ADAPTIVE MULTI-RAT COMMUNICATIONS *for* ULTRA-RELIABLE INTERNET *of* THINGS

Diploma thesis

Vojtěch Hauser

June 2019

This thesis was done at the Department of Telecommunications of the Faculty of Electrical Engineering, Czech Technical University in Prague, as part of the Communication Systems and Networks branch of the Electronics and Communication master's programme under the supervision of Lukáš Vojtěch.

This document was set using the LaTeX typesetting system originally developed by Leslie Lamport, based on TeX designed by Donald Knuth. Vector graphics are written in Ti*k*Z and drawn by PGF by Till Tantau. The bibliography was processed using BibLaTeX created by Philipp Lehman, *et al*.

The body text is set 11pt on a 26pc measure with Robert Slimbach's Minion Pro acting both as the text and display typeface. Monospaced text is typeset in `Inconsolata` designed by Raph Levien. The Computer Modern created by Donald Knuth is used throughout.

The use of sidenotes instead of footnotes and figures spanning both the textblock and fore-edge margin was inspired by *Beautiful Evidence* by Tufte [123].

DECLARATION

I hereby submit for the evaluation and defence the master's thesis elaborated at the CTU in Prague, Faculty of Electrical Engineering.

I have no relevant reason against using this schoolwork in the sense of § 60 of Act No. 121/2000 Coll. on Copyright and Rights Related to Copyright and on Amendment to Certain Acts (the Copyright Act).

I declare I have accomplished my final thesis by myself and I have named all the sources used in accordance with the Guideline on ethical preparation of university final theses.

In Prague, 23rd May, 2019.

Vojtěch Hauser, author

MASTER'S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Hauser Vojtěch** Personal ID number: **420063**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Telecommunications Engineering**

Study program: **Electronics and Communications**

Branch of study: **Communication Systems and Networks**

II. Master's thesis details

Master's thesis title in English:

Adaptive Multi-RAT Communications for the Ultra-reliable Internet of Things

Master's thesis title in Czech:

Adaptivní komunikace s více přístupovými rádiovými technologiemi pro ultra-spolehlivý Internet věcí

Guidelines:

Study the concept of "interface diversity" and assess the distinctive features of radio access technologies (RATs) designed for the Internet of Things (IoT) applications used in practice. Develop a model of a multi-interface system capable of exploiting the "interface diversity" by switching between RATs or using more than one RAT at a time. Explore and extend/design quality-of-service-aware algorithms/strategies enabling efficient utilization of the available RATs and create tools for performance evaluation thereof.

Bibliography / sources:

[1] K. Chebrolu and R. Rao, "Communication using multiple wireless interfaces," in 2002 IEEE Wireless Communications and Networking Conference Record. WCNC2002 (Cat. No.02TH8609), IEEE, 2002. DOI:10.1109/wcnc.2002.993516.14

[2] N. Himayat, S. Yeh, A. Y. Panah, S. Talwar, M. Gerasimenko, S. Andreev, and Y. Koucheryavy, "Multi-radio heterogeneous networks: Architectures and performance," in 2014 International Conference on Computing, Networking andCommunications (ICNC), Feb. 2014, pp. 252–258.doi:10.1109/ICCNC.2014.6785341.

[3] J. J. Nielsen and P. Popovski, "Latency analysis of systems with multiple interfaces for ultra-reliable m2m communication," in 2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), IEEE, Jul. 2016. doi:10.1109/spawc.2016.7536857.

[4] N. Niebert, A. Schieder, J. Zander, and R. Hancock, Ambient Networks. John Wiley & Sons, Ltd, Apr. 2007. DOI:10.1002/9780470511046.

Name and workplace of master's thesis supervisor:

Ing. Lukáš Vojtěch, Ph.D., Department of Telecommunications Engineering, FEE

Name and workplace of second master's thesis supervisor or consultant:

Date of master's thesis assignment: **11.02.2019** Deadline for master's thesis submission: **24.05.2019**

Assignment valid until: **20.09.2020**

Ing. Lukáš Vojtěch, Ph.D. Supervisor's signature	Head of department's signature	prof. Ing. Pavel Ripka, CSc. Dean's signature

III. Assignment receipt

The student acknowledges that the master's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the master's thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt Student's signature

ABSTRACT

Title Adaptive Multi-RAT Communications for the Ultra-reliable Internet of Things

Keywords adaptive scheduling, interface diversity, multi-radio networks, heterogeneous networks, Internet of things

Institution Czech Technical University in Prague

Supervisor Lukáš Vojtěch

The promise of the future Internet is to deliver a new artificial reality dexterously interwoven with the daily business of life. A ubiquitous intelligence to be interacted with, to be touched[1] and felt. Reliably and in real-time.

While every design aspect of a novel wireless technology may be subordinated to the pursuit of ultra-reliability and low-latency, the inherent features of the massively deployed *constrained devices*[2] forming a foundation of these networks beg for a different approach. Following the recent research into the fifth-generation mobile communications, the use of multiple interfaces presents itself as an opportunity to optimise both latency and reliability whilst avoiding the necessity of introducing any engineering constraints into the lowest communication layers.

The utilisation of multiple communication interfaces[3] simultaneously to achieve higher throughput has been in the focus of researchers for decades with robust and scalable solutions making their way into the industry. However, although the notion of *reliability through redundancy* has been adopted in the systems-theoretic literature[4] by the mid-20$^{\text{th}}$ century, adaptive schemes to maintain a defined *quality of service* by controlling the redundancy of connectivity are still scarce and limited.

Despite the general tendency of analytical approaches to modelling complex stochastic systems to be affected by the *curse of dimensionality*, the author succeeds at developing two diverse yet complementary frameworks that by dealing with the fundamental problems of queueing and scheduling enable exploitation of the *interface diversity* in a controlled, adaptive manner. The queueing-theoretic model provides a complete response time analysis of a heterogeneous parallel queueing system while the scheduling algorithmic solutions present an efficient approach applicable to a general class of multi-interface systems.

By supporting a wide range of system configurations, the provided[5] high-performance numerical simulation tools form an infrastructure employable for network planning and dimensioning purposes.

1. See the ITU's vision of the *Tactile Internet* [60].

2. As defined by the RFC 7228 [14]. The possibility of high density deployments raises the issues of coexistence, interference mitigation, etc.

3. Often termed as *link aggregation*, *port trunking*, *link bundling*, *channel bonding*, etc.

4. The majority of the pioneering works published in the 1950s were related to the aerospace engineering.

5. Prior to the open-source release, the tools used throughout the work are available upon request.

RÉSUMÉ

Titre Communications multi-RAT adaptatives pour l'Internet des objets à haute fiabilité
Mots-clés ordonnancement adaptative, diversité d'interfaces, réseaux multiradio, réseaux hétérogènes, Internet des objets
Société Université Technique de Prague
Encadrant Lukáš Vojtěch

La promesse de l'Internet futur est de délivrer une nouvelle réalité artificielle adroitement entremêlée avec les activités quotidiennes de la vie. Une intelligence ubiquitaire avec laquelle interagir, qu'on pourra toucher[6] et sentir. De manière fiable et en temps réel.

Tandis que chaque aspect de conception d'une nouvelle technologie sans fil peut être subordonné à la poursuite de l'extrême fiabilité et d'une faible latence, les fonctions inhérentes aux *appareils contraints* massivement déployés[7] formant le fondement de ces réseaux sollicitent une approche différente. Suite aux récentes recherches sur la cinquième génération des communications mobiles, l'usage d'interfaces multiples se présente comme une opportunité pour optimiser la latence et la fiabilité tout en évitant la nécessité d'introduire toute contrainte d'ingénierie au sein des couches de communication les plus basses.

L'utilisation d'interfaces de communication multiples[8] pour réaliser simultanément un débit plus élevé a été un point focal pour les chercheurs pendant des décennies, avec des solutions robustes et évolutives qui arrivent dans le secteur. Cependant, bien que la notion de *fiabilité à travers la redondance* ait été adoptée dans la littérature sur la théorie des systèmes au milieu du XXème siècle, les schémas adaptatifs pour maintenir une *qualité de service* définie en contrôlant la redondance de connectivité sont encore rares et limités.

Malgré que la tendance générale des approches analytiques pour modéliser de complexes systèmes stochastiques est d'être affectée par *le fléau de la dimension*, l'auteur réussit à développer deux cadres divers mais complémentaires qui, en traitant des problèmes fondamentaux de file d'attente et de planification, permettent l'exploitation de *la diversité d'interfaces* d'une manière contrôlée et adaptée. Le modèle théorie de file d'attente fournit l'analyse du délai de réponse d'un système parallèle hétérogène de file d'attente tandis que les stratégies algorithmiques de planification présentent une approche efficace applicable à une classe générale de systèmes d'interfaces multiples.

En supportant une vaste gamme de configurations systèmes, les outils[9] de simulation numérique à haute performance forment une infrastructure employable dans les cadres de dimensionnement et de planification des réseaux.

6. Se référer à la vision de l'ITU sur *l'Internet tactile* [60].

7. Tel que défini par le RFC 7228 [14]. La possibilité des déploiements à haute densité soulève les problèmes de la coexistence, de l'atténuation des interférences etc.

8. Ouvent dénommées *link aggregation* (agrégation des liens), *port trunking* (connexion des ports parallèle), *link bundling* (le groupement des liens), *channel bonding* (agrégation de canaux) etc.

9. Avant la diffusion open-source, les outils utilisés à travers le travail sont disponibles sur demande.

ABSTRAKT

Název práce	Adaptivní komunikace s více přístupovými rádiovými technologiemi pro ultra-spolehlivý Internet věcí
Klíčová slova	adaptivní scheduling, diverzita rozhraní, více-rádiové sítě, heterogenní sítě, Internet věcí
Instituce	České vysoké učení technické v Praze
Vedoucí práce	Lukáš Vojtěch

Budoucnost Internetu je spojena s vizí vytvoření nové umělé skutečnosti šikovně provázané s každodenním životem. Všudepřítomné inteligence, se kterou bude možné vstupovat do interakcí, které se bude možné dotknout[10] a na naopak jí cítit. Spolehlivě a v reálném čase.

Zatímco každý aspekt návrhu nové bezdrátové technologie může být podřízen úsilí o dosažení vyjímečné spolehlivosti a nízké latence, bytostné vlastnosti četně rozmísťovaných *omezených zařízení*[11] tvořících základ těchto sítí vyžadují jiný přístup. V návaznosti na současný výzkum v oblasti mobilních sítí páté generace se využití vícero přístupových rádiových technologií jeví jako příležitost pro optimalizaci latence i spolehlivosti, která nevyžaduje zásah do nejnižších komunikačních vrstev.

Využívání více komunikačních rozhraní[12] souběžně k dosažení vyšší propustnosti bylo předmětem výzkumu v průběhu dekád, které přinesly robustní škálovatelná řešení dnes již užívaná v praxi. Ačkoliv se pojem *spolehlivost prostřednictvím redundance* v literatuře věnované teorii systémů[13] začal vyskytovat již v polovině minulého století, přizpůsobivá technologická řešení řídící redundanci konektivity pro udržení požadované *kvality služby* jsou stále spíše vyjímečná a jejich možnosti jsou omezené.

Navzdory obecné náchylnosti analytických technik modelování složitých stochastických systémů trpět *prokletím dimenzionality*, se autorovi zdařilo vyvinout dva rozličné, byť doplňující se, postupy, které prostřednictvím nástrojů teorií front a rozvrhování umožňují využít *diverzitu rozhraní* řízeně a přizpůsobivě. Přístup vycházející z teorie front nabízí úplnou analýzu odezvy systému různorodých souběžných front, zatímto přístup inspirovaný teorií rozvrhování představuje efektivní algoritmucké řešení velmi obecné třídy systémů s více přístupovými rádiovými technologiemi.

Díky podpoře široké škály uspořádání komunikačního systému, poskytnuté[14] numerické simulační nástroje tvoří infrastrukturu použitelnou pro plánování a dimenzování sítí.

10. Vizte vizi tzv. *Hmatového Internetu* zformulovanou ITU [60].

11. Ve smyslu definice “constrained device” v RFC 7228 [14]. Umožnění jejich hustého nasazení je podmíněno vyřešením otázek koexistence, omezení vzájemného rušení, a dalších.

12. V literatuře označováné jako *link aggregation* (agregace linek), *port trunking* (slučování portů), *link bundling* (seskupování linek), *channel bonding* (sdružování kanálů), atd.

13. Většina prvotních prací publikovaných v 50. letech minulého století se věnovala technologií z oblasti aerokosmonoutiky.

14. Do zveřejnění open-source verze jsou nástroje použité v rámci práce dostupné na vyžádání.

CONTENTS

Just-In-Time Performance Optimization

Summary

Appendices

LIST OF FIGURES

LIST OF ABBREVIATIONS

DBPSK Differential Binary Phase Shift Keying . . . 7

DNS Domain Name System . . . 23

DPSK Differential Phase Shift Keying . . . 4

DQPSK Differential Quadrature Phase Shift Keying . . . 4

DRR Deficit Round Robin . . . 44

DRX Discontinuous Reception . . . 8

DSSS Direct Sequence Spread Spectrum . . . 4–6

ECTS European Credit Transfer and Accumulation System . . . xiii

EDF Earliest Deadline First . . . 21, 58

eMTC Enhanced MTC . . . 8

ETSI European Telecommunications Standards Institute . . . 4

E-UTRA Evolved UMTS Terrestrial Radio Access . . . 8

FCFS First Come First Served . . . 19, 21, 26, 53, 54, 60, 63

FEC Forward Error Correction . . . 7

FFD Fully Function Device . . . 4–6

FHSS Frequency Hopped Spread Spectrum . . . 4, 6

FSK Frequency Shift Keying . . . 5

GFSK Gaussian Frequency Shift Keying . . . 4, 5, 7

GTS Guaranteed Time Slots . . . 5

HART Highway Addressable Remote Transducer Protocol . . . 6

HETNET Heterogeneous Network . . . 9, 43

HPC High-Performance Computing . . . 54

ID Identification . . . 3

IEEE Institute of Electrical and Electronics Engineers . . . xii, 4–7, 9, 45, 46

IETF Internet Engineering Task Force . . . 6, 8

IoT Internet of Things . . . 3, 4, 6, 8, 9, 14

IP Internet Protocol . . . ix, 6–8

ISM Industrial, Science, Medical . . . 3, 4, 6

ITU International Telecommunication Union . . . i–iii, 3

RFD Reduced Function Device 4, 6

RFID Radio Frequency Identification . . . . . xxii, 3, 5

RPMA Random Phase Multiple Access 6, 8

RR Radio Regulations 3

SCHC Static Context Header Compression 8

SIG Special Interest Group 4

SRD Short Range Device 4

SUN Smart Utility Network 5

TCP Transmission Control Protocol 23

TDMA Time Division Multiple Access 6

TVWS Television White Space 5

TX Transmission 56

ULP Ultra Low Power 5

UMTS Universal Mobile Telecommunications System x

UNB Ultra Narrow Band 7

URLLC Ultra-Reliable Low Latency Communications 9, 11, 12

UWB Ultra Wide Band 5

WiFi IEEE 802.11*x* protocol 4

WLAN Wireless Home Area Network 4

WMN Wireless Mesh Network 43, 45, 46

WPAN Wireless Personal Area Network . . . . . xi, 4

PREFACE

This is a diploma thesis of 30 ECTS credits[15] following the work that builds upon the preceding internship done at the Connectivity Section of the Department of Electronic Systems, Technical Faculty of IT and Design, Aalborg University (AAU) in Denmark as part of the double degree master's programme between Faculty of Electrical Engineering, Czech Technical University in Prague (CTU) and the French EURECOM as outlined in fig. 1.

15. Prescribed to six months of full-time work.

"Adapt or perish, now as ever, is nature's inexorable imperative."
— H. G. Wells [135, p. 19]

This manuscript is a continuation of author's systematic interest in decentralized adaptive communication systems and complex network modeling. The main objective is to enable development of adaptive communication schemes exploiting multiple-interface-enabled devices by extending the state-of-the-art techniques for analysis of such systems.

For most of the time, the author was able to base his research on the foundations laid down during his studies at CTU and EURECOM which, towards the end of the work, permitted the author to diverge into uncharted territories and to explore various unconventional approaches.

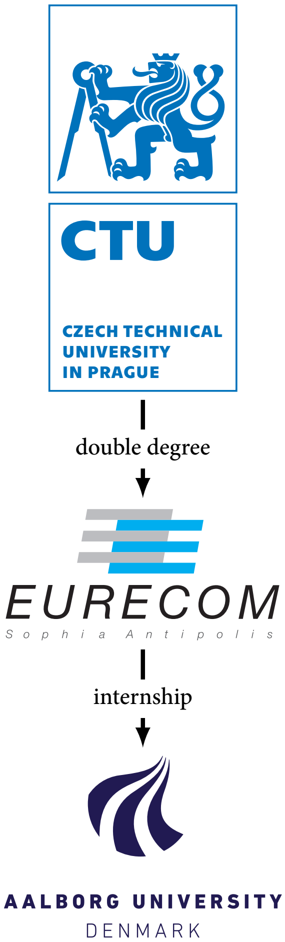


Figure 1: Participating institutions and their relationships.

The author expresses his gratitude to CTU's Lukáš Vojtěch for allowing him to conduct this research under his auspices. The author is especially grateful for the freedom he was given to do this work. Equally, the author is grateful to AAU's Jimmy Jessen Nielsen who has been an never-ending source of encouragement and inspiration both as a scientist and as a teacher. Many other members of the Section were of great help by providing valuable feedback and creating a sense of supportive community. Among others, the author would like to thank Petar Popovski, Kristoffer Stern, Nasrin Razmi, Badiaa Gabr, and René Brandborg for providing thoughtful input.

On a personal level, the author was extraordinarily fortunate to find a kindred spirit in Section's secretary, Charlotte Kattrup Madsen, who's commitment to maintaining warm and welcoming atmosphere within the Section is an inspiration for years to come. The author owes his sincere gratitude to his closest ones for bearing with him through the thick and thin of his studies. If it wasn't for their support, this endeavour would not have been possible.

Last but not least, the author would like to thank the AAU and the CTU for supporting his studies financially.

Vojtěch Hauser
Prague, Czech Republic
Thursday, 23rd May, 2019

PART I

PROLOGUE

1

TECHNOLOGY OVERVIEW

If we had computers that knew everything there was to know about things–using data they gathered without any help from us–we would be able to track and count everything, and greatly reduce waste, loss and cost. (...) We need to empower computers with their own means of gathering information, so they can see, hear and smell the world for themselves, in all its random glory.

— Kevin Ashton [8]

"*When wireless is perfectly applied the whole earth will be converted into a huge brain, which in fact it is, all things being particles of a real and rhythmic whole.*"

— Nikola Tesla [70]

The term INTERNET OF THINGS coined by Kevin Ashton, a co-founder and executive director of the MIT Auto-ID Center, at a 1999 presentation for Procter & Gamble linking the introduction of Radio Frequency Identification (RFID) technology to the supply chain[1] with the Internet [8], has become an umbrella term for several technologies that enable provision of Internet-based services through a global network infrastructure supported by uniquely identifiable, programmable electronic devices[2] attached to physical things which provide sensing and actuation capabilities. As the technology and the visions behind it are continuously evolving, a common or unified definition is regrettably still lacking[3].

As the discussion of the whole Internet of Things (IoT) technology stack is far beyond scope of this thesis, the subsequent introductory sections merely overview the most prominent access technologies and refer the interested reader to the brilliant view of the "big picture" by Porter and Heppelmann [106].

Restricting the focus to Radio Access Technologies (RATs), the candidates suitable for IoT applications are manifold. Each with inherent strengths and weaknesses that predestine them for particular use or, as the author argues, empower a creative designer to exploit the diversity of using multiple technologies at once.

1.1 SHORT RANGE TECHNOLOGIES

Short-range RATs typically operate in unlicensed frequency range[4]. To ensure fair access, the regulations may impose limits on transmitted power, duty cycle and/or to require additional Medium Access Control (MAC) protocols such as Listen Before Talk (LBT). Though some wireless standards supplement the

1. See *"The history of* RFID*"* [81].

2. Taxonomy [25] and typology [73] of smart objects is studied extensively in the literature.

3. As demonstrated by a systematic literature review [55] following the evolution of the concept.

4. Unlicensed bands are portions of the radio spectrum reserved internationally for the use of radio frequency without a license, as would otherwise be required. In Europe, unlicensed bands fall into two categories: *Industrial, Science, Medical (ISM) bands* designated by the International Telecommunication Union (ITU) Radio Regulations (RR), RR 5.138 [29, p. 60], RR 5.150 [29, p. 65] and RR 5.280 [29, p. 90] for the "operation of equipment or appliances designed to generate and use locally radio frequency energy for industrial, scientific, medical, domestic or similar purposes, excluding applications in the field of telecommunications"...

…Due to the increasing congestion of the radio spectrum the ISM bands are nowadays shared with telecommunication systems (which are required to be tolerant to the ISM emissions). In Europe, the operation of non-ISM devices is regulated by technical recommendations by European Conference of Postal and Telecommunications Administrations (CEPT) and standards by European Telecommunications Standards Institute (ETSI). *Short Range Device (SRD) bands* defined by ETSI EN 300 220 [26] which defines additional bands in the 25 MHz to 1000 MHz range. In Europe, their use is regulated by CEPT.

efforts by interference detection and avoidance mechanisms (e.g. collaborative IEEE 802.19.1 [57], non-collaborative Bluetooth Adaptive Frequency Hopping (AFH) [13, sec. 7.2]), detection of other protocols is challenging [137, 138].

The short-range wireless standards can be grouped by operating range into Wireless Personal Area Network (WPAN) technologies centered around an individual person's workspace (tens of meters, e.g. Bluetooth), Wireless Home Area Network (WLAN) technologies operated within a building (lower hundreds of meters, e.g. WirelessHART) and Wireless Local Area Network (WLAN) technologies operated within a boundary up to a kilometre in radius (e.g. WiFi HaLow).

1.1.1 *Bluetooth*

The development of Bluetooth defined as "a short-range radio link between a cellular phone and nearby electronic devices, supporting both voice and data" began at Ericsson in 1994 as a replacement for an earlier unpromising project Cornelius [46]. Over time, the Bluetooth IoT-oriented offering has diversified, adding Alternate MAC / PHY (AMP) in 2009, Bluetooth Low Energy (BLE) in 2010 and explicit IoT support in Bluetooth 5 (2016) offering faster, longer range and connectionless operations for low energy devices [110].

To meet the regulatory requirements of the 2.4 GHz ISM radio band and to provide robustness against fading and interference the Bluetooth employs Frequency Hopped Spread Spectrum (FHSS)[5] GFSK modulation in the basic rate and $\pi/4$-DQPSK or 8DPSK in enhanced data rate mode.

5. Direct Sequence Spread Spectrum (DSSS) is used in BLE.

In addition to core protocols, the Bluetooth Special Interest Group (SIG) defines *application profiles* layer that resides on top of the Bluetooth Core Specification [13]. Bluetooth profiles describe how a subset of core functionality can be adopted to support specific types of operations (services).

1.1.2 *IEEE 802.15.4-based technologies*

IEEE 802.15.4 defines the PHY and MAC layers for "low-data-rate wireless connectivity with fixed, portable, and moving devices with no battery or verylimited battery consumption requirements typically operating in the personal operating space of 10 m" [59].

The initial 2003 version defined two DSSS-based PHYs. The subsequent revisions and amendments introduce multiple PHYs, modulation techniques, regional and application-specific extensions as summarized[6] by table 1.1 on the next page.

6. This paragraph and the table highlights key points from the work by Ramonet and Noguchi [109] which provides a more detailed overview.

The IEEE 802.15.4 standard classifies devices into two categories:

Fully Function Device (FFD) has all the capabilities such as routing, association and formation of a network.

> **Coordinator** is a specific FFD responsible for network coordination, time synchronization and association services.

Reduced Function Device (RFD) is typically a very simple end-device with reduced capabilities rendering it unable to operate as a coordinator. A RFD associates only with a single FFD.

Revision	Year	Features
IEEE 802.15.4-2003	2003	Initial release. Two DSSS-based PHYS: global O-QPSK at 2450 MHz and two regional using BPSK in 868 MHz (Europe) and 915 MHz (United States) bands.
IEEE 802.15.4-2006	2006	New modulation schemes: ASK and O-QPSK (regional PHYS).
IEEE 802.15.4a	2007	New PHYS: Ultra Wide Band (UWB) using Burst Position Modulation (BPM) and BPSK, and Chirp Spread Spectrum (CSS).
IEEE 802.15.4c	2009	New PHYS in 780 MHz band using O-QPSK and MPSK, respectively, to be used in China.
IEEE 802.15.4d	2009	New PHYS in 950 MHz band using GFSK and BPSK, respectively, to be used in Japan.
IEEE 802.15.4-2011	2011	Compiles amendments a through d.
IEEE 802.15.4f	2012	New UWB-based PHY optimized for low complexity RFID transmitters (tags).
IEEE 802.15.4g	2012	New Smart Utility Network (SUN) PHYS optimized for smart grid systems.
IEEE 802.15.4j	2013	New PHY optimized for medical applications.
IEEE 802.15.4k	2013	New PHYS optimized for Low Energy, Critical Infrastructure Monitoring (LECIM) applications.
IEEE 802.15.4m	2014	New PHY operating in Television White Space (TVWS) frequencies.
IEEE 802.15.4p	2014	New PHYS optimized for Rail Communications Control (RCC) systems.
IEEE 802.15.4-2015	2015	Compiles amendments f through p.
IEEE 802.15.4p	2016	New PHYS optimized for Ultra Low Power (ULP) systems (peak power consumption for the PHY below 15 mW).
IEEE 802.15.4u	2016	New PHY in 865 MHz to 867 MHz band using either SUN FSK, OFDM or O-QPSK to be used in India.
IEEE 802.15.4t	2017	New PHYS capable of supporting data rates up to 2 Mbit s^{-1}.
IEEE 802.15.4v	2017	Regional frequency changes in SUN, LECIM and TVWS PHYS.
IEEE 802.15.4x	2019	Update of the SUN OFDM PHY to support data rates up to 2.4 Mbit s^{-1} and definition of additional channel plans.

Table 1.1: Evolution of the IEEE 802.15.4 PHY specification. See [109] for more details.

The standard describes network formation procedures for star (master-slave), peer-to-peer (FFDS form a multi-hop network) and mesh[7] topologies. The network operates with or without the help of periodic beacon messages transmitted by the coordinator which provide synchronization using *superframes*. Each superframe consists of sleep period and an active period which is further divided into Contention Access Period (CAP) when the nodes contend to access the channel using slotted Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) and Contention Free Period (CFP) divided into Guaranteed Time Slots (GTS) assigned by the coordinator. Since IEEE 802.15.4e[8], the MAC protocol is defined by "MAC behaviours" corresponding to various application domains[9].

7. Although mesh topology is said to be supported, little to no details are provided.

8. See [78] for in depth survey of IEEE 802.15.4e.

9. E.g. Low Latency Deterministic Networks (LLDN) optimized for factory automation.

Thread [122] is a low-power mesh networking technology designed to securely connect low-power IoT / smart home applications to the Internet (cloud) – i.e. Thread nodes are IP-addressable. Thread employs the encapsulation and header compression mechanisms[10] developed by Internet Engineering Task Force (IETF) IPv6 over LoWPAN (6LoWPAN) group that enable packet transport over IEEE 802.15.4 (and other networks[11]).

10. RFCs: 4919 [79] (overview, assumptions, problem statement, and goals), 4944 [95] (IPv6 over IEEE 802.15.4), 6282 [54] (header compression), 6775 [118] (neighbor discovery optimization).

11. E.g. RFC 7668 [103], IPv6 over BLE.

ZigBee is a low-data-rate, low-power, wireless technology targeted at automation and remote control applications. ZigBee builds on top of the IEEE 802.15.4 DSSS PHY – using the global 2.4 GHz (BPSK) and the regional (O-QPSK) ISM bands – and MAC. ZigBee defines the network layer specifications for star, cluster tree, peer-to-peer and mesh network topologies and provides a framework for application programming in the application layer. Notably, the network layer provides multi-hop routing, route discovery and maintenance, security and joining/leaving a network, with consequent short (16 bit) address assignment to newly joined devices[12]. To encourage interoperability, ZigBee Alliance and OEM vendors define application profiles as domain spaces of related applications and devices[13].

12. Device classification is simmilar to IEEE 802.15.4 – ZigBee end-device corresponds to a RFD, ZigBee router to a FFD and ZigBee coordinator to the IEEE 802.15.4 coordinator.

13. E.g. the "Home Automation" profile defines devices such as lights and switches, remotes, wall outlets and thermostats. More recently, the alliance has begun unifying the profiles into a cross-platform (e.g. Thread) open interoperability and certification standard "Dotdot" [143].

WirelessHART is an adaptive, self-organizing mesh[14] technology based on the Highway Addressable Remote Transducer Protocol (HART) industrial automation protocol that provides backwards compatibility with the legacy analogue 4–20 mA systems. WirelessHART uses the IEEE 802.15.4 2.4 GHz FHSS-based O-QPSK PHY. In contrast to, e.g. ZigBee, it defines a custom Time Division Multiple Access (TDMA)-based MAC whose distinctive features include strict 10 ms time slot, network wide time synchronization, channel hopping, channel blacklisting, and AES-128 security [121].

14. The network, however, maintains a centralized network manager entity responsible for configuring the network, maintaining routing information and traffic scheduling.

Other IEEE 802.15.4-based technologies include ISA 100.11a and MiWi.

1.2 LONG RANGE TECHNOLOGIES

Multi-hop routing based on inherently short-range technologies over large area may reduce energy consumption [131]. However, the complexity required to ensure sufficient reliability and low latency may prove restrictive [3]. The development of Low Power Wide Area Network (LPWAN) technologies[15] is, therefore, key enabler of extensive IoT applications such as agricultural and large-scale industrial monitoring and smart cities.

15. See RFC 8376 [30].

Due to the propagation characteristics of lower frequencies that allow signals to propagate further and offer superior building penetration than higher frequencies, making lower frequency systems potentially less costly to deploy [63], most LPWAN technologies use the regional sub-GHz unlicensed bands[16]. To further extend the operating range, high sensitivity receivers[17] and more efficient antennae may be used. However, since typical regulation imposes restrictions on the radiated power, the link budget asymmetries between the simple, inexpensive end-devices and "base stations" with high-gain antennas may allow only uplink connectivity [3].

16. Ingenu / On-Ramp Wireless, a proponent of 2.4 GHz LPWAN technology Random Phase Multiple Access (RPMA), argues that the 9 dB advantage of an 900 MHz system is easily offset by antena diversity (estimates 8 dB), global availability of the ISM band, more relatex duty cycle restrictions and more bandwidth available (80 MHz) [41].

17. LoRa and SigFox allow Maximum Coupling Loss (MCL) of 157 and 160 dB, respectively [104].

Since the current LPWAN networks are topologically similar to cellular networks, the possibility integration of LPWAN"base stations" into existing cellular deployments makes the subscription-based business model viable for traditional telecom operators.

1.2.1 LoRa

Long Range (LoRa) is an emerging technology operating in the regional sub-GHz unlicensed bands based on a proprietary[18] modulating scheme derived from a CSS modulation[19]. In addition to resistance to Doppler effect, multipath fading, inherent narrowband interference mitigation and high processing gain of the CSS modulation, by ensuring the inter-symbol phase continuity which simplifies timing and frequency synchronization, the LoRa modulation enables the use of low-cost devices inevitably exhibiting frequency and/or timing offset [97]. The technology supports variation of the modulation parameters[20] and thus enables trading the data rate[21] for robustness, or coverage [115]. LoRa gateways may further exploit the (quasi-)orthogonality of spreading factors to process multiple signals simultaneously.

In contrast to LoRa modulation, LoRa Wide Area Network (LoRaWAN) [88] is an *open* MAC standard inspired by IEEE 802.15.4, albeit much simpler. Topologically, LoRaWAN-based networks form a star-of-starts topology in which *gateways*[22] relay messages between *end-devices* (LoRa link) and a central *network server* (IP link) which routes the packets from each device of the network to the associated *application server*. To make the system more robust to interference, the end-devices exploit channel-hopping [132] at each transmission.

LoRaWAN defines three classes of operation according to the trade-off between downlink network latency and battery life:

Class-A (All) Following a uplink transmission initiated by the end-device (ALOHA protocol), the end-device opens two short downlink receive windows. That is, downlink traffic must wait for an uplink.

Class-B (Beacon) In addition to class-A operation, the end-device opens extra receive windows at times specified by a beacon message.

Class-C (Continuously listening) End-device is almost[23] continuously listening downlink traffic implying the highest power consumption.

As analysed in author's earlier work [48], a prominent feature of LoRaWAN is the Adaptive Data Rate (ADR) capability enabling the network to optimize the rate-robustness trade-off by adjusting the data rate of individual end-devices. The LoRaWAN provides AES-based security fundamentals (symmetric cryptography) enabling secure join procedures and end-to-end encrypted communication, however LoRaWAN is susceptible to several classes of attacks [16].

18. Patents: [113], [97], [114].

19. See [43] for overview of the technique.

20. spreading factor, $\text{SF} \in \{7, \dots, 12\}$, which determines length of the chirp; bandwidth BW whose values are regionally dependent; code rate of the Forward Error Correction (FEC) code, $\text{CR} \in \{1, \dots, 4\}$

21. $R_b = \frac{\text{SF} \cdot \text{BW}}{2^{\text{SF}}} \frac{4}{4 + \text{CR}}$

22. Gateways are transparent to the end-devices and there is no coordination – e.g. uplink packets are received by all gateways and later de-duplicated at the network server.

23. Not when transmitting.

1.2.2 SigFox

SigFox is an proprietary LPWAN technology optimized for uplink communications that operates in the regional sub-GHz unlicensed bands. The PHY relies on a Ultra Narrow Band (UNB)[24] modulation[25] which allows for low-power operation. Each uplink message is subsequently retransmitted on two different frequencies across a much larger band to provide diversity and to enable high-density deployments.

The *modus operandi* of *the* SigFox network is very similar to LoRa – the device-initiated transmissions are received by nearby base stations[26], which are transparent to the end-devices, and forwarded through the global cloud-based core network to the applications via Application Programming Interfaces (APIs)

24. As low as $100\,\text{bit}\,\text{s}^{-1}$ – this allows a SigFox device to carry only up to six 12 B messsages per hour.

25. GFSK for the downlink and more bandwidth-efficient and robust Differential Binary Phase Shift Keying (DBPSK) for the uplink [42]

26. On average, by three base stations [119] – which enables cooperative reception and spatial diversity.

or a web portal of the Operations Support Systems (OSS). While the technology initially supported only unconfirmed uplink transmissions (as described above), the recent releases support bidirectional communications. After a defined time interval following an uplink transmission, a receive window is opened at the frequency calculated[27] from the first frequncy used for the "uplink". In contrast to LoRa, loss of "confirmed" messages does not result in retransmissions.

27. A known delta is added.

In contrast to LoRaWAN, SigFox focuses solely on commercial networks where the infrastructure is managed by the SigFox company and offered as a service on a subscription-basis. To allow IP-based transport over its network, Sigfox has implemented Static Context Header Compression (SCHC) of IPv6 headers which is currently being standardized by the IETF [93, 144].

1.2.3 *Cellular Technologies*

This section is based on the brilliant chapter following the evolution of LTE connectivity for IoT by [24].

The interest to integrate the IoT-related massive Machine-Type Communicationss (MTCs) into the existing based cellular networks driven by the desire to reuse the infrastructure has been recognized in 3rd Generation Partnership Project (3GPP) Release 8 (i.e. Long-Term Evolution (LTE)). MTC was based on Category 1 providing the lowest capabilities, but failing to meet the power consumption and cost requirements of a typical IoT application. Release 12 has attempted to address these concerns by defining Category 0 providing power-saving mode and adding support for half-duplex communications potentially reducing transceiver complexity.

Release 13 introduced a new category (LTE-M1) for Enhanced MTC (eMTC) supporting reduced bandwidth, transmit power and support for downlink transmission modes while achieving longer battery life through extension of period when the device may sleep[28], extended coverage[29], and significant transceiver cost reduction through restriction in system bandwidth. Furthermore, two competing initiatives aimed at developing a LTE-based narrowband technologies were started only to be merged in November 2015 into a single standard – Narrow Band LTE (NB-LTE)[30].

28. That is, allowing for longer Discontinuous Reception (DRX) timers.

29. Providing improvement up to 15 dB which allows deployments in remote locations, indoors, etc.

30. Standardized in June 2011

To a great extent, NB-LTE is based on a non-backward-compatible variant of Evolved UMTS Terrestrial Radio Access (E-UTRA), that provides improved indoor coverage, support for massive number of low throughput devices, low delay sensitivity, ultra low device cost[31], low device power consumption and optimised network architecture.

31. The reduction of device complexity, compared with Cat. 1, is up to 90%.

A large scale[32] simulation of NB-LTE, LoRa and SigFox networks performed at the AAU [82] has shown that NB-LTE provides superior coverage compared to its competitors.

32. The simulation is based on the Telenor's 2G, 3G and 4G deployment across North Jutland region in Denmark (7800 km^2).

Other long-range technologies include the Weightless "family of technologies"[33], RPMA highlighted earlier as an LPWAN technology operating in the 2.4 GHz band, and Telensa aimed at smart sities. An overview of the former two is provided in [43].

33. Weightless-N, Weightless-P and Weightless-W variants.

2

WORK OVERVIEW

"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."

