

Hardware-orientierte Untersuchung von drahtlosen Kommunikationsmechanismen im Kontext der Intragebäudekommunikation

Von der Fakultät für Elektrotechnik und Informatik
der Gottfried Wilhelm Leibniz Universität Hannover
zur Erlangung des akademischen Grades
Doktor-Ingenieur
(Dr.-Ing.)
genehmigte Dissertation

von Herrn
M.Sc. Daniel Pfefferkorn

2021

1. Referent: Prof. Dr-Ing. Holger Blume
2. Referent: Prof. Dr-Ing. Rüdiger Kays
Tag der Promotion: 09. Juni 2021

Durch die rapide voranschreitende Miniaturisierung, die kontinuierlich schrumpfende Verlustleistung und den fortwährenden Preisverfall, halten drahtlos kommunizierende Geräte Einzug in vormals drahtgebundenen Kommunikationsdomänen oder völlig neuen Einsatzgebieten. Diese Entwicklung spiegelt sich auch in der zunehmenden Automatisierung von Gebäuden wider, für welche die drahtlose Kommunikation eine Schlüsseltechnologie darstellt. Der Entwurfsprozess für entsprechende Kommunikationssysteme stellt aufgrund der Vielzahl der verwendeten Hard- und Softwarekomponenten sowie deren Parametrierung, vor dem Hintergrund oftmals gegensätzlicher Entwurfskriterien eine komplexe Herausforderung dar. Um dennoch beispielsweise anwendungsrelevante Güten sicherzustellen oder anwendungsspezifische Anpassungen vorzunehmen, ist es notwendig, das Systemverhalten in einer Vielzahl von Betriebszuständen präzise vorherzusagen zu können. Vor dem Hintergrund des Abtauschs zwischen Komplexität und Präzision können hier analytische Verfahren, bedientheoretische Modelle und Simulationstechniken eingesetzt werden. Um ein hohes Maß an Präzision zu erreichen, ist es dabei notwendig, zeitliche Zusammenhänge innerhalb der eingesetzten Hardware sowie der Protokolle abbilden zu können. Diese Anforderungen werden vom Ansatz der diskreten, ereignisbasierten Simulation vollständig erfüllt und erlauben es, ein umfassendes und zugleich realistisches Bild vom Verhalten von Kommunikationssystemen für die Intragebäudekommunikation gewinnen zu können.

Die vorliegende Arbeit stellt diese und andere Methoden zur Entwurfsraumexploration vor und bewertet sie hinsichtlich ihrer inhärenten Modellierungskomplexität sowie der resultierenden Abbildungspräzision. Anhand von zwei Anwendungsbeispielen (elektronische, funkvernetzte Schließanlage und optisches Kommunikationssystem) mit gegensätzlichen Entwurfszielen wird dabei der Prozess der diskreten, ereignisbasierten Modellierung veranschaulicht. Neben der exakten Wiedergabe des tatsächlichen Systemverhaltens – insbesondere der Hardware-Aspekte –, erlaubt ein hohes Maß an Modularität die schnelle Anpassung, respektive Erweiterung, der Modelle. Bei der anschließenden Untersuchung des Entwurfsraums basierend auf verschiedenen real existierenden Hardwarekomponenten sowie aktuellen Kommunikationsprotokollen, zeigt sich zum einen, dass Änderungen einzelner Betriebsparameter aufgrund der starken Kopplung der einzelnen Systemelemente oftmals schichtübergreifende, d.h. systemweite Auswirkungen haben. Zum anderen offenbaren Parameterstudien, dass die entstehenden Lösungen häufig nichtlinear im Entwurfsraum verteilt sind, weswegen eine linear approximative Systemauslegung zu suboptimalen Konfigurationen führen kann. Ebenso ist es mit Hilfe des erarbeiteten Ansatzes sowie statistischer Methoden möglich, diejenigen Parameter zu bestimmen, die für ein Entwurfskriterium, z.B. den Datendurchsatz, die größte Bedeutung haben, so dass gezielte Anpassungen möglich sind.

