsf{BT}}_{n,k}$. Its cardinality derives directly from the number of possible transmission set selection combinations.

Lemma 3.4 (Cardinality of the t-Slots communication event sample space)**.**

$$\left|\Omega^{\mathsf{tSlots}}_{n,k,t}\right| = \binom{n}{t}^k \tag{3.26}$$

Proof. There are $\binom{n}{t}$ combinations of the selection of t out of total n slots into the single transmission set σ_i, $i \in \{1, \ldots, k\}$. Transmission set selections are independent between k agents, resulting $\left|\Omega^{\mathsf{tSlots}}_{n,k,t}\right| = \binom{n}{t}^k$. □

Albeit the t-Slots communication sample space has lesser cardinality than its BT counterpart, their asymptotics are still comparable. Cardinality of $\Omega^{\mathsf{tSlots}}_{n,k,t}$ grows rapidly, and for the sample setup of $n = 64$ slots, $k = 8$ agents, $t = 8$ selections per agent, its value (1.47×10^{77}) has already an extreme order of magnitude, close to the one mentioned in Section 3.2.2. Iterative counting of synchronization events is infeasible, even for relatively small network sizes.

3.3.3 Synchronization probability, 2 agents

Aiming to derive a formula for synchronization probability, the t-Slots all-agents synchronization is defined in accordance to Definition 3.1.

Definition 3.7 (t-Slots all-agents synchronization)**.** $\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}$ *denotes a synchronization of k agents implementing t-Slots strategy with transmission set size t, in the communication round of n slots:*

$$\mathcal{E}^{\mathsf{tSlots}}_{\sigma_1,\ldots,\sigma_k} \in \mathsf{Success}^{\mathsf{tSlots}}_{n,k,t} \iff (\forall i \in \{1, \ldots, k\}) \left(\sigma_i \nsubseteq \bigcup_{j \neq i} \sigma_j\right) \tag{3.27}$$

The observation that two agents must match their transmission sets to fail, leads to the first synchronization probability formula. The below theorem addresses the special case of two synchronizing agents.

Theorem 3.3 (t-Slots all-agents synchronization probability, 2 agents)**.** *For $k = 2$ agents, the synchronization probability of t-Slots agents is:*

$$\Pr\left[\mathsf{Success}_{n,2,t}\right] = 1 - \frac{1}{\binom{n}{t}} \tag{3.28}$$

Proof. The synchronization fails if agents' transmission sets σ_1, σ_2 are exactly the same. Given the $\binom{n}{t}$ transmission set combinations, the chance of convergent choice is $\binom{n}{t}^{-1}$. This gives $\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{n,2,t}\right] = 1 - \binom{n}{t}^{-1}$. □

When $k = 2$, the $\Pr[\mathsf{Success}_{n,2,t}] = \frac{1}{2}$ for the minimal reasonable setup of $n = 2$ slots and $t = 1$ selections; as the n rises, the probability approaches 1. The optimal $t = \frac{n}{2}$ and the probability declines symmetrically when t diverges from this value.

3.3.4 Synchronization probability, k agents

Unfortunately, when $t > 2$ calculation gets significantly more complicated compared to Theorem 3.3. Transmissions in different slots are not independent; as every transmission set has a certain cardinality, arrangements of all slots have to be analyzed together.

The approach could be partitioning a communication round into subsets of the successful transmission slots $\mathcal{S}_{n,k,t}$ (single transmission) and the failed transmission slots $\mathcal{F}_{n,k,t}$ (zero or multiple transmissions). However, it is not trivial to group the feasible transmission patterns and count the cardinality of these mutually-dependent sets. The more agent-centric approach is necessary to find a probability of all-agents synchronization [23]; the formula along with its detailed proof is provided.

Theorem 3.4 (t-Slots all-agents synchronization probability)**.** *For $n, k, t \in \mathbb{N}_+$, the probability of communication success for the t-Slots strategy is:*

$$\Pr[\mathsf{Success}_{n,k,t}] = \frac{G_{n,k,t}}{\binom{n}{t}^k} \tag{3.29}$$

Where:

$$G_{n,k,t} = \sum_{\overline{\alpha} \in C_{k,t}} \binom{n}{\|\overline{\alpha}\|} \binom{\|\overline{\alpha}\|}{\alpha_1, ..., \alpha_k} (-1)^{k+\|\overline{\alpha}\|} \prod_{i=1}^{k} \binom{n - \|\overline{\alpha}\|}{t - \alpha_i} \tag{3.30}$$

for:

$$C_{k,t} = \big\{ \langle \alpha_1, ..., \alpha_k \rangle : (\forall i \in \{1, ..., k\}) \; (1 \leqslant \alpha_i \leqslant t \;\wedge\; \alpha_i \in \mathbb{N}) \big\}.$$

Proof. The denominator in the main formula represents the cardinality of the sample space and comes directly from Lemma 3.4.

The challenging part is finding the $G_{n,k,t}$, which stands for the cardinality of synchronization events set in the setup of k agents, transmitting in t out of total n communication slots. The precise conditions determining such events are stated in Definition 3.7. For each $i \in \{1, 2, \ldots, k\}$:

1. each transmission set σ_i contains exactly t elements;

2. the family of transmission sets is dominant [4], i.e. each transmission set σ_i contains at least one unique slot, not contained by any other σ_j, $j \neq i$.

Determining the exact number of events matching the above criteria requires the application of algebraic and combinatoric apparatus; thus, the right form of representation is preliminarily introduced.

Each slot S_j is considered separately as a agents' activity vector $\overline{x}$, where the single x_i, represents A_i's activity for $i \in \{1, 2, \ldots, k\}$. S_j is then represented by a multinomial ${x_1}^{\alpha_1} {x_2}^{\alpha_2} \ldots {x_k}^{\alpha_k}$, where $\alpha_i = 1$ for A_i transmitting their message and $\alpha_i = 0$ for A_i remaining silent. There are 2^k possible combinations of such multinomial, where:

0. no agent's transmission is expressed by 1 (or: ${x_1}^0 {x_2}^0 \ldots {x_k}^0$);
1. k possible successful transmissions of the single agent are: x_1, x_2, …, x_k;
2. $\binom{k}{2}$ possible transmissions of 2 agents are expressed by $x_i x_j$, $1 \leqslant i < j \leqslant k$;

   ⋮

k. all agent's simultaneous transmission is $x_1 x_2 \ldots x_k$ term.

The alternative of all possible combinations is expressed with a sum over these combinations:

$$1 + (x_1 + ... + x_k) + ... + x_1 x_2 ... x_k = \prod_{i=1}^{k} (1 + x_i) \tag{3.31}$$

Notably, all the individual variables are special, demarcating a desirable successful transmission; to highlight these the elements, sum is modified:

$$\prod_{i=1}^{k} (1 + x_i) - \sum_{i=1}^{k} x_i + \sum_{i=1}^{k} x_i y_i \tag{3.32}$$

The same possible combinations may occur in every communication slot independently. Consequently, the representation of the n-slot communication round $g_{n,k}(\overline{x}, \overline{y})$ is defined:

$$g_{n,k}(\overline{x}, \overline{y}) = \left(\prod_{i=1}^{k} (1 + x_i) - \sum_{i=1}^{k} x_i + \sum_{i=1}^{k} x_i y_i \right)^n \tag{3.33}$$

Observe, that the $g_{n,k}(\overline{x}, \overline{y})$ – when expanded – is a multinomial of terms being the form of: $c \;\; {x_1}^{\alpha_1} {x_2}^{\alpha_2} \ldots {x_k}^{\alpha_k} \;\; {y_1}^{\gamma_1} {y_2}^{\gamma_2} \ldots {y_k}^{\gamma_k}$. Each $\alpha_i, \gamma_i \in \{1, 2, \ldots, n\}$, while c is an individual coefficient. In the above multinomial ${x_i}^{\alpha_i}$ stands for agent A_i transmitting α_i times as well as ${y_i}^{\gamma_i}$ stands for agent A_i having γ_i individual (successful) transmissions.

Every term either represents the possible t-SLOTS communication event $\mathcal{E}^{\mathsf{tSlots}}_{\sigma_1,\ldots,\sigma_k}$ (Definition 3.5) or not. Also, every term either signifies the t-SLOTS all-agent synchronization event $\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}$ (Definition 3.7) or not. The conditions determining above are respectively:

1. $(\forall i \in \{1,2,\ldots,k\})(\alpha_i = t)$, that ensures the term represents $\mathcal{E}^{\mathsf{tSlots}}_{\sigma_1,\ldots,\sigma_k}$;
2. $(\forall i \in \{1,2,\ldots,k\})(\gamma_i > 0)$, that ensures the term represents $\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}$.

The coefficient c denotes the number of the actual distinct events represented by each term. Therefore, to find $G_{n,k,t}$ (the number of distinct synchronization events), coefficient c values must be summarized from all the terms matching the above conditions. These terms are selected by two consecutive extraction operations (introduced with Fact 8). Firstly, the application of $[{x_1}^t \ldots {x_k}^t]$ extracts only the terms representing actual t-SLOTS communication events, providing every agent transmit exactly t times. Secondly, the terms representing synchronization events are extracted by application of $[y_1^{\gamma_1} \ldots y_k^{\gamma_k}, \gamma_1, \ldots, \gamma_k \in \mathbb{N}_+]$ (in a compact form: $\left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \succeq 1\right]$[5]), providing at least one successful transmission per agent.

According to the strategy sketched above, initially $g_{n,k}(\overline{x},\overline{y})$ is transformed:

$$\begin{aligned} g_{n,k}(\overline{x},\overline{y}) &= \left(\prod_{i=1}^{k}(1+x_i) - \sum_{i=1}^{k} x_i + \sum_{i=1}^{k} x_i y_i\right)^n \\ &= \left(\prod_{i=1}^{k}(1+x_i) + \sum_{i=1}^{k} x_i(y_i-1)\right)^n \\ &= \sum_{a=0}^{n}\left(\binom{n}{a}\left(\sum_{i=1}^{k} x_i(y_i-1)\right)^a \left(\prod_{i=1}^{k}(1+x_i)\right)^{n-a}\right) \\ &= \sum_{a=0}^{n}\left(\binom{n}{a}\mathbf{M}\mathbf{F}\right) \end{aligned} \tag{3.34}$$

$\mathbf{M}$ is transformed further following Multinomial Theorem (Fact 2):

$$\mathbf{M} = \left(\sum_{i=1}^{k} x_i(y_i-1)\right)^a = \sum_{\substack{\|\overline{\alpha}\|=a \\ \overline{\alpha} \succeq 0}}\left(\binom{a}{\overline{\alpha}}\overline{x}^{\overline{\alpha}}\prod_{i=1}^{k}(y_i-1)^{\alpha_i}\right) \tag{3.35}$$

while $\mathbf{F}$ is transformed following the Fact 9:

$$\mathbf{F} = \left(\prod_{i=1}^{k}(1+x_i)\right)^{n-a} = \sum_{\substack{\overline{\beta} \preceq n-a \\ \overline{\beta} \succeq 0}}\left(\overline{x}^{\overline{\beta}}\prod_{i=1}^{k}\binom{n-a}{\beta_i}\right) \tag{3.36}$$

Combined together, $\mathbf{M}\cdot\mathbf{F}$ gives:

$$\mathbf{M}\cdot\mathbf{F} = \sum_{\substack{\|\overline{\alpha}\|=a \\ \overline{\alpha} \succeq 0}}\ \sum_{\substack{\overline{\beta} \preceq n-a \\ \overline{\beta} \succeq 0}}\left(\binom{a}{\overline{\alpha}}\overline{x}^{\overline{\alpha}+\overline{\beta}}\prod_{i=1}^{k}\binom{n-a}{\beta_i}(y_i-1)^{\alpha_i}\right). \tag{3.37}$$

[5] See Section 1.4 for the notation details and the "$\succeq$" operator definition.

Application of the first extraction $[{x_1}^t ... {x_k}^t]$ to $g_{n,k}(\overline{x},\overline{y})$ retains only the sum elements meeting $\alpha_i + \beta_i = t$ criterion. For each $i \in \{1, 2, \ldots, k\}$ this consequences with $\alpha_i \leqslant t$ and $\beta_i = t - \alpha_i$. Effectively:

$$\left[{x_1}^t ... {x_k}^t\right] g_{n,k}(\overline{x},\overline{y}) = \sum_{a=0}^{n} \left(\binom{n}{a} \sum_{\substack{\|\overline{\alpha}\|=a \\ 0\preceq\overline{\alpha}\preceq t}} \left(\binom{a}{\overline{\alpha}} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} (y_i - 1)^{\alpha_i} \right) \right) \tag{3.38}$$

Proceeding, with application of $\left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \succeq 1\right]$, the $G_{n,k,t}$ formula is obtained:

$$\begin{aligned} G_{n,k,t} = & \left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \succeq 1\right] \left[{x_1}^t ... {x_k}^t\right] g_{n,k}(\overline{x},\overline{y}) \\ = & \left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \succeq 1\right] \sum_{a=0}^{n} \left(\binom{n}{a} \sum_{\substack{\|\overline{\alpha}\|=a \\ 0\preceq\overline{\alpha}\preceq t}} \left(\binom{a}{\overline{\alpha}} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} (y_i - 1)^{\alpha_i} \right) \right) \\ = & \sum_{a=0}^{n} \left(\binom{n}{a} \sum_{\substack{\|\overline{\alpha}\|=a \\ 1\preceq\overline{\alpha}\preceq t}} \left(\binom{a}{\overline{\alpha}} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} \left(\left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \succeq 1\right] \prod_{i=1}^{k} (y_i - 1)^{\alpha_i} \right) \right) \right) \\ = & \sum_{a=0}^{n} \left(\binom{n}{a} \sum_{\substack{\|\overline{\alpha}\|=a \\ 1\preceq\overline{\alpha}\preceq t}} \left(\binom{a}{\overline{\alpha}} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} \mathbf{Y} \right) \right) \end{aligned} \tag{3.39}$$

where the last element $\mathbf{Y}$ is further transformed to extract the count of the synchronization events exclusively:

$$\begin{aligned} \mathbf{Y} &= \left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \succeq 1\right] \prod_{i=1}^{k} (y_i - 1)^{\alpha_i} \\ &= \left[{y_1}^{\gamma_1} ... {y_k}^{\gamma_k}, \gamma_1, ..., \gamma_k \in \mathbb{N} \wedge \gamma_1, ..., \gamma_k > 0\right] \prod_{i=1}^{k} (y_i - 1)^{\alpha_i} \\ &= \prod_{i=1}^{k} \left[{y_i}^{\gamma_i}, \gamma_i > 0\right] (y_i - 1)^{\alpha_i} = \prod_{i=1}^{k} \left[{y_i}^{\gamma_i}, \gamma_i > 0\right] \sum_{r=0}^{\alpha_i} \binom{\alpha_i}{r} {y_i}^r (-1)^{\alpha_i - r} \\ &= \prod_{i=1}^{k} \sum_{\substack{\gamma_i \geqslant 1 \\ \gamma_i \in \mathbb{N}}} \binom{\alpha_i}{\gamma_i} (-1)^{\alpha_i - \gamma_i} = \prod_{i=1}^{k} \left(\sum_{\substack{\gamma_i \geqslant 0 \\ \gamma_i \in \mathbb{N}}} \binom{\alpha_i}{\gamma_i} (-1)^{\alpha_i - \gamma_i} - (-1)^{\alpha_i} \right) \\ &= \prod_{i=1}^{k} \left((1-1)^{\alpha_i} - (-1)^{\alpha_i} \right) = \prod_{i=1}^{k} (-1)^{\alpha_i + 1} = (-1)^{\|\overline{\alpha}\| + k} \end{aligned} \tag{3.40}$$

Finally, given the $\mathbf{Y}$ obtained above, the formula for $G_{n,k,t}$ is derived:

$$\begin{aligned} G_{n,k,t} &= \sum_{a=0}^{n} \left(\binom{n}{a} \sum_{\substack{\|\overline{\alpha}\|=a \\ 0 \preceq \overline{\alpha} \preceq t}} \left(\binom{a}{\overline{\alpha}} \left(\prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} (-1)^{\|\overline{\alpha}\|+k} \right) \right) \right) \\ &= \sum_{1 \preceq \overline{\alpha} \preceq t} \left(\binom{n}{\|\overline{\alpha}\|} \binom{\|\overline{\alpha}\|}{\alpha_1, \dots, \alpha_k} (-1)^{\|\overline{\alpha}\|+k} \prod_{i=1}^{k} \binom{n-\|\overline{\alpha}\|}{t-\alpha_i} \right) \end{aligned} \tag{3.41}$$

□

3.3.5 Optimal transmission set cardinality

The configured value of t shall be greater than zero and not greater than $n-k+1$. When $t-1 > n-k$, there is no chance of t-Slots all agents synchronization due to the Pigeonhole Principle (Fact 10). Finding the precise formula for $\arg\max_{t \in \{1,2,\dots,n-k+1\}}$ is perplexing, as the t-Slots synchronization probability formula is complex and not continuous. Despite that, with Theorem 3.4 the optimal t calculation is straightforward.