— Mark Weiser [133]

The utilisation of multiple communication interfaces to achieve higher throughput and fault tolerance has been a focus of researchers for decades with robust and scalable solutions making their way into the telecommunication industry. Historically, both physical (e.g. MIMO systems or IEEE 802.11n channel bonding [56]) and logical (e.g. IEEE 802.1AX link aggregation [58] or 3GPP dual/multi connectivity [64]) approaches were part of integrated, homogeneous network technologies. To satisfy the exponential increase in demand for capacity driven by the widespread adoption of new generation of devices and services in a cost-efficient way, a new paradigm of Heterogeneous Networks (HETNETS) [4] consisting of nodes with different characteristics (e.g. frequency, coverage) has emerged. More recently, the topics of multi-RAT HETNETS and introduction of heterogeneous networking into 5G are gaining momentum [5].

The promise of the future Internet is to deliver a new artificial reality dexterously interwoven with the daily business of life. A ubiquitous intelligence to be interacted with, to be touched[1] *and felt. Reliably and in real-time.*

In the effort to deliver on this vision, the development of the 5th evolution of the mobile networks has become the single most significant driving factor of research into low-latency high-reliability communications. The air latency requirements[2] for 5G in Ultra-Reliable Low Latency Communications (URLLC) services are as stringent as 1 ms and 0.5 ms in IMT-2020 and 3GPP [62, 1, sec. 7.5], respectively. Although, currently, 3GPP defines reliability requirement as $1 - 10^{-5}$ for a short packet with user-plane latency of 1 ms [1, sec 7.9], the envisioned use-cases for URLLC may impose requirements up to $1 - 10^{-9}$, e.g. in the context of industrial automation [53].

1. *See the ITU's vision of the* Tactile Internet *[60].*

2. Market leaders' such as Ericsson [28] and Qualcomm [108] foresee a stricter requirement of "1 ms end-to-end latency".

Figure 2.1: Distance reachable in 1 ms by signals starting in Prague and traveling at the speed of light. Other types of delay in the network are not taken into account. MAP: [124], IDEA: [125]

As the industry transition towards massive, adaptive, opportunistic networks the opportunity to exploit multiple interfaces as an additional *degree of diversity* to achieve significant improvements in network capacity, latency, and fault tolerance while avoiding the need to design dedicated 5G wireless interfaces becomes compelling. However, especially in the context of constrained networks forming the foundations of the IoT, the requirements of coexistence and scalability have to be factored in. Since the "number of connected devices is forecasted to grow at 109% per year until 2023, reaching more than one billion active connections" [61] with most of the prominent technologies operating in the unlicensed spectrum, the efficient utilization of the spectrum[3] is evermore vital and thus injection of redundancy must be performed in a controlled manner to avoid overall network performance degradation through interference, unfair medium access and increased energy consumption.

3. The use of the radio spectrum (e.g. transmission power and duty cycle) is regulated in most countries. E.g. [22, 26].

2.1 LITERATURE REVIEW

This section provides a basic review of the literature related to the arguments set forth above. The main parts of this thesis are prefaced by individual introductory chapters providing an overview of the relevant literature.

2.1.1 *Redundancy–Latency Trade-Off*

1	Title	📰 More is less [130]
	Author/s	Vulimiri, Michel, Godfrey, and Shenker
	Year	2012

In their early work Vulimiri et al. make the general case for exploiting redundancy in communication networks. Although the analysis is simplistic, the intuitive insights including the following are worth noting.

The power of redundancy is to reduce uncertainty without having to anticipate the cause of that uncertainty. *(...) [W]e may not have multiple truly independent options, so that exceptional conditions are correlated across options. (...) Instead of choosing paths (or servers) based on their mean performance, it may be beneficial to pick options that are as independent as possible.*

Although redundancy enables system designers to overcome greater uncertainties, the benefit it may provide is bound by the measure of correlation between the degrees of diversity[4]. Therefore, knowledge of the dependence structure of the "causes of uncertainty" is vital should redundancy be exploited efficiently.

4. Both endogenous (e.g. antenna element spacing [84])) and exogenous (e.g. correlated fading [19] and interference [47]) phenomena were studied in the literature.

[R]edundancy involves more work and expense. (...) [T]he overall increase in utilization may be small, since latency-sensitive tasks are often a small fraction of the total network load. (...) [Redundancy] may be worthwhile to automate.

Redundancy implies higher utilization (and, in turn, may lead to increase in latency, energy consumption, interference, etc.). Careful evaluation of the trade-offs possibly employing traffic classification is therefore warranted.

Just as redundancy makes it harder for nature to cause a problem, so it is harder for attackers to cause a problem.

When coupled with diversity[5], redundancy provides an opportunity to ensure intrusion tolerance, detection [20], interception avoidance [85] and other security services.

5. For discussion about relationship of the two, see [86].

2	Title	Ambient Networks [99]
	Author/s	Niebert, Schieder, Zander, and Hancock
	Year	2007

The AN IST was one of the most notable large-scale projects aiming to update the 1980s notion of being "always connected" [45] for the 21th century[6].

6. That is, towards being "Always Best Connected (ABC)" over multiple access technologies that best suit the user's needs or profile at any time.

The eighth chapter on multi-radio access broadly introduces the concept of heterogeneous connectivity based on cooperation of actors across RATs and different user/operator domains. The AN project considers a possibility of joining existing networks into a composed network where the user agents would "continuously evaluate different access offerings from a technical as well as from a business perspective to obtain the best 'value for money'."

Section 8.2 outlines the research areas related to multi-RAT:

– *access selection* – development of Quality of Service (QoS)-aware adaptive RAT selection strategies possibly allowing multi-hop access

– *load sharing and admission control* – load management

– *vertical handover* procedures including context transfer mechanisms, in addition to optimized signaling mechanisms

– *network discovery and advertising* – development of energy-efficient network/service advertisement (signaling) enabling the competitive networking model as well as reducing the "time and effort required for terminals to scan for new candidate networks" and services

The authors emphasizes the trade-off between generality of proposed solutions and the possibilities for optimization. Solutions "should be general enough to be applicable to any combination of RATs, while at the same time allowing the use of sufficiently detailed information to deliver high performance".

Section 8.2.4 of the work may be of special interest to a reader interested in related early research projects.

3	Title	Wireless Access for Ultra-Reliable Low-Latency Communication: Principles and Building Blocks [105]
	Author/s	Popovski, Nielsen, Stefanovic, Carvalho, Strom, Trillingsgaard, Bana, Kim, Kotaba, Park, and Sorensen
	Year	2018

Popovski et al. provide an overview of communication-theoretic URLLC. URLLC traffic is expected to be sporadic and low-throughput, yet demanding an exceptional reliability ($1 - 10^{-5}$ to $1 - 10^{-9}$) and time localization (sub-1 ms latency). As such URLLC is envisioned to be used in mission-critical domains such as safety, industrial automation and vehicular communications.

To achieve ultra-reliability, the authors call for use of proper stochastic channel models to estimate the "known unknowns" (e.g. channel estimation) and to bound the "unknown unknowns" (e.g. estimating noise variance upper bound). Furthermore, a proper MAC rules must be designed and enforced (activity modelling, grant-free access, QoS-aware scheduling).

Next, the authors focus on protocol-level analysis and argue that the joint probability of success is higher than when the data transmission, metadata transmission and the auxiliary procedures are executed independently. Following a results from blocklength information, the authors suggest that joint encoding multiple messages into larger blocks could benefit from higher achievable rates and thus enable the use of shorter frames giving rise to a power-efficiency and frame duration trade-off. The latest developments by 3GPP including the introduction of minislot-scheduling (downlink) and pre-configured/semi-persistent scheduling (uplink) are outlined, as well as two approaches for less predictable traffic patterns.

- *grant-free access* – random access exploiting either use of higher power/wider bandwidth or the successive interference cancellation to achieve multi-packet reception. Compressed sensing with "training" sequences prepended to downlink transmissions "may be used for activity detection and channel estimation".

- *coordinated grant-free access* – users are assigned access patterns according to the scheduling policy and these are infrequently updated

The benefits of massive MIMO are discussed. While there are promising features such as high SNR links due to the array gain and quasi-deterministic links with spectacular spatial multiplexing capacity, if the environment allows for enough scattering (→ diversity paths), the requirements for instantaneous Channel State Information (CSI) acquisition and complex signal processing lead to trade-off "between spatial diversity and multiplexing, as well as latency".

Finally, following benefits of base station densification are outlined.

- *short association distance* – reduced propagation loss

- *per-user resource allocation increase* – can be utilized for latency reduction

- *multiple associations* – extra associations for URLLC users

4	Title	Ultra-Reliable and Low-Latency Wireless Communication: Tail, Risk and Scale [9]
	Author/s	Bennis, Debbah, and Poor
	Year	2018

Bennis, Debbah, and Poor review the state-of-the art literature related to URLLC, identify key enablers of URLLC and their inherent trade-offs. An overview of relevant tools and methodologies used in various fields from finance to particle physics applicable to analysis of URLLC systems is provided. Their usefulness is illustrated by a set of case studies.

The crucial argument made by Bennis, Debbah, and Poor is that in the age of URLLC the QoS metrics can no longer been analysed in terms of simple averages and that the focus to more complex statistical analysis is warranted.

2.2 THE PROBLEM

This section provides a concise overview of the system model, discussion of mapping of its abstract components to the features of realistic multi-RAT systems and a general overview of the analysis techniques explored later in the text.

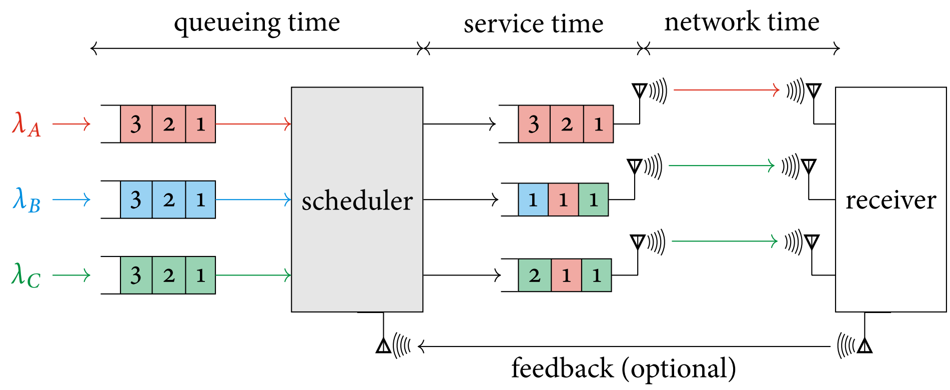


Figure 2.2: Multi-RAT system overview

A multi-RAT system can be interpreted as a sequence of per-stream input queues, a scheduler assigning input blocks to output interfaces and per-interface output queues. The scheduler may introduce redundancy to improve latency, redundancy, throughput, etc. The input blocks, which generally may be heterogeneous with respect to length may be fragmented (e.g using rateless codes). The scheduler may consider specific per-stream QoS requirements and prioritize certain traffic classes.

In addition, one may consider the feedback channel from the receiver. The introduction of feedback channel enables reactive schemes such as early cancellation of pending transmissions of replicas after reception of one replica has been acknowledged or, on the other hand, introducing redundancy after a reception has not been acknowledged after certain time interval. Nevertheless, feedback may be exploited only in specific circumstances and given the increased complexity, its cost may outweigh the potential benefit.

The main aim of this work is to analyse the system from QoS perspective and to develop strategies that optimize selected metrics. The main approach is to control and adapt redundancy introduced in the scheduler. QoS optimization strategies aim to answer following two fundamental questions:

- Which interfaces to use for each input block (fragment)?
- How much redundancy should be introduced given some constraints?

To analyse such system, it is necessary to develop a tractable model which is "general enough to be applicable to any combination of RATs, while at the same time allowing the use of sufficiently detailed information to deliver high performance"[7].

7. General objective of the Ambient Networks IST project [99].

Although the concept of multi-connectivity has been studied extensively in the recent years as part of the efforts towards "5G networks", the latency-redundancy trade-off has been analysed only sporadically and such works are limited to empirical analysis. Surprisingly, much of tangential analytical work is being carried out in the context of high-performance parallel and cloud computing where controlled introduction of redundancy and redundancy-aware scheduling has been shown to be beneficial (e.g. in systems MapReduce-based systems). Parallel processing systems, unfortunately, typically exhibit features such as homogeneity and exponential service times which greatly simplify their modelling.

To author's best knowledge, the relevant literature on QoS optimization of multi-RAT systems including earlier works by author's supervisors do not take in account queueing delay within the receiver due to e.g. limited throughput of its interfaces or limitations arising from MAC protocols or retransmission schemes. Especially in the context of IoT applications utilizing the unlicensed spectra, the impact of the medium access procedures may often be the dominant factor.

The author proposes to view the latency of a datagram as a sum of

queueing time corresponding to application layer datagram queues managed by a "scheduler" which assigns datagrams to lower layer queues (interfaces) according to a "strategy",

service time corresponding to the delay due to queueing in the service queues (buffers) and the time spent executing the physical or media access layer protocols at respective interfaces and

network time equal to the time elapsed after the actual transmission has completed at the transmitter side and the reception of the datagram at the application layer of the receiver side. For example, such delay may include routing delays for connections over multi-hop networks.

From the perspective of the transmitter, the service queues are blocked by each datagram for the respective service time duration. On the other hand, the queues are not not directly affected by the network times.

The author has explored two approaches to analyse such system which can be broadly classified as *predictive* and *reactive*, respectively.

2.3 PROPOSED APPROACHES

2.3.1 *Predictive Approach*

The predictive approach relies on queueing-theoretic description of the system which enables per-stream analysis of the expected system response time and reliability. These insights are subsequently used to select the "optimal" scheduling strategy.

While, due to the inherent complexity of the problem, the scheduling strategies were restricted to a family of first-come-first-served disciplines, the scheduler remains free to control the allocation of the interfaces to traffic streams. E.g., in the figure above, stream A is served by all three interfaces while the class B is served only by the middle one.

For a fixed allocation, both the simulation tool, `quesi`, and the analytical framework developed and discussed later in the text, can be used to determine the expected service time of a stream i, $T^{(i)}$, and the probability that a class-i

datagram will be served at the interface s, $P_{i \to s}$. Ultimately, the expected per-class latency $L^{(i)}$ and reliability $R^{(i)}$ can be expressed as

$$L^{(i)} = T^{(i)} + T_s^{(i)} P_{i \to s}, \tag{2.1}$$

$$R^{(i)} = R_s^{(i)} P_{i \to s}, \tag{2.2}$$

where $T_s^{(i)}$ and $R_s^{(i)}$ are the expected latency and reliability, respectively, of a class-i datagram while being served by the interface s.

The simulation framework enables numerical analysis of a very broad range of redundancy queueing systems and as such is well suited for network planning purposes. In order to deliver a tractable model, the analytical framework makes the following assumptions about the system and as such is not as general.

- Per-stream arrival times follow exponential distribution(s) with parameter(s) λ_i which are either known or may be estimated "on-the-fly";
- Per-interface service times follow exponential distribution(s)[8] with mean(s) μ_i which are either known or may be estimated "on-the-fly";
- Per-interface-and-class network times follow general distribution(s) with known or estimable parameters. The means to estimate the network times include (but are not limited to) exploiting a dedicated feedback control channel, protocol features such as immediate acknowledgement replies or occasional transmission of "pilot" datagrams.

8. Error performance of systems where this condition is not fulfilled *exactly* is analysed empirically. In short, the analysis implies that a wide range of non-exponential systems may be modelled using this technique with low approximation error.

Nevertheless, it provides computationally inexpensive closed-form expressions for the quantities listed above and as such may be easily used for adaptive QoS optimization even in significantly computationally constrained systems.

2.3.2 *Reactive Approach*

As the performance characteristics of more behaviourally complex or more dynamic systems could not be predicted "satisfactorily", a reactive approach is warranted.

The general objective is to provide an algorithm solving the combinatorial optimisation problem of allocating "processing" time at a set of interfaces to the individual datagrams entering the system based on the knowledge of the current state of the system and its expected stochastic performance characteristics.

In addition to the predictive approach outlined previously which attempts to completely model the expected behaviour, the problem could be formulated as stochastic resource allocation and scheduling problems. The resources to be allocated represent the time dedicated to serving the traffic by an interface or combination thereof. Once the set of interfaces capable of serving the traffic is identified and the per-interface time allocation required to meet the specified QoS requirements is known, the problem reduces to a classical scheduling problem.

The structural system model is equivalent to the fig. 2.2 on page 13. Internally, the solutions described below view the per-class input queues as a single interleaved queue. In contrast to the predictive approach the system may be required to cancel (remove) certain datagrams from the input queues without ever allocating them to the outbound interfaces.

The reactive approach is an well suited for more dynamic systems and these not compatible with the set of assumptions imposed by the predictive analytical framework. However, the advantage of flexibility is counterbalanced by the cost during execution due to the need to constantly (re-)evaluate scheduling decisions.

2.4 STRUCTURE

The remainder of this thesis follows the structure of the section 2.3 on page 14.

Part II on page 19 discusses the queueing-theoretic system model. Section 4.1 provides formal set of definitions and assumptions used extensively for the response time analysis in section 4.2. As a practical system may not exhibit "exactly exponential" service times, the applicability of the model to non-exponential systems is analysed in section 4.3. In section 4.4 the author provides an empirical insight into the transient behaviour of the queueing system. Section 4.5 is dedicated to answering the crucial question of interface selection – i.e. what is the probability that a given interface will serve the job first if multiple interfaces operate in parallel? Section 4.6 makes a case for using preemptive systems that enable job cancellations. Finally, section 4.7 discusses redundancy structure representation and makes concluding remarks regarding employing the technique for construction of adaptive QoS-optimizing algorithms.

Part III on page 43 follows the development of the just-in-time scheduling model. As the system exhibits qualitatively different behaviour when a queueing effect is significant the section 6.1 on page 48 outlining scheduling considerations consists of two parts: the brief first part discusses scheduling in under-loaded systems. Subsequently, a statistical framework used to develop an adaptive filtering mechanism to cope with excessive traffic is introduced. A scheduling decision process is developed in section 6.2 on page 53. Finally, in section 6.3 the concepts are demonstrated and evaluated using numerical simulation.

Part IV on page 63 summarizes the work and, finally the section 7.1 on page 64 provides an outlook on the future work.

Both main parts II and III are prefaced with introductory sections outlining the state of the art and providing concise summaries of selected, particularly relevant, scholarly works.

The appendices[9] include miscellaneous notes referenced throughout the text to illuminate various concepts (i.e. examples and derivations) collected in appendix A on page A1; and implementation details of the queueing simulator in appendix B on page A5.

9. Page numbers prefixed with the letter "A".

PART II

EX-ANTE PERFORMANCE OPTIMIZATION

3

STATE OF THE ART

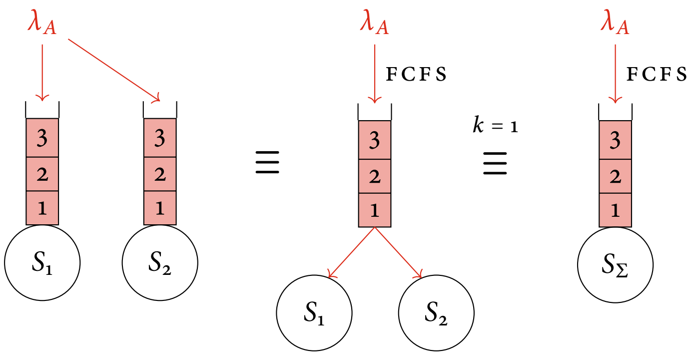


Figure 3.1: "Central queue" analogy of a parallel queueing system.

A multi-RAT system can be seen as a queueing system where each interface is represented as a processor and datagrams scheduled onto an interface are interpreted as being enqueued in a queue leading up to the processor.

Analyses of similar systems found in the literature build upon a simple insight about parallel systems with replicated jobs. If a job is replicated across N interfaces with associated service time random variables $S_1, S_2, \ldots, S_N$, the system will behave as if there was only a central First Come First Served (FCFS) queue with N processors attached to its head. Furthermore, if a job is removed from the system after any of its replicas has been served, the processors could be seen as a single processor with service time random variable $S_\Sigma = \min\{S_1, S_2, \ldots, S_N\}$, i.e. the service time of the fastest server.

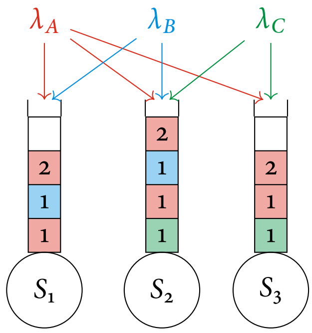


Figure 3.2: General redundancy system

Practical systems may involve multiple traffic classes (colours in the figure above) and arbitrary redundancy structures (coloured arrows *ibidem*) representing of combinatorial assignments of interfaces to traffic classes. Since the response times of each particular class depend on the redundancy structure, a more robust analysis is necessary.

Unfortunately, closed form formulae describing such systems are not known. Assuming similar analogy as in the simplistic case outlined above, the general system may be viewed as being structurally similar to the G/G/k[1]. Very few results for general k are known or even can be approximated (e.g. it can be shown that even under simplifying assumption of exponentially distributed inter-arrival times, the waiting time could not be approximated using first two moments of the service time distribution [44]). Many of the very few approximate solutions for $k > 2$ rely on very restrictive assumptions[2] which are unlikely satisfied by practical systems in question. Another set of approaches is based on the diffusion technique – i.e. interpolation between simpler systems[3]. Extension of these results is not yet possible since exact analysis of simple systems with replication and job cancellations is not available.

1. G/G/k is a Kendall's notation for a queueing system with k servers where inter-arrival times and service times are generally distributed (the respective distribution may be different).

2. Commonly, all processors are assumed to be identical [136] or the utilization (and thus the queueing time) is assumed to be very high. Such assumption allows for modelling the queue length as a reflected Brownian motion [72].

3. For example, Kimura [71] models the G/GI/k system as a combination of D/M/k, M/D/k and M/M/k systems. The "D" and "M" denote deterministic distribution and exponential distribution (M as in Markovian), respectively.

3.1 LITERATURE REVIEW

3.1.1 *Multi-RAT QoS optimization*

5	Title	📰 Communication using multiple wireless interfaces [17]
	Author/s	Chebrolu and Rao
	Year	2002

The work by Chebrolu and Rao is one of the earliest works on QoS optimization through wireless interface diversity.

The authors note that in multi-RAT systems, delay-sensitive applications may suffer from out-of-order delivery (c.f. paper 7). To address the problem, two algorithms are proposed:

- *interface selector* solves minimizes the "cost function" related to bandwidth utilization (appropriate model referenced) constrained on meeting specified QoS goals.

- *earliest delivery first* (EDF) scheduling policy is used in conjunction with the weighted fair queuing algorithm. EDF schedules packets based on the delivery time across the interfaces. Upper bounds on throughput, packet delay, jitter and buffer size required for in-order-delivery to higher layers at client side are derived. (note: proofs to theorems are available in later work [18])

6	Title	Performance Evaluation of Multi-Radio Transmission Diversity: QoS Support for Delay Sensitive Services [140]
	Author/s	Yaver and Koudouridis
	Year	2009

Yaver and Koudouridis discuss the following traffic splitting schemes (packet level) and their impact on various QoS metrics – average packet delay, packet drop probability and goodput.

1. *interface switching* – at each time slot, at most one interface is scheduled to transmit a packet. Traffic scheduling decisions are based on feedback information from the receiver side.

2. *parallel transmission* – the scheduler schedules outbound packets in a round-robin fashion.

3. *parallel transmission with redundancy* – the scheduler schedules each outbound packet on all interfaces (*cloning*).

The system model adheres to the architecture proposed for the Ambient Networks (FP6) project [99] as summarized by Koudouridis et al. [75].

Although simplistically, the authors raise the issue of feedback-based adaptive scheduling. A sophisticated NS2-based HSDPA model is used (code available [11, 27]).

7	Title	Delay optimal concurrent transmissions in multi-radio access networks [142]
	Author/s	Yu, Hua, Li, and Ni
	Year	2015

Yu et al. model traffic flow through a multi-RAT system as an M/G/1 queue. The authors identify the out-of-order delivery as a major contributor to the system latency (c.f. paper [reference not included in this report]) and formulate an optimization problem (DOCT) minimizing the maximum delay across the interfaces. Inbound traffic flow is split into M sub-flows according to the weights obtained as a solution to the DOCT problem and scheduled using a FCFS scheduler. Closed-form solutions are provided for $M = 2$ and a dual decomposition technique is discussed for $M \geq 3$.

The optimization problem formulation requires that the inbound and the net outbound rates be *equal* (no redundancy).

8	Title	📰 Latency analysis of systems with multiple interfaces for ultra-reliable M2M communication [101]
	Author/s	Nielsen and Popovski
	Year	2016

Nielsen and Popovski consider an M2M device that has $N = 3$ (generally not independent) communication interfaces at its disposal. Assuming prior knowledge of QoS characteristics (quasi-CDF of the reliability in terms of latency) of each interface and a Continous Time Markov Chain (CTMC) modeling failure interdependence, both obtained empirically, the authors analyze three traffic splitting schemes complementing those discussed by Yaver and Koudouridis [140]:

1. *cloning* across *all* interfaces – whole message is cloned
2. *selecting 2 out of the* 3 interfaces – each transmits 2/3 of the information
3. *weighted splitting* across two interfaces (cloning on the most reliable third)

9	Title	📰 Optimized Interface Diversity for Ultra-Reliable Low Latency Communication (URLLC) [100]
	Author/s	Nielsen, Liu, and Popovski
	Year	2017

The article builds upon the system model proposed in earlier work by Nielsen and Popovski [101] and generalizes the proposed traffic splitting schemes for N interfaces. The *weighted splitting* approach is formulated as a optimization problem. A closed-form solution under the assumption of Gaussian interface latency is derived for $N = 2$, however a solution for $N \geq 3$ is only obtained by exhaustive search[4].

4. The author notes that as has been demonstrated during the internship, even a simplistic genetic algorithm-based solver outperforms the exhaustive search for $N = 3$ and, in contrast, is applicable for $N > 3$.

The message splitting is achieved by encoding the source message opening the possibility for employing mechanism discussed in [reference to discussion of network coding in final report here]. Splitting of short messages may be inefficient or even infeasible.

10	Title	📰 Ultra-Reliable Low Latency Communication Using Interface Diversity [102]
	Author/s	Nielsen, Liu, and Popovski
	Year	2018

The article further extends the earlier work by Nielsen, Liu, and Popovski [100]. The authors analyze the bounds on achievable latency gain in a multi-RAT systems with similar and dissimilar interfaces, the impact of payload length and the latency-reliability trade-off. The reliability model is extended to account for reliability-utilization trade-off.

Furthermore, the impact of interface failure dependence is analyzed. In the case of dependent failures, the performance of both the *cloning* and *interface selection* schemes degrades while the weighted splitting scheme is unaffected.

3.1.2 *Redundancy–latency trade-off*

11	Title	Low latency via redundancy [129]
	Author/s	Vulimiri, Godfrey, Mittal, Sherry, Ratnasamy, and Shenker
	Year	2013

Vulimiri et al. raise the fundamental question of optimization of the trade-off between redundancy and latency by providing response time analysis of a queueing system with N independent, identical servers as a function of redundancy (number copies of transmitted data).

The authors show that a certain *threshold load* below which traffic replication always improves mean latency and provide values for simple system configurations under the assumption that the introduction of redundancy does not add no transmitter-side cost and briefly discuss implications of a such overhead. Interestingly, the authors recognize that a receiving-side overhead may depend on the amount of redundancy due to correlated arrivals.

Finally, the authors provide thorough discussion[5] of multiple possible applications ranging from disk-backed databases to Domain Name System (DNS) service and Transmission Control Protocol (TCP) connection establishment procedures.

5. Supplemented by numerous empirical analyses.

3.1.3 *Tangential works in other areas*

Analysis of MapReduce (fork-join) systems

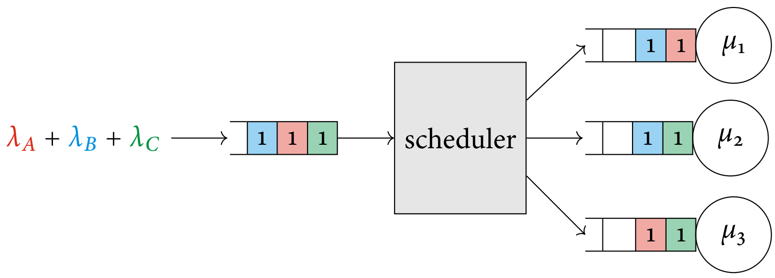


Figure 3.3: Fork-join system model

In the parallel processing context, a *(n, r, k) fork-join system* forks each job into r tasks across r out of n servers such that completion of any k tasks is sufficient to finish the job. Due to inherent analytical complexity and limited applicability of the results, the analysis was limited to $k = 1, n = r$ – i.e. systems where each job is split across *all* available servers and all replicas are removed upon first completion. More recently, bounds on expected response time of fork join systems with general k have been obtained. In the case of the general problem, only a heuristic-based estimation techniques with asymptotically diverging estimation error.

The multi-RAT system may may be seen as a general *(n, r, k) fork-join system* where service times are distributed according to the per-interface latency distribution. The analysis may be simplified by setting $k = 1$ in cases where successful transmission of one replica is sufficient or where there is a reception acknowledgement scheme that can be used to to determine successful reception[6].

6. In this case, the service times are distributed according to the round-trip-time latency distribution. Since any device can not compute one-way latency without an external synchronization (e.g. GPS), such approach may simplify the device design.

12		
	Title	📰 When do redundant requests reduce latency ? [116]
	Author/s	Shah, Lee, and Ramchandran
	Year	2013

13		
	Title	📰 When Do Redundant Requests Reduce Latency? [117]
	Author/s	Shah, Lee, and Ramchandran
	Year	2016

Shah, Lee, and Ramchandran provide a queuing-theoretic analysis of a system composed of n servers processing batches of k jobs. The authors examine the impact of assigning each of the jobs may be to $1 \leq r \leq n$ servers at once on the sojourn time for various configurations of k, service time distribution, system load, etc.

For arbitrary k and "light-everywhere"[7] service times the authors observe that redundant requests help when the arrival rates are low, but start hurting beyond a certain threshold on the arrival rate. The value of the threshold is, however, not provided.

14		
	Title	📰 Efficient Redundancy Techniques for Latency Reduction in Cloud Systems [67]
	Author/s	Joshi, Soljanin, and Wornell
	Year	2017

Joshi, Soljanin, and Wornell analyze (n, r, k) fork-join systems where each job is forked into r tasks across r out of n servers such that completion of any k tasks is sufficient to finish the job. The authors consider general service time distribution.

15		
	Title	📰 Synergy via Redundancy [66]
	Author/s	Joshi
	Year	2018

Joshi explores the general problem for by designing adaptive redundancy schemes, where tasks are replicated if the original task did not finish in given time, and provides two heuristic-based replication policies. The authors focus on the impact on throughput, latency distribution analysis is not provided.

[7] "[U]nder a light-everywhere distribution, waiting for some time brings you closer to completion, resulting in a smaller additional waiting time."

16	Title	📰 On Scheduling Redundant Requests With Cancellation Overheads [83]
	Author/s	Lee, Pedarsani, and Ramchandran
	Year	2017

Lee, Pedarsani, and Ramchandran consider parallel system with redundant requests focusing on the impact of job cancellation overhead. The authors analyze a M/M/2 queuing system under the customary assumptions of Poisson-distributed arrivals and exponential i.i.d. service times.

The authors analyze three job cancellation scenarios: infeasible, slow and immediate. In addition to thresholds on amount of redundancy works 12, 13 and 11, the authors observe that even in the scenario with infeasible cancellation, when the arrival rate is smaller than a certain threshold, one can still reduce job latency by using redundant requests.

The analysis is based on insightful observation that Little's law can be generalized to redundant systems simply by considering only distinct jobs in the system (excluding redundant replicas; proof provided). The authors prove

17	Title	📰 Reducing Latency via Redundant Requests [39]
	Author/s	Gardner, Zbarsky, Doroudi, Harchol-Balter, and Hyytia
	Year	2015

18	Title	📰 Queueing with redundant requests: exact analysis [38]
	Author/s	Gardner, Zbarsky, Doroudi, Harchol-Balter, Hyytiä, and Scheller-Wolf
	Year	2016

Gardner et al. provide an exact analysis of a redundancy system consisting of k servers serving ℓ classes of jobs. A closed form of limiting probability distribution is derived for the most general case where each class of jobs is replicated across all servers which can serve it. Three simplified models are analyzed in greater detail:

$\mathbb{N}$ model with $k-1$ non-redundant classes and 1 redundant class. Non-redundant classes arrive with rate λ_{C_i} and join the queue at $k-1$ servers, one server each. Redundant class arrives with rate λ_R and is replicated across queues at all servers.

$\mathbb{W}$ model with k non-redundant classes and 1 redundant class. Non-redundant classes arrive with rate λ_{C_i} and join the queue at k servers, one server each. Redundant classes arrive with rate λ_R and is replicated across queues at all servers.

$\mathbb{M}$ model with $k-1$ non-redundant classes, $k-1$ dedicated servers and 1 server serving redundant requests. Redundant classes arrive with rate λ_{C_i}

and are replicated across queues at $k-1$ dedicated servers, one server each, and the queue at redundant server.

Unfortunately, the solutions provided assume Exponentially distributed service times and thus could not be applied directly to multi-RAT systems. However, the intuitive interpretation of the solutions may prove useful even with generally distributed service times.

19	Title	Scheduling for efficiency and fairness in systems with redundancy [40]
	Author/s	Gardner, Harchol-Balter, Hyytiä, and Righter
	Year	2017

The authors, building upon the previous work of Gardner et al. [39, 38], provide closed form expressions for per-class response times as well as intuitive interpretations, which might be applied to systems with general service times. Next, the authors focus on design of scheduling policies minimizing the response time while guaranteeing fairness across job classes. The most evolved strategy achieving fairness, Primaries First – under which one of the replicas designated as the primary is given preemptive priority over secondaries, regardless of job class, while the primaries (respectively, secondaries) are served in FCFS order – is shown to perform well with regards to latency regardless of service time distribution.

The works 17, 18 and 19 are aggregated in thesis proposal [37] and the final thesis [36] of Kristy Gardner.

4

QUEUEING MODEL

This chapter provides an alternative derivation[1] of the analytical description of a M/M/k system assuming "nested" structure (see below) with replication and job cancellations obtained by Gardner et al. [40]. Although the ultimate result is equivalent, the author believes that the subsequent sections may still provide more intuitive insight and may be viewed as experimental verification.

In addition, formulae for system overhead due to job replication and a closed-form expression for probability of job of class i being served at server s, enabling analysis of systems with per-interface additive delay, are derived. For comparison, numerical results are given for more general systems. Finally, the rate of convergence of the asymptotic solution is examined.

1. The author has misinterpreted one of the definitions (highlighted in the following section) provided by the authors of the referenced publication and incorrectly assumed that there was a mistake in the publication. Consequently, the author has "corrected" and validated the solution to match his alternative set of definitions.

4.1 DEFINITIONS

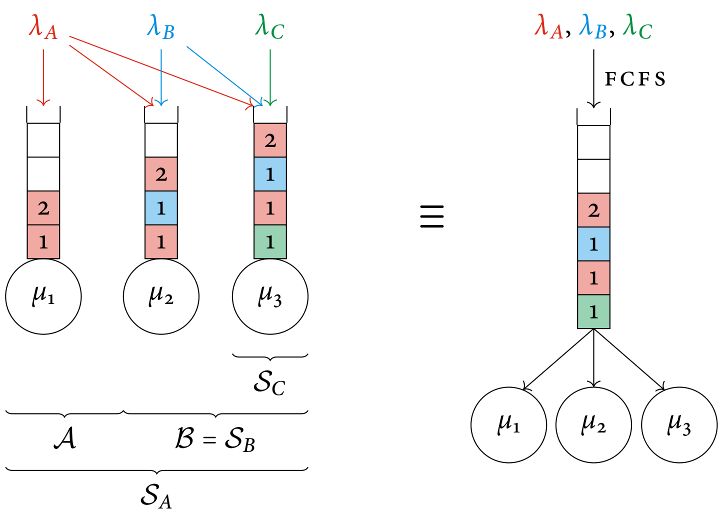


Figure 4.1: Nested queueing system model

The following four paragraphs are primarily drawn from the aforementioned work by Gardner et al. [40] for reader's convenience.

The authors consider a general multi-server system consisting of k servers and ℓ classes of jobs, as shown in the figure above. Arrivals form an Poisson process with average rate λ. Each job belongs to class j independently with probability p_j; the arrival rate of class j is $\lambda_j = \lambda \cdot p_j$. Each job of class j replicates itself upon arrival by joining the queues at a fixed subset of the servers $S_j = \{s \mid \text{server } s \text{ can serve class } j\}$. A job is allowed to be in service at multiple servers simultaneously. Each job departs the system immediately as soon as its first replica completes service, at which time all remaining replicas are immediately cancelled. The k servers are heterogeneous, where each server s provides service rate μ_s. Service times are assumed to be exponential and independent across jobs on the same server and across the same job on different servers.

In a *nested system*, for all job classes i and j, we have either $S_i \subset S_j$, $S_i \supset S_j$, or $S_i \cap S_j = \varnothing$, where $\subset$ and $\supset$ refer to proper subset and proper superset, respectively. If we have multiple identical servers (identical in the sense that they can serve the same classes of jobs, not necessarily in service rates), we can think of these servers as a single fast server. This is because a single job will occupy all the identical servers, releasing them all at the same time for the next job to be assigned. In this sense, relation $S_i = S_j$ is also permitted.