Schlagworte: Netzwerksimulation, drahtlose Kommunikation, optische Kommunikation, Entwurfsraumexploration

Through the rapidly advancing miniaturization, the continuously decreasing power consumption and the perpetual price decline, wirelessly communicating devices find their way into formerly wired communication domains and entirely new fields of application. This development is reflected in the increasing automation of buildings, for which wireless communication is a key technology. Due to the multitude of hard- and software components, including their respective parameters, the design process of communication systems with often contrary design goals, such as high throughput with very low power consumption, poses a complex challenge. Nevertheless, in order to guarantee communication quality at application level or to make application-specific adjustments, it is necessary to precisely predict system performance for a variety of operating conditions. Against the background of trading complexity and precision, analytical methods, queueing theory and simulation techniques can be employed. In order to reach a high degree of precision, it is necessary to be able to model temporal contexts or relations, existing within the used hardware and associated protocols. These requirements are entirely fulfilled by the approach of discrete event-based simulation and allow forming of a comprehensive and realistic picture of indoor communication system performance.

The presented thesis introduces this and other methods for design space exploration and assesses them with regard to their inherent modelling complexity as well as resulting modelling precision. Based on two application examples (radio-networked door locking system and optical communication system) with contrasting design goals the process of discrete event-based modelling of a communication system is illustrated. Next to the exact reproduction of actual behavior – particularly the hardware aspects –, a high degree of modularity enables quick adaptation, as well as extension, of the models. The subsequent design space exploration based on real-life hardware as well as state of the art communication protocols reveals on the one hand, that a change in a single parameter value can have a layer-spanning or system-wide impact due to the tight coupling of individual system elements. On the other hand, parameter studies showed that resulting solutions are often distributed non-linearly within the design space, which is why an approximative system design can lead to negative results. Furthermore, the developed approach combined with statistical methods allows determination of those parameters with the highest impact on a specified design goal, e.g. throughput, enabling targeted adjustments.

Keywords: Network simulation, wireless communication, optical communication, design space exploration

Kurzfassung	iii
Abstract	v
Abkürzungsverzeichnis	xi
Symbolverzeichnis	xiii
1 Einleitung	1
1.1 Drahtlose Kommunikation in Gebäuden	2
1.2 Einordnung	4
1.3 Zielsetzung	6
1.4 Aufbau der Arbeit	7
2 Grundlagen drahtloser Datenübertragung	9
2.1 Funkbasierte Datenübertragung	9
2.1.1 Beschreibung des Übertragungskanals mittels Pfadverlustmodellen	9
2.1.2 Aufbau eines funkbasierten Datenübertragungssystems	13
2.1.3 Aufbau und Funktion Sende- & Empfängereinheit	14
2.1.4 Medienzugriffsschicht	15
2.1.5 Vermittlungsschicht	19
2.2 Optische Datenübertragung	28
2.2.1 Aufbau eines optischen Datenübertragungssystems	28
2.2.2 Genutzte Modulationsverfahren	33
3 Methoden zur Entwurfsraumexploration für Kommunikationssysteme	35
3.1 Entwurfsraumexploration	35
3.2 Modellierung des Verhaltens von Kommunikationssystemen	36
3.2.1 Analytische Modelle	37
3.2.2 Simulation	38
3.2.3 Simulationsbasierte Entwurfsraumexploration	40
4 Werkzeuge für die simulationsgestützte Analyse von Netzwerkkommunikation	41
4.1 Überblick über vorhandene Simulationswerkzeuge	41
4.1.1 Network Simulator/ns	41