If $t > \frac{n}{k}$, transmission sets must certainly overlap; thus, further increasing the t above this threshold will cause more conflicts, effectively deteriorating the chance of synchronization. Knowing this upper bound, simple linear Algorithm 5 is proposed to find the optimal argument t value for t-Slots synchronization probability.

Algorithm 5 t-Slots: The optimal transmission set cardinality t [23]

given procedure $prob(n, k, t)$ to calculate $\Pr[\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}]$
given n, k
$t \leftarrow \lfloor \frac{n}{k} \rfloor$
$p \leftarrow prob(n, k, t)$
while $t > 1 \land prob(n, k, t-1) > p$ **do**
    $p \leftarrow prob(n, k, t-1)$
    $t \leftarrow t - 1$
end while
return t ▷ returns the optimal t for given n, k network preset

A compelling and counter-intuitive phenomenon is observed, that often the optimal cardinality of transmission set is noticeably smaller than $\lfloor \frac{n}{k} \rfloor$. As shown in Table 3.2, given the 30-slot communication round: for $k = 4$ agents the optimal $t = 6$ (1 less than $\lfloor \frac{30}{4} \rfloor$), for $k = 8$ agents the optimal $t = 1$ (2 less than $\lfloor \frac{30}{8} \rfloor$).

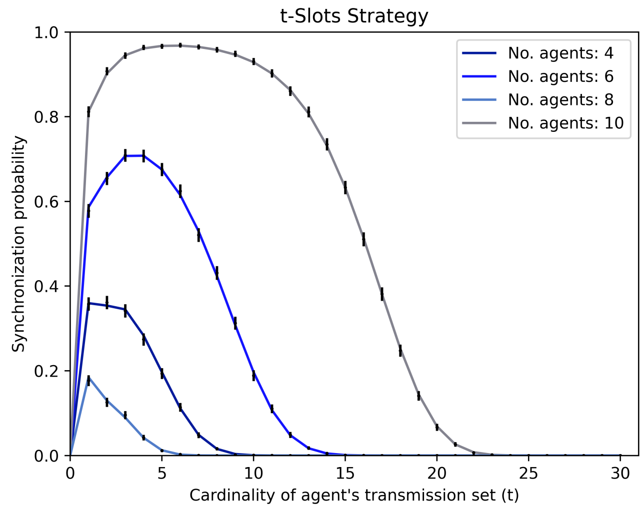


Figure 3.3: The t-SLOTS strategy synchronization probability (Theorem 3.4) as a function of the transmission set size t, linear interpolation. Results for $k \in \{4, 6, 8, 10\}$ agents synchronizing in the communication round of $n = 30$ slots. Vertical ticks represent the statistical test data, with a margin of error (10,000 tests per sample; $MOE_{0.999}$ [53]).

3.3.6 Characteristics and statistical experiments

An open-source Python library developed by author [85] includes the implementation of Theorem 3.4 formula. Similarly to the BT strategy, the t-SLOTS synchronization probability formula has been confronted with the statistical tests (raw test results are available [86]). The sample experiment data is exposed in Figure 3.3 and the corresponding Table 3.2.

Contrary to the BT, in the t-SLOTS strategy domain of the probability formula is not continuous, as the transmission set cardinality t may take only integer values. The calculated synchronisation probability values in Figure 3.3 are represented as a linear interpolation to expose them from the experiment results. In addition, the results of the experiment match the values calculated with the formula of Theorem 3.4.

A particular case of $k = 2$ was already characterized in Section 3.3.3. For $k > 2$, the function of $\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}\right]$ is asymmetric, concave, and has a single maximum point. The function grows rapidly from zero for $t = 0$ to the maximum reached for $t \leqslant \lfloor \frac{n}{k} \rfloor$; then it recedes at an accelerating rate after some close-optimal values decline to smoothly reach the zero values. When the number of agents k is significantly smaller than the number of available slots n, the wide plateau of suboptimal values is formed around the maximum point; with an increase of k, the maximum value is decreased, and the function shape reveals

Table 3.2: t-Slots strategy synchronization probability, $n = 30$, $k \in \{4,6,8,10\}$.

t	$\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{30,4,t}\right]$	$\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{30,6,t}\right]$	$\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{30,8,t}\right]$	$\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{30,10,t}\right]$
1	0.81200000	0.58644444	**0.35968593**	**0.18463878**
2	0.90193317	0.65685418	0.35400007	0.12994318
3	0.94513736	0.70734075	0.34495816	0.08957469
4	0.96134225	**0.70793747**	0.28364982	0.04192549
5	0.96720406	0.67534335	0.19692091	0.01238292
6	**0.96786719**	0.61510080	0.11152155	0.00208525
7	0.96481001	0.52977732	0.04883826	0.00018456
8	0.95802394	0.42406846	0.01557733	0.00000838

the pronounced peak.

Given the example of $n = 30$ slots and $k \in \{4, 6, 8, 10\}$ agents presets, how quickly does the probability of t-Slots all-agents synchronization decrease when the agents' congestion rises? For $k = 4$, the maximum t-Slots synchronization probability is almost 97%, and retains above 0.5 for $1 \leqslant t \leqslant 16$ (half of the domain). For $k = 6$, the maximum probability value is just 70%, while 0.5 is achieved only for $1 \leqslant t \leqslant 7$ (less than a quarter of t configuration domain). For $k = 8$, the probability values already do not reach 36%.

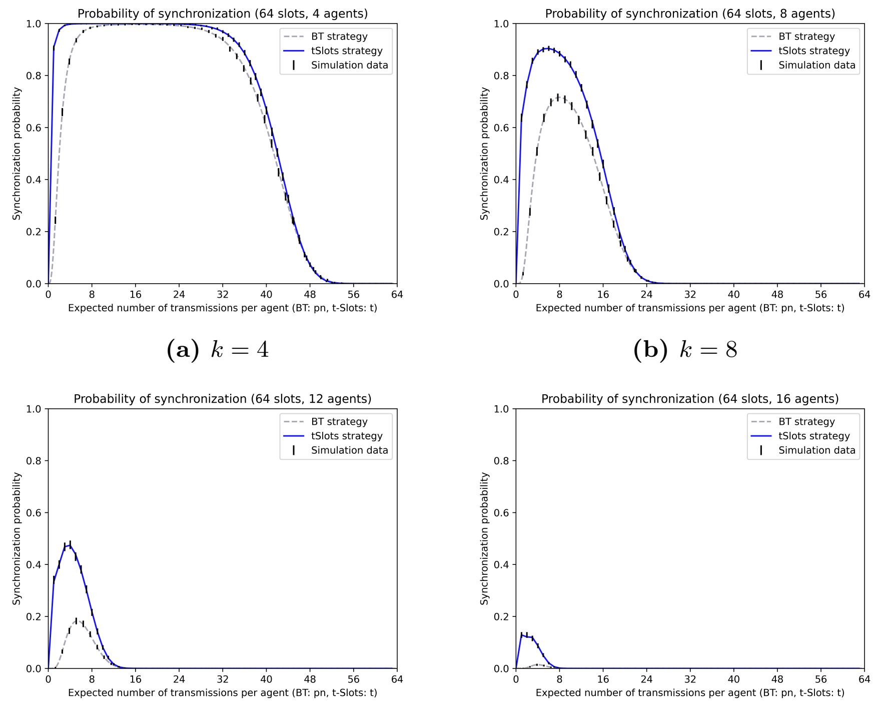
3.4 Comparison

Both BT and t-Slots strategies address the same problem slightly differently; the multi-factor comparison is presented to expose their strengths and weaknesses. The comparison is made for pragmatic network sizes, as stated in Definition 2.10 and the all-agents synchronization setup (Definition 3.1). The following subsections expose the t-Slots strategy advantage, which aligns with the research hypothesis (Hypothesis 1).

3.4.1 Probability of the synchronization

The primary comparison factor is the probability of synchronization that can be achieved in the network when adopting the BT or t-Slots strategy. The strategy must guarantee the highest possible probability of synchronization to be practical. The overall chance of synchronization depends on the number of slots in the communication round n, the number of synchronizing agents k, and the correctness of the strategy configuration. That is, respectively, the probability of a transmission in the single slot p for BT and the cardinality of a transmission set for t-Slots.

Naturally, the higher the congestion of agents in the communication round $\frac{k}{n}$, the lower the chance of all-agent synchronization; both strategies' performance must be compared for the same setup k, n. The setups of the examples from



(a) $k = 4$

(b) $k = 8$

(c) $k = 12$

(d) $k = 16$

Figure 3.4: Compared BT and t-Slots agents synchronization probabilities, they are given as a function of the expected number of individual agent's transmissions (pn for BT, t for t-Slots). The communication round of $n = 64$ slots and (a) 4, (b) 8, (c) 12, (d) 16 synchronizing agents are presented. The synchronization probability of t-Slots is displayed as a linear interpolation; vertical ticks represent the statistical test data, with a margin of error (10,000 tests per sample; $MOE_{0.999}$ [53]).

the previous sections match together; thus, a juxtaposition of Figure 3.2 and Figure 3.3 already reveals the advantage of the t-Slots strategy.

To highlight the difference in performance of both strategies, Figure 3.4 presents four plots that illustrate the synchronization probability achieved by strategies for setup of the $n = 64$ slot communication round and $k \in \{4, 8, 12, 16\}$ agents. Aside from calculated synchronization probabilities, the plots show the statistical test results for the respective configurations. As the configuration domains of $p \in [0, 1]$ and $t \in 1, 2, \ldots, n$ diverge, the data is organized so that the horizontal axis scale presents the expected number of agent transmissions. For the BT strategy, it is the transmission probability p multiplied by n number of slots in the communication round; for the t-Slots strategy, it is the transmission set cardinality t directly. When $np = t$ under the same n, k setup, it is told that BT's and t-Slots's configurations are *adjacent*.

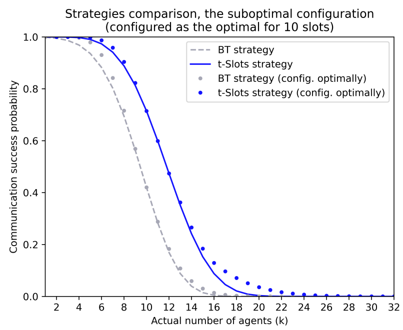
Regardless of the setup n, k, for the same expected value of the agent's transmissions number t-SLOTS strategy guarantees the higher probability of all-agents synchronization. When the probability of synchronization converges to 1 (Figure 3.4a), the difference becomes negligible. For low congestion, both strategies offer almost certain synchronization for a wide margin of p/t configuration value. t-SLOTS strategy agents reveal a slightly higher tolerance to the imperfect configuration that has a broader plateau of near-one values; synchronization probability decreases rapidly for both strategies, reaching zero for the expected number of zero transmissions per agent and 60 transmissions per agent.

The advantage of t-SLOTS becomes truly observable as the congestion increases and the environment becomes more challenging. For $k = 8$ (Figure 3.4b) t-SLOTS's maximum synchronization probability is 1.26 times higher than BT's (0.904 vs. 0.716), for $k = 12$ (Figure 3.4c) it is already 2.59 times higher (0.474 vs. 0.183), and for $k = 16$ (Figure 3.4d) it is 9.21 times higher (0.129 vs. 0.014). This example also exposes how quickly the achievable synchronization probability decreases along with the increasing congestion in the communication round. On the other hand, the setup of 12 agents communicating in the round of 64 slots is already immensely challenging, giving roughly only 5 slots per agent, and every agent must transmit in at least one slot as the only one. BT strategy becomes non-pragmatic earlier: for the $n = 64$ slots it does not achieve a synchronization probability close to 0.5 for $k > 9$, while the t-SLOTS strategy still achieves a synchronization probability of approximately 0.5 for $k = 12$. In the round of 64 communication slots, for $k = 12$, a double execution is expected to result with the synchronization in the t-SLOTS strategy. In contrast, 12 BT agents have an expected number of more than five synchronization attempts before reaching synchronization.

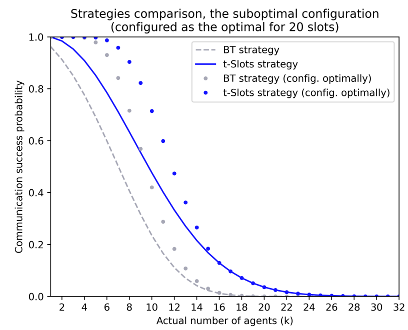
The preference of the t-SLOTS strategy comes not only with the higher synchronization probability and the better resilience to agents' congestion. Figure 3.4 also reveals the desired properties of the $\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}\right]$ function shape. Although for low $k = 4$ (Figure 3.4a) function shapes of both BT and t-SLOTS strategies are similar, when the number of agents increases, they observably diverge. For the BT strategy, it is a slightly asymmetric cone with a single peak and a rapid decline to zero; for $k = 16$ agents in Figure 3.4d, the rise of the synchronization probability above near-zero values is barely observable. The function shape for the t-SLOTS strategy synchronization probability is also concave, asymmetric with a single peak, which is visible even for 16 communicating agents; notably, the function has steeper slopes and forms a broader area of suboptimal values, which is mainly observable in Figure 3.4b, where the values for $4 \leqslant t \leqslant 9$ are within 5% from the function maximum.

3.4.2 Imperfect configuration resilience

A remarkably significant property of both synchronization strategies is a narrowness of the non-zero synchronization probabilities range. The above range



(a) $p = 0.1$, $t = 4$



(b) $p = 0.05$, $t = 1$

Figure 3.5: BT and t-SLOTS agents synchronization probabilities as a function of the number of agents k. The communication round of $n = 64$ slots and the configuration np, t optimal for a network of (a) $k = 10$, (b) $k = 20$ agents is presented. The achieved synchronization probabilities are displayed as a linear interpolation; dots represent the maximum strategy's synchronization probability for a given k.

shrinks as the congestion of agents ($\frac{k}{n}$) rises; this is highlighted in Figure 3.4. For higher congestion, even a tiny misconfiguration can drift the synchronization probability to impractical values.