Let class-A be the most redundant job class in the nested system (i.e., class-A jobs replicate to all k servers). Then the remaining $\ell - 1$ job classes can be partitioned into two nested subsystems, denoted $\mathcal{A}$ and $\mathcal{B}$, by the following construction such that for all classes $i \in \mathcal{A}$, $j \in \mathcal{B}$, $S_i \cap S_j = \varnothing$. That is, none of the classes in $\mathcal{A}$ have any servers in common with any of the classes in $\mathcal{B}$.

Let $\mathcal{S}_i$ denote the subsystem in which class-i is fully redundant. That is, the servers in the subsystem $\mathcal{S}_j$ are the servers in S_j. For any subsystem $\mathcal{S}$, we use $\mu_{\mathcal{S}}$ to denote the total service rate of all servers in $\mathcal{S}$, and $\lambda_{\mathcal{S}}$ to denote the total arrival rate of all job classes *in*[2] $\mathcal{S}$.

2. The author incorrectly assumed that $\lambda_{\mathcal{S}_j}$ referred to the total arrival rate of all job classes entering $\mathcal{S}_j$ while Gardner et al. meant the sum of arrival rates of all job classes i such that, $S_i \subseteq S_j$. Unfortunately, the original publication contains a "typo" in the equation defining this quantity which prevented correct resolution of the ambiguity.

For example, in the figure above, $\lambda_{\mathcal{S}_B} = \lambda_A + \lambda_B + \lambda_C$ when assuming the author's definition (i.e. the total arrival rate *into* the subsystem $\mathcal{S}_B$) while $\lambda'_{\mathcal{S}_B} = \lambda_B + \lambda_C$ when assuming the original definition.

The author's interpretation is assumed throughout the remainder of the text.

4.2 RESPONSE TIME ANALYSIS

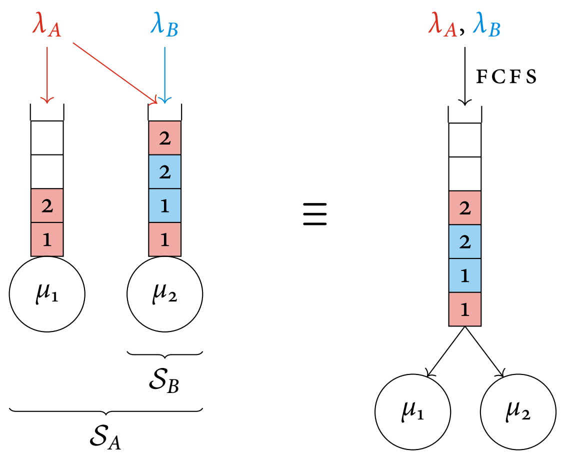


Figure 4.2: Response time analysis example configuration

In the system depicted above, the subsystem $\mathcal{S}_B$ corresponding to the job class B is nested within $\mathcal{S}_A$. For the purposes of illustration, the following paragraphs will follow the "lifetime" of a particular job, however as a consequence of the arrival theorem, on average, the behaviour of the system will be identical.

The duration until the job 2 leaves service can be seen as a sum of two components:

1. Jobs of the enclosing class A in front of it must leave the queue, i.e. the *queueing time* of class A must elapse. Since jobs of both classes are being processed in $\mathcal{S}_A$, the exclusive "capacity" available for processing class-A jobs is obtained by subtracting the "capacity" required by class-B jobs, i.e. $\mu_{\mathcal{S}_A} - \lambda_B = \mu^{\mathrm{R}}_{\mathcal{S}_A}$.

   Note that this step must be repeated for each enclosing subsystem.

2. Subsequently, the system, as seen from the perspective of the class-B jobs, consists structurally only of the subsystem $\mathcal{S}_B$. Since the jobs of classes corresponding to enclosing subsystems (i.e. class A) have "already" left the queue, the total residual arrival rate into $\mathcal{S}_B$ obtained by subtracting the effect of these classes, i.e. $\lambda_{\mathcal{S}_B} - \lambda_A = \lambda^{\mathrm{R}}_{\mathcal{S}_B}$. Therefore, the job leaves service after the *response time* of the adjusted system elapses.

Ultimately, the response time of job class i is given by

$$T^{(i)} = \underbrace{\sum_{j:\mathcal{S}_i \subset \mathcal{S}_j} T_{\mathrm{Q}}^{\mathrm{M/M/1}}(\lambda_j, \mu^{\mathrm{R}}_{\mathcal{S}_j})}_{\text{1. as many times as necessary}} + \underbrace{T^{\mathrm{M/M/1}}(\lambda^{\mathrm{R}}_{\mathcal{S}_i}, \mu_{\mathcal{S}_i})}_{\text{2.}}, \tag{4.1}$$

where $T^{\mathrm{M/M/1}}(\lambda, \mu)$ and $T_{\mathrm{Q}}^{\mathrm{M/M/1}}(\lambda, \mu)$ are expressions for the response time and the queueing time of a M/M/1 system; $\lambda^{\mathrm{R}}_{\mathcal{S}_i} = \lambda_{\mathcal{S}_i} - \sum_{j:\mathcal{S}_i \subset \mathcal{S}_j} \lambda_j$ denotes the residual rate *entering* subsystem $\mathcal{S}_i$ after subtracting the effect of classes corresponding to subsystems enclosing $\mathcal{S}_i$, and $\mu^{\mathrm{R}}_{\mathcal{S}_i} = \mu_{\mathcal{S}_i} - \sum_{j:\mathcal{S}_j \subset \mathcal{S}_i} \lambda_j$ denotes the residual "processing capacity" available for jobs of class j within $\mathcal{S}_i$ after subtracting the the effect of classes corresponding to subsystems nested within $\mathcal{S}_i$.

A numerical example is provided in appendix A.2.1 on page A2 in the appendix. The results obtained through simulation[3] closely match the analytical solution given enough simulation time. The convergence rate and applicability of the general approach to systems with "almost-exponential" service time distribution is discussed next.

4.3 APPLICABILITY TO NON-EXPONENTIAL SYSTEMS

As the service time distribution may not be exactly exponential, the approximation error due to "distributional mismatch" restricts the applicability of the model to general systems. Two generalized approaches are analysed:

1. *exponential fitting* – service rates are approximated by an exponential distribution and the response time is modelled directly by equation (4.1);

[3] See appendix B.2 on page A18.

2. *Pollaczek-Khinchine modelling* – a “response-time structure” identical to the equation (4.1) is assumed, yet the formulae for M/M/1 queues are replaced by Pollaczek-Khinchine relations for M/G/1 queues with parameters as appropriate (see below).

4.3.1 *Error Performance Evaluation*

Error performance of both generalized approximation techniques was evaluated against a reference obtained by simulating the system displayed in fig. 4.3 which is structurally equivalent to the examples given in sections above.

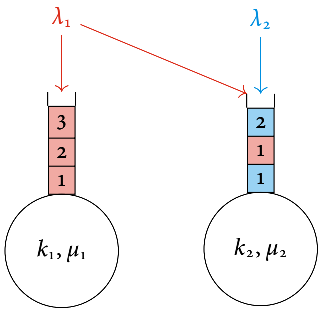


Figure 4.3: System structure used for approximation error evaluation.

The inter-arrival times were distributed exponentially with rates λ_1 and λ_2 while the service times were distributed as a distribution with Probability Density Function (PDF) given as

$$W'(k, \mu) \equiv W\left(k, \frac{1}{\mu\Gamma\left(1 + k^{-1}\right)}\right)$$

where $W(k, s)$ is the Weibull distribution with shape k and scale s.

The configuration was simulated for all combinations of parameters for $i \in \{1, 2\}$:

λ_i 1 to 5 with a step of 0.5;

k_i 0.25 to 1.75 with a step of 0.25;

μ_i 2 to 10 with a step of 1.

The mean of the modified Weibull distribution is equal to the mean of an exponential distribution with rate μ. The modified Weibull distribution interpolates between the exponential distribution with rate μ when $k = 1$ and a Rayleigh distribution when $k = 2$. The modified Weibull thus captures wide range of common latency distributions, as does the similar Erlang distribution.

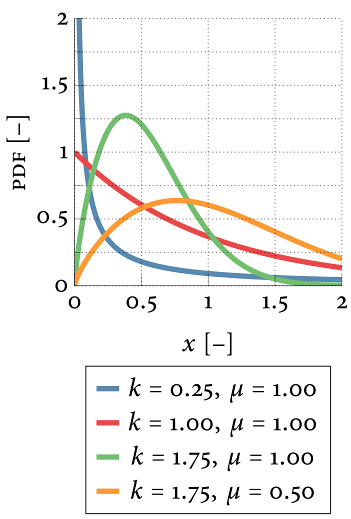


Figure 4.4: PDF of the modified Weibull distribution.

Results presented below express the relative approximation error as a function of the combination of service time distribution shape parameters, k_i.

4.3.2 *Exponential fitting technique*

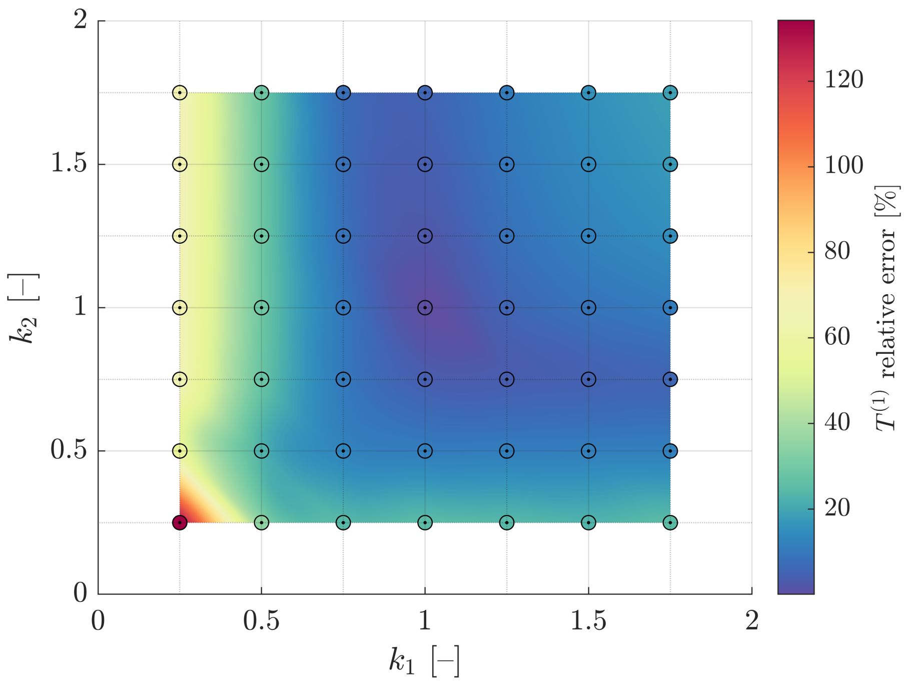


Figure 4.5: Relative absolute error of $T^{(1)}$ aggregated (median) over all λ_i and μ_i.

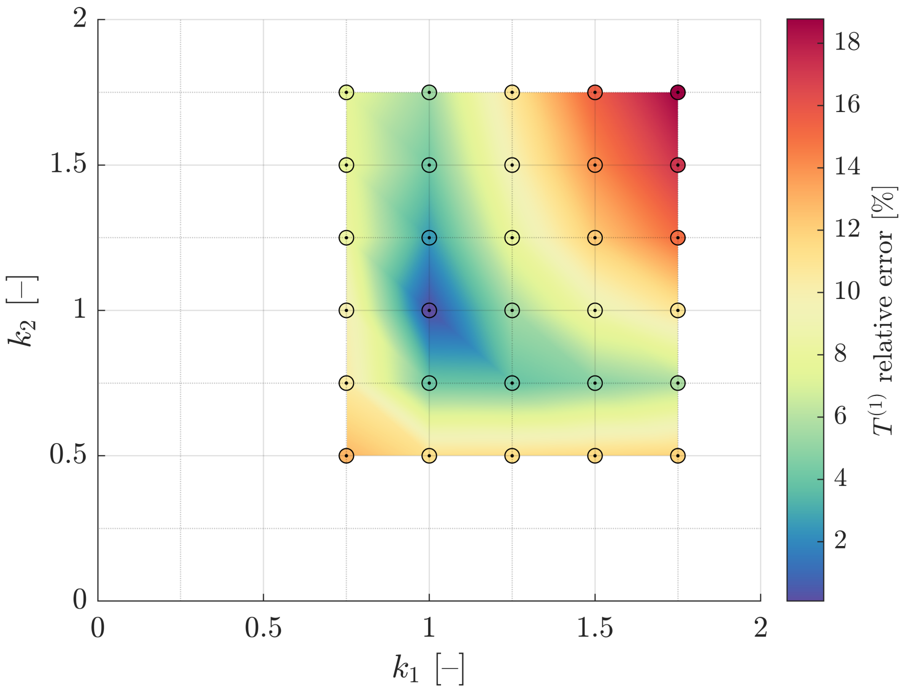


Figure 4.6: Relative absolute error of $T^{(1)}$ aggregated (median) over all λ_i and μ_i. Configurations resulting in relative error in excess of 20% were removed.

The figures 4.5 and 4.6 on the preceding page and on the facing page, respectively, show the result (and a detail view thereof) for the fully redundant class 1, while the figures 4.7 and 4.8 on the current page and on the following page, respectively, present the result for the nested class 2.

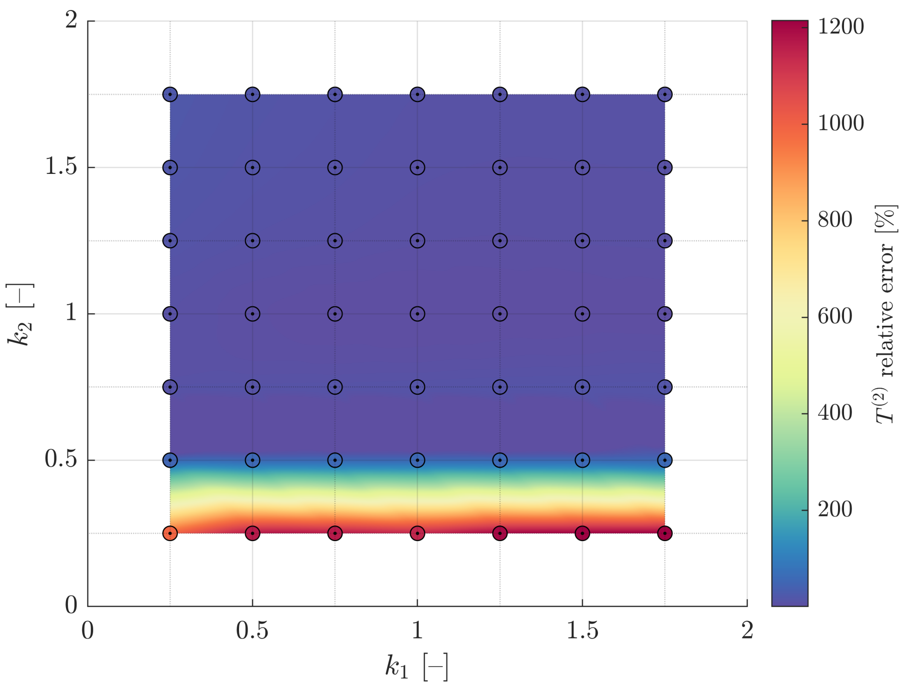


Figure 4.7: Relative absolute error of $T^{(2)}$ aggregated (median) over all λ_i and μ_i.

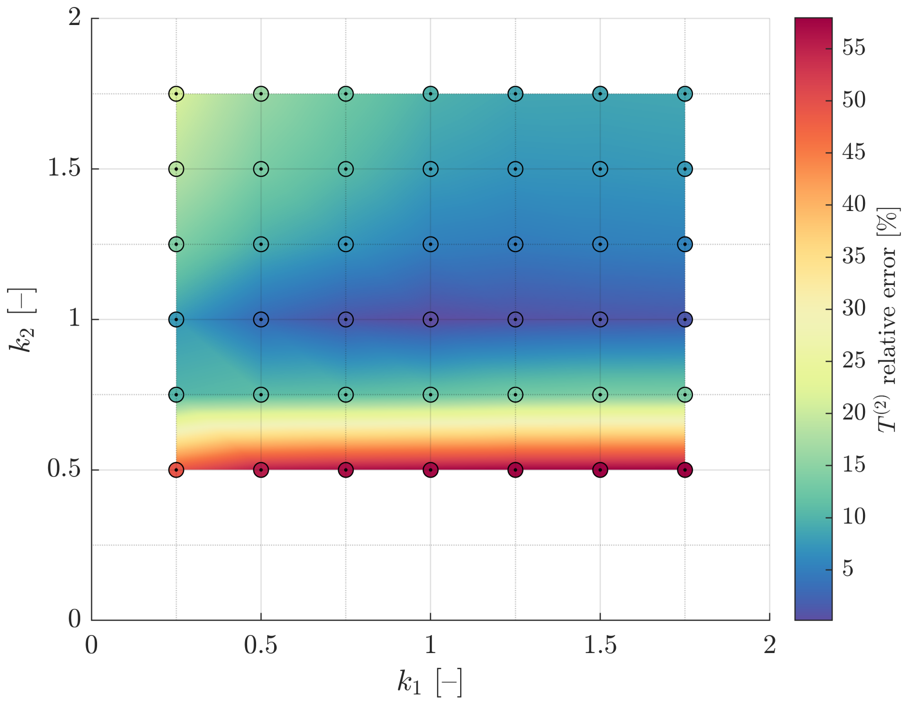


Figure 4.8: Relative absolute error of $T^{(2)}$ aggregated (median) over all λ_i and μ_i. Configurations resulting in relative error in excess of 60% were removed.

The results indicate that the error is exacerbated by nesting. While the error of fully redundant class 1, $T^{(1)}$, increases moderately as shape parameters approach 0.25, the error of nested class 2, $T^{(2)}$, in this area grows dramatically towards values in the order of 100s %. The error of $T^{(2)}$ is influenced primarily by the values of k_2 since server 1 does not process jobs of class 2.

The results exhibit low to moderate approximation error for systems whose service time probability mass is *not* concentrated around a certain value[4].

4. That is, higher values of the parameter k corresponding to "close-to-exponential" or more "Rayleigh-like" distributions. Still, more effort focused on understanding the shapes shown above and improving the estimator is needed.

Approximation Error Distribution

Some insight could be obtained by observing relative approximation errors corresponding to combinations of parameters μ_i and λ_i, aggregated by the combination of shape parameters (k_i) of the service time distributions.

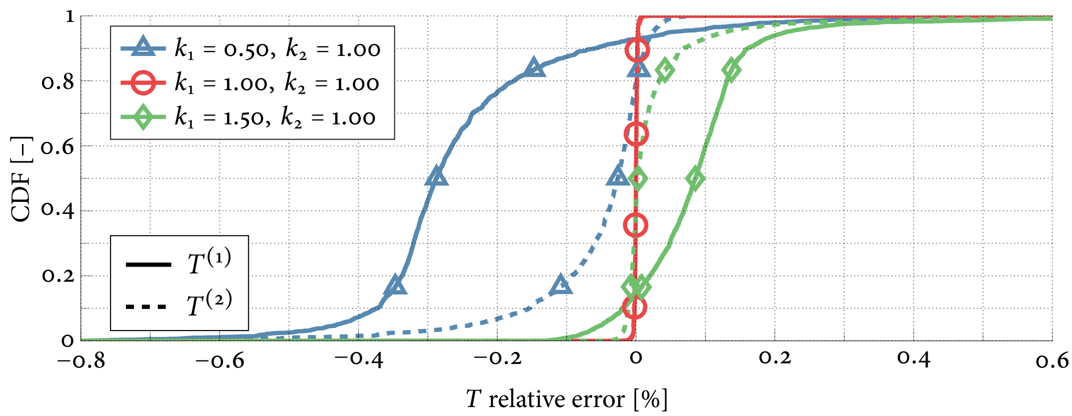


Figure 4.9: CDF of the relative error of $T^{(i)}$.

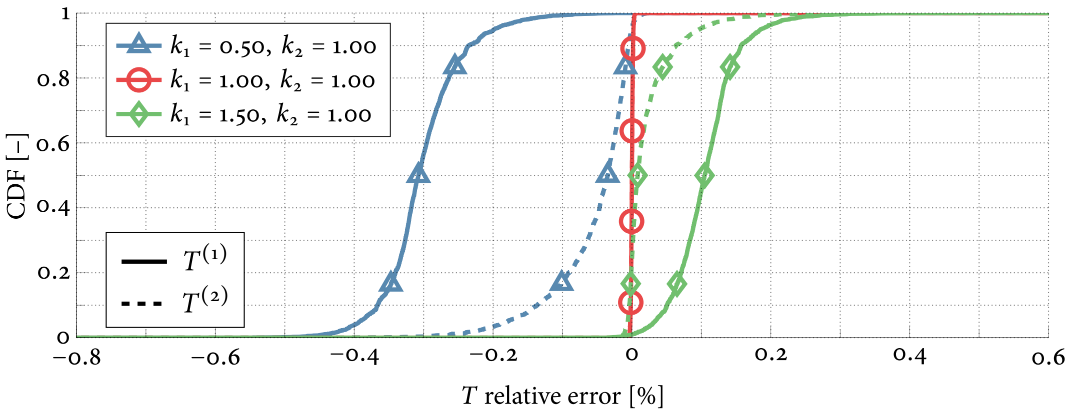


Figure 4.10: CDF of the relative error of $T^{(i)}$ after removing configurations exhibiting high utilization.

In the fig. 4.9 on the preceding page, the service time distribution of server 1 is fixed to exponential ($k_1 = 1$) as k_2 varies. The results suggest that a system with more "Rayleigh-like" service time distribution (green) exhibits greater response times than "predicted" and vice-versa. If this relationship could be generalized, the insight could be exploited to improve the approximation.

The behaviour of a simpler system with a more significant portion of the probability mass of the service time distribution concentrated around zero is generally more deterministic[5] and queueing is less likely.

Figure 4.10 obtained by removing configurations for which the utilization is high ($\frac{\lambda_1}{\mu_1+\mu_2} \geq 0.5$ and $\frac{\lambda_2}{\mu_2} \geq 0.5$). The figure reveals that the data points at the tails of the error distributions correspond primarily to configurations where the utilization is high[6].

5. However, the mutual influence of multiple classes in more complex systems is yet to be understood.

6. In highly utilised systems, the queueing time typically dominates the response time.

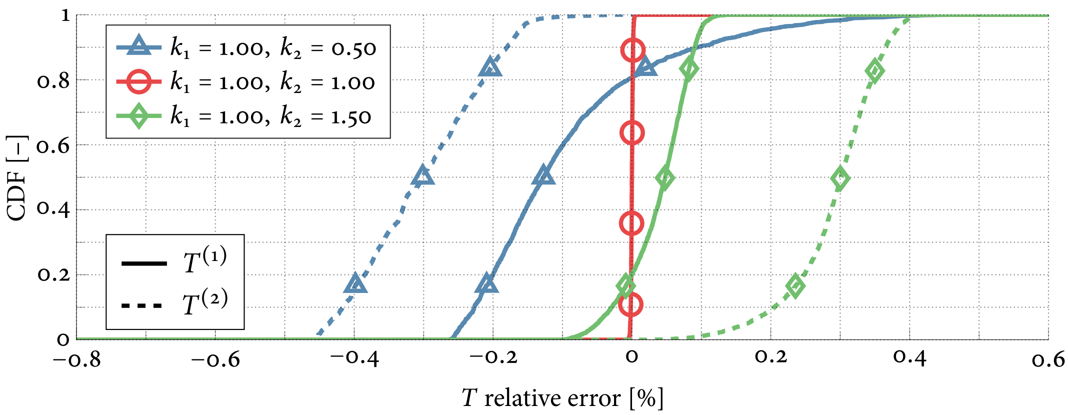


Figure 4.11: CDF of the relative error of $T^{(i)}$ after removing configurations exhibiting high utilization..

In the fig. 4.11, the "opposite" situation is presented. The service time distribution of server 2 remains fixed to exponential ($k_2 = 1$) as k_1 varies. In addition to the effects discussed above (all apply), the dramatic shift of the blue curves towards higher absolute values further confirms that the nested job classes are more severely affected by a distributional "mismatch".

7. Both service time distributions distributed exponentially.

In contrast to the trivial case[7], in this scenario, class-2 jobs are more likely to be blocked by class-1 jobs and as a result, the approximated response times are significantly lower shifting the CDFs to the left.

4.3.3 *Pollaczek-Khinchine modelling technique*

As the general formula for M/G/1 queues takes into account second moments of the service time distributions, the performance of an approximation obtained by formulating equation eq. (4.1) on page 29 in terms of M/G/1 queues is evaluated.

Let $T_\mathrm{P}^{(1)}, T_\mathrm{P}^{(2)}, \dots, T_\mathrm{P}^{(n)}$ denote the service time random variables, $T_\mathrm{P}^{(\mathcal{S}_i)} = \min_{s \in \mathcal{S}_i} T_\mathrm{P}^{(s)}$ the minimum of the service times across servers in $\mathcal{S}_i$, $\mu_{\mathcal{S}_i} = \mathbb{E}\, T_\mathrm{P}^{(\mathcal{S}_i)}$ its mean and $\sigma^2_{\mathcal{S}_i} = \operatorname{Var} T_\mathrm{P}^{(\mathcal{S}_i)}$ variance. Similarly to the earlier definition of $\lambda^\mathrm{R}_{\mathcal{I}_1}$, let $r^\mathrm{R}_{\mathcal{S}_i} = (\mu_{\mathcal{S}_i})^{-1} - \sum_{j:\mathcal{S}_j \subset \mathcal{S}_i} \lambda_j$ denote the residual "processing rate" of jobs of class j within $\mathcal{S}_i$ after subtracting the the effect of classes corresponding to subsystems nested within $\mathcal{S}_i$. Then,

$$T^{(i)} \simeq \sum_{j:\mathcal{S}_i \subset \mathcal{S}_j} T_\mathrm{Q}^{M/G/1}\left[\lambda_j, \left(r^\mathrm{R}_{\mathcal{S}_j}\right)^{-1}, \left(r^\mathrm{R}_{\mathcal{S}_j}\right)^{-2}\right] + T^{M/G/1}(\lambda^\mathrm{R}_{\mathcal{S}_i}, \mu_{\mathcal{S}_i}, \sigma^2_{\mathcal{S}_i}), \qquad (4.2)$$

where $T^{M/G/1}(\lambda, \mu, \sigma^2)$ and $T_\mathrm{Q}^{M/G/1}(\lambda, \mu, \sigma^2)$ are given by the Pollaczek-Khinchine formula for M/G/1 queue.

The second term was obtained merely by expressing the M/M/1 queue in the original formulation by an equivalent expression for an M/G/1 queue. The classical queueing theory, unfortunately, does not suggest a sensible way to generalise beyond the particular case of exponentially distributed service times. Therefore, the approach for obtaining the second term was intentionally simplistic: the mean was obtained as the inverse of the rate while the variance was set so that the coefficient of variation is equal to the rate as in the exponential case. Such construction is indeed suboptimal in the general case; however, at the time of writing, the author lacks the insight necessary for the development of a better approximation.

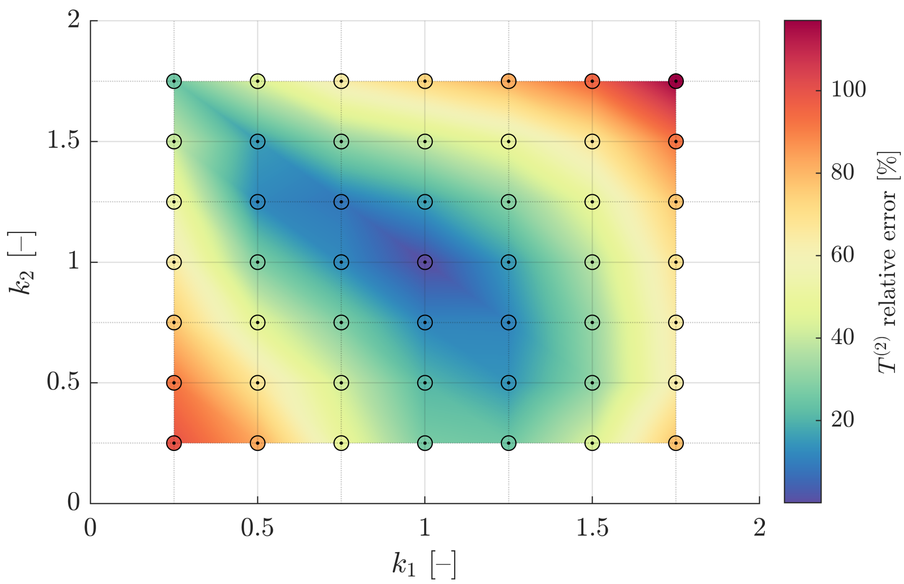


Figure 4.12: Relative absolute error of $T^{(1)}$ averaged over all λ_i and μ_i.

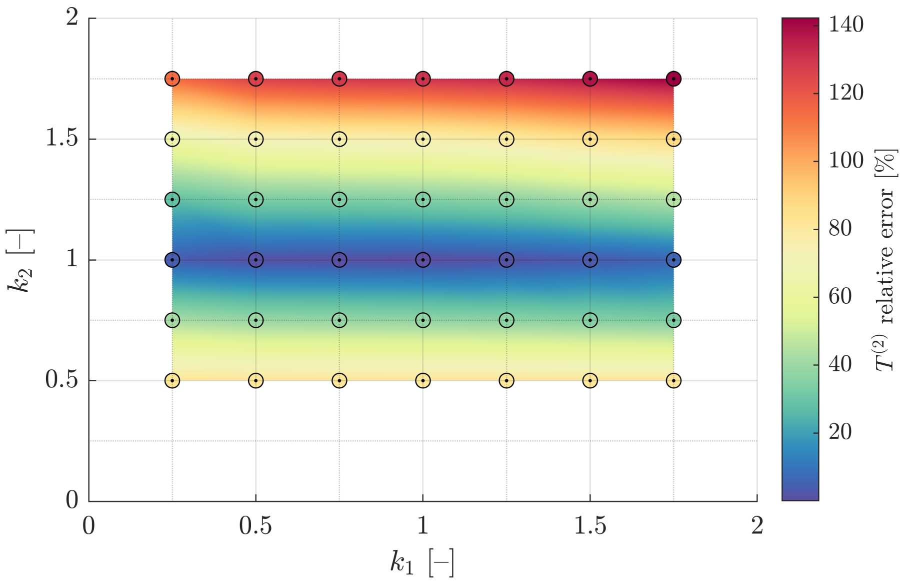


Figure 4.13: Relative absolute error of $T^{(2)}$ averaged over all λ_i and μ_i.

In comparison to the exponential fitting technique, the approximation error is worse in most cases. Since to author's best ability, the shape of the contours shown in fig. 4.12 on the preceding page cannot be easily explained; the author considers this approach as not applicable unless more insight enabling improvement of the approximation becomes available.

4.4 CONVERGENCE ANALYSIS

Since the technique derived in previous sections relies on mean value analysis, the solutions hold only on (long-time) average which, as mentioned in the literature review entry 4 on page 12, may not be sufficient for some applications.

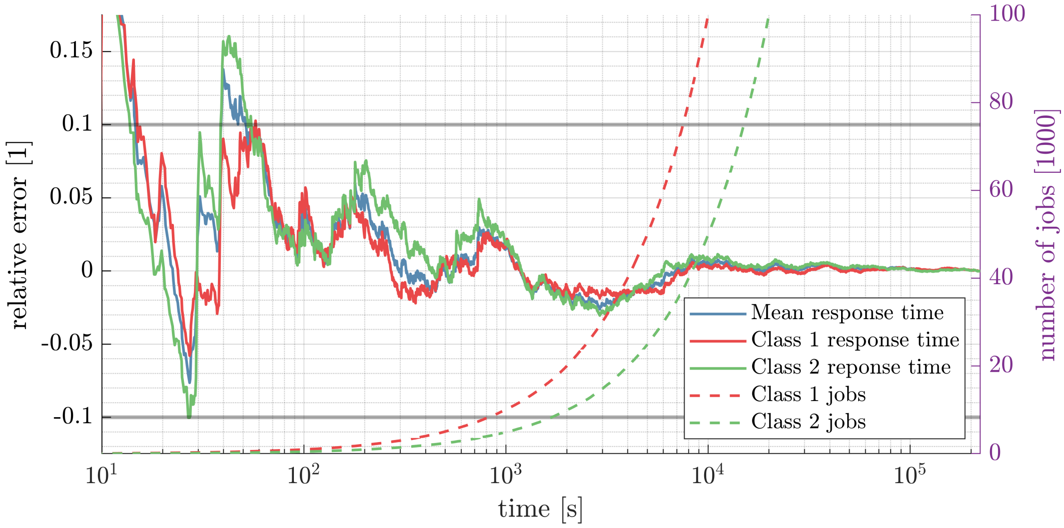


Figure 4.14: Convergence of the mean response time expressed as error relative to the asymptotic value. The left axis refers to the number of finished jobs. The grey lines correspond to the 10% bounds for the stationarity stopping criterion with the threshold of 0.1.

The fig. 4.14 on the previous page obtained by simulating the example shown in fig. A.1 on page A3 (numerical example for the response time analysis)[8] illustrate the typical shape of the relative error (Δ★) of the simulated mean response time, $\widehat{T}^{(i)}$, as a function of the simulation time[9], t, i.e.

8. Structurally identical to the system shown in fig. 4.3 on page 30.

9. For purposes of this section, the rates shown in the fig. A.1 on page A3 are assumed to have units of Hz.

$$\Delta\widehat{T}^{(i)}(t) = \frac{\widehat{T}^{(i)}(t) - \widehat{T}^{(i)}_{\text{asymptotic}}}{\widehat{T}^{(i)}_{\text{asymptotic}}} \tag{4.3}$$

Clearly, to achieve a relative error lower than 10%, a number of jobs in the order of thousands required which suggests that the technique may not be ideal for applications operating in moderately to rapidly time-varying environments.

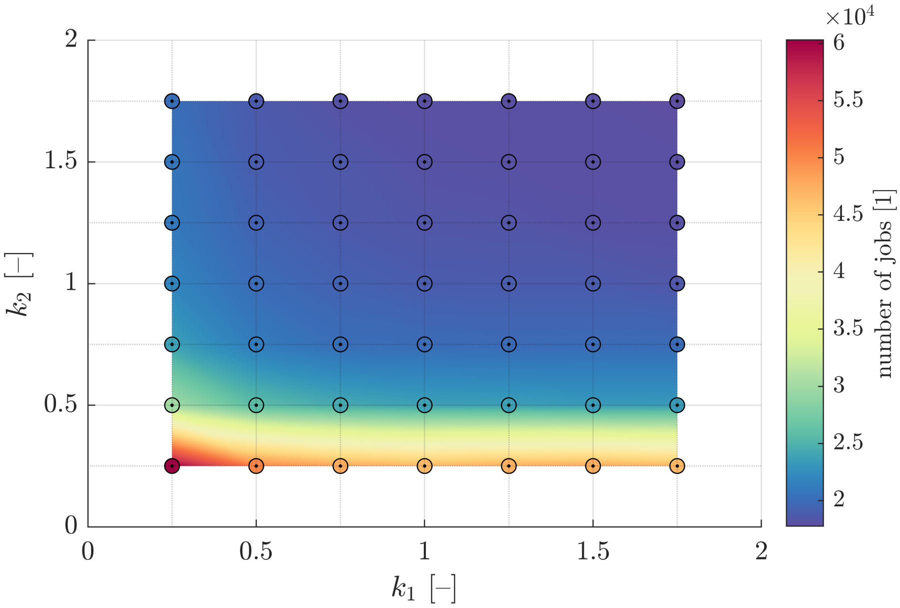


Figure 4.15: Number of class-1 jobs finished when the stationarity stopping criterion with 0.01 threshold was triggered[10]. The plot for class-2 is visually identical, yet the values are lower by approximately 7%.

10. Note: the simulation was stopped after 5 consecutive observations of the $\Delta\widehat{T}^{(i)}(t)$ with observation period of 10^4 simulation steps were lower than the threshold. In the future version of the simulator, the option to trigger directly on the number of jobs will be added and the figure will be modified to show the number of jobs finished at the time of the first of the observations.

The fig. 4.15 obtained by simulating the system shown in fig. 4.3 on page 30 with identical parameters as previously shows that systems exhibiting more "Rayleigh-like" service time distributions require significantly less time to converge as opposed to their counterparts with low shape parameters corresponding to PDFs concentrated tightly around zero with a long tails, although deeper analysis is needed.

4.5 PROBABILITY OF PROCESSOR SELECTION

Following the decomposition of the response time described above, on average, a job of class i spends possibly multiple queueing times corresponding to enclosing subsystems in addition to its own before entering service at any of the servers. The probability of leaving service at server $s \in S_i$ therefore may only depend on processing times of classes j such that $\mathcal{S}_j \subseteq \mathcal{S}_i$.

Upon reaching head of the queue, the jobs replicate across all servers. The server with smallest service time serves the job. Therefore, the server selection probability is

$$P_{i \to s} = \Pr\{\text{job of class } i \text{ served at server } s \in S_i\}$$
$$= \Pr\left\{ s = \underset{s \in S_i}{\operatorname{argmin}}\, T_{\mathrm{P}}^{(s,i)} \right\}, \tag{4.4}$$

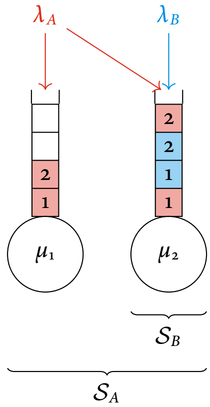


Figure 4.16: Example of the structure of a nested queueing system model.

where $T_{\mathrm{P}}^{(s,i)}$ denotes the random variable corresponding to processing time of class i at the server s.