4.1.2	OPNET Modeler/Riverbed Modeler	42
4.1.3	NetSim	42
4.1.4	OMNeT++	42
4.1.5	Vergleich und Auswahl der Simulationsumgebung	43
4.2	Netzwerksimulationsframework OMNeT++	44
4.2.1	Komponenten und Struktur	44
5	Modellierung von drahtlosen Kommunikationssystemen zur diskreten Netzwerksimulation	49
5.1	Entwurfsprozess	50
5.1.1	Anforderungen	50
5.1.2	Implementierung	53
5.1.3	Verifikation	53
5.2	Modellierung eines funkbasierten Kommunikationssystems mit geringer Verlustleistungsaufnahme	54
5.2.1	Bitübertragungsschicht	54
5.2.2	Sicherungsschicht	56
5.2.3	Vermittlungsschicht	57
5.2.4	Anwendungsschicht	58
5.3	Modellierung eines optischen Kommunikationssystems	58
5.3.1	Bitübertragungsschicht	58
5.3.2	Höhere Schichten	59
6	Modellgestützte Parameteroptimierung von Drahtloskommunikationssystemen	61
6.1	Anwendungsbeispiel funkvernetzte Schließanlage	61
6.1.1	Systemaufbau	62
6.1.2	Simulationsszenario	65
6.1.3	Optimierungen in der Bitübertragungsschicht	67
6.1.4	Optimierungen in der Vermittlungsschicht	71
6.1.5	Optimierungen in der Anwendungsschicht	80
6.1.6	Simulationsperformance	83
6.1.7	Diskussion der Ergebnisse	86
6.2	Anwendungsbeispiel beleuchtungsintegriertes Drahtloskommunikationssystem	87
6.2.1	Systemaufbau	89
6.2.2	Simulationsszenario	89
6.2.3	Experimentelle Verifikation des Simulationsmodells	92
6.2.4	Optimierungen der Umgebungsbedingungen	97
6.2.5	Optimierungen in der Bitübertragungsschicht	97
6.2.6	Optimierungen in der Datensicherungsschicht	103
6.2.7	Optimierungen in der Vermittlungsschicht	103
6.2.8	Einfluss des Transportprotokolls	105
6.2.9	Simulationsperformance	113
6.2.10	Diskussion der Ergebnisse	114
6.3	Auswertung der Optimierungsergebnisse der Anwendungsbeispiele	116
7	Zusammenfassung	119
A	Anhang	123
A.1	Parameter für das empirische Pfadverlustmodell der ITU-R	123

B	Aufbau von Nachrichten verwendeter Protokolle der Vermittlungsschicht	127
B.1	Aufbau der Nachrichten des Protokolls „Ad-Hoc On-Demand Distance Vector Routing“ (AODV)	127
B.2	Aufbau der Nachrichten des Protokolls „Energy-efficient Multi-Path Routing“ (EEMP)	129

Abkürzungsverzeichnis

AODV	Ad hoc On-Demand Distance Vector Routing
AP	Access Point
ARP	Address Resolution Protocol
AWGN	Additive White Gaussian Noise
BAN	Body Area Network
BDP	Bandwidth-Delay Product
BER	Bit Error Rate
BLE	Bluetooth Low Energy
BPSK	Binary Phase Shift Keying
CAP	Contention Access Period
CFP	Contention-free Period
CPU	Central Processing Unit
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
DMA	Direct Memory Access
DMT	Discrete Multitone Transmission
DSP	Digital Signal Processor
DSR	Dynamic Source Routing
DSSS	Direct Sequence Spread Spectrum
EEMRP	Energy-efficient Multipath Routing
FEC	Forward Error Correction
FFD	Full Function Device
FHSS	Frequency Hopping Spread Spectrum
FOV	Field of View
ICMP	Internet Control Message Protocol
IP	Internet Protocol
ISI	Intersymbol Interference
JTAG	Joint Test Action Group