Despite the visibly higher maximum and wider suboptimal area of the t-SLOTS strategy's synchronization probability, the functions of both strategies plummet at some points, reaching near-zero values at the same expected number of transmissions per agent. Figures 3.4b and 3.4c expose that the advantage of t-SLOTS strategy becomes meaningless when the cardinality of the transmission set t is far misconfigured. The right p and t preset of the analyzed strategies is difficult - as it must be done ahead of the synchronization, either pre-programmed for agents or announced to them before the synchronization starts; in both cases, it must be estimated. This assignment is equally difficult for both strategies.

Figure 3.5 presents how the wrong estimation of p or t affects the synchronization probability of each strategy. Linear interpolations in the plots represent the actual probabilities of synchronization achieved for a given number of participating agents while being fixed for a specific configuration, optimal for 10 (Figure 3.5a) and 20 (Figure 3.5b) agents. The dots represent the maximum probability of synchronization, which is achievable if the agents are perfectly configured.

The optimal configuration synchronization probabilities illustrated by dots, identical in both plots of Figure 3.5, is a frame to compare the impact of misconfiguration. This data already shows that while t-SLOTS indicates a higher resilience, so its achievable synchronization probability starts to decline for larger k, the rising agents' congestion causes both strategies to lose their synchronization capabilities almost at the same pace (the function steepness for t-SLOTS is just minimally lower). The phenomenon is regular, regardless of the network n,

k setup, and manifests itself in Figure 3.5 with the t-Slots function having a shape similar to BT's but shifted right. The application of the t-Slots strategy for $n = 64$ slots and $k = 10$ agent allows to achieve the synchronization probability of 0.715, for the BT strategy achievable at most for $k = 8$ agents; for $k = 15$ agents t-Slots allows to achieve the synchronization probability of 0.184, BT is close to this at most for $k = 12$ agents.

In real scenarios, the transmission probability p or the transmission set cardinality t is adjusted for the network size k, which is usually under- or overestimated, this causes a deviation from the theoretical optimal synchronization probability. The loss is proportional to the difference between the number of agents for which the strategy is configured and the actual number of synchronizing agents; results indicate that the loss is negligible when this difference is small. When the configurations aim for $k = 10$ agents (Figure 3.5a), BT starts visibly underperforming for $k < 8$ and $k > 12$, t-Slots starts underperforming for $k < 8$ and $k > 13$. Considerably, the t-Slots strategy with this fixed configuration guarantees a higher probability of synchronization than the perfectly adjusted BT strategy for arbitrary k. When the configurations aim $k = 20$ agents (Figure 3.5b), both strategies perform almost optimally for any $k \geqslant 15$ number of agents; however, the probability of synchronization there is already close to zero. At a low number of agents, synchronization probability drops (relative to the maximum achievable) significantly worse for BT than for t-Slots, for example $\Pr\left[\mathsf{Success}^{\mathsf{BT}}_{64,6,p}\right]$ drops from 0.930 to 0.599, while $\Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{64,6,t}\right]$ drops from 0.987 to 0.786.

Two important conclusions may be drawn from the analyzed data. The first is an approach to the setup p and t, with a given communication round size n but an unknown number of agents k. As strategies are characteristic of the intensive decline of their maximum synchronization probability from 1 to 0 at some point of network congestion and reveal some tolerance to the misconfiguration, p/t should be fixed to the optimal value for k marking the middle of strategy's synchronization probability decline; in case of the $n = 64$ communication slots in Figure 3.5 it would be $k = 9$ for BT, $k = 13$ for t-Slots. In this way, the least is sacrificed at the boundaries of the strategy's capabilities, while slightly decreasing the probability of synchronization when it is already almost certain or nearly impossible. The second conclusion is that the t-Slots strategy is similarly fragile to the misconfiguration as the BT– but due to its higher resilience, for a fixed configuration described above and the arbitrary k t-Slots may result with the higher synchronization probability than the perfectly-adjusted BT strategy.

3.4.3 Algorithm

All the agent algorithms proposed introduced in Section 3.2.1 and Section 3.3.1 are comparably straightforward. Considering the optimal of the suggested algorithms, their time complexities (measured in the number of generated random bits) are respectively $O\left(n \log_2\left(\frac{1}{\varepsilon}\right)\right)$[6] and $O(t \log_2 n)$, while their space complexities are respectively $O(1)$ and $O(t)$. Therefore, the t-SLOTS agent's algorithm has a better time performance and BT, assuming it can be adapted for streaming, offers better space complexity.

Although the BT algorithm's advantage is the streaming ability, the cardinality of its returned transmission set is subject to considerable variance, negatively impacting the protocol's fairness. On the other hand, homogeneously configured t-SLOTS guarantees equal chances of channel access for all agents. Furthermore, a constant size of the t-SLOTS transmission set allows to testify agents against malicious behaviors, like transmitting in more slots than expected; Algorithm 4 offers even a tool to legitimate the transmission slot choices.

The presented algorithms fit in the purposed environment of the networked embedded systems and the real-time systems. Algorithm 1 and Algorithm 2 source code in C is available [84].

3.4.4 Number of transmissions and energy consumption

Considering the applications based on battery-powered vehicles and devices[7] mentioned in Section 2.3, energy management becomes essential. As highlighted in the previous section, the t-SLOTS Algorithm 3 offers a slight advantage of time complexity, improving the processing cost. However, the heavy-energy-consuming operations are message transmission and channel listening. Here, the outcomes of both strategies differ significantly.

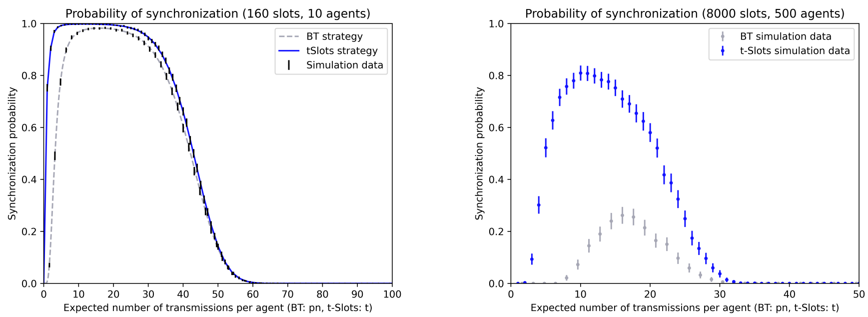
Since the expected number of BT agent's transmissions is np and the number of t-SLOTS agent's transmissions is exactly t, the overall number of communicates sent by a k-agent network under these strategies is, respectively, approximately knp and exactly kt. The advantage of the t-SLOTS strategy is threefold in this field.

Firstly, the general advantage of the achievable synchronization probability allows t-SLOTS to perform the same well as BT in the smaller communication round, reducing the channel listening time of the protocol. In addition, the higher chance of synchronization in the same setup n, k reduces the plausibility of repeating the protocol (if admissible in the particular use case), vitally reducing the energy cost of the synchronization.

Secondly, the optimal number of transmissions per agent is generally lower in t-SLOTS than in BT. In the optimally configured BT strategy $p = \frac{1}{k}$ (Theorem 3.2), expected number of network transmissions is always n. In the t-SLOTS strategy, the optimal t appears to be significantly lower than $\frac{n}{k}$; thus, the number of network transmissions kt is less than n, which conserves energy. In the examples presented in Figure 3.5, for $k = 10$ agents it is 40 versus 64 transmissions (37.5% reduction), for $k = 20$ agents it is 20 versus 64 transmissions (68.75% reduction).

[6] Here the ε is arbitrary depending on the required precision of the parameter p; it could be estimated roughly by $\lfloor \frac{1}{k} \rfloor$, giving the time complexity $O(n \log_2 k)$.

[7] The intermittent power sensor networks (Subsection 2.3.5), in particular.



(a) $k = 10$ **(b)** $k = 500$

Figure 3.6: Compared BT and t-Slots agents synchronization probabilities for the same congestion $\frac{k}{n} = \frac{1}{16}$ but a different scale of (a) 10 and (b) 500 synchronizing agents, given as a function of the expected number of individual agent's transmissions (pn for BT, t for t-Slots). The left-hand plot presents the calculated results juxtaposed with the statistical test data. The right-hand plot presents the results approximated by the statistical tests. Vertical ticks represent a margin of error ((a) $10,000$ and (b) $2,000$ tests per sample; $MOE_{0.999}$ [53]).

Finally, yet not less important, t-Slots is preferred due to the predictability of its energy consumption. Contrary to BT, the t-Slots agent's number of transmissions is not prone to variate; this determinism enables a better agents' power management [74, 89] and results in a more uniform discharge of agents, balancing the power consumption in the network.

3.4.5 Scalability

Although all-agents synchronization aims to mainly address a few to dozens of synchronizing local agents and the short communication rounds, the assumed pragmatic-size ad hoc networks (per Definition 2.10) raise the scope of analysis to hundreds and even low thousands of agents. The performance of both strategies is considered in the scaled case of $k = 500$ agents and $n = 8000$ slots. Communication round of this size is extreme but still rational for wide-bandwidth and high-frequency multichannel networks, such as Wi-Fi (IEEE 802.11 family [48]) or 5G [40, 87]. As calculating the exact synchronization probabilities for a network of this size would require excessive computational power, the synchronization probability is estimated by the statistical tests.

In this setup, the general properties of both strategies are preserved and the t-Slots advantage is intensely amplified. Naturally, as the number of synchronizing agents increases, the chance of synchronization gradually decreases, albeit preserving the $\frac{k}{n}$ ratio. Figure 3.6 shows the comparison of the probabilities of synchronization of the BT and t-Slots strategies for different network sizes with the same congestion $\frac{k}{n}$. For 10 agents and 160 communication slots in Figure 3.6a, both strategies perform very well; when properly configured, their synchronization probabilities converge to one, so the t-SLOTS is only 1.6% better than the BT. For $k = 500$ agents and 8000 communication slots (withstanding congestion) in Figure 3.6b the t-SLOTS performs significantly better, achieving a decent synchronization probability of approximately 0.8, three times higher than the BT.

The results show that when the network scales 50 times, as in Figure 3.6, the t-SLOTS strategy loses 0.2 of its maximum synchronization probability, demonstrating a better resilience than the BT strategy, which maximum drops by 0.72. The t-SLOTS maintains a wide area of suboptimal values around the optimal $t = 10$, while the BT declines quite sharply from the peak reached at expected 16 transmissions per agent for $p = 0.002$. It is noteworthy that the scale of the x-axis differs between Figure 3.6a and Figure 3.6b – the tolerance to imperfect configurations is much smaller in larger networks. In the example of t-SLOTS: for 10 agents, any $3 \leqslant t \leqslant 29$ guarantees the chance of successful synchronization within 5% of maximum; for 500 agents, this range is nearly four times smaller $8 \leqslant t \leqslant 14$.

Chapter 4

Fault-tolerant synchronization

4.1 Objective

The preceding chapter focused on the rigorous all-agents synchronization. Naturally, as the communication round is limited, partial network synchronization failure is inevitable. The results of the previous chapter enable the adjustment of network parameters to decrease the risk above the desired probability. The real systems, depending on application, may endure some number of agents that failed to transmit their message.

To cover the case of unsynchronized agents, a new parameter f is introduced that thereafter represents the number of agents allowed to fail. The quality measure is still the probability of the synchronization, but this time the *synchronization* is defined as follows:

Definition 4.1 (Fault-tolerant synchronization)**.** *Given agents $A_1, ..., A_k$ and the fault tolerance parameter f, the synchronization succeeds if at least $k - f$ agents succeed in transmitting their message M_i in at least one collision-free slot of the communication round.*

Analysis of the time-critical ad hoc network synchronization (Problem 1) with the above-defined success criterion is decisive for the research hypothesis confirmation (Hypothesis 1), proving the coherence of the t-Slots strategy advantage. Results presented in this chapter are also valuable for analyzing fault-tolerant systems and protocols.

In order to facilitate the analysis of the generalized synchronization case defined above, the formulas for the fault-tolerant success probability are derived in this chapter for both the BT and the t-Slots strategy. As in the previous chapter, these results are juxtaposed and validated against the results of statistical experiments. The open-source Python library implementation is provided

[85] along with the open-source analyzer, including the statistical experiment results [84].

4.2 Bernoulli trials strategy

The strategy remains exactly the same as in Algorithm 1, along with all related remarks. Consequently, the sample space is the same as in Definition 3.3. All that differs is the liberalized synchronization criterion, which impacts the synchronization probability.

4.2.1 Synchronization probability formula

This section is devoted to the BT communication scenario, where a new parameter $f \in \mathbb{N}$ is introduced that defines the allowed margin of agents faced with communication failure (Definition 2.7). The fault-tolerant synchronization event is specified according to Definition 4.1.

Definition 4.2 (BT fault-tolerant synchronization event). $\mathsf{SuccessFt}^{\mathsf{BT}}_{n,k,p}(f)$ *is an event of* k *agents following the* BT *strategy with transmission probability* p*, communicating in a round consisting of* n *communication slots, where at most* f *agents have communication failure.*

The parameter f is noted differently from n, k, p to express that it does not belong to the network setup, instead it determines whether the sample event represents a synchronization.

To represent the outcome of the algorithm, a matrix $T_{k,n,p}$ with k rows and n columns is considered, where the rows correspond to agents and columns correspond to slots (see Figure 3.1 in Subsection 3.2.2). $T_{k,n,p}$ contains a 1 in the row i and the column j, if the agent A_i decides to transmit in the slot j. Otherwise, $T_{k,n,p}$ contains a 0 at this place. So, an agent A_i succeeds to communicate in the slot j, if and only if the column j of $T_{k,n,p}$ contains a single 1 and this 1 is in the row i.

The matrix $T_{k,n,p}$ can be seen as a random variable with values in the set of all matrices with elements from the set $\{0, 1\}$ of dimension $k \times n$, where for each position of the matrix a 1 is chosen with probability p, independently of other choices.

Let W denote the set of agents with a communication success, and let L represent the set of agents that have communication failure. Both sets, L and W, are also random variables; by Definition 2.6:

$$W = \{i \in \{1, 2, \ldots, k\} : (\exists i)(T[i, j] = 1 \wedge (\forall u \neq i)(T[u, j] = 0))\} \tag{4.1}$$

and $L = \{1, \ldots, k\} \setminus W$. Of course, $W \cup L = \{1, 2, \ldots, k\}$ and $W \cap L = \emptyset$.

Recall that Theorem 3.1 states that

$$\Pr[W = \{1, 2, \ldots, k\}] = \sum_{l=0}^{k} \binom{k}{l} (-1)^l (1 - lp(1 - p)^{k-1})^n \ . \tag{4.2}$$

An auxiliary function is defined.

Definition 4.3. *For $n, k, a \in \mathbb{N}_+$, such that $a \leqslant k \leqslant n$, and $p \in [0,1]$ we define*

$$f(n,k,a,p) = \sum_{c=0}^{a} \binom{a}{c} (-1)^c \big(1 - cp(1-p)^{k-1}\big)^n \ . \tag{4.3}$$

Theorem 3.1 can be reformulated with function f in the following way:

$$\Pr[W = \{1,2,\ldots,k\}] = f(n,k,k,p) \ . \tag{4.4}$$

In fact, this result can be generalized.

Theorem 4.1. *For any $n, k \in \mathbb{N}_+$, $p \in [0,1]$ and $A \subseteq \{1,2,\ldots,k\}$:*

$$\Pr[A \subseteq W] = f(n,k,|A|,p) \ . \tag{4.5}$$

The proof of Theorem 4.1 follows directly the same lines as the proof of Theorem 3.1, hence it is skipped.