In the trivial case of a "innermost" subsystem $\mathcal{S}_i$ (i.e. such that for any j, $\mathcal{S}_j \notin \mathcal{S}_i$), the processing time random variable is equivalent to a random variable with the service time distribution.

Similarly as seen during derivation of the response time, in general case job class i experiences only residual "processing capacity" due to subsystems nested within $\mathcal{S}_i$. The residual "capacity" of server s available for serving class-i jobs is

$$\mu^{\mathrm{R}}_{s,i} = \begin{cases} \mu_s - \sum\limits_{\substack{j: s \in \mathcal{S}_j, \\ \mathcal{S}_j \subset \mathcal{S}_i}} \lambda_j \cdot P_{j \to s}, & s \in S_i \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

The server selection probability term ensures that, on average, only a fraction of the required capacity by class j proportional to the likelihood of class-j job ($\mathcal{S}_j \subset \mathcal{S}_i$) being served by the server s is subtracted.

For independent exponentially distributed processing times with mean $\mu^{\mathrm{R}}_{s,i}$, the probability of s's service time being the minimum across S_i, is[11]

11. See appendix A.1.1 on page A1 in the appendix for derivation.

$$P_{i \to s} = \frac{\mu^{\mathrm{R}}_{s,i}}{\sum\limits_{t: t \in S_i} \mu^{\mathrm{R}}_{t,i}} \tag{4.6}$$

Formulae eq. (4.5) and eq. (4.6) form a recursive expression of finite depth (iff the redundancy structure is finite). The innermost subsystems have to be computed first as their effects impact the server selection probabilities of classes corresponding to enclosing subsystems.

Numerical example illustrating the calculation and the verification of the results using a numerical simulation is provided in appendix A.2.2 on page A3 in the appendix.

4.6 PREEMPTION & EFFICIENCY

Throughout the text, the author has analysed systems where jobs may be preempted (interrupted) during service. The analysis of the non-preemptive systems where jobs may be removed neither from the queues nor the servers effectively reduces to a mere response time analysis of parallel isolated queueing systems[12]. The third option is to perform hybrid preemption where jobs may be interrupted only before entering service[13] as discussed next.

12. As, trivially, those systems are bound to perform poorly and thus are not practical, the author will not discuss them further.

13. I.e. when a replica has left service before another replica enters.

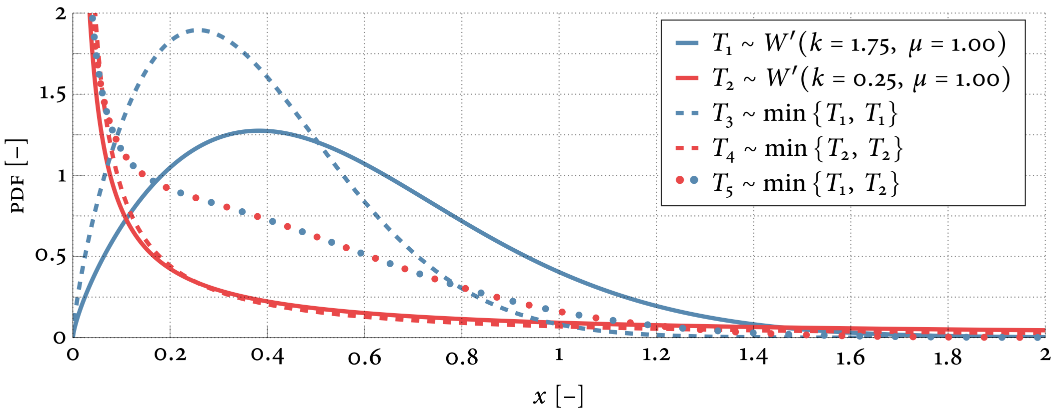


Figure 4.17: PDFs of random variables X, Y and Z distributed as Exponential ($\mu = 10$), Normal ($\mu = 4, \sigma = 10$) and Rayleigh ($B = 8$), respectively, and their minimums.

14. Not referring to the convolution operator.

Figure 4.17 illustrates that the probability of short service occurring at one server while the other experiences longer service is higher when at least one of the distributions is heavy tailed.

Since the min operator concentrates the probability mass around zero, the jobs in the queues of a preemptive system perceive the convolved[14] service time distributions (non-solid lines in fig. 4.17) whereas the jobs in the hybrid preemptive systems see only the individual distributions (solid lines). The preemptive system blocks for the duration of a single sample drawn from the convolved distribution while the hybrid system, each of the servers serving the job blocks for the duration drawn independently from the respective distributions which implies a loss in efficiency if serving the job once is sufficient.

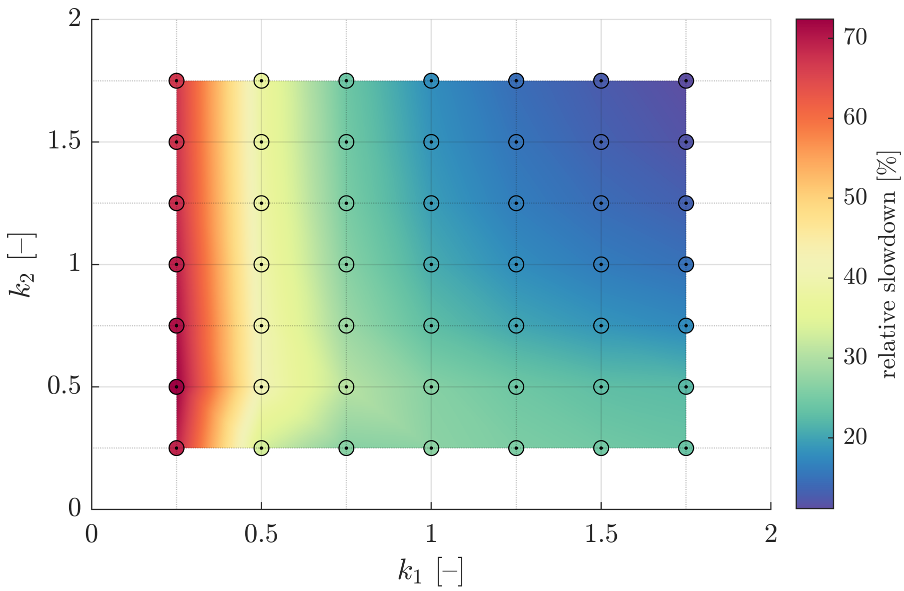


Figure 4.18: Relative slowdown of class-1 jobs due to lack of preemption.

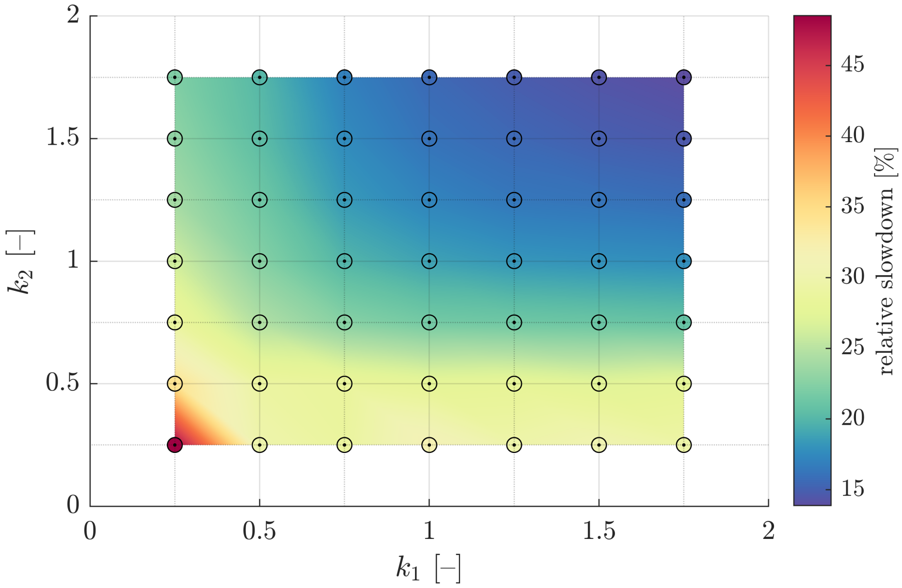


Figure 4.19: Relative slowdown of class-2 jobs due to lack of preemption.

The figures 4.18 and 4.19 obtained by simulating the system shown in fig. 4.3 on page 30 with identical parameters as previously shows the relative (median) slowdown $\varsigma^{(i)}$ expressed as

$$\widehat{\varsigma}^{(i)} = \mathrm{Med}\left[\frac{\widehat{T}^{(i)}_{\text{non-preemptive}} - \widehat{T}^{(i)}_{\text{preemptive}}}{\widehat{T}^{(i)}_{\text{non-preemptive}}}\right]. \tag{4.7}$$

As illustrated by the fig. 4.18 on the facing page, the loss of efficiency while processing class-1 jobs is largest when the shape parameter of the first server (serving exclusively class-1 jobs) is low which corresponds to the conclusions concerning tail distributions reached above.

Figure 4.19 shows that the class-2 jobs are impacted less severely as they are processed only on server 2 and thus do not benefit from the multi-server diversity in either case. In addition to the most notable impact of the parameter k_2, the effect of class-1 jobs blocking class-2 jobs becomes more apparent as the tail of the service time distribution of the first server grows heavier.

A complementary approach of analysing the impact of *cancellation overhead* (delay) taken by Lee, Pedarsani, and Ramchandran is summarized in the literature review entry 16 on page 25.

4.6.1 Structural inefficiencies

$$\eta^{(i)} = \frac{\text{time spent processing a class-}i\text{ job } j}{\sum\limits_{s \in S_i} \text{time spent processing the job } j \text{ by the the server } s} \tag{4.8}$$

In addition to possible inefficiency due to limited support for preemption by the target system, the redundancy structure itself is a fundamental factor reducing the system efficiency. In the worst case scenario all replicas of a class-i job enter service at all servers in S_i simultaneously. By the time t the fastest replica leaves service, in addition, all remaining servers have already wasted $(n-1) \cdot t$ time units leading to the worst case efficiency

$$\eta^{(i)}_{\text{preemptive}} \le \frac{t}{n \cdot t} = \frac{1}{n}. \tag{4.9}$$

In hybrid preemptive servers, the servers which ultimately do not serve the job waste the whole service time samples drawn from their individual service time distribution; moreover the non-preemptive systems' queueing times are additionally adversely impacted by redundant jobs already served elsewhere and thus

$$\eta^{(i)}_{\text{non-preemptive}} \le \eta^{(i)}_{\text{hybrid}} \le \eta^{(i)}_{\text{preemptive}}. \tag{4.10}$$

General formulae for η in the preemptive and hybridly preemptive cases are hard to derive as one would have to determine the positions (in queue or receiving service at a server) of all replicas of each job leaving service.

4.7 REDUNDANCY STRUCTURE (OPTIMIZATION)

The redundancy structures can be seen as a *matching* in an undirected bipartite graph where each vertex is either one of n input streams or one of m queues. For any such graph, there are $2^{m \cdot n}$ such matching. For practical purposes, only the structures where all job classes are assigned and none of the servers is idle. By inclusion-exclusion principle (applied twice), the number of such matchings is

$$N = \sum_{k=0}^{n} \binom{n}{k} (-1)^k (2^{n-k} - 1)^m. \tag{4.11}$$

Furthermore, structures which are not *nested structures* may not be analysed using the technique set forth above, and thus are not to be considered.

A redundancy structure may be represented by an adjacency matrix of the corresponding graph representation, however since the adjacency matrix of any such bipartite graph if of the form

$$A = \begin{bmatrix} 0_{n,n} & B \\ B^T & 0_{m,m} \end{bmatrix},$$

definition of the *allocatiom matrix B* is sufficient.

A simplistic algorithm to generate all valid allocation matrices is to begin with a full graph matrix and recursively attempt to remove edges and verify that the resulting structure is satisfies all properties discussed above. The algorithm is equivalent to a depth-first (or breath-first) search of a tree where each child vertex is a valid redundancy structure formed by removing an edge from the parent redundancy structure so that no two vertices are identical.

The optimization algorithm would compute the full response times and reliabilities using eq. (2.1) on page 15 for all valid structures and, if available, selects a structure satisfying the defined per-job class requirements or other optimization goal (e.g. minimum of the response times averaged across all job classes). For systems with small number of interfaces and job classes, which is the likely scenario for the IoT devices this work primarily focuses on, the number of combinations permits exhaustive search. For more complex applications an approximation technique (e.g. a genetic algorithm) may be more suitable.

PART III

JUST-IN-TIME PERFORMANCE OPTIMIZATION

5

STATE OF THE ART

The JUST IN TIME approach is an umbrella term arising in many areas of human activity including manufacturing, supply-chain and general management as vividly discussed in the work of one of the pioneers of the paradigm often credited for introduction in the method to the western world, Monden [94].

The Toyota Production System was developed and promoted by Toyota Motor Corporation [author: in the 1950s] and is being adopted by many Japanese companies in the aftermath of the 1973 oil shock. The main purpose of the system is to eliminate through improvement activities various kinds of waste lying concealed within a company.

TPS enables one-piece production that allows products to be made fluidly, one unit at a time, just as water flows through a river.

—— Yasuhiro Monden [94, pp. 3 and xxxiv]

The general applicability of the additional optimization objectives introduced into the classical theory of scheduling has led to its adoption in general systems-theoretic literature [80]. Amongst these objectives, the minimization of the overall tardiness and reduction of "waste" can easily be seen as crucial for general computer [12] and communication systems [87, 91]. In this context the interference (i.e. waste) reduction is a key enabler of the massive deployment envisioned as the foundation of the future Internet. As discussed in the prologue, the requirements for low latency follow directly from the intended applications including mission-critical systems such as factory automation and autonomous driving.

The problem of QoS-aware scheduling in telecommunications [6, 31, 7] and communications systems especially in the context of hardware design (e.g. [90]) is a broad research field of its own. Similarly, the problem of channel assignment often analysed within the contexts of Wireless Mesh Networks (WMNs) [2] and to lesser extent in HETNETS [141] has received much attention in past two decades. With the advent of multiple radio interfaces on a single device, the availability of affordable performant hardware suitable for massive deployment and the demand for ultra-reliable low-latency networking, the obvious generalization lies in scheduling across multiple interfaces to achieve enhance system reliability and reduce latency. Unfortunately, the literature is scarce and solutions addressing constrained networks are lacking.

5.1 LITERATURE REVIEW

5.1.1 *Local scheduling*

20	Title	📰 Opportunistic Scheduling: Generalizations to Include Multiple Constraints, Multiple Interfaces, and Short Term Fairness [77]
	Author/s	Kulkarni and Rosenberg
	Year	2005

As with many other works on scheduling with multiple constraints done in the early 2000s, the work of Kulkarni and Rosenberg focuses on achieving objective and directly quantifiable QoS measures (e.g. fairness, energy consumption, data rate, etc.) in a time-slotted wireless system. Inspired by earlier information-theoretic works on multi-user diversity in time-varying channel conditions [128], the authors propose to generalise the classical "water-filling" solution to power allocation problems and further argue that an optimal solution for a multi-interface system is also an argmax policy with a water-filling method.

21	Title	📰 Scheduling Packets over Multiple Interfaces While Respecting User Preferences [139]
	Author/s	Yap, Huang, Yiakoumis, Chinchali, McKeown, and Katti
	Year	2013

Yap et al. develop a fair scheduler that respects the users' preferences (e.g. assigning more capacity to using only a subset of interfaces for a particular application) by extending the Deficit Round Robin (DRR)[1] technique. The authors develop a *pareto efficient* scheme requiring almost no coordination among the interfaces. As the work is primarily focused on optimising throughput of an ideal system, the stochastic phenomena such as latency distribution, and reliability are not discussed. Furthermore, the algorithm operates on the premise of having a continuous input streams defined by their rates rather than making decisions on datagram by datagram basis.

1. DRR is a $\mathcal{O}(1)$ approximation of a Fair Queueing scheduler by Demers, Keshav, and Shenker [23] (based on the RFC 970 by Nagle [96]) which requires $\mathcal{O}(n)$ work, where n is the number of datagrams in the queue.

22	Title	📰 Multi-interface Communication: Interface Selection under Statistical Performance Constraints [68]
	Author/s	Kar, Rizk, and Fidler
	Year	2018

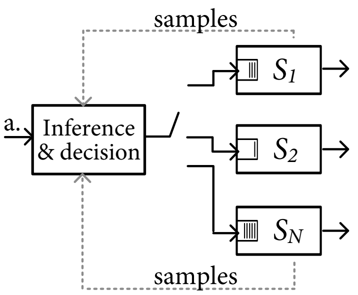


Figure 5.1: Transition based system. SOURCE: [68].

Kar, Rizk, and Fidler consider a multi-RAT *transition based system* where each interface with its associated input queue form a subsystem S_i as shown in fig. 5.1 defined by its service curve [21] which are either known or inferred from observation. The system decides at the beginning of every epoch which subsystem to assign the arrivals (a.) to – i.e. each datagram is processed by a single interface.

The authors employ stochastic network calculus[2] to develop a scheduling strategy minimising the waiting time. Further, the authors develop an envelope for general discrete-time Markov modulated service processes and from the numerical evaluation in the special case of Gilbert-Elliot channel observe that a "adaptive strategy takes the statistical characteristics of the arrival and service processes into account" outperforms both Round-Robin and Join the Shortest Queue policies.

2. See their earlier work [32].

23	Title	Multiple Interface Scheduling System for HeterogeneousWireless Vehicular Networks: Description and Evaluation [111]
	Author/s	Roman, Ball, and Ou
	Year	2017

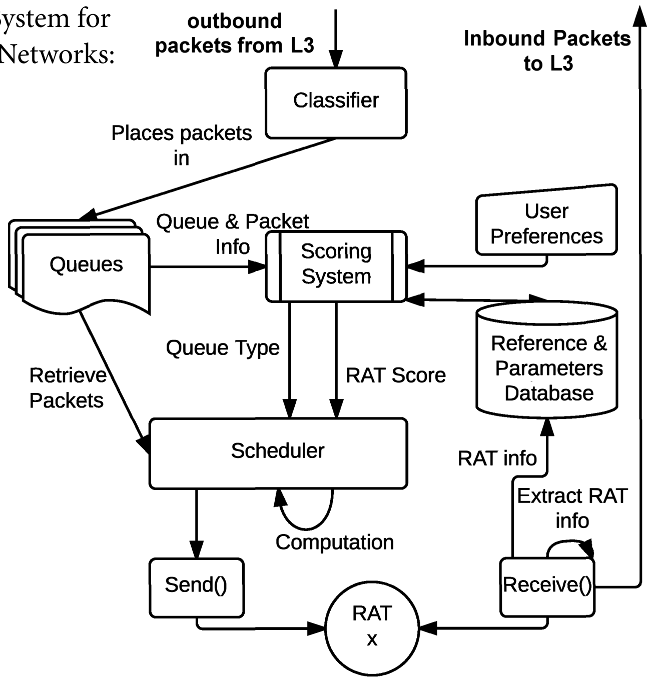


Figure 5.2: SOURCE: [111]

The work of Roman, Ball, and Ou focuses on QoS-aware scheduling in multiple-interface vehicular (IEEE 802.11-based) systems. Following the IEEE 802.11e-based QoS classification separating the traffic into queues by QoS-class priority, the proposed scheduler employs an intricate scoring system (see fig. 5.2) that "grades" available RATs based on datagram metadata, queue state, user preferences (e.g. energy consumption), etc. Finally, the datagram is sent out using the RATs scoring positively. Although the authors give special attention to delivery of safety-critical traffic the possibility of employing multiple interfaces in parallel is not explored[3].

3. Cf. Figure 5 in [111] – as its first step, the scoring system eliminates each RAT with higher average delay than the requirement.

5.1.2 *Network scheduling (in wireless mesh networks)*

The evolution towards contemporary WMN began in the early 1980s with the US military funded research programme DARPA SURAN[4] [10] into adaptive self-configuring, multi-hop networks. Throughout the 1990s the development was primarily limited by the availability of affordable hardware. By the turn of the century the significant increase in availability of powerful yet affordable equipment accompanied with the maturation of core projects within the IEEE 802 family[5] heralded a dramatic surge in research work worldwide.

4. Defense Advanced Research Projects Agency SURvivable Adaptive Networks

5. Most importantly, IEEE 802.11 and the IEEE 802.14.

24	Title	Channel Assignment Techniques for Multi-Radio Wireless Mesh Networks: A Survey [2]
	Author/s	Alim Al Islam, Islam, Nurain, and Raghunathan
	Year	2016

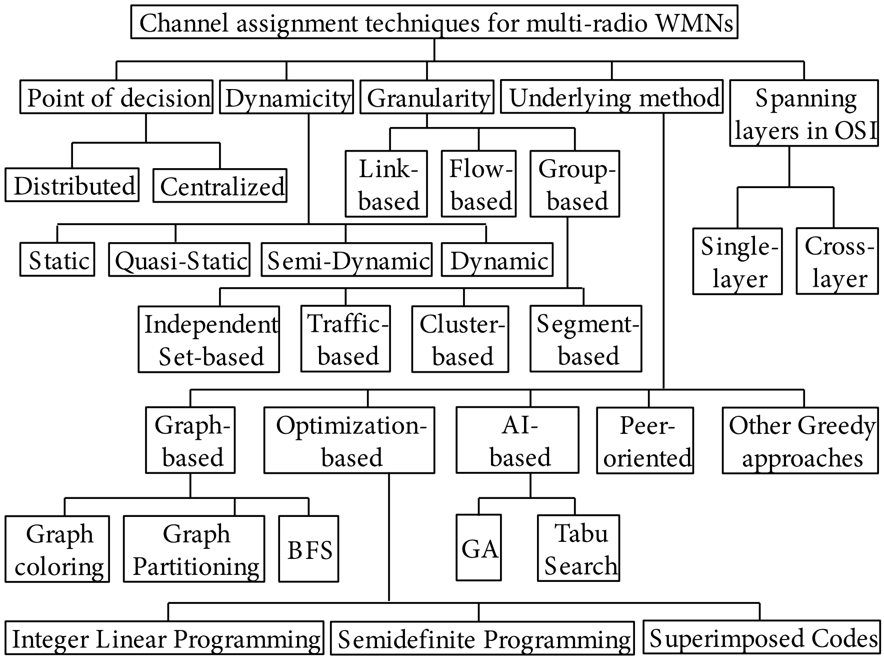


Figure 5.3: Classification of channel assignment techniques. SOURCE: [2].

After decades of research into WMNs, even the range of works focused on addressing the joint problem of channel[6] *and* interface selection in multi-RAT WMNs is beyond brief summary.

Fortunately, Alim Al Islam et al. have provided an excellent survey of over 150 articles solely focused on channel assignment techniques in multi-radio WMNs. Instead of commenting on individual approaches the author would like to highlight that the vast majority of the works neglects the QoS (e.g. latency and reliability) aspects and remain an open research topic.

6. E.g.; there are up to 14 (13 in Europe) overlapping IEEE 802.11 channels in the 2.4 GHz band. Depending on the region, it is possible to specify up to 4 non-overlapping channels, i.e. whose spectral overlap is so small that it does not affect the performance of networks which use them. [92]

6

JUST-IN-TIME SCHEDULING

In contrast to the queueing-theoretic approaches discussed earlier, the algorithmic scheduling solutions attempt to gather as much information about the system under control during runtime and to make scheduling decisions reactively rather than predictively.

"The fact that any operation, no matter how complicated, can be resolved into a series of simple operations, is the key to the solution of many problems. (...) The natural method, then, of studying a complex operation is to study its component elementary operations."
— H. L. Gantt [35, p. 41]

JIT scheduling can be illustrated by a Gantt-like chart where each interface is associated with a time axis and scheduled packet are shown as bars placed on the corresponding axes according to the instantaneous allocation.

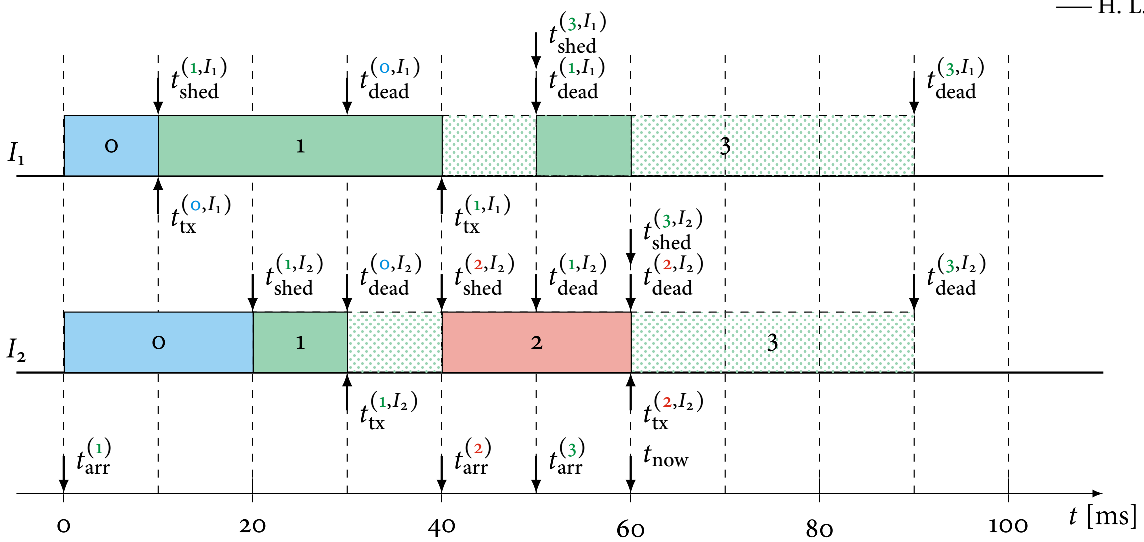


Figure 6.1: JIT scheduling chart. The colours denote different traffic classes.

Figure 6.1 shows a sample view of the JIT scheduling chart of an device with two interfaces (I_1, I_2) at time t_{now}. The solid bars correspond to the time intervals between the scheduled time, t_{sched} and, ultimately, the transmission time, t_{tx}. The dotted bars span the remainder of the allocated time slots up to the corresponding deadline time, t_{dead}. To obtain these using the per-interface service time distributions, the scheduler needs to calculate the the sum of the arrival time, t_{arr}, and the per-class latency requirement of the traffic class, t_{hard}[1].

As suggested by the fig. 6.1, an allocation of a packet may begin at different time and differ in duration for each interface. Furthermore, the scheduling process may decide to use only a subset of interfaces to serve a packet.

1. The packet must be delivered before this *hard deadline*. As the network time distribution by be shifted by a constant representing fixed minimum delay, the scheduler may exploit this to constraint the deadline time.

Depending on the RAT performance and the packet arrival processes the system may be either *under-loaded* or *over-loaded*. In the former case, most of the arriving packets may be scheduled soon enough so that information ageing due to buffering in the input queue is not significant. An over-loaded system, on the other hand, can be viewed as unstable in the queueing-theoretic sense. That is, unable to serve all inbound traffic in time prompting the need for filtering / discrimination to reduce the *effective arrival rate*. The crucial question is which packets should be removed to ensure a given QoS.

6.1 SCHEDULING CONSIDERATIONS

6.1.1 *Scheduling in the under-loaded region*

In the under-loaded region, a simple scheduling algorithm may not require knowledge of the underlying system performance metrics (e.g. the delay distributions)[2] to deliver stable performance as, by definition, the processing capacity exceeds the inbound flow rate. However, to control the performance, to adapt to varying traffic patterns, etc., such information may still be valuable.

2. The implementation `NoFilter` discussed in the section 6.3 on page 54 is an example of a such algorithm.

6.1.2 *Scheduling in the over-loaded region*

In order to operate in the under-loaded region a single hyperparameter – delay margin D – may be sufficient. The answer to the question raised above clearly depends on the overall objective. The author proposes a definition of a reliability target as the probability of successful delivery of each allocated packet, i.e. $R = \Pr\,(\text{delivered} \mid \text{allocated})$ with the concrete value imposed by the application/user.

Especially in the context of massive deployments of power-constrained devices – as is often the case in IoT applications – ensuring that the use of radio modules is not "wasteful" is paramount. Minimizing the amount of transmissions unlikely to provide any value due to high *age of the contained information*[3] may additionally reduce the overall network interference and thus contribute to its scalability.

3. The notion of the , as introduced by Kaul et al. [69], quantifies the freshness of knowledge a system has about a process observed remotely. For a vivid overview of the concept see the work by Kosta, Pappas, and Angelakis [74].

6.1.3 *Single interface considerations*

The hatched area in fig. 6.2 represents earlier allocations blocking the interfaces.

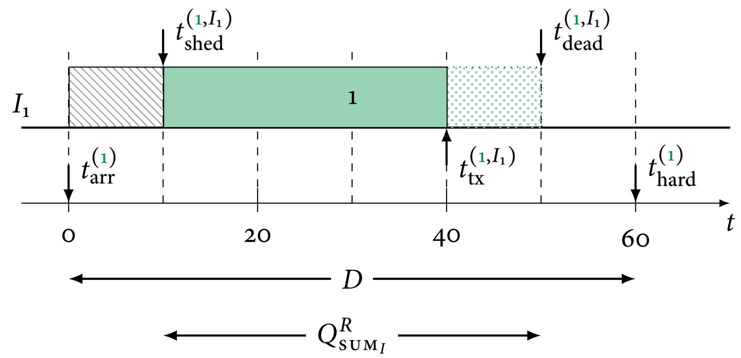


Figure 6.2: JIT scheduling with a single interface.

As the delay of a packet served by an interface I can be considered as draws from the (possibly dependent) service time distribution SVC_I and the network time distribution NET_I, the delay requirement is effectively specified in terms of the distribution of the sum of random variables associated with the respective distributions, $\textsc{sum}_I \equiv \textsc{svc}_I + \textsc{net}_I$. A packet is scheduleable if the delay seen as the quantile function Q of the sum distribution evaluated at a probability R is less (or equal) than the delay requirement D. That is, $Q^R_{\textsc{sum}_I} = \inf\{d \in \mathbb{R} \mid R \le \Pr(\textsc{sum}_I \le d)\} \le D$.

Quantile sum decomposition

The scheduler needs to determine the latest time of allocation of the packet so that the probability (in the classical sense) of successful delivery is R. In the general case where the dependency structure of the service and network time distributions is unknown, the quantile functions of the addend-distributions, $Q^u_{\textsc{svc}}$ and $Q^v_{\textsc{net}}$, are evaluated at appropriate probability values u, v, such that the condition may be rewritten as

$$Q^R_{\textsc{sum}_I} \le Q^u_{\textsc{svc}_I} + Q^v_{\textsc{net}_I} \le D \tag{6.1}$$

The problem of existence of the upper bound on the sum-distribution (6.1) is related to the problem formulated and conjectured by A. N. Kolmogorov (as cited and convolutely proved by Makarov [89]). The solutions to the problem are limited to the sum of $n = 2$ variables[4] – further decomposition of the packet delay within the model is therefore impractical.

4. Although recently, some interesting results with respect to a dual problem have been obtained by Puccetti and Rüschendorf [107], "computing the best-possible bounds for arbitrary n is an NP-hard (computationally intractable) problem" [76].

In parallel to the work of Makarov, from the theory of copulas can be used to derive (upper and lower) bounds on algebraic operations over random variables as done by Schweizer and Sklar [112, ch. 7]. Specifically, following the outlined steps for binary operator +, it follows that for $w \in [0,1]$, $u \in [w,1]$, and $w = u + v - 1$ we have

$$Q^w_{\textsc{sum}_I} \le Q^u_{\textsc{svc}_I} + Q^{1-u+w}_{\textsc{net}_I}. \tag{6.2}$$

Since the (6.2) holds for any w and u, and the w is fixed to R, the bound may be tightened by testing all values of u

$$Q^R_{\textsc{sum}_I} \le \inf_{u\in[R,1]} \left(Q^u_{\textsc{svc}_I} + Q^{1-u+R}_{\textsc{net}_I}\right). \tag{6.3}$$

The complete derivation of equations (6.2) and (6.3) is provided in the appendix A.1.2 on page A2.

Known dependency structure

In the special case where the dependency structure is given by a copula density $c_{\textsc{svc}_I,\textsc{net}_I}$ and the distributions are jointly continuous, the quantile function $Q^R_{\textsc{sum}_I}$ is the quasi-inverse (see [98, def. 2.3.6]) of

$$\begin{aligned} F_{\textsc{sum}_I} &= \Pr\{\textsc{svc}_I + \textsc{net}_I \le x\} \\ &= \int_{-\infty}^{\infty}\int_{v=-\infty}^{v=x-u} f_{\textsc{svc}_I,\textsc{net}_I}(u,v)\,\mathrm{d}v\,\mathrm{d}u \\ &= \int_{-\infty}^{\infty}\int_{v=-\infty}^{v=x-u} c_{\textsc{svc}_I,\textsc{net}_I}\left(F_{\textsc{svc}_I}(u), F_{\textsc{net}_I}(v)\right) \\ &\qquad f_{\textsc{svc}_I}(u) f_{\textsc{net}_I}(v)\,\mathrm{d}v\,\mathrm{d}u \end{aligned} \tag{6.4}$$

Furthermore, under the assumption of independence, (6.4) reduces to

$$\begin{aligned} F'_{\text{SUM}_I} &= \int_{-\infty}^{\infty} \int_{v=-\infty}^{v=x-u} f_{\text{SVC}_I}(u) f_{\text{NET}_I}(v) \, \mathrm{d}v \, \mathrm{d}u \\ &= \int_{-\infty}^{\infty} F_{\text{NET}_I}(x-u) f_{\text{SVC}_I}(u) \, \mathrm{d}u. \end{aligned} \tag{6.5}$$

6.1.4 *Multiple interfaces*

Note that in fig. 6.3, the interface I_2 failed to transmit the packet 1 before the deadline and thus it was discarded locally. Conversely, the I_1 has successfully transmitted another replica of packet 1. Note further, that neither of the interfaces could have met the requirement independently as both individual quantile functions are higher than the actual scheduling margin.

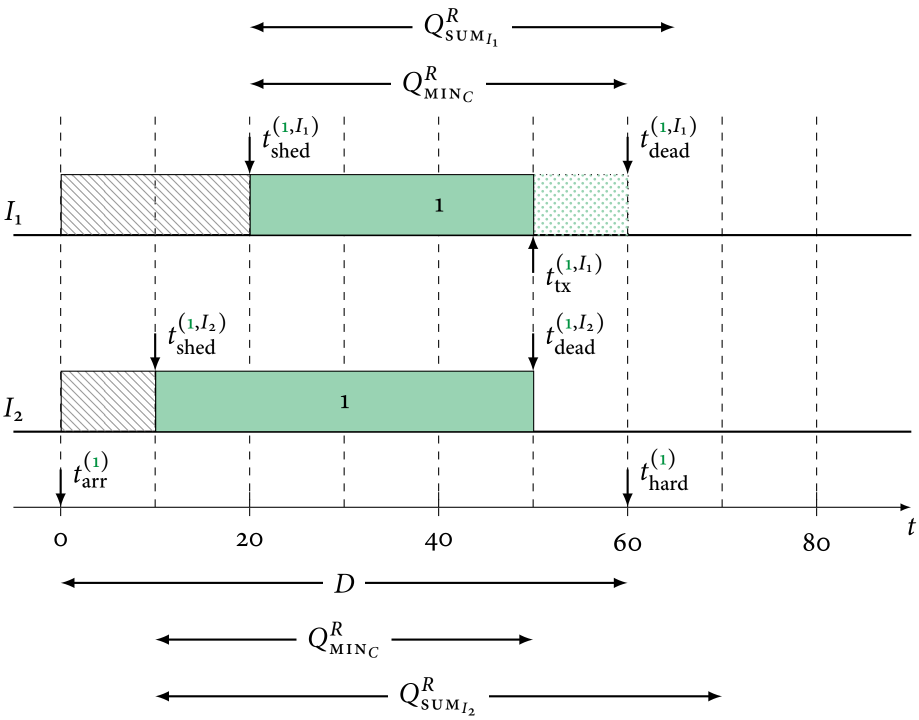


Figure 6.3: JIT scheduling with a multiple interfaces.

In the multi-RAT case the delay constraint effectively expresses the quantile function of the "minimum of per-interface sums of service and network time distributions" across all interfaces $\mathcal{I}_C = \{I_i\}_{i \in C}$ belonging to a combination C, i.e. $\text{MIN}_C \equiv \min \{\text{SUM}_I\}_{I \in \mathcal{I}_C}$.

Quantile min decomposition

Trivially, the quantile function of the minimum distribution is less or equal to the individual quantile functions, i.e.

$$Q^R_{\text{MIN}_C} \leq \min_{I \in \mathcal{I}_C} \left\{ Q^R_{\text{SUM}_I} \right\}. \tag{6.6}$$

Thus, since the overall hard deadline is smaller or equal to any of the hard deadlines calculated independently for any of the relevant interfaces it suffices to require that each interface executes the single interface calculation with reliability target R independently of others following the methods discussed for the single-interface case.