LiFi	Light Fidelity
LLC	Logical Link Control
LOS	Line of Sight
LPWAN	Low Power Wide Area Network
LTE	Long Term Evolution
MAC	Media Access Control
MCPS	MAC common part sublayer
MERR	Minimal Energy Relay Routing
MLME	MAC sublayer management entity
MSS	Maximum Segment Size
MTU	Maximum Transfer Unit
NoC	Network on a Chip
OAP	Optical Access Point
OFDM	Orthogonal Frequency-Division Multiplexing
OOK	On-Off Keying
OSI	Open Systems Interconnection
OWC	Optical Wireless Communication
PAM	Pulse-Amplitude Modulation
PAN	Personal Area Network
PD	Photodiode
PIN	Positive Intrinsic Negative
PLL	Phase-Locked Loop
PPM	Pulse-position Modulation
PSSS	Parallel Sequence Spread Spectrum
PWM	Pulse-Width Modulation
QAM	Quadrature Amplitude Modulation
QPSK	Quadruple Phase Shift Keying
RERR	Route Error
RFID	Radio-Frequency Identification
RREP	Route Reply
RREQ	Route Request
RTC	Real-Time Clock
RTP	Real-Time Transport Protocol
RTT	Round-Trip Time
SNR	Signal-to-Noise Ratio
SoC	System-on-a-Chip
TCP	Traffic Control Protocol
TIA	Transimpedance Amplifier
TTL	Time to live
UDP	User Datagram Protocol
UID	Unique Identifier
VOIP	Voice over IP
WDMA	Wavelength Division Multiple Access

Symbolverzeichnis

\mathbb{A}	Anwendungsraum
A	effektive Fläche einer Antenne oder Photodiode
B	Signalbandbreite
k_B	Boltzmann-Konstante
d_{char}	charakteristische Distanz für die Verfahren „Minimal Energy Relay Routing“
\mathbb{E}	Entwurfsraum
$G_{\mathbf{R}}$	Antennengewinn am Empfänger
$G_{\mathbf{T}}$	Antennengewinn am Transmitter
γ	Verlustleistungskoeffizient im Nahbereich gemäß IEEE 802.15.4
K_{opt}	Anzahl optimaler Hops bei Nutzung von „Minimal Energy Relay Routing“
λ	Wellenlänge
M	Modulationsgrad
N	Verlustleistungskoeffizient des ITU-R-Modells
p	Paketverlustrate/Paketfehlerrate
\mathbb{P}	Parameterraum
$\sigma_{AWGN}^2(\lambda)$	elektrische Leistung additiven, weißen Rauschens in Abhängigkeit der Wellenlänge λ
$P_{\mathbf{R}}$	Vom Empfänger aufgenommene elektromagnetische oder optische Leistung
$P_{\mathbf{T}}$	Vom Transmitter abgegebene elektromagnetische oder optische Leistung
$R_{\mathbf{LED}}(\theta, n)$	Abstrahlcharakteristik/Strahlungsmuster einer LED in Abhängigkeit des Öffnungswinkels θ und der Modenzahl n
\mathbb{S}	Verhaltensraum eines Systems
$\theta_{\mathbf{FOV}}$	Öffnungswinkel von Optiken
S_{PD}	optische Empfindlichkeit einer Photodiode
T_{OF}	Transmittanz eines optischen Filters
X_{σ}	mittelwertfreie, gaussverteilte Zufallsvariable mit Standardabweichung σ

Die Entwicklung der Netzwerktechnik wird maßgeblich von drei Gesetzen getrieben bzw. lässt sich durch diese beschreiben. Das Mooresche Gesetz [1] sagt aus, dass sich die Komplexität integrierter Schaltungen bei identischem Kosten innerhalb eines Zeitraums von ca. 18 Monaten verdoppelt. Es lässt den Schluss zu, dass heute unmöglich erscheinende Aufgaben innerhalb absehbarer Zeit leicht zu bewältigen sein werden. Darüber hinaus werden Komponenten mit identischer Rechenleistung zukünftig zu einem Bruchteil heutiger Kosten verfügbar sein. George Gilder postulierte darüber hinaus 2000, angelehnt an das Mooresche Gesetz, dass sich die zur Verfügung stehende Bandbreite innerhalb eines Zeitraums von ca. sechs Monaten verdoppelt (Gildersches Gesetz) [2]. Als dritte Säule trifft das Metcalfesches Gesetz eine Aussage über die Bedeutung eines Netzwerks. Sie ist proportional zum Quadrat der Anzahl der verbundenen Geräte oder Benutzer (n^2) [3]. Hintergrund ist die Anzahl von $n(n-1)/2$ Verbindungen, die den eigentlichen Wert des Netzwerks darstellen. Im Vergleich dazu wird die Kostenentwicklung als direkt proportional zur Benutzeranzahl n angenommen.