Let B be an arbitrary set of agents. Observe that

$$\Pr[L \subseteq B] = \Pr[(\{1,2,\ldots,k\} \setminus B) \subseteq W] \ . \tag{4.6}$$

Hence

$$\Pr[L \subseteq B] = f(n,k,k-|B|,p) \ . \tag{4.7}$$

However, what is needed in the following considerations is not a formula for $\Pr[L \subseteq B]$ but a formula for $\Pr[L = B]$, where B is a fixed subset of $\{1,2,\ldots,k\}$. This probability is derived with the Möbius Inversion Formula (Fact 11).

Theorem 4.2. *Let $B \subseteq \{1,2,\ldots,k\}$ and $b = |B|$. Then*

$$\Pr[L = B] = \sum_{c=0}^{b} \binom{b}{c} (-1)^{b-c} f(n,k,k-c,p) \ . \tag{4.8}$$

Proof. A direct application of the Möbius Inversion Formula (Fact 11) to the pair of functions $\alpha(B) = \Pr[L = B]$ and $\beta(C) = \Pr[L \subseteq C]$ gives

$$\Pr[L = B] = \sum_{C \subseteq B} \Pr[L \subseteq C](-1)^{|B|-|C|} \ . \tag{4.9}$$

Thus, substituting $\Pr[L \subseteq C]$ with $f(n,k,k-|C|,p)$:

$$\begin{aligned} \Pr[L = B] &= \sum_{C \subseteq B} f(n,k,k-|C|,p)(-1)^{|B|-|C|} \\ &= \sum_{c=0}^{b} \binom{b}{c} (-1)^{b+c} f(n,k,k-c,p) \ . \end{aligned} \tag{4.10}$$

□

Corollary 4.1. *Let $b \leqslant k$, then*

$$\Pr[|L| = b] = \binom{k}{b} \sum_{c=0}^{b} \binom{b}{c} (-1)^{b+c} f(n, k, k-c, p) \ . \tag{4.11}$$

Proof. Notice that

$$\Pr[|L| = b] = \sum_{\substack{B \subseteq \{1,2,\ldots,k\} \\ |B|=b}} \Pr[L = B] \ , \tag{4.12}$$

so the required formula is a direct consequence of the previous theorem. □

Finally, $\Pr[|L| \leqslant f]$ is the probability of agents' synchronization according to Definition 4.1, where no more than f agents have a communication failure.

Corollary 4.2 (BT fault-tolerant synchronization probability)**.** *For any $n \in \mathbb{N}$, $k \in \mathbb{N}_+$, $p \in [0, 1]$, and the fault tolerance $f \in \mathbb{N}$, $f \leqslant k$, the probability of successful Bernoulli trials fault-tolerant synchronization (Definition 4.2) is*

$$\Pr\left[\mathsf{SuccessFt}^{\mathsf{BT}}_{n,k,p}(f)\right] = \Pr[|L| \leqslant f] \ , \tag{4.13}$$

where:

$$\Pr[|L| \leqslant f] = \sum_{b=0}^{f} \sum_{c=0}^{b} \sum_{d=0}^{k-c} \binom{k}{b} \binom{b}{c} \binom{n-k}{d} (-1)^{b+c+d} (1 - d \cdot p \cdot (1-p)^{k-1})^{n} \tag{4.14}$$

Proof. Since events $|L| = f_1$ and $|L| = f_2$ are mutually exclusive when $f_1 \neq f_2$, $\Pr[|L| \leqslant t] = \sum_{b=0}^{t} \Pr[|L| = b]$. The thesis follows from Corollary 4.1, replacing the term $f(n, k, k-c, p)$ by its direct form. □

4.2.2 Optimal probability choice

Although the expression derived in the previous subsection enables calculation of the BT fault-tolerant synchronization probability, its complicated shape does not provide an easy insight of what value of p is the optimal agent's transmission probability. Any direct calculation seems to be at least extremely tedious. However, as shown in the following, there is a shortcut that does not refer to these complicated expressions.

The idea is to consider an alternative way to generate the probability space that describes the result of the execution of the Algorithm 1 (that implements the BT strategy) and *couples* the results for algorithms with different probabilities p and p^* so that one of them *dominates* in some sense. The idea is related to the coupling technique applied to show rapid mixing of Markov chains [28].

As only a presence of the communication success is important, it is not considered whether the communication failure in the slot occurred due to a conflict or a lack of any transmission. In this case, an outcome of the algorithm execution is a vector M of length n, where $M[j]$ represents the outcome in the j-th slot:

$$M[j] = \begin{cases} 0 & \text{if there is a transmission failure in slot } j, \\ i & \text{if agent } A_i \text{ succeeds in slot } j. \end{cases} \tag{4.15}$$

So M is a random variable where, due to the construction of the algorithm BT, the entries at different positions are stochastically independent.

The vector M might be generated in a slightly different way, but so that the result has the same probability distribution as for the execution of the original Algorithm 1. Let $\phi(p)$ denote $p(1-p)^{k-1}$. Note that $\phi(p)$ is the probability of communication success in a slot. The process of generating an instance of $M = M_p$ is as follows. For $j \leqslant n$, do:

1. choose $\zeta_j \in [0,1]$ uniformly at random,
2. if $\zeta_j > \phi(p)$, then set $M[j] = 0$,
3. otherwise choose $i \in \{1,\ldots,k\}$ uniformly at random and set $M[j] = i$.

It is obvious that vectors M obtained this way have the same probability distribution as those resulting from the execution of the Algorithm 1.

Now, consider two probabilities: an arbitrary p and $p^* = \frac{1}{k}$. So p^* is the probability that maximizes the probability of success when a single slot is concerned (Lemma 3.2). The aim is to show that p^* is better than p from the point of view of BT agents' synchronization.

The main idea is to *couple* the random variables M_p and M_{p^*}. Consider two instances of the above-defined generation procedure. For each $j \leqslant n$, do:

1. choose $\zeta_j \in [0,1]$ uniformly at random,
2. choose $i \in \{1,\ldots,k\}$ uniformly at random,
3. if $\zeta_j > \phi(p)$, then set $M_p[j] = 0$, otherwise set $M_p[j] = i$,
4. if $\zeta_j > \phi(p^*)$, then set $M_{p^*}[j] = 0$, otherwise set $M_{p^*}[j] = i$.

Recall that $\phi^*(p) \geq \phi(p)$. This leads to the following result:

Lemma 4.1. *If $M_p[i] > 0$, then $M_{p^*}[i] = M_p[i]$.*

Let $W(M_q)$ denote the set of non-zero entries from M_q. So $W(M_q)$ is a set of *winners* - it corresponds to the set of agents that succeed in communicating during the algorithm execution represented by M_q. Note that the situation might be complicated: some repetitions may occur on M_q, which are useless from the point of view of the algorithm target.

Corollary 4.3. *$W(M_p) \subseteq W(M_{p^*})$.*

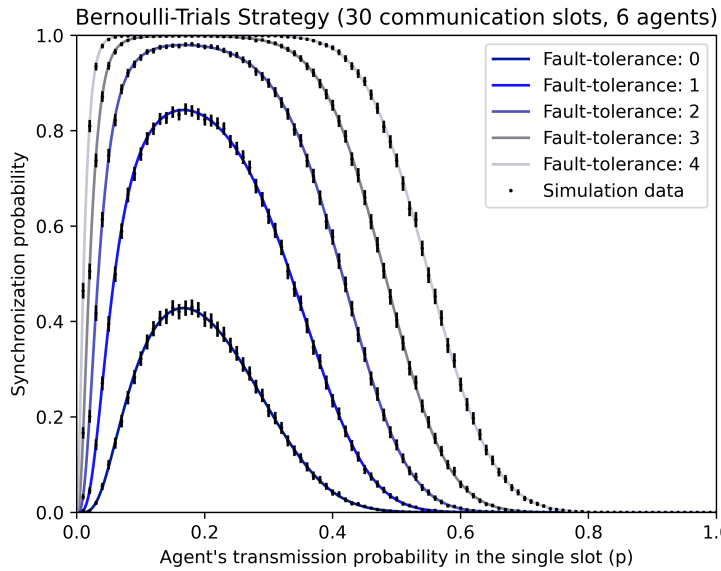


Figure 4.1: The fault-tolerant BT strategy synchronization probability (Corollary 4.2) as a function of the single-slot agent transmission probability p. $k = 6$ agents synchronizing in $n = 30$ slots communication round, $f \in \{0, 1, 2, 3, 4\}$. Vertical ticks represent the statistical test data, including a margin of error (10,000 tests per sample; $MOE_{0.999}$ [53]).

The corollary 4.3 is the key observation: each result of the algorithm for probability p and the choice of $\langle \zeta_1, \ldots, \zeta_n \rangle$ may be coupled with the result of the algorithm run for probability p^* and the same $\langle \zeta_1, \ldots, \zeta_n \rangle$ so that the winners for M_{p^*} form a superset of the winners for M_p. This means that the instance of M for p^* is always not worse than the instance of M for an arbitrary p, which leads to the conclusion:

Theorem 4.3. *For $n, k \in \mathbb{N}_+$, $f \in \mathbb{N}$, regardless of the number of slots n, agents k and the fault tolerance level f*

$$\arg\max_{p \in [0,1]} \Pr\left[\mathsf{SuccessFt}^{\mathsf{BT}}_{n,k,p}(f)\right] = \frac{1}{k} \tag{4.16}$$

4.2.3 Characteristics and statistical experiments

The BT fault-tolerant synchronization probability formula presented in Corollary 4.2 was implemented by the author and delivered as the open-source Python library [85].

The synchronization probability values, which represent the likelihood of successful synchronization in a given configuration, are illustrated in Table 4.1, which refers to the configuration presented in Figure 4.1.

Table 4.1: The fault-tolerant BT strategy synchronization probability value for the optimal configuration $p = \frac{1}{6}$ in the setup of $n = 30$ slots, $k = 6$ agents.

Fault-tolerance (f)	$\Pr\left[\mathsf{SuccessFt}^{\mathsf{BT}}_{30,6,\frac{1}{6}}(f)\right]$
0	0.42822021
1	0.84346411
2	0.97981428
3	0.99881770
4	0.99997184

The results of increasing fault tolerance f, are compared with the results of the statistical tests, as presented in Figure 4.1. The case $f = 0$ is actually the synchronization of all the BT agents, presented in Section 3.2. Increasing fault tolerance involves an overall increase in the probability of synchronization throughout the p domain, while for close-zero function values this rise is negligible.

As the fault tolerance increases, so the maximum function value approaches 1, a plateau of near-optimal values starts forming, so the BT strategy becomes less fragile to the non-optimal p configuration. As is observable in Figure 4.1 for non-optimal $p = 0.35$, after accepting the failure of up to two agents, the BT synchronization probability rises seven times (from 0.11 when $f = 0$, to 0.78 when $f = 2$). If the increase f continues, $\Pr\left[\mathsf{SuccessFt}^{\mathsf{BT}}_{n,k,p}(f)\right]$ converges to 1 across the whole domain, expanding the plateau of close-optimal values.

4.3 tSlots strategy

Similarly to the BT strategy, the algorithm implemented by the agents does not change; for the t-Slots agents, it is one of the algorithms introduced in Section 3.3.1 of the previous chapter. Also, the sample space is the same as specified by Definition 3.6. Only the synchronization event is redefined to capture the case that up to f agents have communication failure (Definition 4.1). The problem considered in this chapter is to determine the probability of synchronization and the optimal choice of the cardinality of the transmission set t, depending on other algorithm parameters.

4.3.1 Synchronization probability formula

Definition 4.4 (t-Slots fault-tolerant synchronization event)**.** $\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)$ *is an event of k agents following the t-Slots strategy, transmitting in exactly t slots of the n-slot communication round, where at most f agents have a communication failure.*

The auxiliary event representing the synchronization with *exactly* v agent communication failures is defined first.

Definition 4.5. *$T_{n,k,t}(v)$ is an event in which exactly v out of k agents have a communication failure in the n-slots communication round, while all agents followed the t-*SLOTS *strategy, that is, each agent transmits in exactly t slots.*

Based on Theorem 3.4 the following probability formula is derived:

Theorem 4.4. *For any $n \in \mathbb{N}$, $k, t \in \mathbb{N}_+$, $t \leqslant n$ and fault tolerance $v \in \mathbb{N}$, $v \leqslant k$*

$$\Pr\left[T_{n,k,t}(v)\right] = \frac{G_{n,k,t,v}}{\binom{n}{t}^k} \tag{4.17}$$

where

$$G_{n,k,t,v} = \binom{k}{v}(-1)^{k-v} \sum_{\overline{\alpha} \in C_{k,t,v}} \binom{n}{\|\overline{\alpha}\|}\binom{\|\overline{\alpha}\|}{\overline{\alpha}}(-1)^{\|\overline{\alpha}\|} \prod_{i=1}^{k} \binom{n-\|\overline{\alpha}\|}{t-\alpha_i} \tag{4.18}$$

and

$$C_{k,t,v} = \left\{ \langle \alpha_1, ..., \alpha_k \rangle \in \mathbb{N}^k : \alpha_i \in \begin{cases} \{0, 1, \ldots, t\} & \textit{if } 1 \leq i \leq v \\ \{1, 2, \ldots, t\} & \textit{if } v < i \leq k \end{cases} \right\}.$$

Proof. Assumptions of the theorem are similar to those of Theorem 3.4, with only one significant difference: exactly v agents have no successful transmission in the communication round. Consequently, the proof follows the same initial steps with adjusted requirements:

1. Each agent A_i has the transmission set consisting of exactly t slots;
2. Exactly v agents have a communication failure within the communication round; i.e., transmission sets of exactly $k - v$ agents have at least one slot not contained by any other agent's transmission set.

Possible combinations of the transmission sets are formulated algebraically, reproducing steps of the Theorem 3.4 up to the equation 3.38.

The first requirement (each agent's transmission set consists of exactly t slots) holds both in this theorem and in the Theorem 3.4. In the equation 3.38 an extraction related to this obligation is applied: $[{x_1}^t {x_2}^t \ldots {x_k}^t]$.

From this point, the proof diverges. An extraction related to the second requirement must retain only these monomials that contain exactly $k-v$ variables $y_i^{\gamma_i}$, where $\gamma_i \geqslant 1$. The extraction is defined as $\left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \in \Gamma\right]$, where $\Gamma = \{\langle \gamma_1, \ldots, \gamma_k \rangle : \gamma_i \geqslant 0 \wedge |\{\gamma_i : \gamma_i = 0\}| = v\}$. Applied to the formula from equation 3.38 leads to the formulation of $G_{n,k,t,v}$:

$$\begin{aligned} G_{n,k,t,v} &= \left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \in \Gamma\right] \left[{x_1}^t {x_2}^t \ldots {x_k}^t\right] g_{n,k}(\overline{x}, \overline{y}) \\ &= \left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \in \Gamma\right] \sum_{a=0}^{n} \binom{n}{a} \sum_{\substack{\|\overline{\alpha}\|=a \\ 0 \preceq \overline{\alpha} \preceq t}} \left(\binom{a}{\overline{\alpha}} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} (y_i - 1)^{\alpha_i} \right) \\ &= \sum_{a=0}^{n} \binom{n}{a} \left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \in \Gamma\right] \sum_{\substack{\|\overline{\alpha}\|=a \\ 0 \preceq \overline{\alpha} \preceq t}} \left(\binom{a}{\overline{\alpha}} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} (y_i - 1)^{\alpha_i} \right) \end{aligned} \tag{4.19}$$

Only v agents are allowed to have zero successful transmissions (y_i^0), while the remaining must have at least one successful transmission ($y_i^{\gamma_i}$, $\gamma_i \geqslant 1$). The v agents having communication failure may be chosen from all k agents in $\binom{k}{v}$ ways. Thus, the $G_{n,k,t,v}$ may be expressed as:

$$G_{n,k,t,v} = \sum_{a=0}^{n} \binom{n}{a}\binom{k}{v} \sum_{\substack{\|\overline{\delta}\| \le a \\ 0 \le \delta_1,\dots,\delta_v \le t}} \sum_{\substack{\|\overline{\beta}\| = a - \|\overline{\delta}\| \\ 1 \le \beta_1,\dots,\beta_{k-v} \le t}} \left(\binom{a}{\overline{\delta}, \overline{\beta}} \mathbf{Y}_{\overline{\delta}} \mathbf{Y}_{\overline{\beta}} \right) \tag{4.20}$$

where $\binom{a}{\overline{\delta},\overline{\beta}} = \binom{a}{\delta_1,\dots,\delta_v,\beta_1,\dots,\beta_{k-v}}$, which is well-defined as $\|\overline{\delta}\| + \|\overline{\beta}\| = a$. Internal expressions $\mathbf{Y}_{\overline{\delta}}$ and $\mathbf{Y}_{\overline{\beta}}$ are elaborated below.