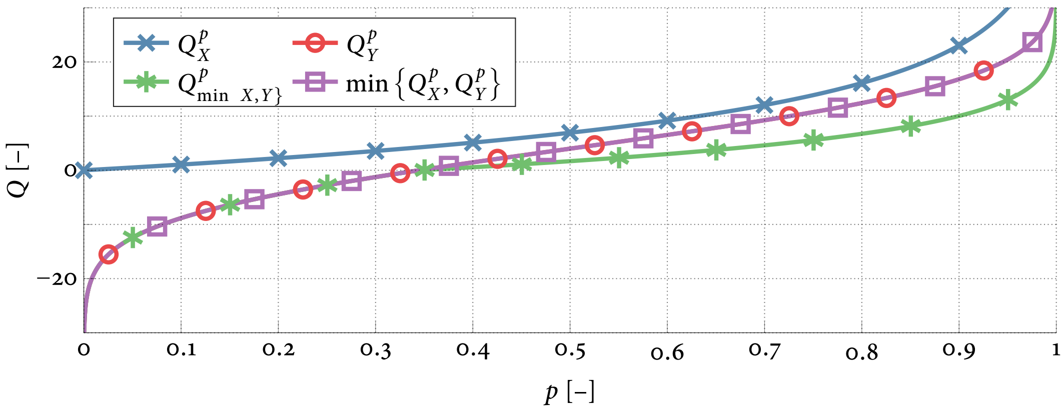


Figure 6.4: Quantile functions of Exponential and Normal distributions and their pairwise minimums. X and Y denote Exponential ($\mu = 10$) and Normal ($\mu = 4, \sigma = 10$), respectively.

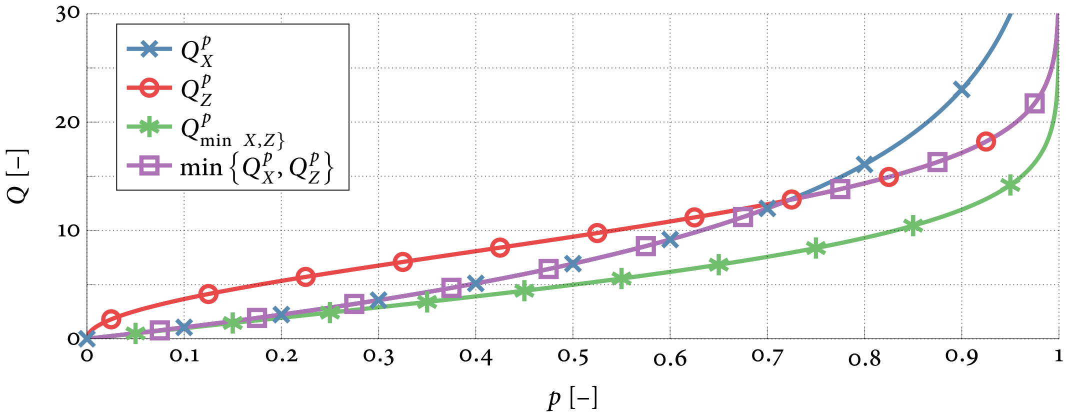


Figure 6.5: Quantile functions of Exponential and Normal distributions and their pairwise minimums. X and Z denote Exponential ($\mu = 10$) and Rayleigh ($B = 8$), respectively.

As illustrated in fig. 6.4 and fig. 6.5, for p where $Q^p > 0$, the upper bound becomes looser which implies that in this case, the overall target $Q^p_{\text{MIN}_C}$ (green) may be achieved even when the per-interface targets $Q^{p_I}_{\text{SUM}_I}$ are less strict (may be evaluated at values p_I lower than p). Further analysis of this feature may unravel some optimisation opportunities.

Known dependency structure

In the special case where the per-RAT delay distributions (i.e. the sum distributions discussed above) and their dependency structure are known, the QoS target $Q^R_{\text{MIN}_C}$ may be obtained as the quasi-inverse of the CDF of the minimum distribution, F_{MIN_C}. The corresponding event $\text{MIN}_C \le x$ can be expressed as the union of $|C|$ events

$$\text{MIN}_C \le x \equiv \bigcup_{I \in \mathcal{I}_C} (\text{SUM}_I \le x) \tag{6.7}$$

and expanded by applying the general form of *inclusion-exclusion p.* [134]

$$
\begin{aligned}
F_{\textsc{min}_C}(x) = &\sum_{I\in\mathcal{I}_C} \Pr\left(\textsc{sum}_I \le x\right) - \sum_{\substack{i<j\\(i,j)\in C^2}} \Pr\left(\max\left\{\textsc{sum}_{I_i}, \textsc{sum}_{I_j}\right\} \le x\right) \\
&+ \sum_{\substack{i<j<k\\(i,j,k)\in C^3}} \Pr\left(\max\left\{\textsc{sum}_{I_i}, \textsc{sum}_{I_j}, \textsc{sum}_{I_k}\right\} \le x\right) \\
&+ \cdots + (-1)^{|C|+1} \Pr\left(\max_{I\in\mathcal{I}_C}\left\{\textsc{sum}_I\right\} \le x\right).
\end{aligned} \tag{6.8}
$$

All terms of (6.8) can be obtained from the joint CDF, $F_{(\textsc{sum}_I |\, I\in\mathcal{I}_C)}(x_1,\ldots,x_{|C|})$. To simplify notation let

$$
F_{\textsc{sum}_J} = F_{(\textsc{sum}_{I_i} |\, i\in J)} = \lim_{\substack{x_j\to\infty\\ j\in C\wedge j\notin J}} F_{(\textsc{sum}_{I_i} |\, i\in C)}, \tag{6.9}
$$

e.g. $F_{\textsc{sum}_{\{2,3\}}}(x,y) = F_{\textsc{sum}_C}(\infty,x,y,\infty,\infty,\ldots) = F_{(\textsc{sum}_{I_2},\textsc{sum}_{I_3})}(x,y)$, where the symbol ∞ denotes the limit to infinity.

$$
\begin{aligned}
F_{\textsc{min}_C}(x) = &\sum_{i\in C} F_{\textsc{sum}_{\{i\}}}(x) - \sum_{\substack{i<j\\(i,j)\in C^2}} F_{\textsc{sum}_{\{i,j\}}}(x,x) \\
&+ \sum_{\substack{i<j<k\\(i,j,k)\in C^3}} F_{\textsc{sum}_{\{i,j,k\}}}(x,x,x) + \cdots \\
&+ (-1)^{|C|+1} F_{\textsc{sum}_C}(x,\ldots,x)
\end{aligned} \tag{6.10}
$$

Furthermore, under the assumption of independence, (6.10) reduces to

$$
\begin{aligned}
F'_{\textsc{min}_C}(x) &= \sum_{i\in C} F_{\textsc{sum}_{\{i\}}}(x) - \sum_{\substack{i<j\\(i,j)\in C^2}} \prod_{\ell\in\{i,j\}} F_{\textsc{sum}_{I_\ell}}(x) \\
&\quad + \sum_{\substack{i<j<k\\(i,j,k)\in C^3}} \prod_{\ell\in\{i,j,k\}} F_{\textsc{sum}_{I_\ell}}(x) + \cdots \\
&\quad + (-1)^{|C|+1} \prod_{\ell\in C} F_{\textsc{sum}_{I_\ell}}(x) \qquad (6.11) \\
&= 1 - \prod_{\ell\in C}\left(1 - F_{\textsc{sum}_{I_\ell}}(x)\right) \qquad (6.12)
\end{aligned}
$$

which can be interpreted as computing the complementary probability of experiencing longer delay than x at each of the interfaces in $\mathcal{I}_C$.

Consequently, the scheduler shall be restricted to select only those combinations which would allow for sufficient time margin for service by scheduling the packet at each of the interfaces in $\mathcal{I}_C$ no later than $t_{\text{arrival}} + D - Q^R_{\textsc{min}_C}$.

Based on the knowledge of the scheduling time constraints, the scheduler may decide to schedule packets in different order, schedule packets later to make space for other packets or to perform other similar operations to make the final schedule more efficient. Analysis of these more advanced features remains an open question to be addressed in future work.

6.1.5 *Interface release times*

When a processing time slot on an interface is allocated, the *allocation* is inserted in the interface queue[5]. As the queueing delay in these queues has to be accounted for, the scheduler must monitor the state of the queues – especially the *release time* t_r^I corresponding to the deadline of the last allocation to interface I.

5. In fig. 2.2 on page 13, the interface queues connect the Scheduler and the respective interfaces.

When evaluating the scheduling criteria, the scheduler needs to assume that no unallocated packet may enter service at I prior to t_r^I, i.e. $t_{\text{sched}}^I \geq t_r^I$. The release time may decrease when a packet is transmitted prior to its allocated deadline and so the scheduler shall not decide to discard packets due to instantaneous interface unavailability.

6.2 SCHEDULING PROCESS

6.2.1 *Scheduler*

A pivotal feature of JIT scheduling is the deferral of the decision until the last possible time whereby ensuring the amount of uncertainty about the outcome is minimized since more information may obtained by observing the inherently stochastic system under control. In accordance with this principle, the scheduling process should either allocate or discard a packet only if further delay would adversely affect the system performance.

The process is triggered by a new arrival or a notification by an interface. The algorithm extracts head of the scheduler queue (cf. fig. 2.2) and verifies that there are idle interfaces ready to process packet. If not, the decision is deferred, i.e. the packet is (re)inserted into the queue and the process ends.

To illustrate the process, a simplified implementation of the concepts developed earlier assuming FCFS input queue *ordering policy* proceeds as shown in fig. 6.6. The process is triggered by a new arrival or by a notification by an interface. The algorithm extracts head of the scheduler queue (c.f. fig. 2.2 on page 13) and verifies that there are idle interfaces ready to process packet. If not, the decision is deferred, i.e. the packet is (re)inserted into the queue and the process ends.

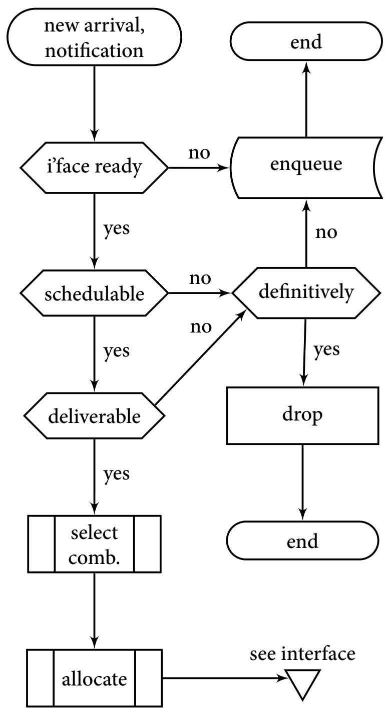


Figure 6.6: Scheduler sub-process.

Next, the scheduler verifies that given the current state of the interfaces (i.e. pending allocations), there is enough time to dispatch the packet before its deadline. Furthermore, if available, the scheduler employs knowledge of the delay distributions to find combinations of interfaces able to satisfy the QoS requirements. Figure 6.6 includes a separate branch element to decide whether, if no combinations are found, the packet may eventually become scheduleable [deliverable] if a busy interface becomes available: definitively unscheduleable [undeliverable] packets are discarded while the possibly scheduleable [deliverable] ones are enqueued.

The optimal combination is selected from the subset of combinations $\mathcal{C}_s$ which satisfies the timing constraints according to the *selection policy* and, finally, the *allocation* consisting of the packet and respective timing constraints are handed to each of the interfaces in the combination.

6.2.2 *Interface*

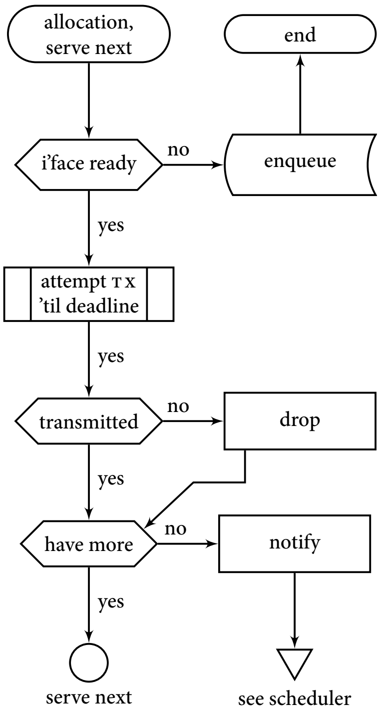


Figure 6.7: Interface sub-process.

The interface part of the scheduling process (fig. 6.7) is executed by each interface independently. In a realistic setting, the scheduling logic is to be executed by a higher layer while the RAT modules are required only to perform packet cancellation (drop) after a defined timeout.

The process is triggered whenever an allocation is assigned to the interface or if there are more allocations in the interface queue after serving an earlier one. If ready, the interface attempts to transmit the packet until the deadline time associated with the allocation. Otherwise, the packet is pushed into the interface queue, and the process terminates.

If the interface fails to transmit the packet "in time", it is dropped. If there are more allocations enqueued in the interface queue, the process is triggered again. If not, the scheduler is notified that the interface became ready for further allocations.

6.3 SIMULATION RESULTS

6.3.1 *Simulator*

To verify and demonstrate the scheduling principles discussed above, the author has developed an MATLAB®-based event-oriented scheduling system simulation tool, `schesi`.

The simulator consists of two core components:

`prob module` implements a basic algebra of random variables (e.g. the min and + operators used above) using technique closely resembling expression templates[6]

`schesi module` provides a general event-based model of the system described above. The internal structure resembles the queuing simulator, `quesi`, discussed in appendix B on page A5.

6. Expression template is a programming idiom most notably used in C++-based scientific computing software that relies on operator overloading in object-oriented code to provide mathematical semantics to common operators whilst enabling efficient optimization. Expression templates were invented independently by Veldhuizen [127] and David Vandevoorde [126, ch. 26] in 1994.

Despite every effort has been made to develop a highly performant code, the overhead of the execution engine, lack of direct memory management and relatively insufficient optimizations performed by the JIT compiler, were found to be limiting factors. To take advantage of the AAU's High-Performance Computing (HPC) infrastructure the simulator is designed to allow massively parallel multi-server execution.

6.3.2 *Configuration*

To explore the behaviour of the algorithm, an example configuration of the system shown in fig. 2.2 with i.i.d. interfaces with service and network times distributed as Exp($\mu = 5$) and Lognormal($\mu = 1, \sigma^2 = 1$), respectively, is simulated. The packets form a single input stream with delay target $D = 50$ with the inter-arrival time distributed as Exp($\mu = \mu_{\text{arr}}$).

The scheduler processes the input FCFS queue and allocates appropriate packets to a combination of interfaces $\hat{C}$ from the set of combinations that satisfy the requirements discussed above, $\mathcal{C}_s$. The selected combination maximizes the metric m_C corresponding to mean time margin computed across all interfaces in combination C, i.e. redundancy is penalized:

$$m_C = t_{\text{arrival}} + D - t_{\text{now}} - \frac{1}{|\mathcal{I}_C|} \sum_{I \in \mathcal{I}_C} t_{\text{r}}^I \tag{6.13}$$

$$\hat{C} = \underset{C \in \mathcal{C}_s}{\operatorname{argmax}} \left(m_C \right) \tag{6.14}$$

Three implementations of the decisions process are compared:

`Filter` implements the complete scheduling process described in section 6.2 on page 53.

`NoFilter` does not have any knowledge of the underlying service and network time distributions. Therefore, the deliverability check in the scheduling process is not performed.

`Hybrid` acts as the `NoFilter` scheduler when the length of the input queue is lower or equal to the number of interfaces and as the `Filter`[7] otherwise.

7. That is, the whole queue is filtered if the decision process is triggered when the length exceeds the threshold.

For the purpose of the simulation, each packet is accounted as a single entity regardless of how many copies are created throughout the process. Moreover, only the "most successful" state counts – e.g. if a packet is allocated to three interfaces and is delivered by the first and the second while the third fails even to transmit it, the result is recorded as a single packet being successfully delivered at the earliest of the delivery times. The results of the simulation are discussed next.

6.3.3 *Results*

Low reliability target

This sub-section concerns with the behaviour of the system when the reliability target is set to 95%, that is *below* the value achievable without filtering in the under-loaded region.

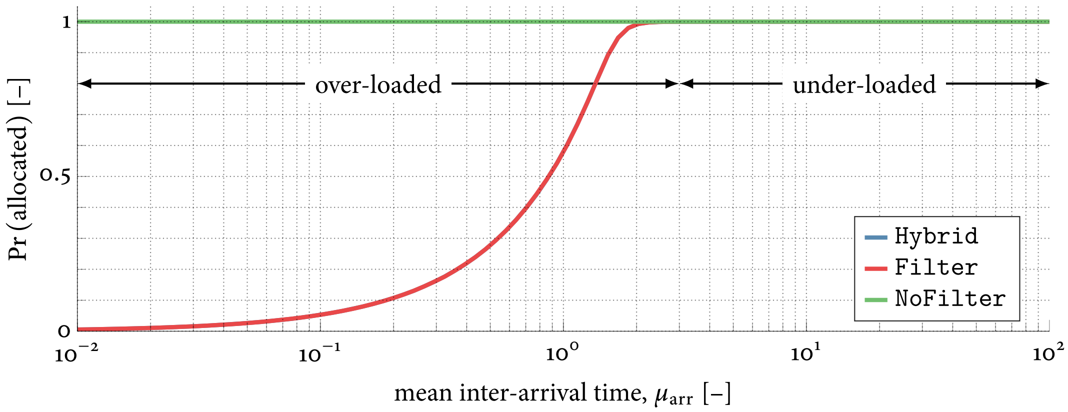


Figure 6.8: Probability of packet allocation (i.e. not discarded by scheduler) and the regions of operation. `Filter` and `Hybrid` overlap.

As the packet inter-arrival time[8] decreases, the implementations performing filtering gradually allocate fewer packets to maintain system stability. Conversely, the naive `NoFilter` implementation allocates all packets since it does not make assumptions about the interface performance and thus attempts to transmit packets even with minuscule time margins.

8. The arrival rate is proportional to the inverse of the mean of the inter-arrival time distribution, μ_{arr}.

All implementations are equivalent in the under-loaded region – i.e. $\mu_{\text{arr}} \gtrapprox 2$ – since the queueing effect (delay) is negligible.

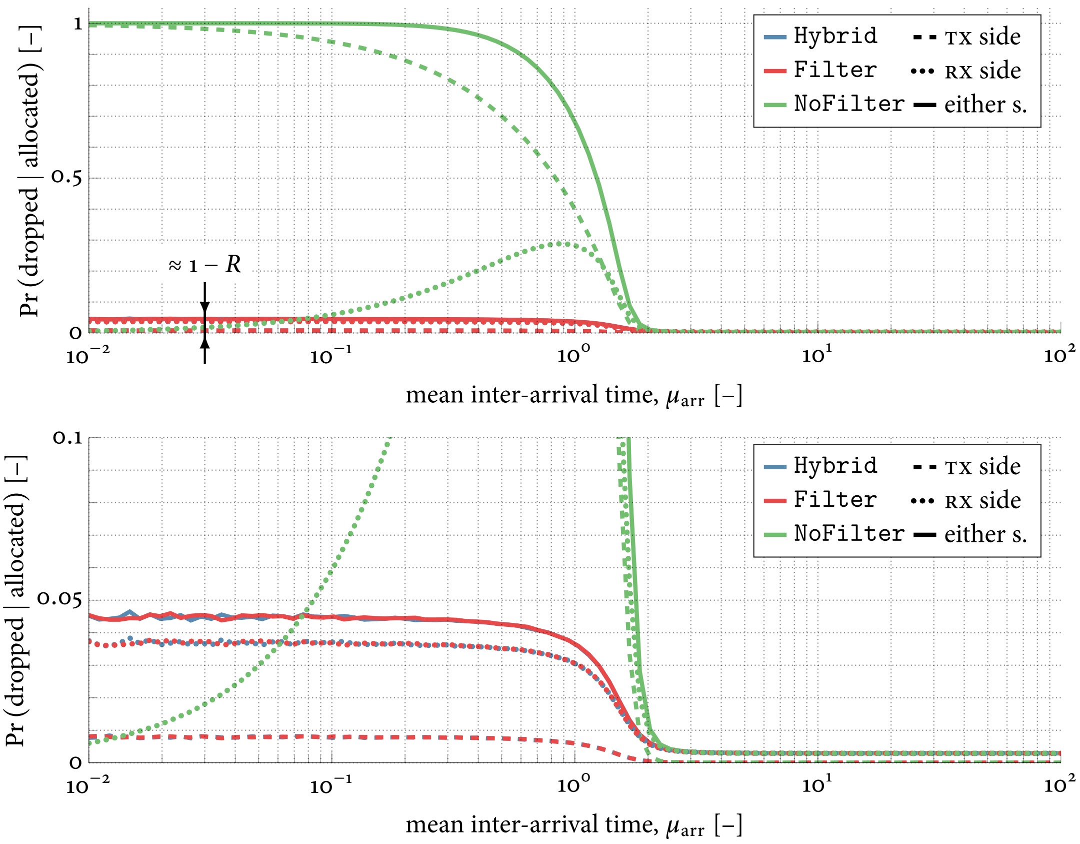


Figure 6.9: Probability of packet drop after allocation.

The "excess packets" allocated by the NoFilter implementation in the overloaded region are, however, unlikely to be served in time. Therefore, as the inter-arrival time decreases, the ratio of those which fail to be served before deadline – and are thus discarded by the TX side as shown in fig. 6.9 – increases.

In the case of the filtering implementations, the probability of discarding a packet is maintained (approximately) lower or equal to $1 - R$ regardless of the inter-arrival time since only packets likely to be delivered are allocated. Furthermore, in contrast to the NoFilter implementation, the overall packet drop ratio is dominated by the proportion of packets transmitted with a positive time margin which ultimately arrive late.

9. Kernel estimate [120, ch. 3] based on a normal kernel function with optimal bandwidth for normal data (applicable to general unimodal distributions [15, sec. 2.4.2]) and support $[0, 50]$ (logarithmic transformation of the support – as per Jones [65] – applied).

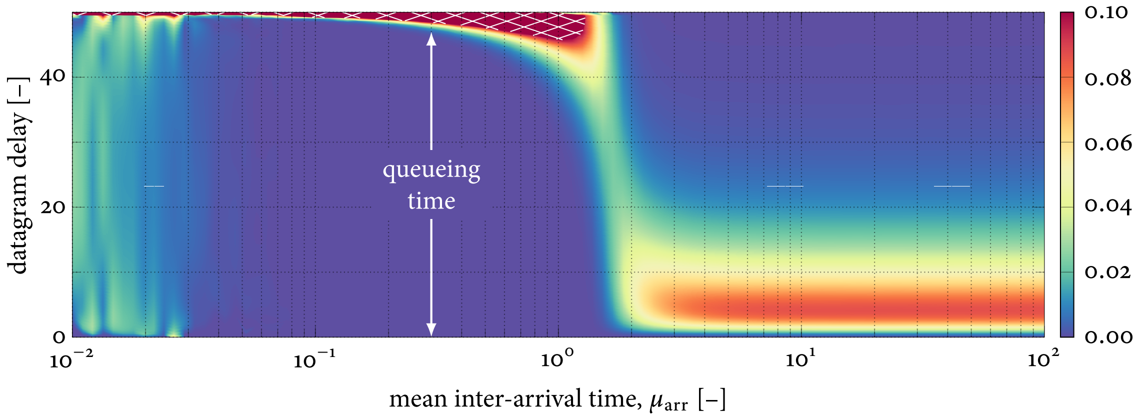


Figure 6.10: Empirical PDF[9] of the delay distribution of successfully delivered packets by the NoFilter implementation. The hatched area exceeds value of 0.1.

As seen in fig. 6.10 on the facing page, in the over-loaded region, most of the successfully delivered packets scheduled by the `NoFilter` scheduler arrive with delay close to the threshold value as the implementation attempts to deliver even those packets which have already incurred significant queueing time.

Furthermore, the density gradually shifts towards higher values with inverse of the inter-arrival time – that is, as the queueing effect intensifies and the proportion of the response time corresponding to queueing time[10] becomes dominant.

10. As usually defined in the queueing-theoretic literature, the *response time* (i.e. delay in this case) consists of the *queueing time* and the *service time*.

Each of the simulation runs[11] consisted of passing 10^7 packets through the simulated system. Nevertheless, the delay variance is noticeable in the leftmost part of the figure fig. 6.10 on the preceding page as the sample size decreases significantly (cf. fig. 6.12).

11. The simulation was run for 80 arrival rate values logarithmically spaced from 10^{-2} to 10^2.

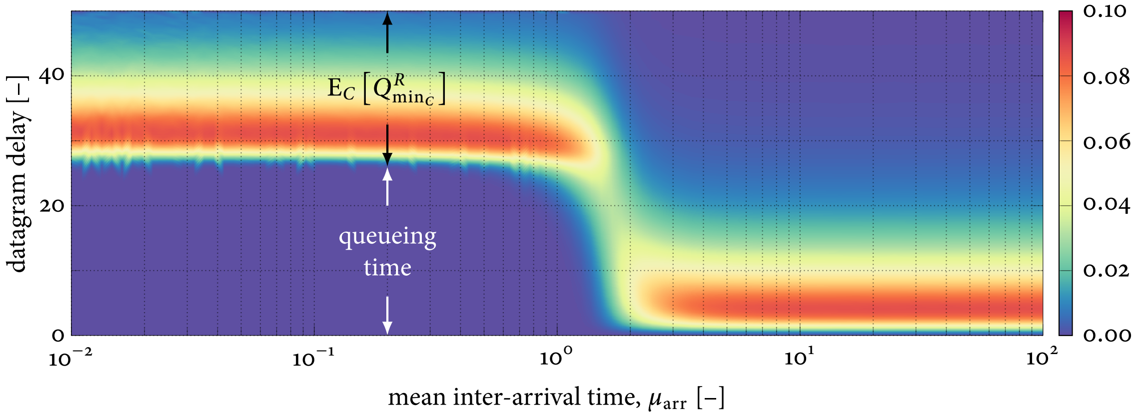


Figure 6.11: Empirical PDF of the delay distribution of successfully delivered packets by the `Filter` implementation. $\mathrm{E}_C[\cdot]$ denotes expectation over $\mathcal{C}$.

Conversely, the `Filter` and `Hybrid`[12] implementations discard "stale" packets to ultimately ensure that the probability of delivery of allocated packets is greater or equal to R and, thereby, constrain the "shift" of the delay distribution density along the delay axis (vertical in fig. 6.11) in the over-loaded region.

12. The fig. 6.11 plotted for the `Hybrid` implementation is nearly identical.

Note that the shape of the probability density function is virtually unchanged, apart from the shift representing the queueing delay component.

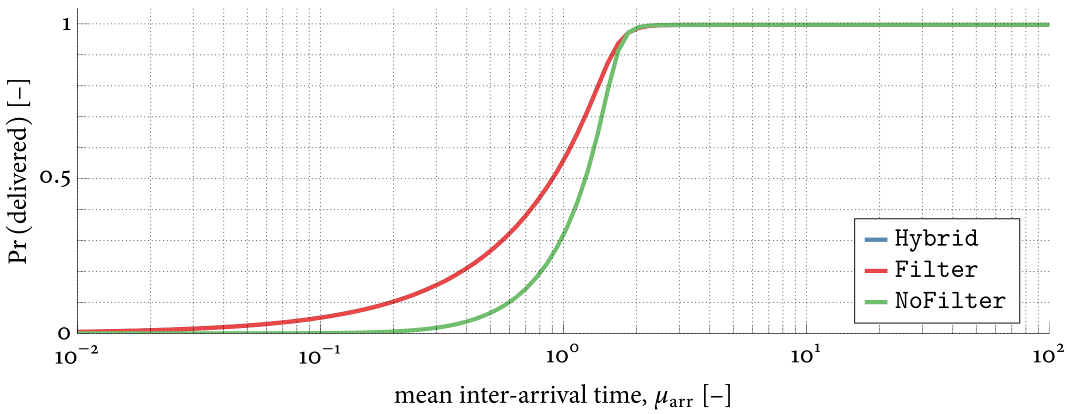


Figure 6.12: Probability of successful delivery. `Filter` and `Hybrid` overlap.

The most straightforward consequence of filtering implementations' effort to avoid squandering of the resources is their ability to deliver more traffic successfully[13]. Intuitively, the texttt Hybrid and `Filter` implementations have

13. For $R = 95\%$, the interface utilisation is virtually identical for all implementations.

more "manoeuvring space" to schedule packets.

Preliminary work on optimising the order and selection policies suggest that the deliverability can be further improved if the scheduler can "cherry-pick" from the input queue instead of following a simple ordering rule such as the EDF.

High reliability target

This sub-section assumes that the reliability is set to 99.9%. That is, *above* the value achievable without filtering in the under-loaded region.

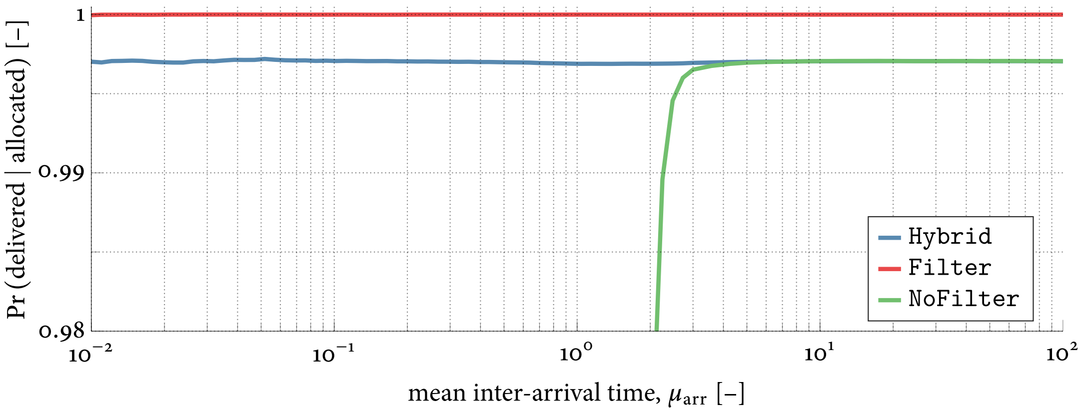


Figure 6.13: Probability of successful delivery of allocated packets for R = 99.9%.

The `Filter` scheduler achieves higher reliability[14] rates than attainable without filtering in the under-loaded region as shown in fig. 6.13. However, as demonstrated later, this behaviour implies worse performance in terms of overall deliverability which may be impractical for some applications.

14. 1 − Pr (dropped | allocated) or Pr (delivered | allocated)

To counter this effect the `Hybrid` implementation is designed to match the performance of the `NoFilter` implementation in the under-loaded region by filtering the queue only if it begins to grow beyond given threshold.

As seen it fig. 6.14, the probability density of the delay distribution obtained by simulating the `Filter` scheduler is concentrated evenly since the queueing delay is effectively eliminated.

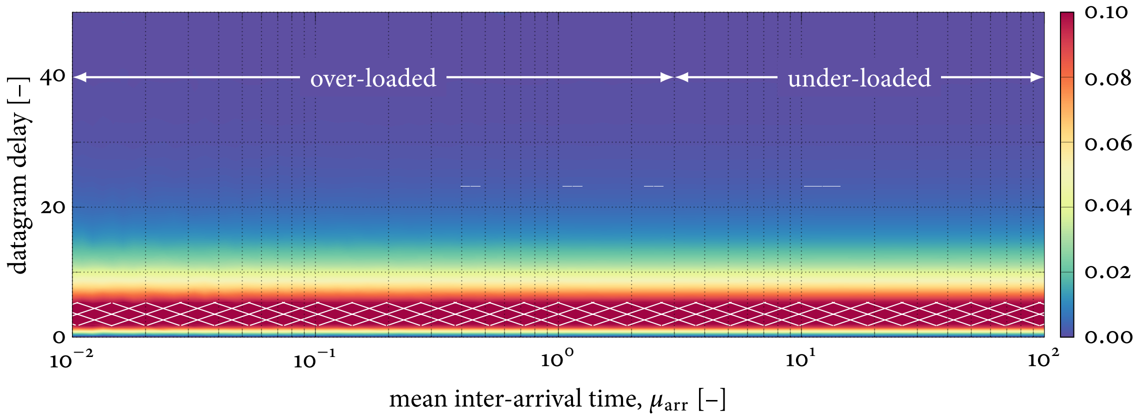


Figure 6.14: Empirical probability density of the delay distribution for the `Filter` implementation and R = 99.9%. The hatched area exceeds value of 0.1.

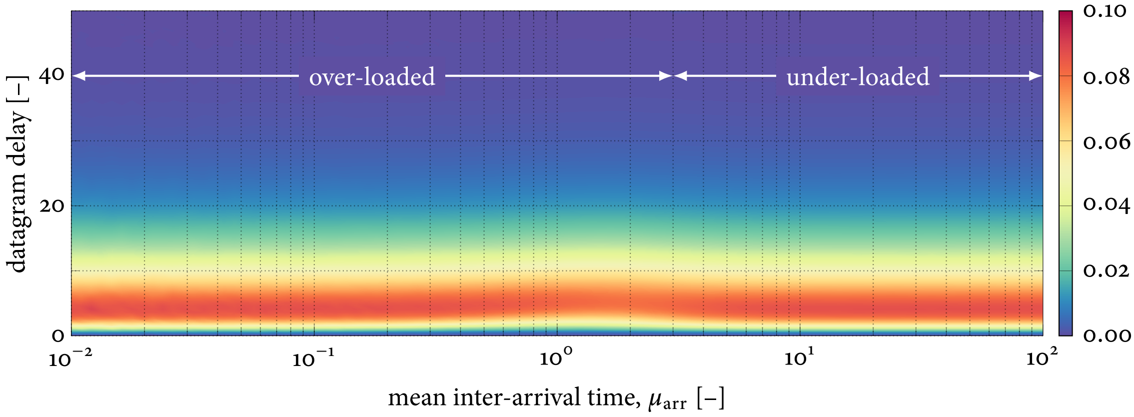


Figure 6.15: Empirical probability density of the delay distribution for the `Filter` implementation and $R = 99.9\%$. The hatched area exceeds value of 0.1.

The `Hybrid` scheduler is designed to imitate the under-loaded scenario across all arrival rates. The delay distribution in the under-loaded region ($\mu_{\text{arr}} \gtrapprox 2$) seen in fig. 6.15 is, therefore, identical to the one observed earlier in fig. 6.11 on page 57. With increasing arrival rate, the distribution undergoes a transition period (related to queue starvation as discussed later) and reverts to the "under-loaded shape".

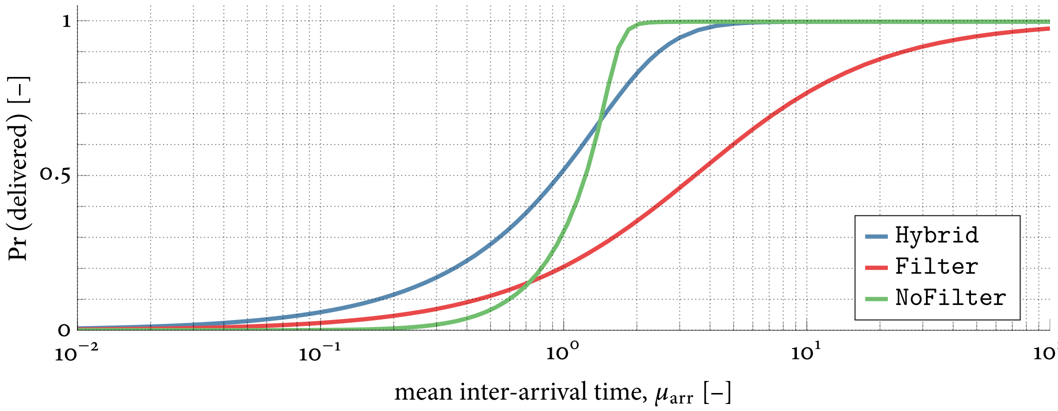


Figure 6.16: Probability of successful delivery for $R = 99.9\%$.

The stringent timing requirements following from evaluation of the quantile functions at 99.9% cause the algorithms to filter excessively effectively forcing the system into starvation (all packets in the input queue are dropped while the interfaces are idle) as illustrated in the fig. 6.17 on the following page. Although the `Hybrid` approach is less susceptible to this effect, an area where the `NoFilter` outperforms the `Hybrid` is clearly visible. Should this effect be eliminated (at the cost of lower energy efficiency), the filtering algorithms should relax the timing constraints when starvation is imminent (as done in the `Hybrid` variant).

Figure 6.17 on the next page further illustrates the under-utilization[15] of the interfaces performing filtering. Note that the utilization achieved by the `Filter` does not approach unity since the "combinations" including only a single interface are never selected[16] since for all interfaces I, $Q_{\text{SUM}_I} \approx 66.01$ which is greater than the required maximum delay of 50.

For $R = 95\%$ all three curves are approximately identical (overlap) and as

15. The mean utilization is obtained as the sum of the durations spent serving all packets divided by the total simulation time.

16. If the probability of selection a 3-interface combination were equal to the one corresponding to any of the 2-interface combinations, the asymptote could be approximated as $\frac{1}{2}\left(\frac{2}{3}+\frac{2}{3}\right) = 0.8\bar{3}$.

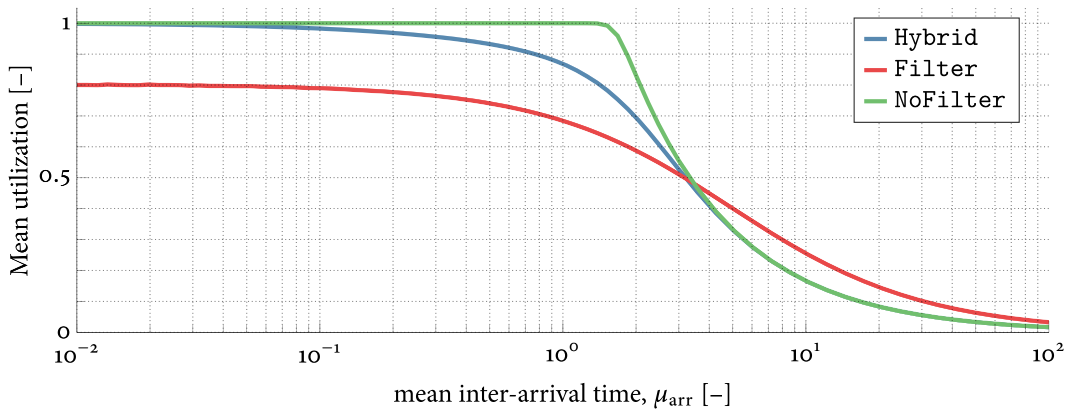


Figure 6.17: Average interface (time) utilization for $R = 99.9\%$.

can be observed from fig. 6.12 on page 57, there is no noticeable degradation of the performance in the vicinity of the knee in that case. The analysis of the inherent trade-off between the *risk* of not delivering allocated packets (lower efficiency) and the overall deliverability (higher-utilization) will be the subject of future work.