Gemeinsam erklären diese drei Gesetze die gegenwärtige Entwicklung im Bereich der drahtlos kommunizierenden Geräte. Durch die rapide voranschreitende Miniaturisierung, respektive den damit einhergehenden kontinuierlich schrumpfenden Leistungsbedarf, sowie den fortwährenden Preisverfall, halten drahtlos kommunizierende Geräte Einzug in vormals drahtgebundene Kommunikationsdomänen oder erschließen völlig neue Einsatzgebiete. Bestehende Netzwerke werden vorzugsweise durch eine Vielzahl neuer Geräte erweitert, anstatt neue Netzwerke zu eröffnen. Insbesondere bei drahtlos operierenden Netzwerken ist dies mit vergleichsweise geringem Aufwand bzw. geringen Kosten möglich. Infolgedessen wird erwartet, dass sich die Anzahl von drahtlos im Nahbereich kommunizierenden Geräten (WLAN, BLE, ZigBee, etc.) innerhalb der nächsten drei Jahre fast verdoppelt (siehe Abbildung 1.1). Für die Anzahl drahtgebundener Geräte, beträgt das erwartete Wachstum im gleichen Zeitraum nur etwa 25 %.

Zur Erfüllung der eigentlichen, durch die Anwendung vorgegebenen, Aufgabe, deren wesentlicher Bestandteil stets auch die Kommunikation selbst ist, müssen diverse Hard- und Softwarefunktionen zusammenspielen. Dabei werden die Softwarefunktionen üblicherweise in Schichten zusammengefasst, die sich am OSI-Modell [5] orientieren und in Form von Protokollen standardisiert. Sowohl die Hard- als auch besonders die Softwarefunktionen weisen dabei eine große Anzahl von Parametern (z.B. Stromaufnahme beim Senden/Empfangen, Schlaf-Wach-Verhältnis, Anzahl parallel zu suchender