It may seem counter-intuitive why the summation over $\overline{\delta}$, related to agents having the communication failure, iterates over the $0, 1, \dots, t$ dedicated slots per agent δ_i, $i \in \{1, 2, \dots, v\}$. The extraction that ensures no successful communication of v agents takes place in $\mathbf{Y}_{\overline{\delta}}$; summations over $\overline{\delta}$, $\overline{\beta}$ are responsible exclusively for generating all the possible combinations of the non-collision slots assignments. The slots assigned to v agents in the sum over the $\overline{\delta}$ of equation 4.20 may be thus interpreted as the non-collision slots dedicated to the failing agents but not belonging to transmission sets of these agents.

The extraction $\left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \in \Gamma\right]$ applied to the representation of agents having a communication failure (those having zero successful transmissions) in $\mathbf{Y}_{\overline{\delta}}$ is actually formulated as $\left[{y_1}^0 {y_2}^0 \dots {y_v}^0\right]$[1]. In a consequence, when $(y_i - 1)^{\delta_i}$ is broken down according to the Binomial theorem (Fact 1), extraction retains only the 0^{th} element of the sum:

$$\begin{aligned}
\mathbf{Y}_{\overline{\delta}} &= \left[{y_1}^0 {y_2}^0 \dots {y_v}^0\right] \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} (y_i - 1)^{\delta_i} \\
&= \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} \left[{y_i}^0\right] (y_i - 1)^{\delta_i} \\
&= \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} \left[{y_i}^0\right] \sum_{r=0}^{\delta_i} \binom{\delta_i}{r} {y_i}^r (-1)^{\delta_i - r} \\
&= \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} \binom{\delta_i}{0} (-1)^{\delta_i} \\
&= \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} (-1)^{\delta_i} \\
&= (-1)^{\delta_1 + \delta_2 + \dots + \delta_v} \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} \\
&= (-1)^{\|\overline{\delta}\|} \prod_{i=1}^{v} \binom{n-a}{t-\delta_i}
\end{aligned} \tag{4.21}$$

[1] As in the underlying proof of the Theorem 3.4, each successful transmission of agent A_i in monomials is represented by $x_i y_i$.

On the contrary, the extraction $\left[\overline{y}^{\overline{\gamma}}, \overline{\gamma} \in \Gamma\right]$ when applied to the agents with an obligatory successful transmission in $\mathbf{Y}_{\overline{\beta}}$ has a form enforcing the presence of $y_{v+1}, y_{v+2}, \ldots, y_k$ in the counted monomials. The extraction is represented as $[{y_{v+1}}^{\gamma_{v+1}} {y_{v+2}}^{\gamma_{v+2}} \ldots {y_k}^{\gamma_k}, \gamma_i > 0]$. Similarly to above, the $\mathbf{Y}_{\overline{\beta}}$ is broken down with the Binomial theorem (Fact 1), but this time all but 0^{th} sum element are counted:

$$
\begin{aligned}
\mathbf{Y}_{\overline{\delta}} =& [{y_{v+1}}^{\gamma_{v+1}} {y_{v+2}}^{\gamma_{v+2}} \ldots {y_k}^{\gamma_k}, \gamma_i > 0] \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} (y_{v+i}-1)^{\beta_i} \\
=& \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} [{y_{v+i}}^{\gamma_{v+i}}, \gamma_{v+i} > 0] \, (y_{v+i}-1)^{\beta_i} \\
=& \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} [{y_{v+i}}^{\gamma_{v+i}}, \gamma_{v+i} > 0] \sum_{r=0}^{\beta_i} \binom{\beta_i}{r} {y_{v+i}}^{r} (-1)^{\beta_i - r} \\
=& \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \sum_{r=1}^{\beta_i} \binom{\beta_i}{r} (-1)^{\beta_i - r} \\
=& \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \left(\left(\sum_{r=0}^{\beta_i} \binom{\beta_i}{r} (-1)^{\beta_i - r} \right) - \binom{\beta_i}{0} (-1)^{\beta_i} \right) \\
=& \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \left((1-1)^{\beta_i} - (-1)^{\beta_i} \right) \\
=& \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \left((-1)^{\beta_i + 1} \right) \\
=& (-1)^{\beta_1 + \beta_2 + \cdots + \beta_{k-v}} (-1)^{k-v} \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \\
=& (-1)^{\|\overline{\beta}\| + k - v} \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \\
=& (-1)^{a - \|\overline{\delta}\| + k - v} \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i}
\end{aligned}
\tag{4.22}
$$

Compiling $\mathbf{Y}_{\overline{\delta}}$ and $\mathbf{Y}_{\overline{\beta}}$ together

$$
\begin{aligned}
\mathbf{Y}_{\overline{\delta}} \mathbf{Y}_{\overline{\beta}} =& (-1)^{\|\overline{\delta}\|} \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} (-1)^{a - \|\overline{\delta}\| + k - v} \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i} \\
=& (-1)^{a + k - v} \prod_{i=1}^{v} \binom{n-a}{t-\delta_i} \prod_{i=1}^{k-v} \binom{n-a}{t-\beta_i}
\end{aligned}
\tag{4.23}
$$

Summation ranges from the equation 4.20 may be composed together by a return to the summation over the single vector, this time $\overline{\alpha} = \langle \overline{\delta}, \overline{\beta} \rangle$:

$$G_{n,k,t,v} = \sum_{a=0}^{n} \binom{n}{a} \binom{k}{v} \sum_{\substack{\|\overline{\alpha}\|=a \\ 0 \le \alpha_1,\ldots,\alpha_v \le t \\ 1 \le \alpha_{v+1},\ldots,\alpha_k \le t}} \binom{a}{\overline{\alpha}} (-1)^{a+k-v} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} \tag{4.24}$$

Analogously to the Theorem 3.4, the iterator a may be eliminated from the equation. It is guaranteed that no extra summands appear after this simplification as all the terms of $\|\overline{\alpha}\| > n$ are zeroed by the binomial $\binom{n}{\|\overline{\alpha}\|}$. Also, $\binom{k}{v}$ and $(-1)^{k-v}$ may be extracted from the sum, since it does not depend on the summation criteria:

$$\begin{aligned} G_{n,k,t,v} &= \sum_{a=0}^{n} \sum_{\substack{\|\overline{\alpha}\|=a \\ 0 \le \alpha_1,\ldots,\alpha_v \le t \\ 1 \le \alpha_{v+1},\ldots,\alpha_k \le t}} \binom{n}{a} \binom{k}{v} \binom{a}{\overline{\alpha}} (-1)^{a+k-v} \prod_{i=1}^{k} \binom{n-a}{t-\alpha_i} \\ &= \sum_{\substack{\overline{\alpha} \\ 0 \le \alpha_1,\ldots,\alpha_v \le t \\ 1 \le \alpha_{v+1},\ldots,\alpha_k \le t}} \binom{n}{\|\overline{\alpha}\|} \binom{\|\overline{\alpha}\|}{\overline{\alpha}} \binom{k}{v} (-1)^{\|\overline{\alpha}\|+k-v} \prod_{i=1}^{k} \binom{n-\|\overline{\alpha}\|}{t-\alpha_i} \\ &= \binom{k}{v} (-1)^{k-v} \sum_{\overline{\alpha} \in C_{k,t,v}} \binom{n}{\|\overline{\alpha}\|} \binom{\|\overline{\alpha}\|}{\overline{\alpha}} (-1)^{\|\overline{\alpha}\|} \prod_{i=1}^{k} \binom{n-\|\overline{\alpha}\|}{t-\alpha_i} \end{aligned} \tag{4.25}$$

where:

$$C_{k,t,v} = \left\{ \langle \alpha_1, ..., \alpha_k \rangle \in \mathbb{N}^k : \alpha_i \in \begin{cases} \{0, 1, \ldots, t\} & \text{if } 1 \le i \le v \\ \{1, 2, \ldots, t\} & \text{if } v < i \le k \end{cases} \right\} \tag{4.26}$$

□

The probability of $\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)$ event may be stated with support of probabilities $T_{n,k,t}(v)$, for $v \in \{0, 1, \ldots, f\}$.

Corollary 4.4 (t-Slots fault-tolerant synchronization probability)**.** *For any $n \in \mathbb{N}$, $k, t \in \mathbb{N}_+$, and the fault tolerance $f \in \mathbb{N}$, $f \leqslant k$, the probability of the t-Slots fault-tolerant synchronization (as per Definition 4.4) is*

$$\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)\right] = \sum_{v=0}^{f} \Pr\left[T_{n,k,t}(v)\right] \tag{4.27}$$

Proof. Analogous to the proof of Theorem 4.2. $\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)$ is satisfied always when $T_{n,k,t}(v)$ occurs, where $v \le f$. According to Definition 4.4, $T_{n,k,t}(v)$ for different values of v are disjoint: the synchronization process must end with a single specific number of agents that did not communicate; thus

$$\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)\right] = \Pr\left[\bigcup_{v=0}^{f} T_{n,k,t}(v)\right] = \sum_{v=0}^{f} \Pr\left[T_{n,k,t}(v)\right] \tag{4.28}$$

□

4.3.2 Optimal transmission set cardinality

Table 4.2: The fault-tolerant t-Slots strategy synchronization probability value for the configurations $t \in \{4, 5\}$ in the setup of $n = 30$ slots, $k = 6$ agents.

Fault-tolerance (f)	$\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{30,6,4}(f)\right]$	$\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{30,6,5}(f)\right]$
0	**0.70793747**	0.67534335
1	**0.94760775**	0.94210863
2	**0.99407694**	0.99381546
3	0.99957906	**0.99960170**
4	0.99998273	**0.99998567**

The optimal value of the cardinality of the agent's transmission set cardinality, t, is still lower than $\frac{n}{k}$; Algorithm 5 from Section 3.3.5, with the adjusted probability function, is also applicable to find the optimal agent configuration in fault-tolerant synchronization. However, an interesting phenomenon occurs when the fault tolerance margin f increases: in t-Slots strategy, the optimal argument value of t tends to converge to $\lfloor \frac{n}{k} \rfloor$. As a consequence, arg max for the same n, k, t configuration increases with the incremented limit of agents having communication failure.

The phenomenon is highlighted in the example presented in Table 4.1; the maximum value of $\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)\right]$ for f in each row is in bold. Initially, the visible peek of the synchronization probability materializes for argument $t = 4$. When $f \geqslant 3$, the probability function already creates a wide plateau of values close to 1 (see Figure 4.2). For $f = 3$, the optimal argument is swapped to $t = 5$, for which the value of the synchronization probability very slightly exceeds the surrounding one.

4.3.3 Characteristics and statistical experiments

The formula of Corollary 4.4 is available as an open-source Python library [85], which can be used for the modeling of communication protocols.

Figure 4.2 presents a sample result of the $\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)\right]$ calculation and the corresponding statistical experiments. The plot shows that the probability of synchronization rises across the t (that is, the agent's configuration) domain when the fault tolerance f increases. As f approaches k, the probability of synchronization converges to 1. However, for the illustrated case of $n = 30$ slots and $k = 6$ agents, for $f = 2$ fault tolerance, the probability of fault-tolerant synchronization is already close to 1 for half of the domain.

Analogously to the fault-tolerant BT synchronization, when the subset of agents is allowed to fail, the t-Slots strategy tolerance to imperfect configuration increases. The functions representing the t-Slots strategy probability of synchronization tend to form a plateau of suboptimal values even for $f = 0$.

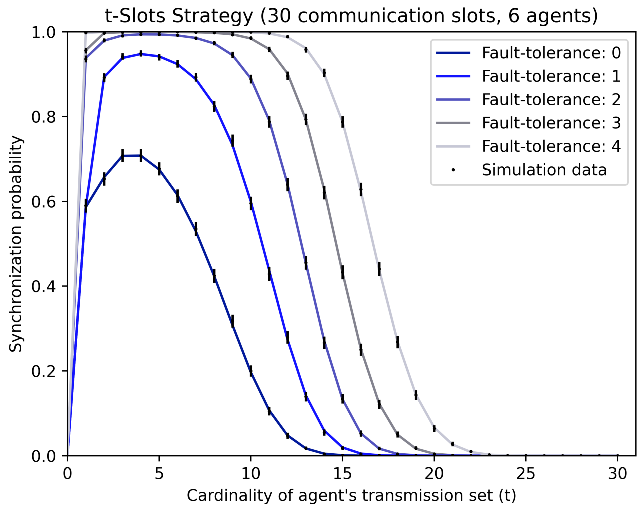


Figure 4.2: The fault-tolerant t-SLOTS synchronization probability (Corollary 4.4) as a function of the transmission set cardinality t (linear interpolation). $k = 6$ agents synchronizing in $n = 30$ slots communication round, failure tolerance $f \in \{0, 1, 2, 3, 4\}$. Linear interpolation of the results for integer argument values. Vertical ticks represent the statistical experiments data, including a margin of error (10,000 tests per sample; $MOE_{0.999}$ [53]).

When $f > 0$, the plateau formed around the maximum value of the function is emphasized, revealing a wide range of configuration values t that almost guarantee synchronization. As the data presented in Figure 4.2 show, the probability of synchronization t-SLOTS can increase even more rapidly than that of BT. When limited fault tolerance is accepted, $\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)\right]$ increases thirteen times for $t = 12$ (from 0.05 when $f = 0$, to 0.64 when $f = 2$).

When $f = 0$, $\Pr\left[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(0)\right] = \Pr\left[\mathsf{Success}^{\mathsf{tSlots}}_{n,k,t}\right]$. The problem collapses naturally into the case of all-agents synchronization, defined in the previous chapter (Definition 3.7):

4.4 Comparison

While the detailed comparison of the two strategies studied is presented in Section 3.4, this section focuses exclusively on the impact made by the limited fault tolerance $f \neq 0$, addressing the synchronization understood according to Definition 4.1. Subsections organizing the juxtaposed strategies' properties reflect the comparison structure present in the preceding chapter, except for an algorithms-dedicated subsection, which is omitted here as the agents' algorithms remain unchanged. The following subsections complete the evidence for the advantage of the t-Slots strategy, confirming the correctness of the research hypothesis.

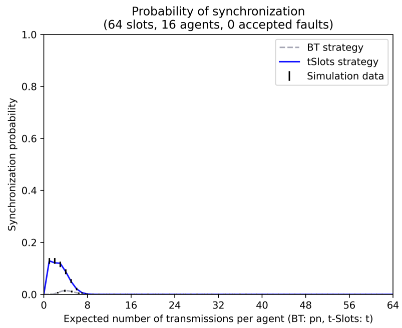
4.4.1 Probability of the synchronization

Expectedly, as the tolerance to f agents with failure communication increases, the synchronization (as per Definition 4.1) probability rises, and the domain of non-zero probability values expands. This may be observed for casual cases in the previous sections, as shown in Figure 4.1 and Figure 4.2.