6.3.4 Queue order

While processing the input traffic queues according to the Last Come First Served (LCFS) instead of the FCFS discussed in the text above is generally impractical, such approach is significantly less susceptible to the queueing effect as the freshest packets are processed first. The author implemented the equivalent schedulers to verify that the queueing effect can be "eliminated" by setting R to high enough level. Since the plots provided in the section 6.3.3 on page 58 are visually identical to those obtained by evaluating the scheduling process with an FCFS policy[17], the author concludes that when R is set sufficiently high, the filtering algorithms presented above perform equivalently to the ideal, queue-less scenario.

17. With R set to 99.9% as discussed in the subsection *High reliability target* starting on page 58.

PART IV

SUMMARY

7

CONCLUSION

This thesis lays the groundwork for the development of adaptive, opportunistic, QoS-aware scheduling algorithms by modelling the multi-RAT device. First, the system is described analytically and solved using the framework of queueing-theory. Second, a complementary algorithmic approach exploiting the concept of reactive just-in-time decision making is explored.

The results of the research which the author has provided in this thesis can be categorised into three venues:

1. The author has given a structured overview of the research on employing multiple interfaces as a means to optimize the quality of service metrics and relevant wireless access technologies.

2. A predictive queuing-theoretic model enabling accurate evaluation of the system performance has been developed. Applicability of the model to non-Markovian systems has been studied.

3. To address the limitations of the aforementioned predictive model a reactive approach employing just-in-time scheduling technique has been adopted and analysed.

PREDICTIVE APPROACH The author provides an exact analytical solution to the queueing-theoretic model under Markovian assumptions and FCFS queueing policy. The structure of the expression implies the existence of an elegant and intuitive decomposition of the inherently complex model into a series of subsequent M/M/1 systems.

Interestingly, a simple closed-form expression for the probability of the event that a given interface will serve the job first if multiple interfaces are serving copies of the job in parallel has been derived which enabled the author to decouple the service time and the network time distributions significantly broadening the class of compatible systems.

An extensive high-performance simulation framework is developed and employed to evaluate some non-Markovian configurations of the model. The results suggest that if a relatively small error is permissible, the Markovian model can be extended to structurally simple systems with more generally distributed service time distribution as long as the system load is low and the tails of the distributions are light.

Unfortunately, yet as expected, the generalisation of the model to a heterogeneous queueing system with general service time distribution has not been derived. Furthermore, since the classical response time analysis relying on mean value analysis has been employed, the solution has been demonstrated not to be suitable for highly dynamic systems.

Reactive approach The author set to explore a novel paradigm of just-in-time scheduling which relies on deferring the scheduling decision for as long as possible while collecting information about the system under control to be used in the decision process.

The author provides a simple algebra on random variables as a framework to decompose the statistical model to enable the use of the model even if only an incomplete description of the system (e.g. the service and network times are known for each interface independently yet the correlation structure is unknown). A matlab implementation of the framework along with a simulation framework optimised for massively parallel execution is developed.

The empirical analysis confirms that the presented technique can be employed to improve system efficiency, datagram delivery ratio and to make the system under control robust against overloading.

7.1 Future work

The author recognizes that the opportunities to further extend the predictive model are limited. However, the addition of transient analysis of the Markovian scenario may provide interesting insights. Furthermore, a possible application of the model to other domains such as the multi-server MapReduce computing will be explored. Finally, a derived scheduler implementation will be compared against the jit scheduler.

The empirical analysis of the jit scheduler revealed the need to optionally allow the algorithm to "risk" scheduling datagrams less likely to be delivered in time to avoid interface idling due to excessive filtering. The author expects to employ network calculus and/or reinforcement learning to model the backlog size and to act accordingly.

The general problem formulation assumes the existence of multiple traffic classes. The handling of multiple classes, however, requires a definition of optimality – e.g. an optimal solution could ensure fair allocation of resources, equal drop probability across traffic classes or follow application-defined preferences. Performance of these variants of the jit scheduler is yet to be evaluated. In addition to the simplistic approaches such as round-robin scheduling, the information about the system performance can be exploited to perform proportional-fair (or similar) scheduling.

The proposed scheduling approach relies internally on two strategies governing the ordering of the input queue(s) and the ultimate selection of the optimal combination of interfaces once the timing constraints are known. To enable demonstration of the concept, the author has implemented simple heuristic. However, a more thorough analysis of those policies is warranted. Concerning the selection policy, the ideal solution could potentially be obtained by solving the associated Markov decision problem.

As noted in the literature[1], the introduction of redundancy increases utilisation of both the transmitter and the receiver sides and therefore may lead to overall network performance degradation due to interference, increased queuing delays in buffers, increased energy consumption, etc. if the resulting knock-on effects are not considered. A new research direction of its own would explore the network-wide consequences of the increased overhead.

1. C.f. literature review entry 11 on page 23.

Although the proposed framework is suitable for arbitrarily distributed systems, an optimisation incorporating predictable variations[2] of the delay distributions – possibly using machine learning – and development of parametric models for the delay distributions based on empirical data would be welcome. Finally, an experimental evaluation of the existing propositions is lacking.

2. E.g., in time.

Finally, an experimental evaluation of the presented concepts and development of a detailed methodology for measuring the service and network time distributions is warranted.

BIBLIOGRAPHY

[1] 3GPP. *Study on scenarios and requirements for next generation access technologies*. Technical Report (TR) 38.913. Version 15.0.0. 3rd Generation Partnership Project (3GPP), July 2018. URL: http://www.3gpp.org/DynaReport/38913.htm (Cited on p. 9).

[2] A. B. M. Alim Al Islam et al. "Channel Assignment Techniques for Multi-Radio Wireless Mesh Networks: A Survey". In: *IEEE Communications Surveys Tutorials* 18.2 (Dec. 2016), pp. 988–1017. ISSN: 1553-877X. DOI: 10.1109/COMST.2015.2510164 (Cited on pp. 43, 45, 46).

[3] Sergey Andreevy et al. "Understanding the IoT connectivity landscape: a contemporary M2M radio technology roadmap". In: *Communications Magazine, IEEE* 53.9 (Sept. 2015), pp. 32–40. ISSN: 0163-6804. DOI: 10.1109/MCOM.2015.7263370 (Cited on p. 6).

[4] J. G. Andrews. "Seven ways that HetNets are a cellular paradigm shift". In: *IEEE Communications Magazine* 51.3 (Mar. 2013), pp. 136–144. ISSN: 0163-6804. DOI: 10.1109/MCOM.2013.6476878 (Cited on p. 9).

[5] J. G. Andrews et al. "What Will 5G Be?" In: *IEEE Journal on Selected Areas in Communications* 32.6 (June 2014), pp. 1065–1082. ISSN: 0733-8716. DOI: 10.1109/JSAC.2014.2328098 (Cited on p. 9).

[6] Matthew Andrews. "A Survey of Scheduling Theory in Wireless Data Networks". In: *Wireless Communications*. Ed. by Prathima Agrawal et al. New York, NY: Springer New York, 2007, pp. 1–17. ISBN: 978-0-387-48945-2. DOI: 10.1007/978-0-387-48945-2_1. URL: https://doi.org/10.1007/978-0-387-48945-2_1 (Cited on p. 43).

[7] A. Asadi and V. Mancuso. "A Survey on Opportunistic Scheduling in Wireless Communications". In: *IEEE Communications Surveys Tutorials* 15.4 (Fourth 2013), pp. 1671–1688. ISSN: 1553-877X. DOI: 10.1109/SURV.2013.011413.00082 (Cited on p. 43).

[8] Kevin Ashton. *That 'Internet of Things' Thing*. 2009. URL: http://www.rfidjournal.com/articles/view?4986 (visited on 05/18/2016) (Cited on p. 3).

[9] Mehdi Bennis, Mérouane Debbah, and H. Vincent Poor. "Ultra-Reliable and Low-Latency Wireless Communication: Tail, Risk and Scale". In: (Jan. 4, 2018). arXiv: http://arxiv.org/abs/1801.01270v2 [cs.IT] (Cited on p. 12).

[10] D. A. Beyer. "Accomplishments of the DARPA SURAN Program". In: *IEEE Conference on Military Communications*. Sept. 1990, 855–862 vol.2. DOI: 10.1109/MILCOM.1990.117536 (Cited on p. 45).

[11] Anders Björsson. “Simulation analysis of RLC/MAC for UMTS in Network Simulator version 2”. Linköping University, Department of Electrical Engineering, 2004, p. 43. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-2105 (visited on 09/12/2018) (Cited on p. 21).

[12] Jacek Blazewicz et al. *Scheduling Computer and Manufacturing Processes*. 2nd ed. Berlin: Springer-Verlag Berlin Heidelberg, 2001, p. 485. ISBN: 9783662043639. DOI: 10.1007/978-3-662-04363-9 (Cited on p. 43).

[13] Bluetooth SIG. *Bluetooth Core Specification*. Version 5.1. Jan. 21, 2019 (Cited on p. 4).

[14] C. Bormann, M. Ersue, and A. Keranen. *Terminology for Constrained-Node Networks*. STD 6. RFC Editor, May 2014. URL: http://www.rfc-editor.org/rfc/rfc7228.txt (Cited on pp. v–vii).

[15] Adrian W. Bowman and Adelchi Azzalini. *Applied Smoothing Techniques for Data Analysis: The Kernel Approach with S-Plus Illustrations*. Oxford Statistical Science Series. OUP Oxford, 1997, p. 204. ISBN: 9780191545696 (Cited on p. 56).

[16] Ismail Butun, Nuno Pereira, and Mikael Gidlund. “Analysis of LoRaWAN V1.1 Security: Research Paper”. In: *Proceedings of the 4th ACM MobiHoc Workshop on Experiences with the Design and Implementation of Smart Objects*. SMARTOBJECTS ’18. Los Angeles, California: ACM, 2018, 5:1–5:6. ISBN: 978-1-4503-5857-6. DOI: 10.1145/3213299.3213304 (Cited on p. 7).

[17] K. Chebrolu and R. Rao. “Communication using multiple wireless interfaces”. In: *2002 IEEE Wireless Communications and Networking Conference Record. WCNC 2002 (Cat. No.02TH8609)*. IEEE, 2002. DOI: 10.1109/wcnc.2002.993516 (Cited on p. 20).

[18] K. Chebrolu and R. R. Rao. “Bandwidth aggregation for real-time applications in heterogeneous wireless networks”. In: *IEEE Transactions on Mobile Computing* 5.4 (Apr. 2006), pp. 388–403. DOI: 10.1109/tmc.2006.1599407 (Cited on p. 21).

[19] Chen-Nee Chuah et al. “Capacity scaling in MIMO wireless systems under correlated fading”. In: *IEEE Transactions on Information Theory* 48.3 (Mar. 2002), pp. 637–650. ISSN: 0018-9448. DOI: 10.1109/18.985982 (Cited on p. 10).

[20] Benjamin Cox et al. “N-variant Systems: A Secretless Framework for Security Through Diversity”. In: *Proceedings of the 15th Conference on USENIX Security Symposium - Volume 15*. USENIX-SS’06. Vancouver, B.C., Canada: USENIX Association, 2006. URL: https://www.usenix.org/conference/15th-usenix-security-symposium/n-variant-systems-secretless-framework-security-through (visited on 05/14/2019) (Cited on p. 10).

[21] R. L. Cruz. “Quality of service guarantees in virtual circuit switched networks”. In: *IEEE Journal on Selected Areas in Communications* 13.6 (Aug. 1995), pp. 1048–1056. ISSN: 0733-8716. DOI: 10.1109/49.400660 (Cited on p. 44).

[22] CTO. *General Authorisation No. VO-R/10/11.2016-13 for the use of radio frequencies and for the operation of short range devices*. Czech Telecommunication Office, May 2016. URL: https://www.ctu.eu/general-authorisation-no-vo-r10112016-13 (visited on 05/16/2019) (Cited on p. 10).

[23] A. Demers, S. Keshav, and S. Shenker. "Analysis and Simulation of a Fair Queueing Algorithm". In: *Symposium Proceedings on Communications Architectures &Amp; Protocols*. SIGCOMM '89. Austin, Texas, USA: ACM, 1989, pp. 1–12. ISBN: 0-89791-332-9. DOI: 10.1145/75246.75248 (Cited on p. 44).

[24] Almudena Díaz Zayas et al. "3GPP Evolution on LTE Connectivity for IoT". In: *Integration, Interconnection, and Interoperability of IoT Systems*. Ed. by Raffaele Gravina et al. Cham: Springer International Publishing, 2018, pp. 1–20. ISBN: 978-3-319-61300-0. DOI: 10.1007/978-3-319-61300-0_1. URL: https://doi.org/10.1007/978-3-319-61300-0_1 (Cited on p. 8).

[25] Bruno Dorsemaine et al. "Internet of Things: A Definition & Taxonomy". In: *Next Generation Mobile Applications, Services and Technologies, 2015 9th International Conference on*. Sept. 2015, pp. 72–77. DOI: 10.1109/NGMAST.2015.71 (Cited on p. 3).

[26] EN 300 220-1. *Electromagnetic compatibility and Radio spectrum Matters (ERM); Short Range Devices (SRD); Radio equipment to be used in the 25 MHz to 1 000 MHz frequency range with power levels ranging up to 500 mW; Part 1: Technical characteristics and test methods*. Version 2.4.1. Sophia Antipolis, France: European Telecommunications Standards Institute, May 2012. URL: https://www.etsi.org/deliver/etsi_en/300200_300299/30022001/02.04.01_40/en_30022001v020401o.pdf (Cited on pp. 4, 10).

[27] Frida Eng. *New Modules for ns-2*. 2004. URL: https://web.archive.org/web/20041208131443/http://www.rt.isy.liu.se:80/~frida/nsmodules/ (visited on 09/12/2018) (Cited on p. 21).

[28] Ericsson. *5G radio access*. en. Tech. rep. Version C. White Paper. Apr. 2016. 10 pp. URL: https://www.ericsson.com/assets/local/publications/white-papers/wp-5g.pdf (visited on 05/14/2019) (Cited on p. 9).

[29] ETSI. *The Radio Regulations*. Geneva, Switzerland: ITU Radiocommunication Sector, 2012 (Cited on p. 3).

[30] S. Farrell. *Low-Power Wide Area Network (LPWAN) Overview*. RFC 8376. RFC Editor, May 2018 (Cited on p. 6).

[31] H. Fattah and C. Leung. "An overview of scheduling algorithms in wireless multimedia networks". In: *IEEE Wireless Communications* 9.5 (Oct. 2002), pp. 76–83. ISSN: 1536-1284. DOI: 10.1109/MWC.2002.1043857 (Cited on p. 43).

[32] M. Fidler and A. Rizk. "A Guide to the Stochastic Network Calculus". In: *IEEE Communications Surveys Tutorials* 17.1 (July 2015), pp. 92–105. ISSN: 1553-877X. DOI: 10.1109/COMST.2014.2337060 (Cited on p. 45).

[33] Maurice René Fréchet. “Remarques au sujet de la note précédente”. In: *Comptes-Rendus de l'Academie des Sciences de Paris*. 1 Math 246 (1958), pp. 2719–2720 (Cited on p. A2).

[34] Maurice René Fréchet. “Sur les tableaux de corrélations dont les marges sont données”. In: *Annales de l'Université de Lyon. Section A, Sciences mathématiques et astronomie*. 4th ser. 14 (1951), pp. 53–77 (Cited on p. A2).

[35] Henry Laurence Gantt. *Work Wages and Profits*. 2nd ed. Works management library. New York: The Engineering magazine co., 1919. 312 pp. (Cited on p. 47).

[36] Kristen Gardner. “Modeling and Analyzing Systems with Redundancy”. PhD Thesis Proposal. Carnegie Mellon University, May 2017. 170 pp. URL: http://www.cs.cmu.edu/~harchol/gardner_thesis.pdf (visited on 09/24/2018) (Cited on p. 26).

[37] Kristen Gardner. “Thesis Proposal: Analyzing Systems with Redundant Requests”. PhD thesis. Carnegie Mellon University, May 2015. 34 pp. URL: http://www.cs.cmu.edu/afs/cs/user/ksgardne/www/KGardner_Proposal.pdf (visited on 09/24/2018) (Cited on p. 26).

[38] Kristen Gardner et al. “Queueing with redundant requests: exact analysis”. In: *Queueing Systems* 83.3-4 (July 2016), pp. 227–259. DOI: 10.1007/s11134-016-9485-y (Cited on pp. 25, 26).

[39] Kristen Gardner et al. “Reducing Latency via Redundant Requests”. In: *Proceedings of the 2015 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems - SIGMETRICS '15*. ACM Press, 2015. DOI: 10.1145/2745844.2745873 (Cited on pp. 25, 26).

[40] Kristen Gardner et al. “Scheduling for efficiency and fairness in systems with redundancy”. In: *Performance Evaluation* 116 (Nov. 2017), pp. 1–25. DOI: 10.1016/j.peva.2017.07.001 (Cited on pp. 26–28).

[41] Kirsten Garvin. *2.4 GHz – The Ideal Unlicensed Spectrum for Long-Range IoT Networking (Part II)*. Ingenu Inc. 2015. URL: https://www.ingenu.com/2015/08/2-4-ghz-the-ideal-unlicensed-spectrum-for-long-range-iot-networking-part-ii/ (visited on 05/18/2019) (Cited on p. 6).

[42] Carles Gomez et al. “A Sigfox Energy Consumption Model”. In: *Sensors* 19.3 (Feb. 2019), p. 681. DOI: 10.3390/s19030681 (Cited on p. 7).

[43] Claire Goursaud and Jean-Marie Gorce. “Dedicated networks for IoT: PHY / MAC state of the art and challenges”. In: *EAI Endorsed Transactions on Internet of Things* 15.1 (Oct. 2015). DOI: 10.4108/eai.26-10-2015.150597 (Cited on pp. 7, 8).

[44] Varun Gupta et al. “On the inapproximability of M/G/K: why two moments of job size distribution are not enough”. In: *Queueing Systems* 64.1 (Aug. 2009), pp. 5–48. DOI: 10.1007/s11134-009-9133-x (Cited on p. 20).

[45] E. Gustafsson and A. Jonsson. “Always best connected”. In: *IEEE Wireless Communications* 10.1 (Feb. 2003), pp. 49–55. DOI: 10.1109/mwc.2003.1182111 (Cited on p. 11).

[46] Jaap Haartsen. "How we made Bluetooth". In: *Nature Electronics* 1.12 (Dec. 2018), pp. 661–661. DOI: 10.1038/s41928-018-0186-x (Cited on p. 4).

[47] M. Haenggi. "Diversity Loss Due to Interference Correlation". In: *IEEE Communications Letters* 16.10 (Oct. 2012), pp. 1600–1603. ISSN: 1089-7798. DOI: 10.1109/LCOMM.2012.082012.120863 (Cited on p. 10).

[48] V. Hauser and T. Hégr. "Proposal of Adaptive Data Rate Algorithm for LoRaWAN-Based Infrastructure". In: *2017 IEEE 5th International Conference on Future Internet of Things and Cloud (FiCloud)*. Aug. 2017, pp. 85–90. DOI: 10.1109/FiCloud.2017.47 (Cited on p. 7).

[49] Wassily Hoeffding. "Masstabinvariante Korrelationsmasse für diskontinuierliche Verteilungen". German. In: *Archiv für mathematischen Wirtschaften und Sozialforschung* 7 (1941). reprinted as "'Scale-Invariant Correlation Measures for Discontinuous Distributions"' [51], pp. 49–70 (Cited on p. A2).

[50] Wassily Hoeffding. "Masstabinvariante Korrelationstheorie". German. In: *Schriften des Mathematischen Seminars und des Instituts für Angewandte Mathematik der Universität Berlin* 5.3 (1940). reprinted as "'Scale-Invariant Correlation Theory"' [52], pp. 181–233 (Cited on p. A2).

[51] Wassily Hoeffding. "Scale-Invariant Correlation Measures for Discontinuous Distributions". In: *The Collected Works of Wassily Hoeffding*. Ed. by N. I. Fisher and P. K. Sen. New York, NY: Springer New York, 1994, pp. 109–133. ISBN: 978-1-4612-0865-5. DOI: 10.1007/978-1-4612-0865-5_5 (Cited on p. xxiii).

[52] Wassily Hoeffding. "Scale-Invariant Correlation Theory". In: *The Collected Works of Wassily Hoeffding*. Ed. by N. I. Fisher and P. K. Sen. New York, NY: Springer New York, 1994, pp. 57–107. ISBN: 978-1-4612-0865-5. DOI: 10.1007/978-1-4612-0865-5_4 (Cited on p. xxiii).

[53] Bernd Holfeld et al. "Wireless Communication for Factory Automation: an opportunity for LTE and 5G systems". In: *IEEE Communications Magazine* 54.6 (June 2016), pp. 36–43. DOI: 10.1109/mcom.2016.7497764 (Cited on p. 9).

[54] J. Hui and P. Thubert. *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*. RFC 6282. http://www.rfc-editor.org/rfc/rfc6282.txt. RFC Editor, Sept. 2011. URL: http://www.rfc-editor.org/rfc/rfc6282.txt (Cited on p. 6).

[55] Jorge Ibarra-Esquer et al. "Tracking the Evolution of the Internet of Things Concept Across Different Application Domains". In: *Sensors* 17.6 (June 2017), p. 1379. ISSN: 1424-8220. DOI: 10.3390/s17061379 (Cited on p. 3).

[56] "IEEE Standard for Information technology – Telecommunications and information exchange between systems Local and metropolitan area networks –Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications". In: *IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)* (Dec. 2016), pp. 1–3534. DOI: 10.1109/IEEESTD.2016.7786995 (Cited on p. 9).

[57] "IEEE Standard for Information technology–Telecommunications and information exchange between systems–Local and metropolitan area networks–Specific requirements–Part 19: Wireless Network Coexistence Methods". In: *IEEE Std 802.19.1-2018* (Nov. 2018), pp. 1–458. DOI: 10.1109/IEEESTD.2018.8520953 (Cited on p. 4).

[58] "IEEE Standard for Local and metropolitan area networks – Link Aggregation". In: *IEEE Std 802.1AX-2014 (Revision of IEEE Std 802.1AX-2008)* (Dec. 2014), pp. 1–344. DOI: 10.1109/IEEESTD.2014.7055197 (Cited on p. 9).

[59] "IEEE Standard for Low-Rate Wireless Networks". In: *IEEE Std 802.15.4-2015 (Revision of IEEE Std 802.15.4-2011)* (Apr. 2016), pp. 1–709. DOI: 10.1109/IEEESTD.2016.7460875 (Cited on p. 4).

[60] International Telecommunication Union. *The Tactile Internet*. 2014, p. 21. ISBN: 9789261184919. URL: http://handle.itu.int/11.1002/pub/80ae6ed7-en (Cited on pp. v–vii, 9).

[61] IoT Analytics. *LPWAN Market Report 2018-2023*. Tech. rep. IoT Analytics GmbH, Sept. 17, 2018. 267 pp. URL: https://iot-analytics.com/product/lpwan-market-report-2018-2023/ (visited on 03/09/2019) (Cited on p. 10).

[62] ITU-R. *IMT Vision. Framework and overall objectives of the future development of IMT for 2020 and beyond*. Rec. M.2083. Geneva: Internation Telecommunication Union, Sept. 2015. 21 pp. URL: https://www.itu.int/rec/R-REC-M.2083 (visited on 09/13/2018) (Cited on p. 9).

[63] ITU-R. *Radiocommunication objectives and requirements for Public Protectionand Disaster Relief (PPDR)*. Tech. rep. M.2377-1. International Telecommunication Union, Nov. 2017. 175 pp. URL: https://www.itu.int/pub/R-REP-M.2377 (visited on 05/17/2019) (Cited on p. 6).

[64] J. Sachs and G. Wikstrom and T. Dudda and R. Baldemair and K. Kittichokechai. "5G radio network design for ultra-reliable low-latency communication". In: *IEEE Network* 32.2 (Mar. 2018), pp. 24–31 (Cited on p. 9).

[65] M. C. Jones. "Simple boundary correction for kernel density estimation". In: *Statistics and Computing* 3.3 (Sept. 1993), pp. 135–146. ISSN: 1573-1375. DOI: 10.1007/BF00147776 (Cited on p. 56).

[66] Gauri Joshi. "Synergy via Redundancy. Boosting Service Capacity with Adaptive Replication". In: *ACM SIGMETRICS Performance Evaluation Review* 45.2 (Mar. 2018), pp. 21–28. DOI: 10.1145/3199524.3199530 (Cited on p. 24).

[67] Gauri Joshi, Emina Soljanin, and Gregory Wornell. "Efficient Redundancy Techniques for Latency Reduction in Cloud Systems". In: *ACM Transactions on Modeling and Performance Evaluation of Computing Systems* 2.2 (Apr. 2017), pp. 1–30. DOI: 10.1145/3055281 (Cited on p. 24).

[68] Sounak Kar, Amr Rizk, and Markus Fidler. "Multi-interface Communication: Interface Selection under Statistical Performance Constraints". English. In: *2018 30th International Teletraffic Congress (ITC 30)* 02 (Sept. 2018), pp. 7–12. DOI: 10.1109/ITC30.2018.10054 (Cited on p. 44).

[69] S. Kaul et al. "Minimizing age of information in vehicular networks". In: *2011 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*. June 2011, pp. 350–358. DOI: 10.1109/SAHCN.2011.5984917 (Cited on p. 48).

[70] John B. Kennedy. "When woman is Boss". In: *Collier's Illustrated Weekly* 77 (1926). An interview with Nikola Tesla by John B. Kennedy, pp. 163–180. URL: http://www.tfcbooks.com/tesla/1926-01-30.htm (visited on 05/16/2019) (Cited on p. 3).

[71] Toshikazu Kimura. "A Two-Moment Approximation for the Mean Waiting Time in the GI/G/s Queue". In: *Management Science* 32.6 (June 1986), pp. 751–763. ISSN: 0025-1909. DOI: 10.1287/mnsc.32.6.751 (Cited on p. 20).

[72] J. F. C. Kingman and M. F. Atiyah. "The single server queue in heavy traffic". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 57.04 (Oct. 1961), p. 902. DOI: 10.1017/s0305004100036094 (Cited on p. 20).

[73] G. Kortuem et al. "Smart objects as building blocks for the Internet of things". In: *IEEE Internet Computing* 14.1 (Jan. 2010), pp. 44–51. ISSN: 1089-7801. DOI: 10.1109/MIC.2009.143 (Cited on p. 3).

[74] A. Kosta, N. Pappas, and V. Angelakis. *Age of Information: A New Concept, Metric, and Tool*. now, 2017. ISBN: 168083360X (Cited on p. 48).

[75] George Koudouridis et al. "Multi-Radio Access in Ambient Networks". In: *IST Workshop*. Nov. 2005. URL: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-28564 (visited on 09/11/2018) (Cited on p. 21).

[76] Vladik Kreinovich and Scott Ferson. "Computing best-possible bounds for the distribution of a sum of several variables is NP-hard". In: *International Journal of Approximate Reasoning* 41.3 (Apr. 2006), pp. 331–342. DOI: 10.1016/j.ijar.2005.06.009 (Cited on p. 49).

[77] Sunil Suresh Kulkarni and Catherine Rosenberg. "Opportunistic Scheduling: Generalizations to Include Multiple Constraints, Multiple Interfaces, and Short Term Fairness". In: *Wireless Networks* 11.5 (Sept. 2005), pp. 557–569. ISSN: 1572-8196. DOI: 10.1007/s11276-005-3512-y (Cited on p. 44).

[78] H. Kurunathan et al. "IEEE 802.15.4e in a Nutshell: Survey and Performance Evaluation". In: *IEEE Communications Surveys Tutorials* 20.3 (thirdquarter 2018), pp. 1989–2010. ISSN: 1553-877X. DOI: 10.1109/COMST.2018.2800898 (Cited on p. 5).

[79] N. Kushalnagar, G. Montenegro, and C. Schumacher. *IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals*. RFC 4919. http://www.rfc-editor.org/rfc/rfc4919.txt. RFC Editor, Aug. 2007. URL: http://www.rfc-editor.org/rfc/rfc4919.txt (Cited on p. 6).

[80] Manuel Laguna and José Luis González Velarde. "A search heuristic for just-in-time scheduling in parallel machines". In: *Journal of Intelligent Manufacturing* 2.4 (Aug. 1991), pp. 253–260. ISSN: 1572-8145. DOI: 10.1007/BF01471113 (Cited on p. 43).

[81] J. Landt. "The history of RFID". In: *IEEE Potentials* 24.4 (Oct. 2005), pp. 8–11. ISSN: 0278-6648. DOI: 10.1109/MP.2005.1549751 (Cited on p. 3).

[82] M. Lauridsen et al. "Coverage Comparison of GPRS, NB-IoT, LoRa, and SigFox in a 7800 km² Area". In: *2017 IEEE 85th Vehicular Technology Conference (VTC Spring)*. June 2017, pp. 1–5. DOI: 10.1109/VTCSpring.2017.8108182 (Cited on p. 8).

[83] Kangwook Lee, Ramtin Pedarsani, and Kannan Ramchandran. "On Scheduling Redundant Requests With Cancellation Overheads". In: *IEEE/ACM Transactions on Networking* 25.2 (Apr. 2017), pp. 1279–1290. DOI: 10.1109/tnet.2016.2622248 (Cited on pp. 25, 39).

[84] William C. Y. Lee. "Antenna Spacing Requirement for a Mobile Radio Base-Station Diversity". In: *The Bell System Technical Journal* 50.6 (July 1971), pp. 1859–1876. ISSN: 0005-8580. DOI: 10.1002/j.1538-7305.1971.tb02585.x (Cited on p. 10).

[85] Xiaohua Li, Juite Hwu, and E. Paul Ratazzi. "Using Antenna Array Redundancy and Channel Diversity for Secure Wireless Transmissions". In: *Journal of Communications* 2 (May 2007), pp. 24–32 (Cited on p. 10).

[86] Bev Littlewood and Lorenzo Strigini. "Redundancy and Diversity in Security". In: *Computer Security – ESORICS 2004*. Ed. by Pierangela Samarati et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 423–438. ISBN: 978-3-540-30108-0. DOI: 10.1007/978-3-540-30108-0_26 (Cited on p. 10).

[87] K. Liu, N. Abu-Ghazaleh, and K. Kang. "JiTS: just-in-time scheduling for real-time sensor data dissemination". In: *Fourth Annual IEEE International Conference on Pervasive Computing and Communications (PERCOM'06)*. Mar. 2006, 5 pp.–46. DOI: 10.1109/PERCOM.2006.33 (Cited on p. 43).

[88] LoRa Alliance. *LoRaWAN™Specification*. Ed. by N. Sornin. Version 1.1. LoRa™Alliance, Jan. 2017. URL: https://lora-alliance.org/resource-hub/lorawantm-specification-v11 (Cited on p. 7).

[89] G. D. Makarov. "Estimates for the Distribution Function of a Sum of Two Random Variables When the Marginal Distributions are Fixed". In: *Theory of Probability & Its Applications* 26.4 (Jan. 1982), pp. 803–806. DOI: 10.1137/1126086 (Cited on p. 49).

[90] Peter Mattson et al. "Communication Scheduling". In: *Proceedings of the Ninth International Conference on Architectural Support for Programming Languages and Operating Systems*. ASPLOS IX. Cambridge, Massachusetts, USA: ACM, 2000, pp. 82–92. ISBN: 1-58113-317-0. DOI: 10.1145/378993.379005 (Cited on p. 43).

[91] Michael P. McGarry et al. "Just-in-Time Scheduling for Multichannel EPONs". In: *J. Lightwave Technol.* 26.10 (May 2008), pp. 1204–1216. URL: http://jlt.osa.org/abstract.cfm?URI=jlt-26-10-1204 (Cited on p. 43).

[92] Peter Miklavčič. “On the number of non-overlapping channels in the IEEE 802.11 WLANs operating in the 2.4 GHz band”. In: *Elektrotehniski Vestnik* 81.3 (Jan. 2014), pp. 148–152. URL: http://www.dlib.si/?URN=URN:NBN:SI:doc-VBGAQPV3 (visited on 02/27/2019) (Cited on p. 46).

[93] Ana Minaburo et al. *LPWAN Static Context Header Compression (SCHC) and fragmentation for IPv6 and UDP*. Internet-Draft draft-ietf-lpwan-ipv6-static-context-hc-16. IETF Secretariat, June 2018. URL: http://www.ietf.org/internet-drafts/draft-ietf-lpwan-ipv6-static-context-hc-16.txt (Cited on p. 8).

[94] Yasuhiro Monden. *Toyota Production System. An Integrated Approach to Just-In-Time*. 4th ed. A Productivity Press book. Taylor & Francis, Oct. 2011. ISBN: 9781439820971. DOI: 10.1201/b11731 (Cited on p. 43).

[95] G. Montenegro et al. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*. RFC 4944. http://www.rfc-editor.org/rfc/rfc4944.txt. RFC Editor, Sept. 2007. URL: http://www.rfc-editor.org/rfc/rfc4944.txt (Cited on p. 6).

[96] John Nagle. *On Packet Switches With Infinite Storage*. RFC 970. RFC Editor, Dec. 1985. URL: https://tools.ietf.org/html/rfc970 (visited on 02/27/2019) (Cited on p. 44).

[97] Nanoscale Labs. *Communications system*. Ed. by Francois Sforza. EP2278724. filed 2009-07-02, published 2011-01-26. Nanoscale Labs, 2011 (Cited on p. 7).

[98] Roger B. Nelsen. *An Introduction to Copulas*. Springer Series in Statistics. Springer-Verlag New York, 2006. 272 pp. ISBN: 0387286594. DOI: 10.1007/0-387-28678-0 (Cited on p. 49).

[99] Norbert Niebert et al. *Ambient Networks*. John Wiley & Sons, Ltd, Apr. 2007. DOI: 10.1002/9780470511046 (Cited on pp. 11, 13, 21).

[100] Jimmy J. Nielsen, Rongkuan Liu, and Petar Popovski. “Optimized Interface Diversity for Ultra-Reliable Low Latency Communication (URLLC)”. In: *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*. IEEE, Dec. 2017. DOI: 10.1109/glocom.2017.8254053 (Cited on p. 22).

[101] Jimmy J. Nielsen and Petar Popovski. “Latency analysis of systems with multiple interfaces for ultra-reliable M2M communication”. In: *2016 IEEE 17th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE, July 2016. DOI: 10.1109/spawc.2016.7536857 (Cited on p. 22).

[102] Jimmy Jessen Nielsen, Rongkuan Liu, and Petar Popovski. “Ultra-Reliable Low Latency Communication Using Interface Diversity”. In: *IEEE Transactions on Communications* 66.3 (Mar. 2018), pp. 1322–1334. DOI: 10.1109/tcomm.2017.2771478 (Cited on p. 22).

[103] J. Nieminen et al. *IPv6 over BLUETOOTH(R) Low Energy*. RFC 7668. RFC Editor, Oct. 2015 (Cited on p. 6).

[104] Nokia Networks. *LTE-M – Optimizing LTE for the Internet of Things*. Tech. rep. 2015 (Cited on p. 6).

[105] Petar Popovski et al. “Wireless Access for Ultra-Reliable Low-Latency Communication: Principles and Building Blocks”. In: *IEEE Network* 32.2 (Mar. 2018), pp. 16–23. DOI: 10.1109/mnet.2018.1700258 (Cited on p. 11).

[106] Michael E. Porter and James E. Heppelmann. “How Smart, Connected Products Are Transforming Competition”. In: *Harvard Business Review* 92.11 (Nov. 2014), pp. 64–88. ISSN: 0017-8012. URL: https://hbr.org/2014/11/how-smart-connected-products-are-transforming-competition (visited on 05/16/2019) (Cited on p. 3).

[107] Giovanni Puccetti and Ludger Rüschendorf. “Sharp Bounds for Sums of Dependent Risks”. In: *Journal of Applied Probability* 50.1 (2013), pp. 42–53. ISSN: 1475-6072. DOI: 10.1239/jap/1363784423 (Cited on p. 49).

[108] Qualcomm Technologies, Inc. *Leading the world to 5G*. Feb. 2016. URL: https://www.qualcomm.com/media/documents/files/qualcomm-5g-vision-presentation.pdf (visited on 05/14/2019) (Cited on p. 9).

[109] A. G. Ramonet and T. Noguchi. “IEEE 802.15.4 Historical Evolution and Trends”. In: *2019 21st International Conference on Advanced Communication Technology (ICACT)*. Feb. 2019, pp. 351–359. DOI: 10.23919/ICACT.2019.8702040 (Cited on pp. 4, 5).

[110] P. P. Ray and S. Agarwal. “Bluetooth 5 and Internet of Things: Potential and architecture”. In: *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*. Oct. 2016, pp. 1461–1465. DOI: 10.1109/SCOPES.2016.7955682 (Cited on p. 4).