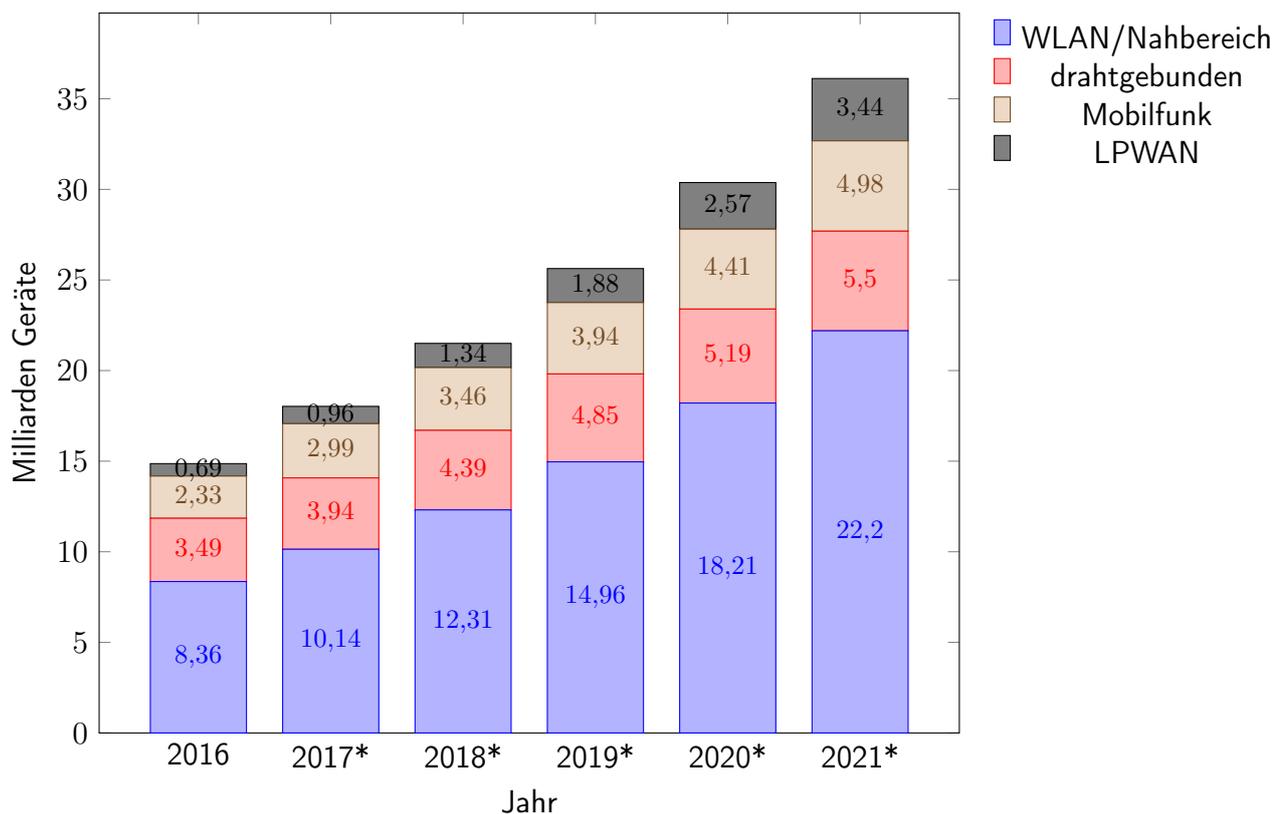


Abbildung 1.1: Entwicklung der weltweiten Anzahl von Kommunikationsgeräten gruppiert nach Verbindungstyp aus dem Jahr 2017 (*Prognose) [4]

Routen zum Ziel, usw.) auf, deren Werte die Funktion des Gesamtsystems maßgeblich beeinflussen. Oftmals stehen auch mehrere Kommunikationsprotokolle zur Verfügung, um die gleiche Aufgabe zu verrichten, wie es beispielsweise beim Routing in der Vermittlungsschicht der Fall ist. Zusätzlich zum so entstehenden hochdimensionalen Parameterraum ergeben sich je nach Anwendungsfall unter Umständen diametral unterschiedliche Anforderungen, so dass eine Standardparametrierung ungeeignet ist, um diese adäquat zu erfüllen. Da der Zusammenhang zwischen Parameterwert und resultierender Systemwirkung darüber hinaus häufig nichtlinear ist, wird die Bestimmung nahezu optimaler Einstellungen deutlich erschwert. Dies verdeutlicht die Notwendigkeit einer Methode zur zielgerichteten Verhaltensprädiktion. Darüber hinaus lässt sich das untersuchte System auf diesem Weg vorab an die Anforderungen im jeweiligen Anwendungsfall anpassen.

1.1 Drahtlose Kommunikation in Gebäuden

Zu Beginn der 90er Jahre des vorigen Jahrhunderts wurden wesentliche Grundsteine für die Gebäudeautomation gelegt. Viele kabelgebundene Standards zur Kommunikation innerhalb von Gebäuden mit dem Ziel der Automation sind auf diese Zeit zurückzuführen. Tabelle 1.1 gibt eine kurze Übersicht über verbreitete Protokolle/Standards deren Informationsübertragung kabelgebunden erfolgt. Von diesen sind heute nur noch KNX, BACnet, DALI und LON relevant.