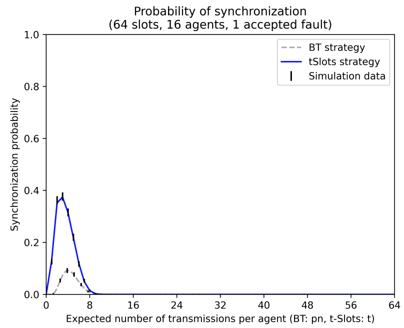
Figure 4.3 visualizes a gradual improvement in the achievable synchronization probability of both t-Slots and BT when the fault tolerance increases. The consecutive plots present the most extreme case from Figure 3.4d, where $k = 16$ agents communicate in the round of $n = 64$ slots, for increasing $f \in \{0, 1, 2, 3, 4, 5\}$. Although all-agent synchronization is unlikely (maximum probability: 0.129 for t-Slots, 0.014 for BT) in this configuration, the maximum synchronization probability of at least 14 agents ($f = 2$) is already higher than 0.647 for t-Slots and higher than 0.276 for BT. Moreover, the probability of synchronization of at least 11 agents ($f = 5$), substantially more than half of the network, converges to one, settled at 0.983 for t-Slots, and 0.911 for BT at their optimal configurations.

Regardless of strategy, the increase in synchronization probability after incrementation of f is proportional to the previous probability value for a given n, k, p/t setup. When the synchronization probability is given as a function of the expected number of agent transmissions, the range of non-zero values expands slowly, whereas the achievable synchronization probabilities rise sharply for both strategies. For every tested n, k setup, p/t resulting in non-zero synchronization probability, and fault tolerance f, the t-Slots strategy exposes a steady advantage over the BT strategy. This advantage diminishes only at the level of f where maximum synchronization probabilities of both strategies converge to 1; even then, t-Slots offers a slightly larger range of close-optimal probability values. In conclusion, since $\Pr[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{n,k,t}(f)]$ remains higher than $\Pr[\mathsf{SuccessFt}^{\mathsf{BT}}_{n,k,p}(f)]$ for the corresponding setups, the t-Slots strategy is also preferred in the problem of fault-tolerant synchronization.

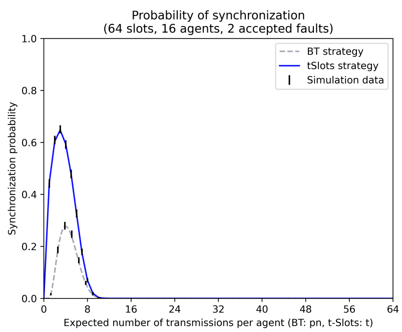
Interesting is a comparison between Figure 3.4c and Figure 4.3e: the former corresponds to the all-agents synchronization scenario of 12 communicating agents, while the latter refers to the scenario of at least 12 agents synchronizing in the network of 16 communicating agents. Comparing the synchronization probabilities, given as functions of the expected number of transmissions per agent, it is observable that the ranges of non-zero values are similar. In contrast, the synchronization probability values soar in the fault-tolerant higher-congestion case; its maximum is doubled for the t-Slots strategy (from 0.474 to 0.943) and quadrupled for the BT strategy (from 0.183 to 0.766). The above phenomenon and the almost certain synchronization of at least half of the communicating agents in the depicted setup suggest that the overall all-agent synchronization probability could be further improved by splitting the synchronization into the multiple rounds, where some portion of agents is synchronized and



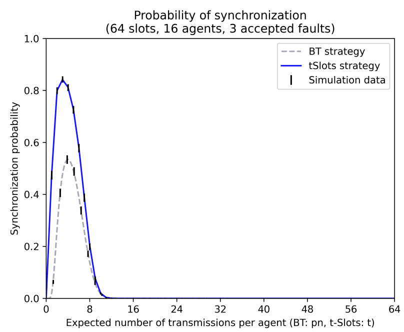
(a) $f = 0$



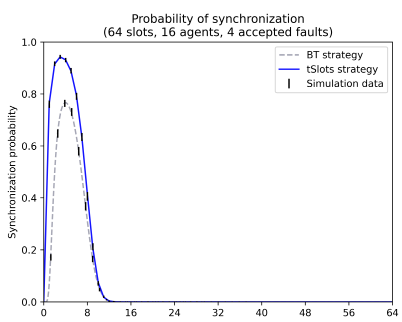
(b) $f = 1$



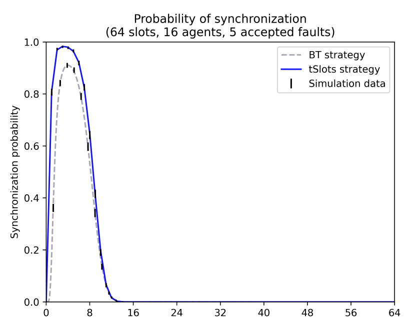
(c) $f = 2$



(d) $f = 3$



(e) $f = 4$



(f) $f = 5$

Figure 4.3: Compared BT and t-Slots agents synchronization probabilities, they are given as a function of the expected number of individual agent's transmissions (pn for BT, t for t-Slots). The communication round of $n = 64$ slots, the network of $k = 16$ agents, and the fault tolerance $f \in \{0, 1, \ldots, 5\}$ are presented. The synchronization probability of t-Slots is displayed as a linear interpolation; vertical ticks represent the statistical test data, with a margin of error ($10,000$ tests per sample; $MOE_{0.999}$ [53]).

eliminated each round. However, the applicability of such a protocol would be limited to specific use cases; the idea deserves a dedicated study.

4.4.2 Imperfect configuration resilience

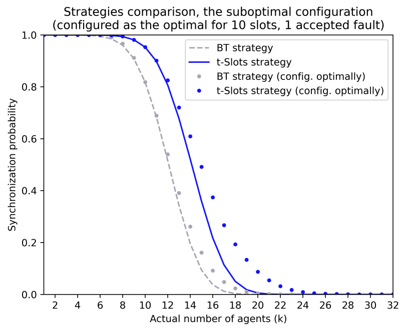
As pointed out in the previous chapter's last paragraph, the range of p/t configuration values leading to non-zero synchronization probabilities expands just slightly when the fault tolerance f rises; thus, the increase of f does not visibly improve the imperfect configuration resilience. Moreover, since the synchronization probability function exhibits a steeper peak shape along with the rising f, as shown in Figure 4.3 – the strategy's misconfiguration is therefore more impactful. Thus, adjusting the strategies properly becomes even more significant in a fault-tolerant synchronization.

Figure 4.4 presents the wrong p/t configuration impact for the BT and t-SLOTS strategies. The dots represent the maximum achievable synchronization probabilities for the given number of communicating agents (k), the linear interpolations represent the synchronization probabilities achieved for a given k, with the configurations fixed as optimal for $k = 10$ (left-hand plots) and $k = 20$ (right-hand plots). Purposely, the setup of the $n = 64$ communication slots aligns with that of Figure 3.5, as Figure 4.4 extends that insight with fault tolerance $f \in \{1, 3, 4\}$.

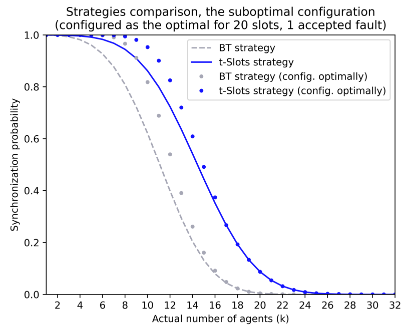
The functions represented by dots are the same for both plots of every row with a specific fault tolerance. Regardless of f, the t-SLOTS strategy exposes a higher resistance to congestion, declining later and slightly slower than BT. Also, along with the increasing value of f, the functions representing the maximum synchronization probabilities shift right; when $f = 1$ decline for the t-SLOTS strategy starts at $k = 7$ reaching zero at $k = 21$, when $f = 4$ it starts at $k = 13$ reaching zero at $k = 32$ (which responds to the extreme congestion of $\frac{k}{n} = \frac{1}{2}$).

The tolerance to the imperfect configuration is analogous for both strategies, with the generally higher t-SLOTS capability demonstrated by the corresponding functions being shifted right respect to the BT functions, as shown in Figure 4.4. The synchronization probabilities achieved when strategies are configured optimally for the network of $k = 10$ agents remain optimal when their values are close to one; however, at the beginning of their declines, their values move apart from the optimum. The above effect is getting stronger as the fault tolerance rises. Interestingly, for $f = 4$ the t-SLOTS optimal configuration for $k = 10$ switches from $t = 5$ to $t = 6$, resulting in the expected number of transmissions being closer to the BT's; at this point, the t-SLOTS strategy visibly loses its advantage over the BT (Figure 4.4e). When strategies' configurations are fixed to be optimal for $k = 20$ agents, they underperform for $k < 15$ and $f = 1$, whereas they only negligibly underperform when the probability values are close to zero or one for $f \in \{3, 4\}$. The reason for such an improvement with regard to $k = 10$ is the fact that $k = 20$ is close to the middle of declines of the shifted optimal synchronization value functions.

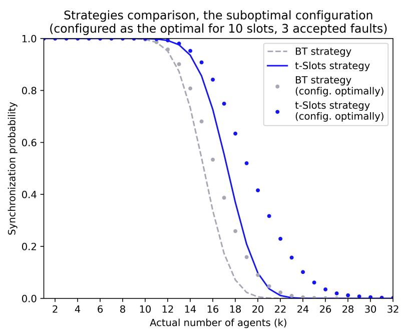
Similarly to Section 3.4.2, the strategy to adjust the p/t configuration for



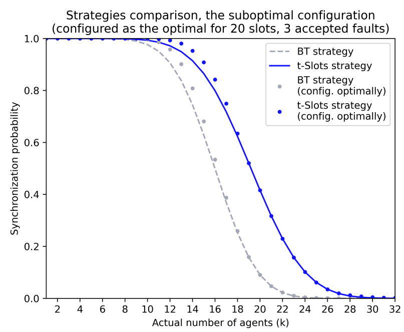
(a) $k = 10, \ f = 1$



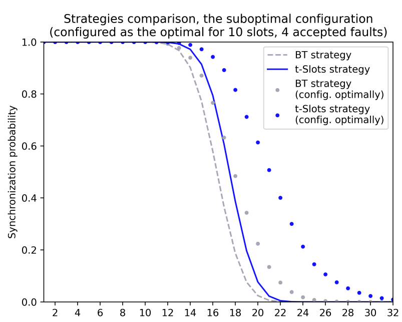
(b) $k = 20, \ f = 1$



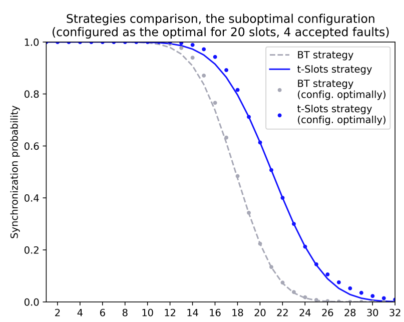
(c) $k = 10, \ f = 3$



(d) $k = 20, \ f = 3$



(e) $k = 10, \ f = 4$



(f) $k = 20, \ f = 4$

Figure 4.4: BT and t-Slots agents synchronization probabilities, given as a function of the number of agents k. The communication round of $n = 64$ slots, the fault tolerance $f = \{1, 3, 4\}$, and the configuration p, t optimal for network of $k = 10$ (left-hand) and $k = 20$ (right-hand) agents are presented. The achieved synchronization probabilities are displayed as a linear interpolation; dots represent the maximum strategy's synchronization probability for a given number of agents k.

the fixed n and known desired f when the number of agents is hard to estimate is to establish the optimal synchronization probabilities as a function of number of agents, then find the k marking the middle of the function descent and configure the strategy optimally for synchronization of k. Since the strategies achieve the synchronization probability close to optimal for the surrounding k, the performance of the strategy configured as above retains the characteristics of the perfectly configured strategy, moving apart only for the probability values already close to one or impractically close to zero. In-advance calculation of the maximum synchronization values for $f > 0$ is possible due to Corollary 4.2 and Corollary 4.4. As noted, the descent shifts towards the higher values of k when the fault tolerance rises; thus, the strategies shall aim to fit a higher congestion when the fault tolerance is higher to maximize the probability of synchronization.

4.4.3 Number of transmissions and energy consumption

All the energy-conserving t-SLOTS strategy advantages from Subsection 3.4.4 hold. Nevertheless, an introduction of fault tolerance comes with changes worth recording.

Firstly, the data collected show that as the fault tolerance increases, the optimal t-SLOTS configuration t for a fixed k also increases slowly toward $\frac{n}{k}$. This phenomenon is observable only when the synchronization probability converges to one (see Table 4.2), and its differences are already of magnitude 10^{-4}. Moreover, the adapted configuration still results in the expected number of transmissions during synchronization not being greater than for the BT strategy.

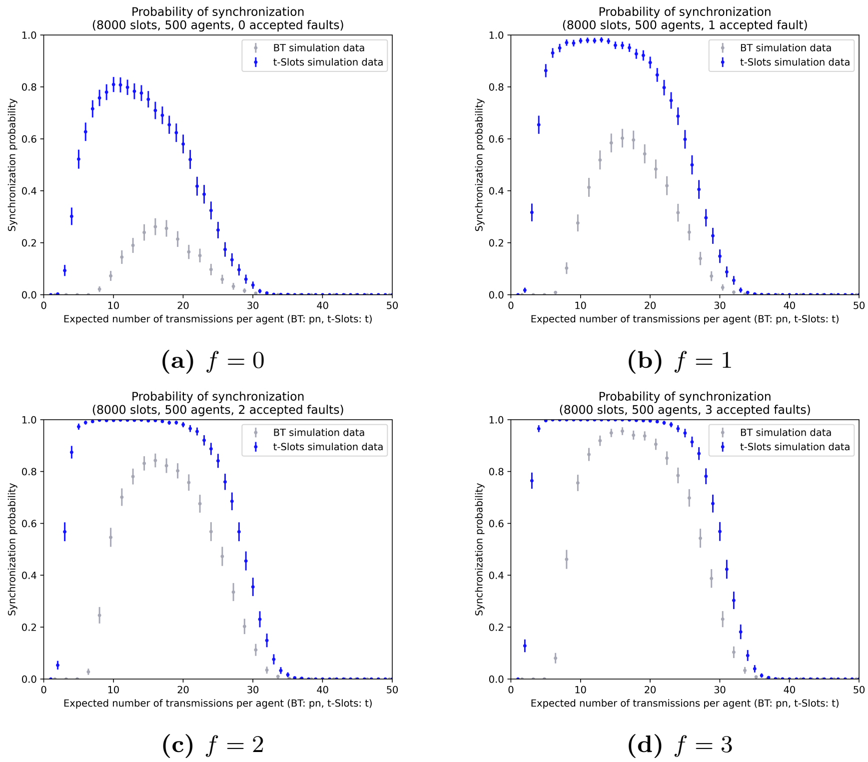
Secondly, as inferred in the previous subsection, the generic agent's configuration (when the network size is unknown) shall aim to be optimal for higher congestion when the fault tolerance rises. This configuration results in a reduced number of transmissions in both strategies, thereby conserving energy.

The key impact of fault-tolerant synchronization is a soaring probability of synchronization (understood as per Definition 4.1). When the network can handle this partial synchronization[2] that excludes some agents, and it neither fails nor has to repeat the round for all participants – significant amounts of energy may be conserved. Splitting the synchronization into smaller fault-tolerant rounds that gradually eliminate the already synchronized agents may lead to further energy conservation.