[111] Cristian Roman, Peter Ball, and Shumao Ou. “Multiple Interface Scheduling System for HeterogeneousWireless Vehicular Networks: Description and Evaluation”. In: *EAI Endorsed Transactions on Wireless Spectrum* 3.11 (Jan. 2017). DOI: 10.4108/eai.9-1-2017.152097 (Cited on p. 45).

[112] B. Schweizer and A. Sklar. *Probabilistic metric spaces*. North Holland series in probability and applied mathematics. North Holland, 1983. ISBN: 9780444006660 (Cited on p. 49).

[113] Semtech Corp. *Low power long range transmitter*. Ed. by Craig A. Hornbuckle. EP2153522. filed 2008-05-16, published 2010-02-17. Semtech Corp, 2008 (Cited on p. 7).

[114] Semtech Corp. *Low power long range transmitter*. Ed. by Olivier Bernard André Seller and Nicolas Sornin. EP2763321. filed 2013-02-05, published 2014-08-06. Semtech Corp, 2013 (Cited on p. 7).

[115] Semtech Corporation. *LoRa™Modulation Basics*. Tech. rep. AN1200.22. Version 2. 2015. URL: https://www.semtech.com/uploads/documents/an1200.22.pdf (visited on 05/17/2019) (Cited on p. 7).

[116] Nihar B. Shah, Kangwook Lee, and Kannan Ramchandran. “When do redundant requests reduce latency ?” In: *51st Annual Allerton Conference on Communication, Control, and Computing, Allerton 2013, Allerton Park & Retreat Center, Monticello, IL, USA, October 2-4, 2013*. 2013, pp. 731–738. DOI: 10.1109/Allerton.2013.6736597 (Cited on p. 24).

[117] Nihar B. Shah, Kangwook Lee, and Kannan Ramchandran. "When Do Redundant Requests Reduce Latency?" In: *IEEE Transactions on Communications* 64.2 (Feb. 2016), pp. 715–722. DOI: 10.1109/tcomm.2015.2506161 (Cited on p. 24).

[118] Z. Shelby et al. *Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)*. RFC 6775. RFC Editor, Nov. 2012 (Cited on p. 6).

[119] SigFox. *Sigfox Technical Overview*. Tech. rep. 2017. URL: https://storage.sbg1.cloud.ovh.net/v1/AUTH_669d7dfced0b44518cb186841d7cbd75/dev_medias/build_technicalOverview.pdf (visited on 05/18/2019) (Cited on p. 7).

[120] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability 26. London: Chapman & Hall CRC, 1986. ISBN: 9780412246203 (Cited on p. 56).

[121] J. Song et al. "WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control". In: *2008 IEEE Real-Time and Embedded Technology and Applications Symposium*. Apr. 2008, pp. 377–386. DOI: 10.1109/RTAS.2008.15 (Cited on p. 6).

[122] Thread Group. *Thread*. 2019. URL: https://www.threadgroup.org/ (visited on 05/17/2019) (Cited on p. 6).

[123] Edward R. Tufte. *Beautiful Evidence*. July 2006. ISBN: 0-961-39217-7 (Cited on p.).

[124] United States Central Intelligence Agency. *Europe*. 803380AI (G00772) 7-08. map. Retrieved from the Library of Congress. Washington, D.C., United States: Central Intelligence Agency, 2008. URL: https://lccn.loc.gov/2008620735 (visited on 05/14/2019) (Cited on p. 9).

[125] D. Van Den Berg et al. "Challenges in Haptic Communications Over the Tactile Internet". In: *IEEE Access* 5 (2017), pp. 23502–23518. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2017.2764181 (Cited on p. 9).

[126] David Vandevoorde, Nicolai M. Josuttis, and Douglas Gregor. *C++ Templates: The Complete Guide (2Nd Edition)*. 2nd ed. Addison-Wesley Professional, 2017. ISBN: 0321714121 (Cited on p. 54).

[127] Todd Veldhuizen. "Expression Templates". In: *C++ Report* 7 (1995), pp. 26–31 (Cited on p. 54).

[128] P. Viswanath, D. N. C. Tse, and V. Anantharam. "Asymptotically Optimal Water-filling in Vector Multiple-access Channels". In: *IEEE Trans. Inf. Theor.* 47.1 (Jan. 2001), pp. 241–267. ISSN: 0018-9448. DOI: 10.1109/18.904525 (Cited on p. 44).

[129] Ashish Vulimiri et al. "Low latency via redundancy". In: *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies - CoNEXT '13*. ACM Press, 2013. DOI: 10.1145/2535372.2535392 (Cited on p. 23).

[130] Ashish Vulimiri et al. “More is less. reducing latency via redundancy”. In: *Proceedings of the 11th ACM Workshop on Hot Topics in Networks - HotNets-XI*. ACM Press, 2012. DOI: 10.1145/2390231.2390234 (Cited on p. 10).

[131] Qin Wang, Mark Hempstead, and Woodward Yang. “A Realistic Power Consumption Model for Wireless Sensor Network Devices”. In: *Sensor and Ad Hoc Communications and Networks, 2006. SECON ’06. 2006 3rd Annual IEEE Communications Society on*. Vol. 1. Sept. 2006, pp. 286–295. DOI: 10.1109/SAHCN.2006.288433 (Cited on p. 6).

[132] Thomas Watteyne, Ankur Mehta, and Kris Pister. “Reliability Through Frequency Diversity: Why Channel Hopping Makes Sense”. In: *Proceedings of the 6th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*. PE-WASUN ’09. Tenerife, Canary Islands, Spain: ACM, 2009, pp. 116–123. ISBN: 978-1-60558-618-2. DOI: 10.1145/1641876.1641898 (Cited on p. 7).

[133] Mark Weiser. “The Computer for the 21st Century”. In: *Scientific American* 265 (Sept. 1991), pp. 94–104. DOI: 10.1038/scientificamerican0991-94 (Cited on p. 9).

[134] Eric W. Weisstein. *Inclusion-Exclusion Principle. From MathWorld–A Wolfram Web Resource*. Visited on 13/04/12. URL: http://mathworld.wolfram.com/Inclusion-ExclusionPrinciple.html (visited on 03/07/2019) (Cited on p. 51).

[135] Herbert George Wells. *Mind at the End of Its Tether*. W. Heinemann Limited, 1945 (Cited on p. xvii).

[136] Ward Whitt. “Approximations for the GI/G/m queue”. In: *Production and Operations Management* 2.2 (Jan. 1993), pp. 114–161. DOI: 10.1111/j.1937-5956.1993.tb00094.x (Cited on p. 20).

[137] D. Yang, Y. Xu, and M. Gidlund. “Coexistence of IEEE802.15.4 based networks: A survey”. In: *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*. Nov. 2010, pp. 2107–2113. DOI: 10.1109/IECON.2010.5675277 (Cited on p. 4).

[138] Yantian Hou, Ming Li, and Shucheng Yu. “Surviving the RF smog: Making Body Area Networks robust to cross-technology interference”. In: *2013 IEEE International Conference on Sensing, Communications and Networking (SECON)*. June 2013, pp. 353–361. DOI: 10.1109/SAHCN.2013.6645005 (Cited on p. 4).

[139] Kok-Kiong Yap et al. “Scheduling Packets over Multiple Interfaces While Respecting User Preferences”. In: *Proceedings of the Ninth ACM Conference on Emerging Networking Experiments and Technologies*. CoNEXT ’13. Santa Barbara, California, USA: ACM, 2013, pp. 109–120. ISBN: 978-1-4503-2101-3. DOI: 10.1145/2535372.2535387 (Cited on p. 44).

[140] Ali Yaver and Georgios P. Koudouridis. “Performance Evaluation of Multi-Radio Transmission Diversity: QoS Support for Delay Sensitive Services”. In: *VTC Spring 2009 - IEEE 69th Vehicular Technology Conference*. IEEE, Apr. 2009. DOI: 10.1109/vetecs.2009.5073791 (Cited on pp. 21, 22).

[141] S. Yeh et al. “QoS Aware Scheduling and Cross-Radio Coordination in Multi-Radio Heterogeneous Networks”. In: *2013 IEEE 78th Vehicular Technology Conference (VTC Fall)*. Sept. 2013, pp. 1–6. DOI: 10.1109/VTCFall.2013.6692384 (Cited on p. 43).

[142] Hongwei Yu et al. “Delay optimal concurrent transmissions in multi-radio access networks”. In: *2015 IEEE International Conference on Communications (ICC)*. IEEE, June 2015. DOI: 10.1109/icc.2015.7248883 (Cited on p. 21).

[143] ZigBee Alliance. *Dotdot*. 2017. URL: https://www.zigbee.org/zigbee-for-developers/dotdot/ (visited on 05/17/2019) (Cited on p. 6).

[144] Juan Zuniga, Carles Gomez, and Laurent Toutain. *SCHC over Sigfox LPWAN*. Internet-Draft draft-ietf-lpwan-schc-over-sigfox-00. IETF Secretariat, Apr. 2019. URL: http://www.ietf.org/internet-drafts/draft-ietf-lpwan-schc-over-sigfox-00.txt (Cited on p. 8).

APPENDICES

A

MISCELLANEOUS NOTES

A.1 DERIVATIONS

A.1.1 *Probability of Being the Minimum of Exponentially Distributed Variables*

Note: Referenced from section 4.5 on page 36.

Let $\{1, 2, \ldots, N\} = \mathcal{N}$ and $(X_i)_{i\in\mathcal{N}}$ be independent exponentially distributed random variables with intensities $(\lambda_i)_{i\in\mathcal{N}}$. Fix some $i, 1 \leq i \leq N$, for every $x \in \mathbb{R}$, the probability of the k^{th} r.v. being the minimum, such that $k \neq i$ ($k \in \mathcal{N} \setminus \{i\} = \mathcal{K}$), is

$$\begin{aligned}
\Pr\left\{\min_{k\in\mathcal{K}} X_k > x\right\} &= \Pr\left\{\bigcap_{k\in\mathcal{K}} \{X_k > x\}\right\} = \prod_{k\in\mathcal{K}} \exp\left(-\lambda_k x\right) \\
&= \exp\left(-\left[\sum_{k\in\mathcal{K}} \lambda_k - \lambda_i\right] x\right) = \exp\left(-\left[\lambda - \lambda_i\right] x\right).
\end{aligned}$$

Since X_i is independent of $\{X_k\}_{k\in\mathcal{K}}$,

$$\Pr\left\{\min_{k\in\mathcal{K}} X_k > X_i \mid X_i\right\} = \exp\left(-\left[\lambda - \lambda_i\right] X_i\right).$$

Taking the expectation of both sides with respect to the distribution of X_i, we obtain

$$\begin{aligned}
\Pr\left\{\min_{k\in\mathcal{K}} X_k > X_i\right\} &= \mathbb{E}_{X_i}\left\{\exp\left(-\left[\lambda - \lambda_i\right] X_i\right)\right\} \\
&= \int_0^\infty \exp\left(-\left[\lambda - \lambda_i\right] X_i\right) \lambda_i \exp\left(-\lambda_i x\right)\, \mathrm{d}x \\
&= \lambda_i \int_0^\infty \exp\left(-\lambda x\right)\, \mathrm{d}x,
\end{aligned}$$

which yields

$$\Pr\left\{i = \operatorname*{argmin}_{j\in\mathcal{N}} X_j\right\} = \frac{\lambda_i}{\lambda}. \qquad \blacksquare$$

A.1.2 Sum Distribution Upper Bound Derivation

Note: Referenced from section 6.1.3 on page 49.

Let X and Y be random variables with unknown dependence structure. Denote their cumulative distribution functions F_X and F_Y, respectively. Let Q_X^u be the u quantile of the distribution of X and let Q_Y^u be the v quantile of the distribution of Y, i.e.

$$F_X\left(Q_X^u\right) = \Pr\left(X \le Q_X^u\right) = u, \tag{A.1}$$

$$F_Y\left(Q_Y^v\right) = \Pr\left(Y \le Q_Y^v\right) = v. \tag{A.2}$$

Borrowing results from the theory of copulas, there is a copula

$$C_{XY}(u,v) = \Pr\left(X \le Q_X^u, Y \le Q_Y^v\right) \tag{A.3}$$

bounded by the Fréchet-Hoeffding[1] bounds

1. Named after the independent works of Wassily Hoeffding [50, 49] and, later, Maurice René Fréchet [34, 33].

$$\max\{1-u+v, 0\} \le C_{XY}(u,v) \le \min\{u,v\}. \tag{A.4}$$

Define an auxiliary variable $w = 1-u+v$. By fixing w, we restrict v to $v = 1-u+w$ while $u \in [w,1]$. Reformulating the equation (A.4) in terms of u and w, we obtain

$$\max\{w, 0\} \le C_{XY}(u, 1-u+w) \le \min\{u, 1-u+w\}, \tag{A.5}$$

where

$$C_{XY}(u, 1-u+w) = \Pr\left(X \le Q_X^u, Y \le Q_Y^{1-u+w}\right). \tag{A.6}$$

That is, the probability that both $X \le Q_X^u$ and $Y \le Q_Y^{1-u+w}$ are satisfied is at least w. Therefore, the probability of $X+Y$ being less than the sum of these quantiles is also upper bounded by w. In other words, for any such w, u, we have that

$$Q_{X+Y}^w \le Q_X^u + Q_Y^{1-u+w} \tag{A.7}$$

because both the values of X and Y must individually be less than their respective quantiles appearing in the above expression. Furthermore, the bound may be tightened by minimizing the right-hand-side across all u conforming to the constraint set forth above – i.e.

$$Q_{X+Y}^w \le \inf_{u\in[w,1]}\left\{Q_X^u + Q_Y^{1-u+w}\right\}, \tag{A.8}$$

which concludes the derivation. ■

A.2 NUMERICAL EXAMPLES

A.2.1 Response Time Analysis

Note: Referenced from section 6.1.3 on page 49.

Substituting the values from the example above into equation (4.1), we obtain

$$T^{(1)} = T^{M/M/1}(10+5, 15+18)$$
$$= 0.0\overline{5} \approx 0.05556$$
$$T^{(2)} = T^{M/M/1}(15-10, 18) + \sum_{j:\{1\}} T_Q^{M/M/1}(10, (15+18)-5)$$
$$\approx 0.09676$$

As shown in the quesi configuration example[2], the results obtained when run with the stationarity threshold of 10^{-6} (achieved at $N_{\text{class 1}} \approx 14.0 \cdot 10^7$ and $N_{\text{class 2}} \approx 7.0 \cdot 10^7$) were

$$T^{(1)} \approx 0.05556 \qquad \text{RAE}\left\{T^{(1)}\right\} \approx 9.2 \cdot 10^{-5}$$
$$T^{(2)} \approx 0.09676 \qquad \text{RAE}\left\{T^{(2)}\right\} \approx 6.4 \cdot 10^{-5}.$$

Numerical simulation of an identical configuration is discussed in appendix B.2 on page A18.

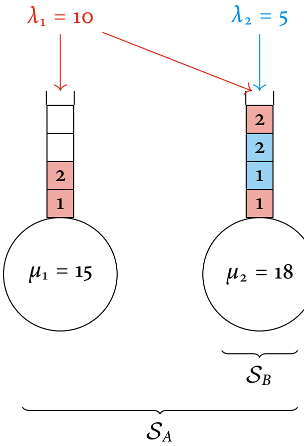


Figure A.1: Response time analysis example with values.

2. See appendix B.2 on page A18.

A.2.2 *Probability of Processor Selection*

Note: Referenced from section 4.5 on page 36.

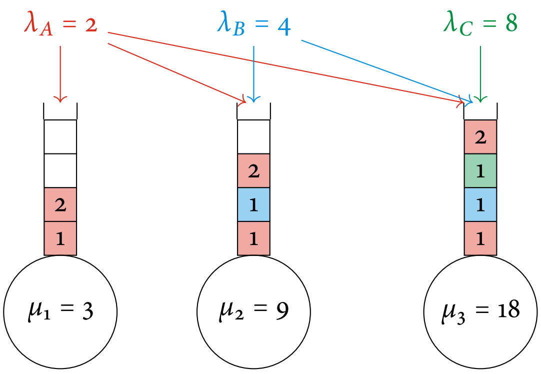


Figure A.2: Probability of processor selection example with values

Since class-3 jobs are server exclusively by the third server, the solution to $P_{3\to s}$ is trivially

$$P_{3\to 1} = \frac{0}{18} = 0,$$
$$P_{3\to 2} = \frac{0}{18} = 0,$$
$$P_{3\to 3} = \frac{18}{18} = 1.$$

As class-3 jobs are served only by a single server, the solution for the class-2 jobs is effectively obtained by a simple comparison of service time r.v.s

$$P_{2\to1} = \frac{0}{9 + (18-8)} = 0,$$
$$P_{2\to2} = \frac{9}{9 + (18-8)} = \frac{9}{19} \approx 0.4737,$$
$$P_{2\to3} = \frac{18-8}{9 + (18-8)} = \frac{10}{19} \approx 0.5263.$$

Conversely, the solution for class-1 jobs must involve the previously computed probabilities for class-2 jobs as the "slowdown" due to processing of class-2 jobs is distributed across all servers in S_2.

$$\mu^{\mathrm{R}}_{1,1} = 3,$$
$$\mu^{\mathrm{R}}_{2,1} = 9 - 4\cdot\frac{9}{19} = \frac{135}{19} \approx 7.1053,$$
$$\mu^{\mathrm{R}}_{3,1} = 18 - 4\cdot\frac{10}{19} - 8 = \frac{150}{19} \approx 7.8947;$$
$$\sum_{t:t\in S_1} \mu^{\mathrm{R}}_{t,1} = 18;$$
$$P_{1\to1} = 3\cdot\frac{1}{18} = \frac{1}{6} = 0.1\overline{6},$$
$$P_{1\to2} = \frac{135}{19}\cdot\frac{1}{18} = \frac{15}{38} \approx 0.3947,$$
$$P_{1\to3} = \frac{150}{19}\cdot\frac{1}{18} = \frac{25}{57} \approx 0.4386.$$

The results can be verified using the `quesi` tool by counting jobs of each class i served at each server s, $N_{i\to s}$, i.e.

$$\widehat{P}_{i\to s} = \frac{N_{i\to s}}{\sum_{t:t\in S_i} N_{i\to t}} \qquad \widetilde{P}_{i\to s} = \left|\widehat{P}_{i\to s} - P_{i\to s}\right|$$

3. See appendix B.2 on page A18.

The following results were obtained[3] by running an equivalent configuration when run with the stationarity threshold of 10^{-5} (achieved at $N_{\text{class 1}} \approx 27.5\cdot10^6$, $N_{\text{class 2}} \approx 13.7\cdot10^6$ and $N_{\text{class 3}} \approx 6.9\cdot10^6$).

$P_{3\to1} = 0$	$\widehat{P}_{3\to1} = 0$	$\widetilde{P}_{3\to1} = 0$
$P_{3\to2} = 0$	$\widehat{P}_{3\to2} = 0$	$\widetilde{P}_{3\to2} = 0$
$P_{3\to3} = 1$	$\widehat{P}_{3\to3} = 1$	$\widetilde{P}_{3\to3} = 0$
$P_{2\to1} = 0$	$\widehat{P}_{2\to1} = 0$	$\widetilde{P}_{2\to1} = 0$
$P_{2\to2} \approx 0.4737$	$\widehat{P}_{2\to2} \approx 0.4737$	$\widetilde{P}_{2\to2} \approx 4.52\cdot10^{-5}$
$P_{2\to3} \approx 0.5263$	$\widehat{P}_{2\to3} \approx 0.5263$	$\widetilde{P}_{2\to3} \approx 4.52\cdot10^{-5}$
$P_{1\to1} \approx 0.1667$	$\widehat{P}_{1\to1} \approx 0.1666$	$\widetilde{P}_{1\to1} \approx 8.74\cdot10^{-5}$
$P_{1\to2} \approx 0.3947$	$\widehat{P}_{1\to2} \approx 0.3948$	$\widetilde{P}_{1\to2} \approx 4.42\cdot10^{-5}$
$P_{1\to3} \approx 0.4386$	$\widehat{P}_{1\to3} \approx 0.4386$	$\widetilde{P}_{1\to3} \approx 4.32\cdot10^{-5}$

B

QUEUEING SIMULATOR

B.1 IMPLEMENTATION

The simulator core library and the command line interface (CLI) are to author's best ability written in modern C++ (incl. features from the ISO/IEC N4778 draft, aka C++20). The library is packaged as a CMake-compatible library target (package) while the CLI application is provided through a CPack installer.

Throughout the development process, several development tools and libraries which are not part of the final program were used. These include:

- *Clang* front-end and tooling infrastructure for the C language family
- *Clion*, a cross-platform IDE for C and C++ by JetBrains
- *Google Benchmark*, a micro-benchmarking tool
- *Valgrind*, an instrumentation framework and dynamic analysis tools (*Callgrind*, *Memcheck* and *Massif* were most commonly used)
- *Catch2*, a unit-test framework

In addition to the Standard library the codebase depends on the following libraries[1]:

- *Args* by Taylor C. Richberger used for parsing command line arguments;
- *RapidJSON* by THL A29 Limited and Milo Yip used for parsing JSON documents;
- *Prio Queue* by Björn Fahller, providing an efficient B-heap implementation.

Although optional, the author suggests linking against the *Thread-Caching Malloc*, *TCMalloc*, developed at Google, for additional performance gain and memory fragmentation reduction. Given the unconventional memory usage patterns, other `malloc` implementation such as *jemalloc* and *Hoard* were not able to provide significant performance benefits over the current *GNU C Library* implementation and are thus not advised (although supported).

1. Licenses to respective libraries and their dependencies can be displayed by passing `quesi_app -license` and will be available in `LICENSE` file once the source is made public.

The simulator CLI application (quesi_app) is designed to run a particular system configuration in a single thread. *GNU Parallel* (or similar tool) is advised to spawn and manage multiple running instances and pass over system configurations / substitutions (see below).

B.1.1 *Modus operandi*

Initialization

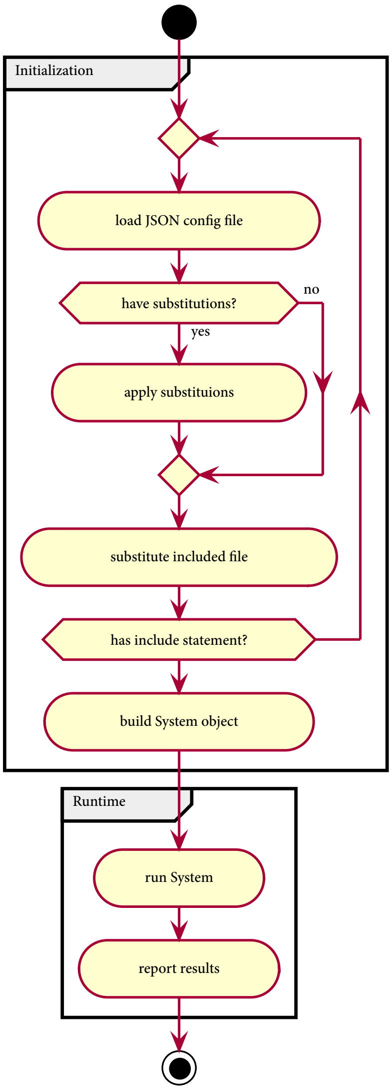


Figure B.1: Activity diagram of the quesi_app.

The quesi_app provides an interface to the quesi library by enabling users to configure the simulated system using JSON files as shown at the end of this section.

Once the user passes path to the configuration file, the program parses and validates its contents. If a substitution token (the syntax is SUB_<TYPE>_<ID>, e.g. SUB_DOUBLE_1) is found, it is replaced by the IDth substitution argument passed to the program command line. Next, if a special key include is found, its value (each of values if it is an array) is taken as a path to another JSON configuration file to be merged with the current JSON document and the process is repeated for the included file. Finally, the resulting JSON documents are sequentially merged.

The JSON configuration document is passed to the *SystemBuilder* instance which constructs and configures the *System* instance (model) and all auxiliary blocks. Descriptive error messages are displayed if an error occurs at any stage of the configuration.

Runtime

The simulation is initiated by the System::run() call. Once finished, the simulation results may be obtained by evaluating the System::report() method which, in turn, calls the report() methods of all user-defined *Reporter* instances.

The *modus operandi* of the quesi runtime can be summarized as follows: a *Source* instance constructs the *Job* object and passes it over to and through a sequence of subsequent blocks representing the *Queue*, an algorithm dispatching (*Dispatcher*) the jobs from the queue to the *Processor*. Throughout its lifetime, the Job reports significant events and statistics to the assigned *Tracker* instance. The flow of time is governed by a global *Scheduler* instance. The fig. B.2 on page A7 provides a simplified view of this process.

1. The Source schedules job construction after delay obtained from the associated *Schedule* (e.g. exponential distribution object in case of M/M/1 queue). From the implementation perspective, a callback structure and a time duration is passed to the Scheduler which, in turn, wraps these as a *Task* stores it in its internal queue.

2. After the time reaches the time point associated with the request 1, the Scheduler calls the associated callback.

3. The Source internally constructs a *Work* object and, if provided, associates it with a Tracker. Subsequently the Source creates a replica of the Work, aka. *Job*.

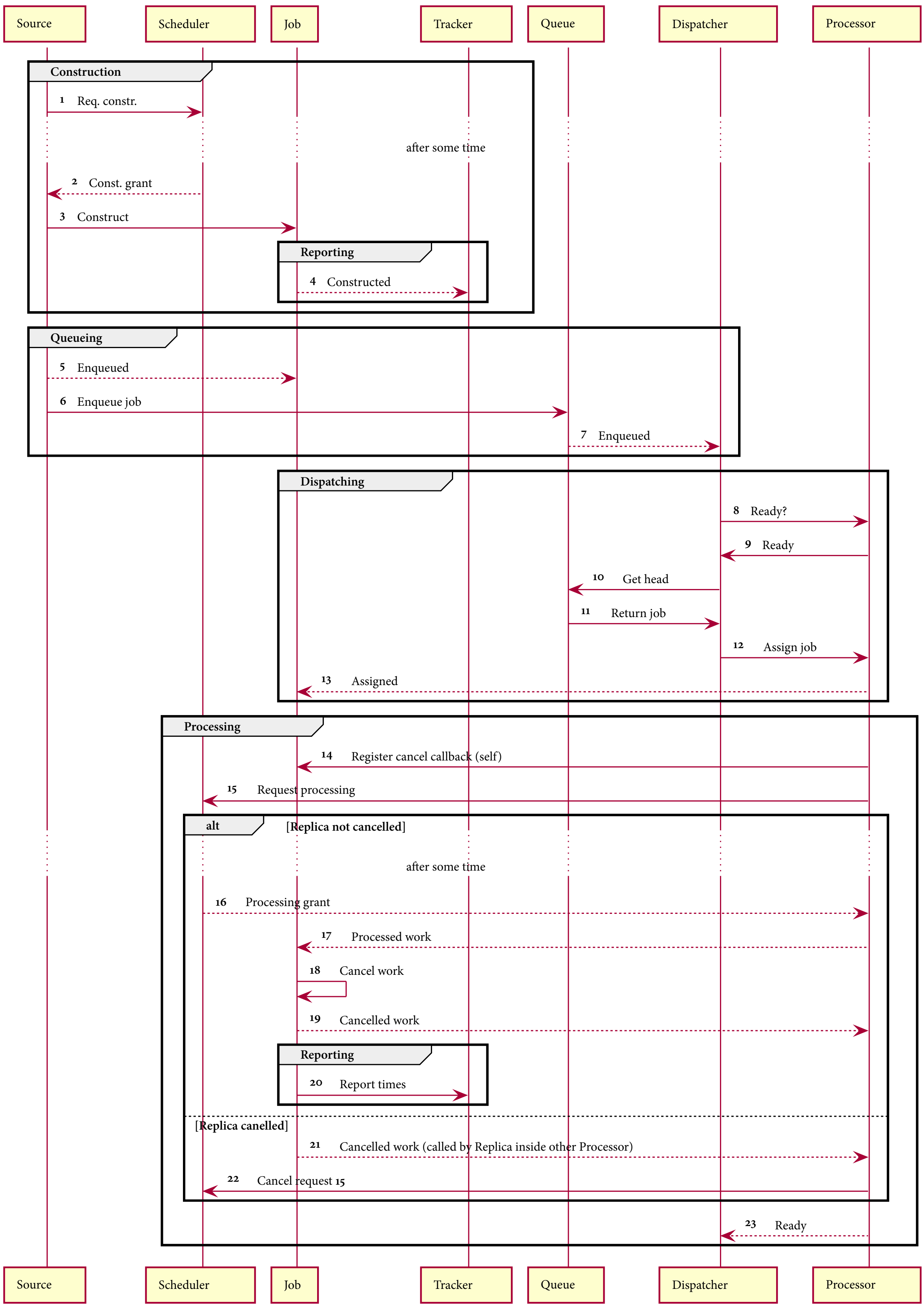


Figure B.2: Simplified job flow through the `quesi` blocks.

4 Upon construction, the Job reports to the associated Tracker. From the implementation perspective, the Job itself does not hold reference to the Tracker but rather delegates the reporting to the associated Work instance.

5 The Source notifies the Job that it is being pushed to a Queue. The Job stores current simulation time. Note that the notification happens in the same time point as the following step.

6 The Source replicates the Job across all associated Queues.

7 The Queue notifies all associated Dispatchers.

8 The Dispatched queries associated Processor to find one ready for new assignment.

9 If a server is ready, it return positive response (as in the example in question).

10 Dispatcher requests head of the Queue from the associated Queue instance.

11 The Job at the head of the Queue is returned.

12 The Job is forwarded to the Processor found in steps **8** and **9**.

13 The Processor notifies the Job that it has been assigned. The Job stores current simulation time.

14 The Processor registers itself as a callback handler for the event of Job cancellation, i.e. the assigned Job will notify the Processor if it gets cancelled.

15 The Processor schedules Job processing after delay obtained from the associated Schedule (see step **1**).

a) If the assigned Job does *not* get cancelled...

   16 After the time reaches the time point associated with the request **15**, the Scheduler calls the associated callback.

   17 The Processor notifies the Job that it has been processed. The Job stores current simulation time.

   18 The Job cancels the associated Work instance which effectively cancels all Jobs derived from it.

   19 Cancellation callbacks are fired for all replicas of the cancelled Work which are in service.

   20 The finished Job reports the saved timestamps to the Tracker. Note that other replicas will follow the alternative flow denoted as "Replica cancelled" and thus will not report.

b) If the assigned Job does get cancelled...

   21 Cancellation callbacks are fired for all replicas of the cancelled Work which are in service.

22 The Processor cancels the processing request **15**.

23 The Processor notifies the Dispatcher that it is ready for new assignments.

B.1.2 *API overview*

Since this section provides solely a overview of the API of the library. Readers not wishing to use the library and/or not interested in its inner workings may skip it. More detailed documentation is provided in respective header files.

The class diagram is broken up into five logical groups, each discussed in separate sub-section and visualized as UML 2.0 class diagrams. For the sake of brevity, various utility portions of the API (e.g. manipulation of the JSON documents, error handling code) are omitted.

Contracts group

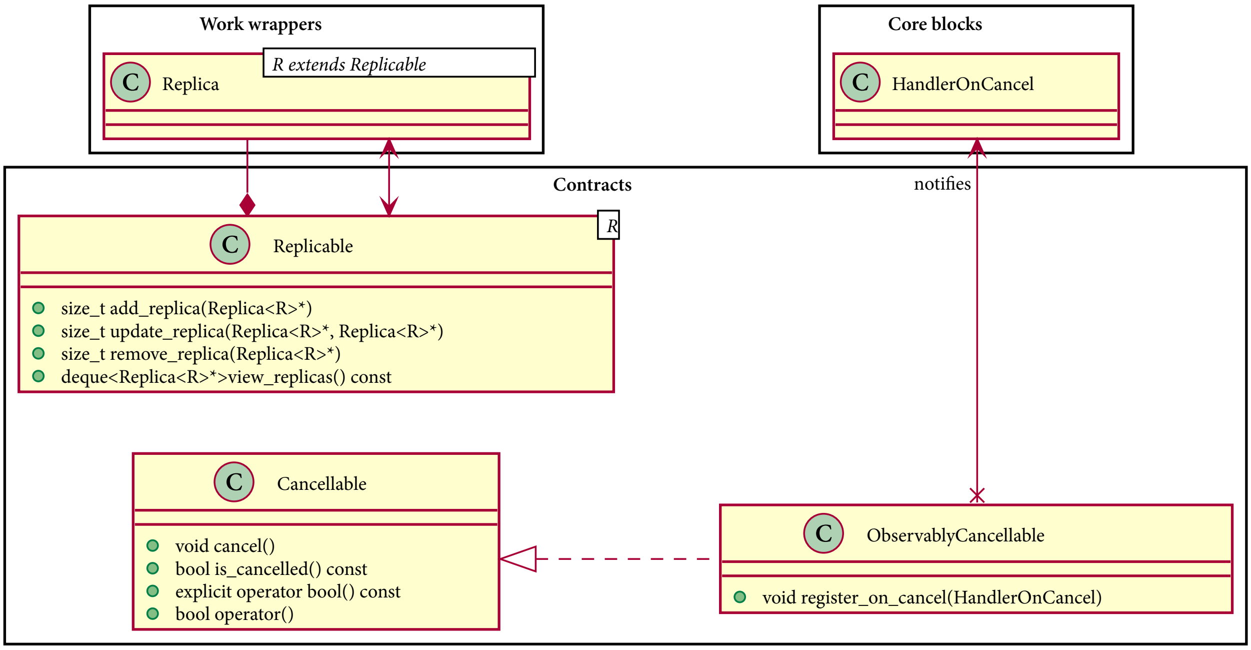


Figure B.3: Class diagram of the Work wrappers group

The contracts groups represents abstract interfaces describing some distinct functionality of a subclass inheriting from those.

CANCELLABLE is an object that may be cancelled. Cancellable exposes its current state via conversion operators and a getter method. Cancellable is effectively a wrapper for a boolean value.

OBSERVABLYCANCELLABLE is a Cancellable that a compatible handler class may subscribe to and be notified when the cancellation event occurs. ObservablyCancellable holds a reference to the associated HandlerOnCancel.

REPLICABLE<R> refers to a concept of a class which may be replicated. Replicable holds a list of references to the respective replicas and provides methods for its management.