4.4.4 Scalability

Synchronization of all agents is possible for high-scale (yet pragmatic, according to Definition 2.10) network sizes, as shown in Subsection 3.4.5. However, this is hard and requires relatively low $\frac{k}{n}$ network congestion. On the contrary, the

[2] Which holds especially in use cases like sensor networks (Subsection 2.3.5) and repeatable protocols of access points (Subsection 2.3.7)



(a) $f = 0$ **(b)** $f = 1$

(c) $f = 2$ **(d)** $f = 3$

Figure 4.5: Compared BT and t-SLOTS agents synchronization probabilities approximated by the statistical tests, they are given as a function of the expected number of individual agent's transmissions (pn for BT, t for t-SLOTS). The communication round of $n = 8000$ slots, the network of $k = 500$ agents, and the fault tolerance $f \in \{0, 1, 2, 3\}$ are presented. Vertical ticks represent a margin of error (2,000 tests per sample; $MOE_{0.999}$ [53]).

limited fault tolerance allows the synchronization of thousands of agents with high probability even for congested setups.

Figure 4.5 presents how far an introduction of fault tolerance impacts the network of $k = 500$ agents communicating in the round of $n = 8000$ slots (analyzed before for all-agents synchronization in Subsection 3.4.5). The tolerance $f = 1$ or $f = 2$ is incredibly low, compared to 500 communicating agents, but it is enough for the well-configured t-SLOTS to achieve almost certain synchronization (as stated in Definition 4.1). For $f = 3$, the BT strategy also achieves the maximum synchronization probability of 0.95. Considerably, for any non-zero fault tolerance f the t-SLOTS reliability is consistent, with a wide range of configuration values guaranteeing the synchronization probabilities close to one; when $f = 1$, this range is $7 \leqslant t \leqslant 17$, when $f = 3$, this range is $4 \leqslant t \leqslant 24$.

The impact of tolerance to just 0.2% - 0.6% not synchronized agents, illustrated in Figure 4.5, shows how remarkably hard absolute synchronization of all agents is. Nonetheless, this case is specific due to the low congestion of agents in the communication round $\frac{k}{n} = \frac{1}{16}$. When congestion increases in high-scale networks, the probability of synchronization quickly converges to zero, even for the t-Slots strategy.

Table 4.3: The high-scale network fault-tolerant synchronization probability values (approximated by the statistical tests) in the setup of $n = 8000$ slots, $k = 2000$ agents.

Fault tolerance (f)	Strategy	Synchronization probability for the expected number of transmissions per agent (pn/t) 1	2	3	4	5	6	7	8
200	BT	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
200	t-Slots	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
300	BT	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
300	t-Slots	0.0	**0.31**	**0.67**	**0.11**	0.0	0.0	0.0	0.0
400	BT	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
400	t-Slots	**0.05**	**1.0**	**1.0**	**1.0**	**0.96**	**0.01**	0.0	0.0
500	BT	0.0	**0.02**	**0.94**	**0.99**	**0.94**	**0.4**	0.0	0.0
500	t-Slots	**0.99**	**1.0**	**1.0**	**1.0**	**1.0**	**0.99**	**0.09**	0.0

Table 4.3 displays the BT and the t-Slots synchronization probabilities estimated by statistical tests for the network of $k = 2000$ agents with the communication round of $n = 8000$ slots and the limits of agents having communication failure $f \in \{100, 200, 300, 400, 500\}$. In this case, the congestion of agents is $\frac{k}{n} = \frac{1}{4}$, as high as in the example shown in Figure 4.3 (while the number of agents is 125 times higher). This level of congestion is already extreme for the absolute synchronization of $k = 16$ agents (Figure 4.3a). However, the tolerance to 25% agents failing to communicate leverages the synchronization probability to 0.943 for the t-Slots and 0.766 for the BT (Figure 4.3e). The fault tolerance of 25% in the network of $k = 2000$ agents is equivalent to 500 accepted communication failures. As presented in Table 4.3, this level of fault tolerance allows the BT strategy to achieve a peek synchronization probability equal to 0.99 for the optimal configuration, for 4 expected transmissions per agent. It also guarantees that the t-Slots strategy consistently achieves even higher synchronization probability for a wide range of configurations $1 \leqslant t \leqslant 6$; When the fault tolerance decreases to $f = 400$ (20% of communicating agents), the t-Slots strategy still allows to achieve an almost certain synchronization for $2 \leqslant t \leqslant 4$, whereas the synchronization probability is commonly zero for any BT configurations. For a stricter $f = 300$ (up 15% of agents with a communication failure), the estimated $\Pr[\mathsf{SuccessFt}^{\mathsf{tSlots}}_{8000,2000,3}(300)]$ still equals 0.67; when $f = 200$, the chance of synchronization is nullified for both strategies.

The data collected in Table 4.3 show that the strategies designed for time-constrained synchronization, the t-Slots in particular, can deal with high congestion among thousands of agents. In the depicted case, with high probability, *t*-Slots leads to the synchronization of 80% of agents. If the use case allows the delegation of the remaining agents to another constrained synchronization round, an application of *t*-Slots can lead to reliable time-constrained absolute synchronization within 2 or 3 rounds, even among thousands of agents.

Chapter 5

Summary

5.1 Confirmation of the hypothesis

In Chapter 3, algorithms for the BT strategy and the t-SLOTS strategy were introduced, and formulas of the respective all-agents synchronization probabilities were derived. That enabled a versatile comparison of both strategies under the restrictive demand of faultless synchronization.

In Chapter 4, the problem was expanded and formulas for the fault-tolerant synchronization probability were derived for the BT strategy and the t-SLOTS strategy. That led to comparing both strategies under the relaxed synchronization requirement, which accepts a predefined margin of agents having communication failure.

As provided in Section 3.4 and Section 4.4, the t-SLOTS strategy is superior to the BT strategy. The t-SLOTS achieves significantly higher synchronization probabilities in adjacent configurations, regardless of the agents' number and the communication round length. It has better scalability, resilience to agents' congestion, and energy conservation compared to the BT strategy. This is achieved without compromising resilience to misconfiguration or increasing the computational complexity of agents' algorithms.

The above findings confirm the Research Hypothesis (stated in Section 2.6). It is compelling how the slight change in the approach to selecting the transmission slots impacts the overall outcome and performance of the time-constrained ad hoc network synchronization.

5.2 Further research directions

This work not only validates the stated research hypothesis, but also provides formulas for synchronization probabilities and comments on optimal configurations for both strategies analyzed, BT and t-SLOTS. The above lays a foundation for designing reliable time-critical ad hoc network protocols, offering tools for a priori synchronization probability calculation and adjustment of the configuration parameters. The research results are also a promising starting point for subsequent studies.

The set of ideas for future research routes is comprehensive. Firstly, the finite multi-round approach, eliminating the synchronized part of agents each round as suggested in Section 4.4, deserves analysis as a more complex, promising alternative to the strategies researched. This idea immediately entails incorporating the t-SLOTS strategy in the random-access channel protocols that are not time-constrained; its superior properties imply a possible improvement it may offer to the random access procedures of massive connectivity systems, like 5G [6, 96].

The research focused on strategies for random access to the communication channel in the restrictive yet idealized communication model. This enabled a study on the fundamental mathematical properties of the BT and t-SLOTS strategies to infer widely applicable conclusions. Further mathematical exploration could lead to a derivation of the expected communication round size that provides a given probability of synchronization; the knowledge of the exact formulas enables the development of estimating functions that would allow rapid calculations, even for high-scale networks.

Conversely, the findings could be adapted and tested in relaxed models, leveraging specific signal processing capabilities, such as channel sense or collision detection. To determine the actual performance of the time-constrained synchronization, studies addressing physical communication limitations (like signal dimming, interference, and delay) shall be held, as these limitations may be impactful in some use cases (such as satellite constellations, Section 2.3.4).

The reliability of time-constrained synchronization could be further improved by deviating from pure channel random access techniques to study protocols. Agents could be adaptive with their p/t configuration, based on the communication channel observation (when the channel remains primarily silent, there is a capacity to raise the number of agent's transmissions; when most of the received signal is distorted, probably the congestion is high, and agent shall transmit fewer messages per round). This approach could improve the probability of synchronization in the decentralized distributed network when the synchronization is repetitive; this occurs in sensor networks (Subsection 2.3.5) or drone swarms (Subsection 2.3.3). Another example could be a protocol in which each agent includes in their message an acknowledgment (ACK) of the last successfully received message and holds from any further transmission in the round when the ACK of their previous message is received. This behavior could further improve capacity and reduce energy consumption in single-hop ad hoc networks.

Finally, since the findings and tools delivered have practical value, their industrial application may have a considerable potential. In the era of an emerging IoT device ecosystem (Subsection 2.3.6) and on the threshold of the autonomous drone revolution (Subsection 2.3.3), new compelling channel access challenges may arise from the implementation of real time-critical synchronization systems.

List of Tables

List of Figures

List of Algorithms

Bibliography

[1] Norman Abramson. Development of the alohanet. *IEEE Transactions on Information Theory*, 31(2):119–123, 1985.

[2] Arshee Ahmed and Haroon Rasheed. Analyzing doppler effects in millimeter wave vanet communications using bch coding. In *2023 International Multi-disciplinary Conference in Emerging Research Trends (IMCERT)*, volume I, pages 1–6, 2023.

[3] Ali N. Akansu and Radha Poluri. Walsh-like nonlinear phase orthogonal codes for direct sequence cdma communications. *IEEE Transactions on Signal Processing*, 55(7):3800–3806, July 2007.

[4] Guillermo Alesandroni. Families of finite sets in which no set is covered by the union of the others. *Examples and Counterexamples*, 3:100095, 2023.

[5] Syed R. Ali and Richard S. Wexler. Army warfighter network-tactical (win-t) theory of operation. In *MILCOM 2013 - 2013 IEEE Military Communications Conference*, pages 1453–1461, 2013.

[6] Alaa Omran Almagrabi, Rashid Ali, Daniyal Alghazzawi, Abdullah Al-Barakati, and Tahir Khurshaid. A poisson process-based random access channel for 5g and beyond networks. *Mathematics*, 9(5), 2021.

[7] 5G Americas. Understanding 5G & time critical services, a 5G americas white paper. https://www.5gamericas.org/wp-content/uploads/2022/08/Understanding-5G-Time-Critical-Services-Aug-2022.pdf, 2022. Accessed: 2024-12-08.

[8] Ahmad R.S. Bahai and Burton R. Saltzberg. *Multi-carrier Digital Communications: Theory and Applications of OFDM (2nd ed.)*, chapter 1 - Introduction to Digital Communications, pages 2–16. Information Technology Series. Kluwer Academic / Plenum Publishers, New York, USA, 2006.

[9] Paulo F. C. Barbosa, Bruna A. da Silva, Cleber Zanchettin, and Renato M. de Moraes. Energy consumption optimization for csma/ca protocol employing machine learning. In *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, pages 1–5, 2020.

[10] Alessandro Bazzi, Alberto Zanella, and Barbara M. Masini. A distributed virtual traffic light algorithm exploiting short range v2v communications. *Ad Hoc Networks*, 49:42–57, 2016.

[11] Mounir Belattar, D Benatia, and Malek Benslama. Aloha system performance in satellite communication. *Canadian Journal on Electrical and Electronics Engineering*, 2, 08 2011.

[12] Mihir Bellare, Thomas Ristenpart, Phillip Rogaway, and Till Stegers. Format-preserving encryption. In *Selected Areas in Cryptography*, pages 295–312, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[13] M. Boulkenafed, B. Hughes, R. Meier, G. Biegel, and V. Cahill. Providing hard real-time guarantees in context-aware applications: Challenges and requirements. In *Fourth IEEE International Symposium on Network Computing and Applications*, pages 119–127, 2005.

[14] Zhijun Cai, Mi Lu, and Xiaodong Wang. Distributed initialization algorithms for single-hop ad hoc networks with minislotted carrier sensing. *IEEE Transactions on Parallel and Distributed Systems*, 14(5):516–528, May 2003.

[15] Enrico Casini, Riccardo De Gaudenzi, and Oscar Del Rio Herrero. Contention resolution diversity slotted aloha (crdsa): An enhanced random access scheme for satellite access packet networks. *IEEE Transactions on Wireless Communications*, 6(4):1408–1419, 2007.

[16] Prabhu Chandhar, Danyo Danev, and Erik G. Larsson. Massive mimo as enabler for communications with drone swarms. In *2016 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 347–354, 2016.

[17] Robert W. Chang. Synthesis of band-limited orthogonal signals for multichannel data transmission. *The Bell System Technical Journal*, 45(10):1775–1796, 1966.

[18] Zhengchuan Chen, Yifan Feng, Zhong Tian, Yunjian Jia, Min Wang, and Tony Q. S. Quek. Energy efficiency optimization for irregular repetition slotted aloha-based massive access. *IEEE Wireless Communications Letters*, 11(5):982–986, May 2022.

[19] Hyun-Ho Choi, Howon Lee, and In-Ho Lee. Carrier sensing multiple access with collision resolution (csma/cr) protocol for next-generation wireless lan. In *2013 Fifth International Conference on Ubiquitous and Future Networks (ICUFN)*, pages 220–225, 2013.

[20] Jinho Choi. Noma-based random access with multichannel ALOHA. *IEEE Journal on Selected Areas in Communications*, 35(12):2736–2743, 2017.

[21] Nikumani Choudhury and Moustafa M. Nasralla. A proposed resource-aware time-constrained scheduling mechanism for dsme based iov networks. In *2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall)*, pages 1–7, 2021.

[22] Jacek Cichoń, Maciej Gebala, and Marcin Zawada. Fault tolerant protocol for data collecting in wireless sensor networks. In *2017 IEEE Symposium on Computers and Communications (ISCC)*, pages 483–486, 2017.

[23] Jacek Cichoń, Mirosław Kutyłowski, and Patryk Stopyra. Creating small ad hoc networks: Swift presence notification strategies. *Vehicular Communications*, 45:100694, 2024.

[24] Jacek Cichoń, Mirosław Kutyłowski, and Marcin Zawada. Adaptive initialization algorithm for ad hoc radio networks with carrier sensing. *Theoretical Computer Science*, 402(1):16–28, 2008. Algorithmic Aspects of Wireless Sensor Networks.

[25] Jacek Cichoń, Mirosław Kutyłowski, and Marcin Zawada. Power of discrete nonuniformity - optimizing access to shared radio channel in ad hoc networks. In *2008 The 4th International Conference on Mobile Ad-hoc and Sensor Networks*, pages 9–15, Dec 2008.

[26] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*, chapter 11 - Hash Tables, pages 224–229. The MIT Press, 2nd edition, 2001.

[27] Grinberg Darji. An introduction to algebraic combinatorics. https://www.cip.ifi.lmu.de/˜grinberg/t/21s/lecs.pdf. Acc: 2023-01-23.

[28] Frank den Hallander. *Probability theory: The coupling method*, chapter 2 - Basic theory of coupling, pages 10 – 15. Mathematical Institute, Leiden University, Leiden, The Netherlands, 2009.

[29] Luc Devroye. *Non-Uniform Random Variate Generation*, chapter 12 - Random Sampling, page 612. Springer-Verlag, 1986.

[30] Christoph Enneking, Felix Antreich, Manuel Appel, and André L. F. de Almeida. Pure pilot signals: How short can we choose gnss spreading codes? *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*, 2019.

[31] José J. Garcia-Luna-Aceves. Carrier-sense multiple access with collision avoidance and detection. In *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, MSWiM ’17, page 53–61, New York, USA, 2017. Association for Computing Machinery.

[32] Mohammad Hossein Sedighi Gilani, Iman Sarrafi, and Maghsoud Abbaspour. An adaptive csma/tdma hybrid mac for energy and throughput improvement of wireless sensor networks. *Ad Hoc Networks*, 11(4):1297–1304, 2013.

[33] Ali Grami. *Introduction to Digital Communications*, chapter 11 - Communication Networks, pages 457–491. Academic Press, Boston, USA, 2016.

[34] Harald Haas and Stephen McLaughlin. A dynamic channel assignment algorithm for a hybrid tdma/cdma-tdd interface using the novel ts-opposing technique. *IEEE Journal on Selected Areas in Communications*, 19(10):1831–1846, 2001.

[35] Sarah-Michelle Hammer, Vamsi Addanki, Max Franke, and Stefan Schmid. Starlink performance through the edge router lens. New York, USA, 2024. Association for Computing Machinery.

[36] Mona Bakri Hassan, Elmustafa Sayed Ali, Rania A. Mokhtar, Rashid A. Saeed, and Bharat S. Chaudhari. 6 - nb-iot: concepts, applications, and deployment challenges. In Bharat S. Chaudhari and Marco Zennaro, editors, *LPWAN Technologies for IoT and M2M Applications*, pages 119–144. Academic Press, 2020.

[37] Mona Bakri Hassan, Elmustafa Sayed Ali, Rania A. Mokhtar, Rashid A. Saeed, and Bharat S. Chaudhari. 6 - nb-iot: concepts, applications, and deployment challenges. In Bharat S. Chaudhari and Marco Zennaro, editors, *LPWAN Technologies for IoT and M2M Applications*, pages 119–144. Academic Press, 2020.

[38] Anne M. Helmenstine. How many atoms exist in the universe? https://www.thoughtco.com/number-of-atoms-in-the-universe-603795, 2019. Accessed: 2024-09-20.

[39] J. Herman, K. Dilcher, R. Kucera, and J. Simsa. *6 - The Inclusion-Exclusion Principle*, pages 66 – 68.

[40] Jeremy Horwitz. The definitive guide to 5G low, mid, and high band speeds. https://venturebeat.com/mobile/the-definitive-guide-to-5g-low-mid-and-high-band-speeds/, 2019. Accessed: 2024-10-15.

[41] Ahmadreza Jame, Marco Rapelli, and Claudio Casetti. Reducing pollutant emissions through virtual traffic lights. *Computer Communications*, 188:167–177, 2022.

[42] Manuel Janssen, Andreas Busboom, Udo Schoon, Gerd Von Colln, and Carsten Koch. A mobile open infrastructure network protocol (moin) for localization and data communication in uwb based wireless sensor networks. pages 409–414, 2014.

[43] Dapeng Jiang, Yongjie Pang, and Zaibai Qin. Cooperation of autonomous underwater vehicle fleet based on moos-ivp. In *The 2010 IEEE International Conference on Information and Automation*, pages 1330–1335, 2010.

[44] D. Karimou and J.F. Myoupo. An energy-saving algorithm for the initialization of single hop mobile ad hoc networks. *2005 Systems Communications (ICW'05)*, pages 153–158, Aug 2005.

[45] Phil Karn. Maca-a new channel access method for packet radio. In *9th ARRL Computer Networking Conference*, pages 134–140, 1990.

[46] Md. Rajibur Rahaman Khan and Vyacheslav Tuzlukov. Null-steering beamforming for cancellation of co-channel interference in cdma wireless communication system. In *2010 4th International Conference on Signal Processing and Communication Systems*, pages 1–5, 2010.

[47] Ahmed M. Khedr, Ahmed Salim, Pravija Raj P V, and Walid Osamy. Mwcrsf: Mobility-based weighted cluster routing scheme for fanets. *Vehicular Communications*, 41:100603, 2023.

[48] Evgeny Khorov, Anton Kiryanov, Andrey Lyakhov, and Giuseppe Bianchi. A tutorial on ieee 802.11ax high efficiency wlans. *IEEE Communications Surveys & Tutorials*, 21(1):197–216, 2019.

[49] Leonard Kleinrock and Fouad A. Tobagi. Packet switching in radio channels: Part i – carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, 23(12):1400–1416, 1975.

[50] Selahattin Kosunalp and Yunus Kaya. Iot-tdma: A performance evaluation of tdma scheme for wireless sensor networks with internet of things. *Concurrency and Computation: Practice and Experience*, 34(21), 2022.

[51] Paras Kumar and Kunwar Babar Ali. Intelligent traffic system using vehicle to vehicle (v2v) vehicle to infrastructure (v2i) communication based on wireless access in vehicular environments (wave) std. In *2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, pages 1–5, 2022.

[52] J.P.S. Kung, G.C. Rota, and C.H. Yan. *Combinatorics: The Rota Way*, chapter 3 - Partially ordered sets and lattices, pages 108 – 109. Cambridge Mathematical Library. Cambridge University Press, 2009.

[53] David C. LeBlanc. *Statistics: Concepts and Applications for Science*, chapter 7 - Hypothesis Testing, Statistical Inference, and Estimation, pages 154–164. Jones and Bartlett, 2004.

[54] Anton Ledergerber, Michael Hamer, and Raffaello D'Andrea. A robot self-localization system using one-way ultra-wideband communication. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3131–3137, 2015.

[55] Carlos Ley-Bosch, Itziar Alonso-González, David Sánchez-Rodríguez, and Miguel A. Quintana-Suárez. Analysis of the effects of the hidden node problem in ieee 802.15.7 uplink performance. In *2015 International Conference on Computer, Information and Telecommunication Systems (CITS)*, pages 1–5, 2015.

[56] Richard T. B. Ma, Vishal Misra, and Dan Rubenstein. An analysis of generalized slotted-aloha protocols. *IEEE/ACM Transactions on Networking*, 17(3):936–949, 2009.

[57] Vivek Mangal and Peter R. Kinget. Clockless, continuous-time analog correlator using time-encoded signal processing demonstrating asynchronous cdma for wake-up receivers. *IEEE Journal of Solid-State Circuits*, 55(8):2069–2081, 2020.

[58] W. Mansouri, K. Mnif, F. Zarai, M. S. Obaidat, and L. Kamoun. Capacity analysis of heterogeneous wireless networks under sinr interference constraints. In *2013 International Conference on Wireless Information Networks and Systems (WINSYS)*, pages 1–7, 2013.

[59] Guowang Miao, Jens Zander, Ki Won Sung, and Slimane Ben Slimane. *3 - Medium access control*, page 27–64. Cambridge University Press, 2016.

[60] Yuqi Mo, Claire Goursaud, and Jean-Marie Gorce. On the benefits of successive interference cancellation for ultra narrow band networks : Theory and application to iot. 2017.

[61] Alaa Muqattash and Marwan Krunz. Cdma-based mac protocol for wireless ad hoc networks. pages 153–164, 2003.

[62] Diaganth R. Murthy. Quora: Is there any limit to the number of users that can connect to a mobile tower (bts)? answered by a software developer of Nokia Networks. https://www.quora.com/Is-there-any-limit-to-the-number-of-users-that-can-connect-to-a-mobile-tower-BTS/answer/Diganth-R-Murthy, 2016. Accessed: 2024-09-29.

[63] T. Nakanishi and T. Ikegami. Throughput performance of cdma-aloha in s-band land mobile satellite and stratospheric platform channels. In *11th IEEE International Symposium on Personal Indoor and Mobile Radio Communications. PIMRC 2000. Proceedings (Cat. No.00TH8525)*, volume 2, pages 1085–1089 vol.2, Sep. 2000.

[64] K. Nakano and S. Olariu. Randomized initialization protocols for ad hoc networks. *IEEE Transactions on Parallel and Distributed Systems*, 11(7):749–759, July 2000.

[65] NXP Semiconductors N.V. Nxp 180 mhz cortex-m3 32-bit mcus lpc18sxx: Microcontrollers with integrated security to protect code and data. https://www.nxp.com/docs/en/brochure/75017637.pdf, 2015. Accessed: 2024-10-02.

[66] Insu Oh, Taeeun Kim, Kangbin Yim, and Sun-Young Lee. A novel message-preserving scheme with format-preserving encryption for connected cars in multi-access edge computing. *Sensors*, 19(18), 2019.

[67] Debashish Pal, M Sai Deepak, N. Sireesha, and S. Sadiya. Implementation of csma/cd and link state routing protocols for dijkstra algorithm. In *2023 1st International Conference on Circuits, Power and Intelligent Systems (CCPIS)*, pages 1–6, 2023.

[68] Athanasios Papoulis. *Probability, Random Variables, and Stochastic Processes*, chapter Bernoulli Trials, pages 43 – 47. McGraw-Hill, New York, 2nd edition, 1984.

[69] Sagar Pawar, Lokesh Bommisetty, and T. G. Venkatesh. A high capacity preamble sequence for random access in 5g iot networks: Design and analysis. *International Journal of Wireless Information Networks*, December 2022.

[70] Sofie Pollin, Mustafa Ergen, Sinem Coleri Ergen, Brano Bougard, Liesbet Van Der Perre, Ingrid Moerman, Ahmad Bahai, Pravin Varaiya, and Francky Catthoor. Performance analysis of slotted carrier sense ieee 802.15.4 medium access layer. *IEEE Transactions on Wireless Communications*, 7(9):3359–3371, 2008.

[71] Tommaso Polonelli, Davide Brunelli, Achille Marzocchi, and Luca Benini. Slotted ALOHA on LoRaWAN-design, analysis, and deployment. *Sensors (Switzerland)*, 19(4), 2019.

[72] Ngan-Cheung Pun. Tdma channel access scheduling with neighbor indirect acknowledgment algorithm (nbia) for ad-hoc networks, CA Patent CA2656656A1, Jun 2007.

[73] Putri Rahmawati, Muhammad Imam Nashiruddin, and Muhammad Adam Nugraha. Capacity and coverage analysis of 5g nr mobile network deployment for indonesia's urban market. In *2021 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT)*, pages 90–96, 2021.

[74] Christian Renner, Florian Meier, and Volker Turau. Policies for predictive energy management with supercapacitors. In *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 314–319, 2012.

[75] Injong Rhee, Ajit Warrier, Mahesh Aia, Jeongki Min, and Mihail L. Sichitiu. Z-mac: A hybrid mac for wireless sensor networks. *IEEE/ACM Transactions on Networking*, 16(3):511–524, 2008.

[76] Lawrence G. Roberts. Aloha packet system with and without slots and capture. *SIGCOMM Comput. Commun. Rev.*, 5(2):28–42, 1975.

[77] Md. Sadman Sakib and Md. Rayhan Ferdous. Performance evaluation of mc-cdma system for the next generation wireless networks. In *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, pages 1–5, Dec 2018.

[78] Balawal Shabir, Muazzam A. Khan, Anis U. Rahman, Asad W. Malik, and Abdul Wahid. Congestion avoidance in vehicular networks: A contemporary survey. *IEEE Access*, 7:173196–173215, 2019.

[79] Bharat Shrestha, Ekram Hossain, and Kae Won Choi. Distributed and centralized hybrid csma/ca-tdma schemes for single-hop wireless networks. *IEEE Transactions on Wireless Communications*, 13(7):4050–4065, 2014.

[80] Omar Shrit, Steven Martin, Khaldoun Alagha, and Guy Pujolle. A new approach to realize drone swarm using ad-hoc network. In *2017 16th Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, pages 1–5, 2017.

[81] Mujdat Soyturk and D. Turgay Altilar. Reliable real-time data acquisition for rapidly deployable mission-critical wireless sensor networks. In *IEEE INFOCOM Workshops 2008*, pages 1–6, 2008.

[82] Murray R. Spiegel, Seymour Lipschutz, and John Liu. *Mathematical Handbook of Formulas and Tables: Third Edition*, chapter The Binomial Formula and Binomial Coefficients, pages 7 – 9. The McGraw-Hill Companies, New York, USA, 3rd edition, 2009. (Originally published 1968).

[83] STMicroelectronics. Introduction to cryptographic library with stm32. https://wiki.st.com/stm32mcu/wiki/Security:Introduction_to_the_cryptographic_library_with_STM32, 2024. Accessed: 2024-10-02.

[84] Patryk Stopyra. Slotted random access channel: Agents TS selection implementation in C. https://github.com/V-I-S/prompt-mac-agents, 2023. Accessed: 2024-09-28.

[85] Patryk Stopyra. Slotted random access channel: Communication success calculation library. https://github.com/V-I-S/prompt-mac-calculator-v2, 2024. Accessed: 2024-12-08.

[86] Patryk Stopyra. Slotted random access channel: Distributed protocol simulator. https://github.com/V-I-S/prompt-mac-simulator-v2, 2024. Accessed: 2024-12-08.

[87] Alain Sultan. 5g system overview. https://www.3gpp.org/technologies/5g-system-overview, 2022. Accessed: 2024-10-15.

[88] Xinghua Sun and Lin Dai. A comparative study of quadratic backoff and binary exponential backoff in ieee 802.11 dcf networks. In *2011 45th Annual Conference on Information Sciences and Systems*, pages 1–6, 2011.

[89] Tomasz Szydło and Robert Brzoza-Woch. Predictive power consumption adaptation for future generation embedded devices powered by energy harvesting sources. *Microprocessors and Microsystems*, 39(4):250–258, 2015.

[90] Vinh-Thong Ta and Amit Dvir. A secure road traffic congestion detection and notification concept based on v2i communications. *Vehicular Communications*, 25:100283, 2020.

[91] Y. Hung Tam, Robert Benkoczi, Hossam S. Hassanein, and Selim G. Akl. Channel assignment for multihop cellular networks: Minimum delay. *IEEE Transactions on Mobile Computing*, 9(7):1022–1034, 2010.

[92] Andrew Thornburg, Tianyang Bai, and Robert W. Heath. Performance analysis of outdoor mmwave ad hoc networks. *IEEE Transactions on Signal Processing*, 64(15):4065–4079, 2016.

[93] Sudarshan Vasudevan, Micah Adler, Dennis Goeckel, and Don Towsley. Efficient algorithms for neighbor discovery in wireless networks. *IEEE/ACM Trans. Netw.*, 21(1):69–83, feb 2013.

[94] Tamas Wágner, Tamas Ormándi, Tamas Tettamanti, and Istvan Varga. Spat/map v2x communication between traffic light and vehicles and a realization with digital twin. *Computers and Electrical Engineering*, 106:108560, 2023.

[95] Mingjun Xu, Yizhou He, Cheng Wang, Gaofeng Cui, and Weidong Wang. Doppler rate estimation scheme for ufmc based leo satellite communication system. In *2016 IEEE International Conference on Communication Systems (ICCS)*, pages 1–5, 2016.

[96] Bei Yang, Fengsheng Wei, Xiaoming She, Zheng Jiang, Jianchi Zhu, Peng Chen, and Jianxiu Wang. Intelligent random access for massive-machine type communications in sliced mobile networks. *Electronics*, 12(2), 2023.

[97] R. K. Rao Yarlagadda and John E. Hershey. *Hadamard matrix analysis and synthesis: with applications to communications and signal/image processing*, chapter 7 - The Hadamard Transform and Its Fast Implementation, pages 17–22. Springer Science+Business Media, New York, USA, 1997.

[98] Kasım Sinan Yıldırım, Amjad Yousef Majid, Dimitris Patoukas, Koen Schaper, Przemyslaw Pawelczak, and Josiah Hester. Ink: Reactive kernel for tiny batteryless sensors. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, SenSys '18, page 41–53, New York, USA, 2018. Association for Computing Machinery.