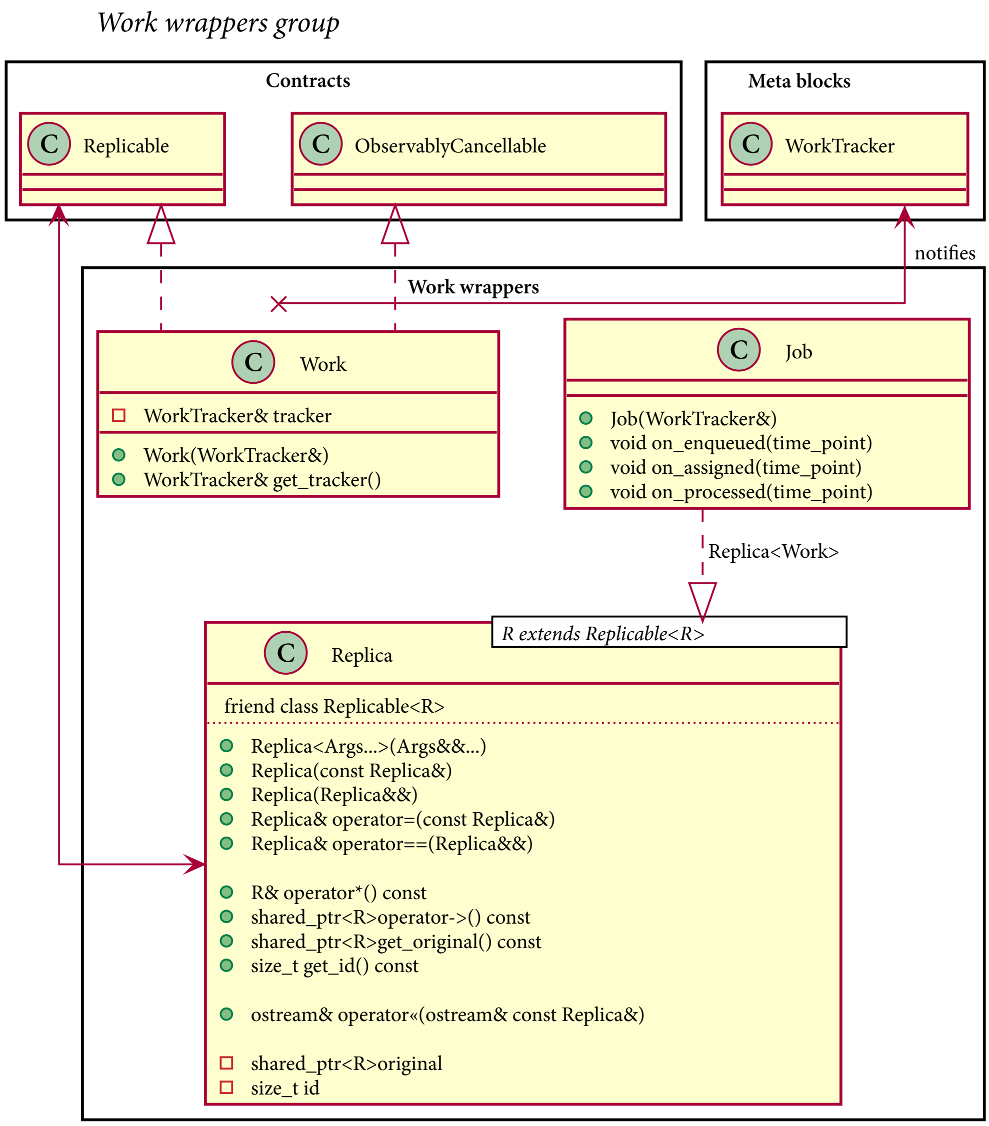


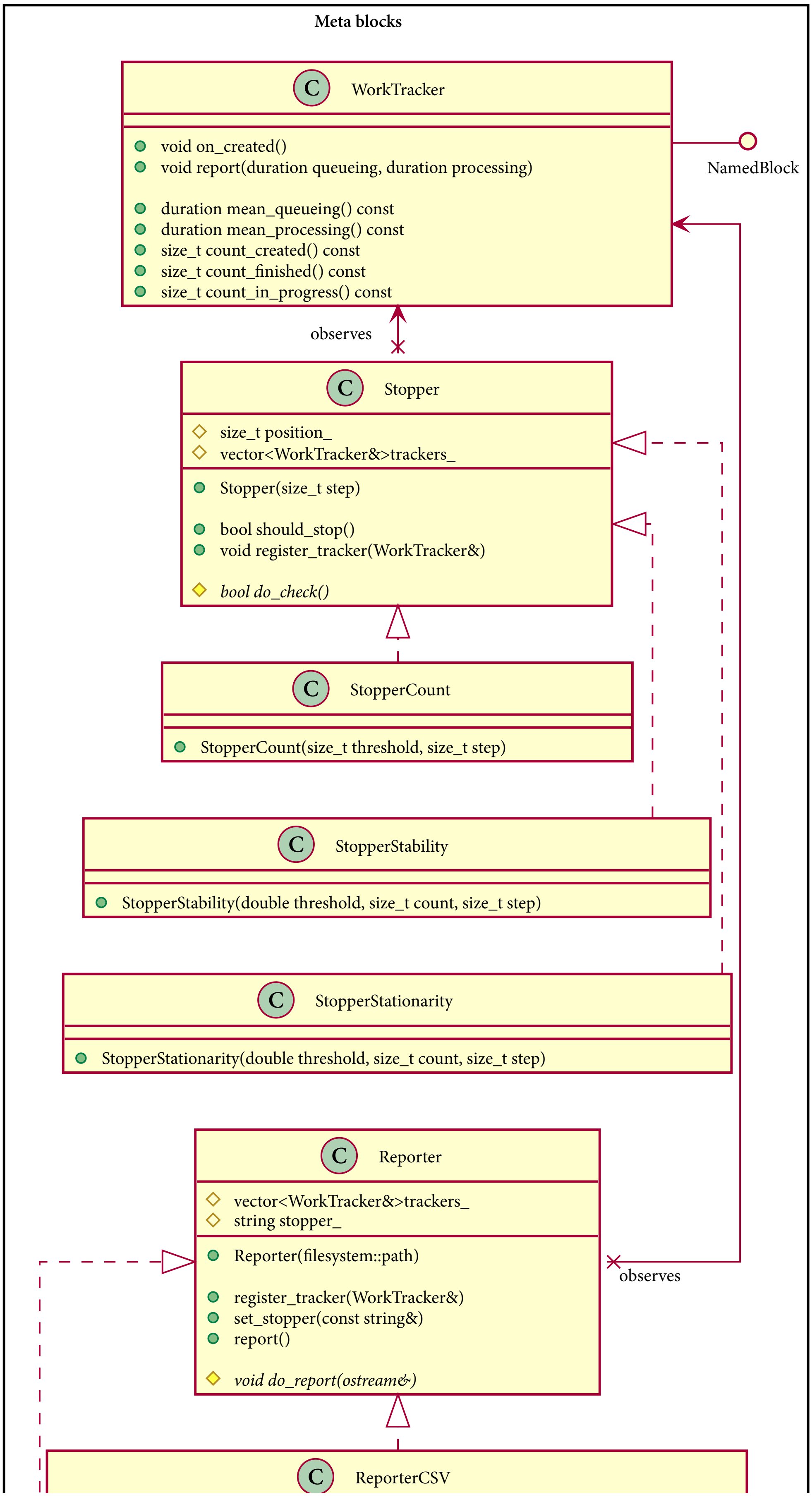
Figure B.4: Class diagram of the Work wrappers group

The Work wrappers model the jobs passing through the queueing system.

Replica wraps a Replicable. Its constructors and its destructor abstract away the replica list management operations of the underlying Replicable – e.g. copying a Replica seamlessly adds a reference to the the newly constructed Replica to the list. Finally, the reference to the underlying Replicable is exposed enabling navigability between the two.

Work models the unique job entering the system. A reference to the underlying Work is held by its replicas. Reference to a WorkTracker enables is intended to be used for reporting as shown in the example above.

Job is a Replica of a Work. Job objects are passed through the simulator and notified using the `on_<event>()` methods about their current owner. Since Replicable and Replica are bidirectionally navigable, the same applies to Job and Work.


Meta blocks
C WorkTracker
void on_created()
void report(duration queueing, duration processing)
duration mean_queueing() const
duration mean_processing() const
size_t count_created() const
size_t count_finished() const
size_t count_in_progress() const
NamedBlock
observes
C Stopper
size_t position_
vector<WorkTracker&>trackers_
Stopper(size_t step)
bool should_stop()
void register_tracker(WorkTracker&)
bool do_check()
C StopperCount
StopperCount(size_t threshold, size_t step)
C StopperStability
StopperStability(double threshold, size_t count, size_t step)
C StopperStationarity
StopperStationarity(double threshold, size_t count, size_t step)
C Reporter
vector<WorkTracker&>trackers_
string stopper_
Reporter(filesystem::path)
register_tracker(WorkTracker&)
set_stopper(const string&)
report()
void do_report(ostream&)
observes
C ReporterCSV

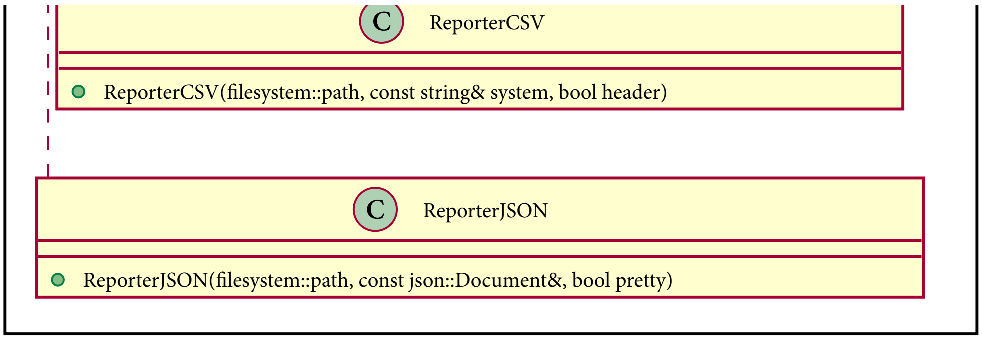


Figure B.5: Class diagram of the Work wrappers group

Meta blocks group

Meta blocks do not model any part of the queueing system but rather observe its runtime statistics.

WorkTracker receives notifications from Work instances upon construction and completion and aggregates these data to provide runtime statistics (mean queueing and processing times and job counts corresponding to distinct stages of the system).

Stopper defines stopping criteria of the simulation. Runtime statistics are observed through a WorkTracker every `step`th simulation time step (the default value[2] is 10^4). Three basic Stoppers are defined:

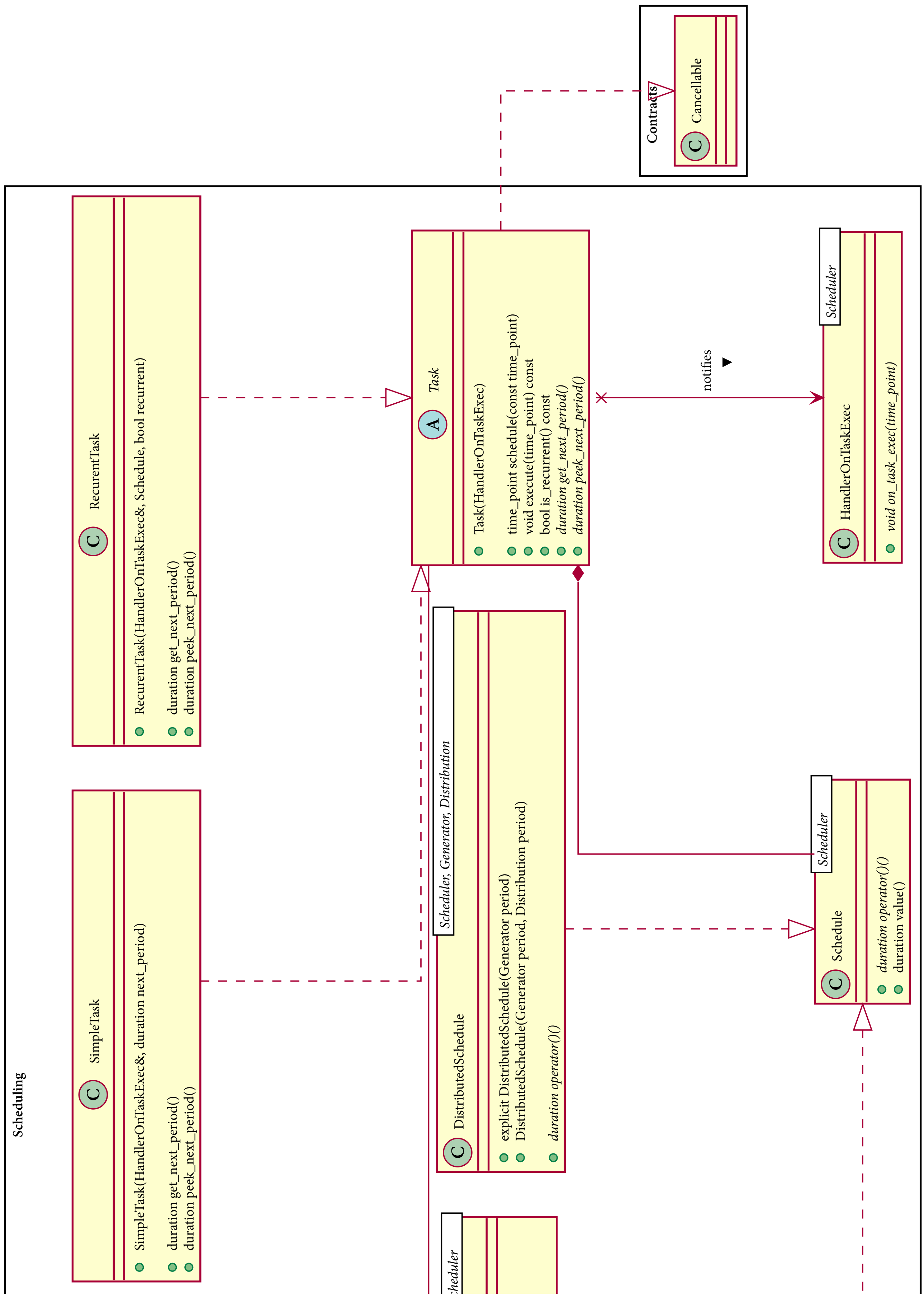
2. At the time of writing, the default values apply only to the `quesi_app`.

- StopperCount stops the simulation if the number of Jobs created exceeds a defined `threshold`.
- StopperStability stops the simulation if the average queueing time increases by a factor greater or equal to a defined `threshold` (the default valuefnote:defaults. is 1.25), `count` number of times in a row (the default valuefnote:defaults. is 5).
- StopperStationarity stops the simulation if the average queueing time changes by a factor lower or equal to a defined `threshold`, `count` number of times in a row (the default valuefnote:defaults. is 5). For stable systems, the `threshold` effectively determines the precision of the results.

Reporter defines formatting of the values stored in a WorkTracker for output to the user. Two basic Reporters are defined:

- ReporterCSV prints the output in a CSV format.
- ReporterJSON adds the values to the system configuration JSON document and the resulting structure is outputted. If the `pretty` flag is set to `false` [`true`], prints the output in compressed [pretty-printed] JSON.

If no or an empty path to the output file is provided, the output is sent to the standard output.



Scheduling
Contracts
Cancellable
RecurentTask
RecurentTask(HandlerOnTaskExec&, Schedule, bool recurrent)
duration get_next_period()
duration peek_next_period()
SimpleTask
SimpleTask(HandlerOnTaskExec&, duration next_period)
duration get_next_period()
duration peek_next_period()
Task
Task(HandlerOnTaskExec)
time_point schedule(const time_point)
void execute(time_point) const
bool is_recurrent() const
duration get_next_period()
duration peek_next_period()
notifies
Scheduler
HandlerOnTaskExec
void on_task_exec(time_point)
Scheduler, Generator, Distribution
DistributedSchedule
explicit DistributedSchedule(Generator period)
DistributedSchedule(Generator period, Distribution period)
duration operator()()
Scheduler
Schedule
duration operator()()
duration value()

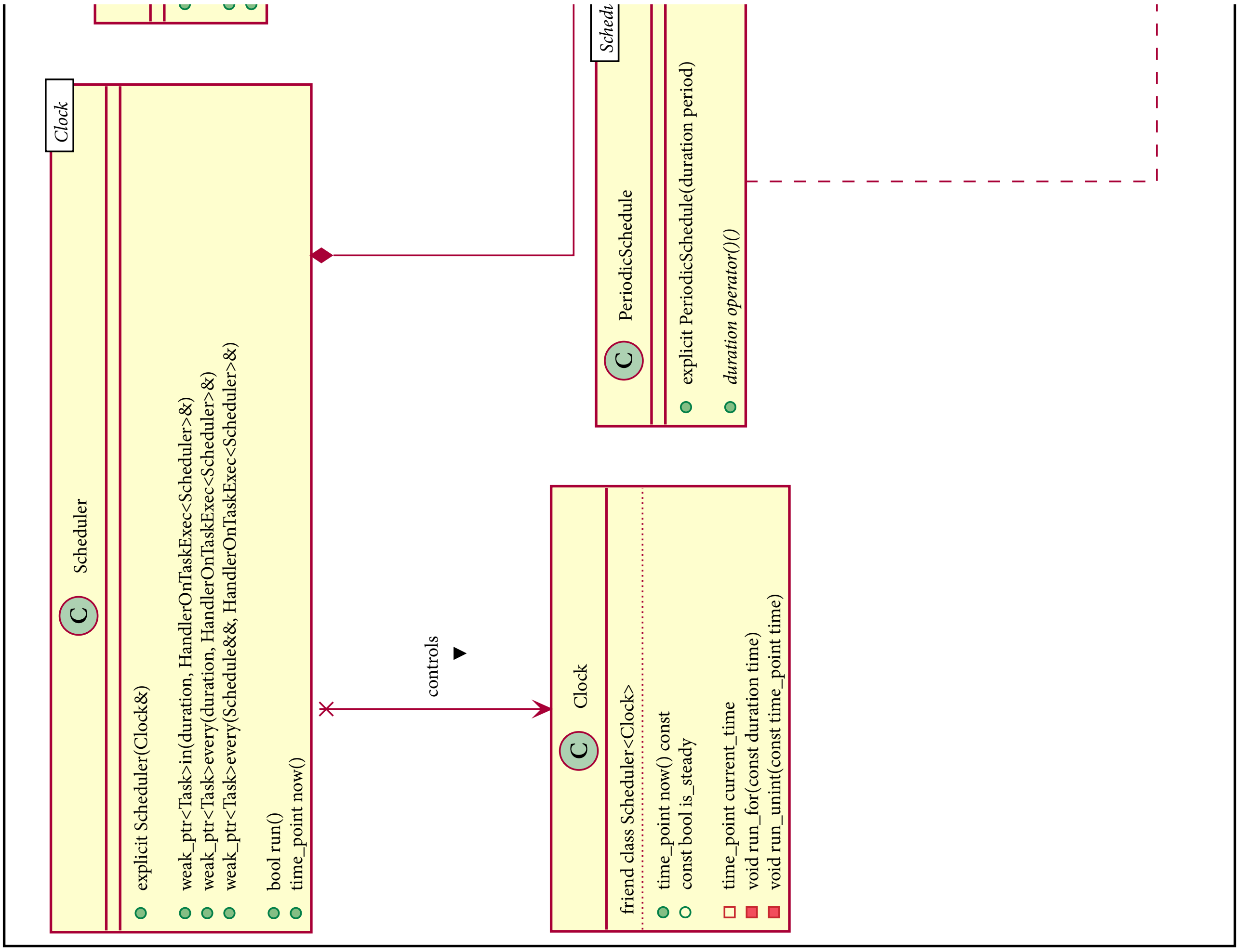


Figure B.6: Class diagram of the Scheduling group

Scheduling group

Members of the Scheduling group handle the time-related aspects of the simulation.

CLOCK stores the simulation time. Its modifiers are accessible exclusively by a Scheduler.

SCHEDULE is an abstraction of a function that returning time until a task should be (re-)scheduled. Two Schedules are defined:

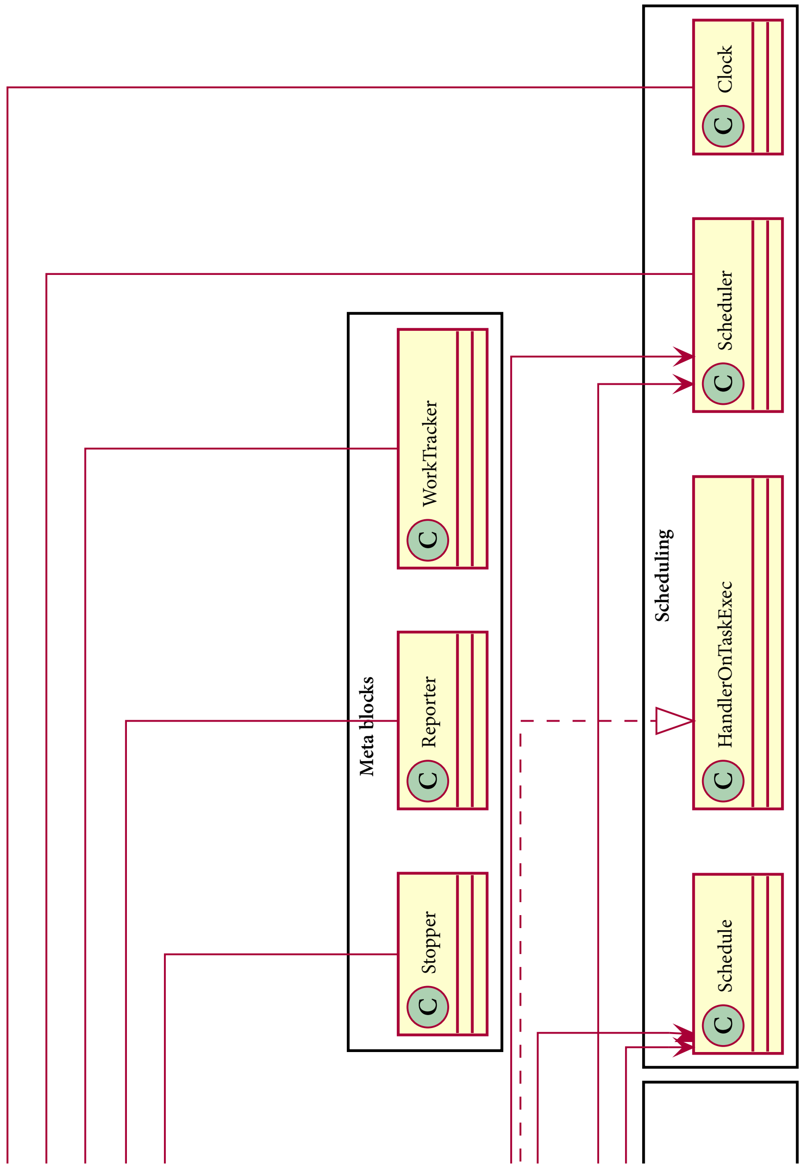
- PeriodicSchedule returns a constant value (duration).
- DistributedSchedule returns a value drawn from a given probability distribution object. Entropy is obtained from a provided uniform generator instance.

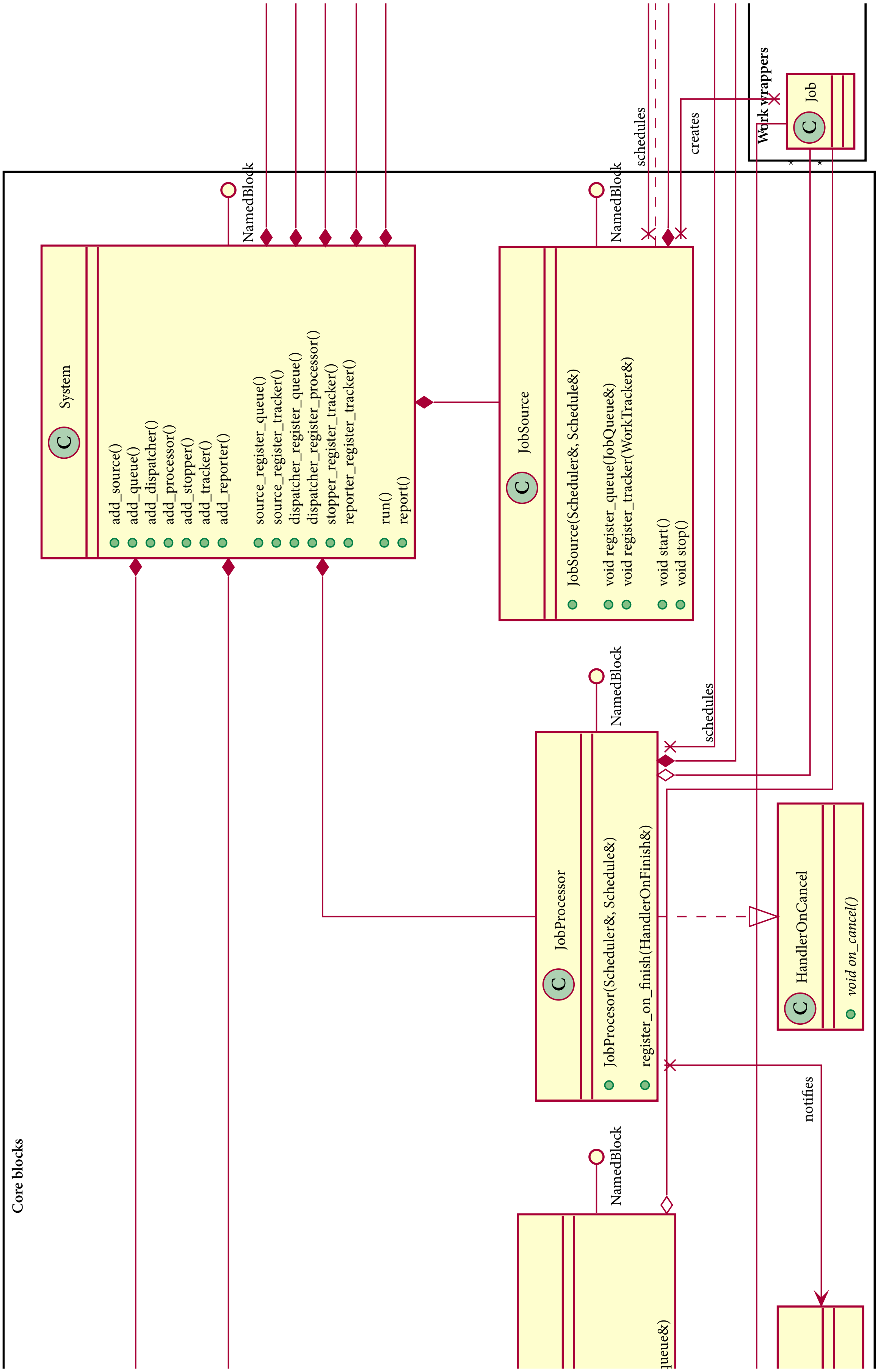
Task is an wrapper of a callback and a Schedule. If provided, the associated HandlerOnTaskExec is notified upon execution. Two Tasks are defined:

- SimpleTask is discarded after execution.
- RecurrentTask is rescheduled after execution according to its Schedule.

HandlerOnTaskExec is notified by an associated Task upon execution.

Scheduler maintains a priority queue (on top of a B-heap) of Tasks ordered by the scheduled time. A `run()` of a Scheduler may advance the simulation time if no task is scheduled to run at the current time. The Scheduler provides method for scheduling callbacks at a given time point, after a given delay or according to a Schedule. These methods return a weak reference to the Task in the queue, i.e. the reference object provides a methods to verify that the Task still exists and obtaining a proper reference if it does.
Tasks scheduled at infinity (or maximal time point representable by the time point type defined in Clock) or cancelled prior to execution are ignored.





Core blocks
Work wrappers
System
add_source()
add_queue()
add_dispatcher()
add_processor()
add_stopper()
add_tracker()
add_reporter()
source_register_queue()
source_register_tracker()
dispatcher_register_queue()
dispatcher_register_processor()
stopper_register_tracker()
reporter_register_tracker()
run()
report()
NamedBlock
JobSource
JobSource(Scheduler&, Schedule&)
void register_queue(JobQueue&)
void register_tracker(WorkTracker&)
void start()
void stop()
NamedBlock
schedules
creates
Job
JobProcessor
JobProcesor(Scheduler&, Schedule&)
register_on_finish(HandlerOnFinish&)
NamedBlock
schedules
HandlerOnCancel
void on_cancel()
notifies
NamedBlock

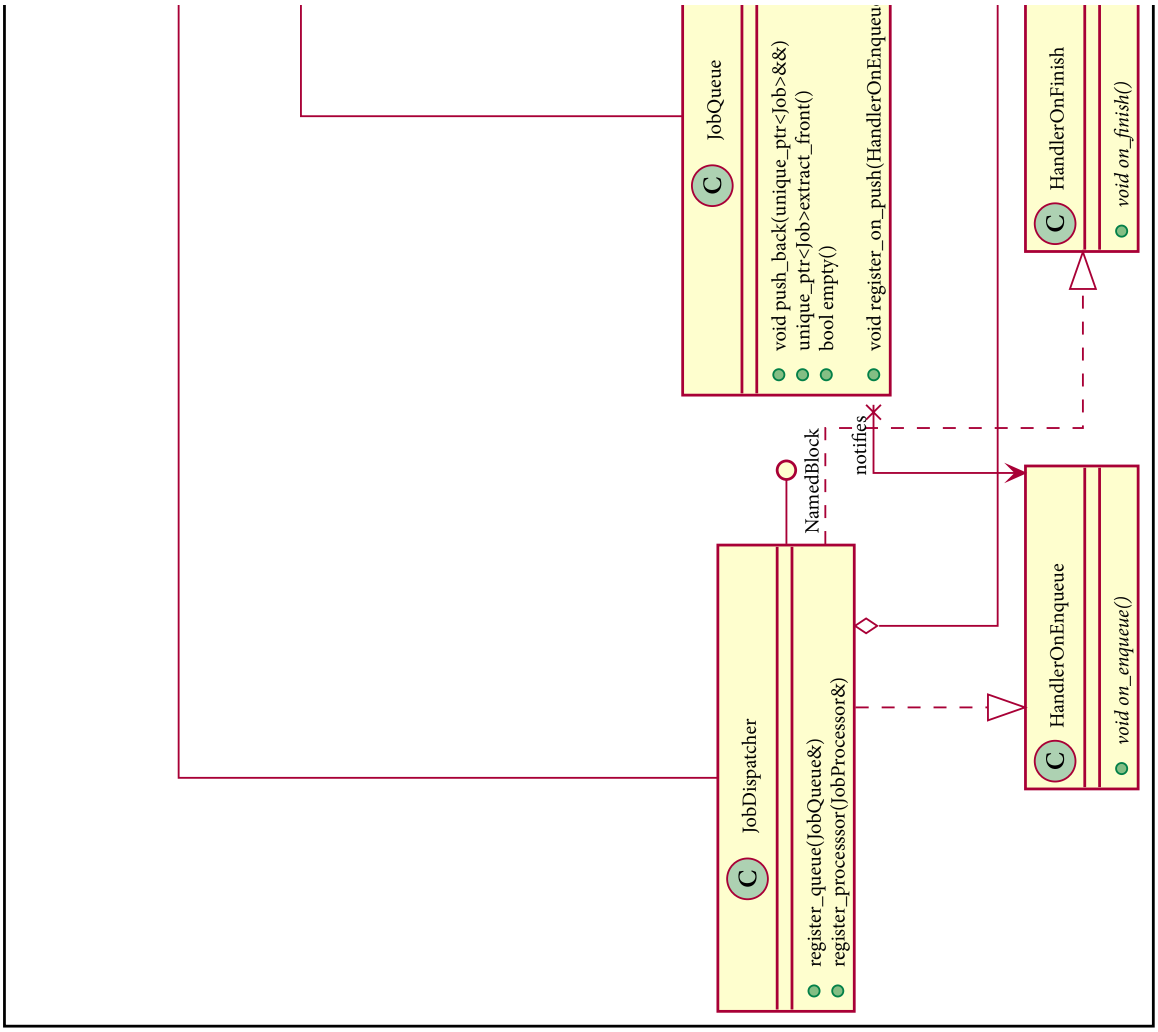


Figure B.7: Class diagram of the Core blocks group

Core blocks group

Meta blocks model concrete parts of the queueing system .

Systemowns all other core and meta blocks as well as a Scheduler and a Clock instance.
For purposes of configuration, methods for adding blocks and associating pairs of already added blocks are provided. Blocks are stored as key-value pairs where the keys represent unique block names.
The `run()` method steps the Scheduler in a loop until a stopping condition defined by any of Stoppers is satisfied. The `report()` method executes all Reporters.

HandlerOn<Event> is class receptive to notifications about <Event>.

JobSource constructs Jobs at times determined by an associated Schedule instance using a Scheduler reference.

JobQueue stores Jobs in a "first-in-first-out" queue. Upon insertion, an associated HandlerOnEnqueue is notified.
If a Job is cancelled while inside a JobQueue, subsequent calls to `extract_front()` and `empty()` behave as if it was never inserted.

JobProcessor processes Jobs at rate determined by an associated Schedule instance using a Scheduler reference. The associated HandlerOnFinish is notified when a Job is successfully served.

JobDispatcher is a HandlerOnEnqueue and HandlerOnFinish, i.e. it is notified when a job enters an associated JobQueue or leaves serviced at an associated JobProcessor. Based on these notification, the JobDispatched extracts Jobs from the JobQueue and assigns them to JobProcessors which are ready for assignment.

Utility group

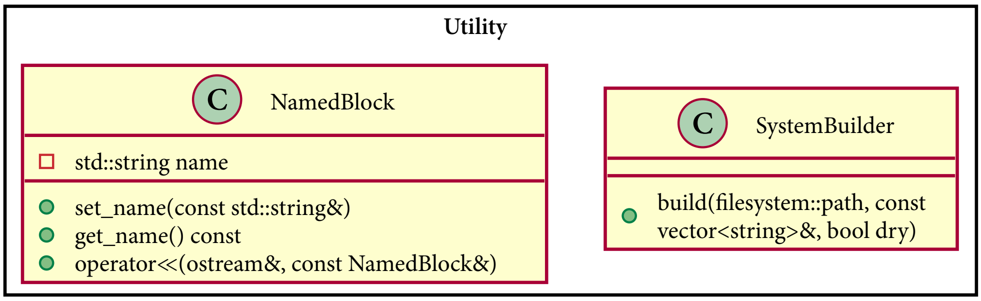


Figure B.8: Class diagram of the Utility group

Although many utility portions of the library were not mentioned in the commentary for the sake of brevity, two deserve a mention.

NamedBlock is a wrapper for a string intended to be interpreted as a name of a block. All blocks which can be added to a System are NamedBlocks and for purposes of configuration (and reporting) are referenced by their names.

SystemBuilder is a utility class which parses JSON configuration files and "builds" a System – i.e. configures all requested blocks as prescribed and populates a newly constructed System instance. If `dry` is set to `true`, resulting configuration structure is printed and the SystemBuilder returns.

B.2 Configuration Example

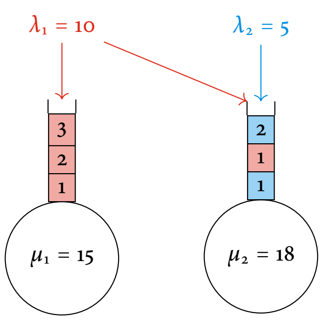


Figure B.9: Example queueing system.

The simulator is configured by a description of the simulated system passed along with other parameters as a JSON file. One possible configuration equivalent to the system shown in figure B.9 that showcases some of the more advanced features of the SystemBuilder's parser would be the following.

Main configuration file (`example_base.json`) declares and defines properties the user wishes parametrize using command line arguments and includes another file with other declarations and definitions, as follows.

```json
{
  "include": "example_include.json",
  "blocks": {
    "sources": [
      {
        "name": "input-1",
        "schedule": {
          "type": "exponential",
          "lambda": "SUB_DOUBLE_0"
        }
      },
      {
        "name": "input-2",
        "schedule": {
          "type": "exponential",
          "lambda": "SUB_DOUBLE_1"
        }
      }
    ],
    "processors": [
      {
        "name": "processor-1",
        "schedule": {
          "type": "exponential",
          "lambda": "SUB_UINT_2"
        }
      },
      {
        "name": "processor-2",
        "schedule": {
          "type": "exponential",
          "lambda": "SUB_UINT_3"
        }
      }
    ]
  }
}
```

Listing B.1: Main configuration file.

The included file `example_include.json` provides additional declarations and definitions which will not be modified by the command line arguments.

```json
{
  "blocks": {
    "queues": [
      {
        "name": "queue-1"
      },
      {
        "name": "queue-2"
      }
    ],
    "sources": [
      {
        "name": "input-1",
        "outputs": [
```

```
          "queue-1",
          "queue-2"
        ],
        "trackers": [
          "tracker-1"
        ]
      },
      {
        "name": "input-2",
        "outputs": [
          "queue-2"
        ],
        "trackers": [
          "tracker-2"
        ]
      }
    ],
    "dispatchers": [
      {
        "name": "dispatcher-1",
        "inputs": [
          "queue-1"
        ],
        "outputs": [
          "processor-1"
        ]
      },
      {
        "name": "dispatcher-2",
        "inputs": [
          "queue-2"
        ],
        "outputs": [
          "processor-2"
        ]
      }
    ],
    "processors": [
      {
        "name": "processor-1"
      },
      {
        "name": "processor-2"
      }
    ],
    "trackers": [
      {
        "name": "tracker-1"
      },
      {
        "name": "tracker-2"
      }
    ],
    "stoppers": [
      {
        "name": "stopper-count",
```

```
        "type": "count",
        "count": 200000000,
        "step": 1000000,
        "inputs": [
          "tracker-1",
          "tracker-2"
        ]
      },
      {
        "name": "stopper-stationarity",
        "type": "stationarity",
        "threshold": 0.000001,
        "inputs": [
          "tracker-1",
          "tracker-2"
        ]
      },
      {
        "name": "stopper-stability",
        "type": "stability",
        "tolerance": 1.25,
        "step": 1000000,
        "inputs": [
          "tracker-1",
          "tracker-2"
        ]
      }
    ],
    "reporters": [
      {
        "name": "reporter-1",
        "type": "json-pretty",
        "inputs": [
          "tracker-1",
          "tracker-2"
        ]
      }
    ]
  },
  "name": "example_oct"
}
```

Listing B.2: Included configuration file.

The configuration corresponds to a system containing:

- two job sources (`input-1` and `input-2`, respectively) assigned to two job-queues as shown in the figure; the inter-arrival times are drawn from the exponential distributions with parameters as designated
- two job queues (`queue-1` and `queue-2`, respectively)
- two dispatchers (`dispatcher-1` and `dispatcher-2`, respectively) that act as schedulers for the processors – in this scenario they monitor the state of the assigned queue and processor and assign jobs from the queue to the processor whenever it is available

- two job processors (processor-1 and processor-2, respectively), the service times are drawn from the exponential distributions with parameters as designated

When run with appropriate number of substitutions passed in, the parser merges the configuration files and starts the simulation. Since no output path is set for the reporter-1, the output will be sent to the command line.

```
$ ./quesi_app -c example_base.json -s 10 -s 5 -s 15 -s 18
{
  "include": "example_include.json",
  "blocks": {
    "sources": [
      {
        "name": "input-1",
        "schedule": {
          "type": "exponential",
          "lambda": 10
        }
      },
      {
        "name": "input-2",
        "schedule": {
          "type": "exponential",
          "lambda": 5
        }
      }
    ],
    "processors": [
      {
        "name": "processor-1",
        "schedule": {
          "type": "exponential",
          "lambda": 15
        }
      },
      {
        "name": "processor-2",
        "schedule": {
          "type": "exponential",
          "lambda": 18
        }
      }
    ]
  },
  "results": [
    {
      "reporter": "reporter-1",
      "tracker": "tracker-1",
      "queueing_time": 0.022355003353266116,
      "processing_time": 0.03320212808539431,
      "num_created": 155384700,
      "num_finished": 155384698,
      "stopper": "stopper-stationarity"
    },
    {
```

```
      "reporter": "reporter-1",
      "tracker": "tracker-2",
      "queueing_time": 0.041212011600649676,
      "processing_time": 0.055552398223901685,
      "num_created": 77690301,
      "num_finished": 77690300,
      "stopper": "stopper-stationarity"
    }
  ]
}
```

Listing B.3: Command line output.

The provided substitution parameters have been converted to appropriate types and substituted and that a new object `results` has been appended to the configuration object.
Due to very low setting of the StopperStationarity threshold, the number of jobs created is significant, nevertheless, the simulation converged (within the limits) with the following per-traffic-class sojourn times:

$$\hat{T}^{(1)} \approx 0.05556 \qquad \hat{T}^{(2)} \approx 0.09676$$