Für den Gebäudeeinsatz ist die drahtlose Netzwerktechnik von besonderer Bedeutung. Zum einen stellen Verkabelungsarbeiten einen signifikanten Kostenfaktor beim Neubau dar und können im Laufe des Gebäudelebens unter Umständen nicht an veränderte Anforderungen angepasst werden

Tabelle 1.1: Übersicht aktuell zur Gebäudeautomation eingesetzter kabelgebundener Bussysteme

Name	Jahr	max. Datenrate	OSI-Schichten	Vorteile	Nachteile
CAN [6]	1983	1 Mbit/s	1	sehr weit verbreitet	lineare Topologie
EIB	1990	1 Mbit/s	1 - 7	große Verbreitung	veraltet & inzwischen Bestandteil von KNX
KNX [7]	2002	9,6 kbit/s	1 - 7	variable Topologien	geringe Datenrate & hohe Leistungsaufnahme
LON [8]	1990	78 kbit/s	1 - 7	dezentral organisiert	lineare Topologie
PLC-Bus	2002	200 bit/s	?	geringer Installationsaufwand	geringe Verbreitung & Datenrate
DALI [9]	1994	1,2 kbit/s	1 - 2	bidirektionale Übertragung	geringe Datenrate
BACnet [10]	1987	100 Mbit/s	1 - 7	robust & herstellerunabhängig	kaum abgesichert

Tabelle 1.2: Übersicht drahtloser Kommunikationsstandards mit Relevanz für die Gebäudeautomation

Name	Jahr	max. Datenrate	OSI-Schichten	Vorteile	Nachteile
IEEE 802.11ac [12]	2013	6,9 Gbit/s	1 - 2	weit verbreitet	hohe Verlustleistung
IEEE 802.11ah [13]	2016	8,7 Mbit/s	1 - 2	große Reichweite	kaum verbreitet
IEEE 802.15.4 [14]	2003	250 kbit/s	1 - 2	sehr weit verbreitet	hohe Verlustleistung
Z-Wave [15]	2005	100 kbit/s	1 - 2	verschlüsselte Übertragung	nur 232 Teilnehmer pro Segment
KNX-RF [7]	2006	16,4 kbit/s	1 - 2	sehr weit verbreitet	hohe Verlustleistung
Enocean [16]	1997	120 kbit/s	1 - 2	energieautarke Knoten	proprietär & HW-Herstellermonopol
Bluetooth Low Energy [17]	2009	1 Mbit/s	1 - 2	sehr weit verbreitet	begrenzte Reichweite
LoRaWAN [18]	2010	50 kbit/s	1 - 2	niedrige Verlustleistung	(noch) kaum verbreitet
IEEE 802.15.7 [19]	2011	96 Mbit/s	1 - 2	hohe Datenrate	sehr begrenzte Reichweite
LiFi	2011	224 Gbit/s	1 - 2	sehr hohe Datenrate	sehr begrenzte Reichweite

[11]. Andererseits ist die nachträgliche Installation drahtgebundener Systeme, z.B. im Rahmen einer Nachrüstung, oftmals nicht möglich, weswegen dann zwangsläufig drahtlose Lösungen zum Einsatz kommen müssen. Weiterhin kann es sich um Netzwerkteilnehmer handeln, die innerhalb des Gebäudes mobil sein müssen, so dass die Nutzung einer ortsfesten Kabelverbindung ebenfalls unmöglich ist.

Abbildung 1.2 gibt einen exemplarischen Überblick über Geräte bzw. Funktionen, die durch eine Vernetzung automatisiert werden können. Sie lassen sich den folgenden Funktionsgruppen zuordnen:

- Licht- und Klimasteuerung (Heizung, Klimaanlage, Belüftung, Wettersensoren)
- Elektrik (ansteuerbare Steckdosen/Schalter und Verbrauchserfassung)
- Sicherheit (Zutrittskontrollsysteme, Brand- und Einbruchmelder)
- Komfort (zentrale Steuerungseinheiten, Automatisierung, Musik, Haushaltsgeräte)

Außer im Fall der Videoübertragung von Überwachungskameras, stellen die genannten Anwendungsfelder nur geringe Anforderungen an die zur Verfügung stehende Datenrate. Besonders für Steuerungs- oder Sicherheitsfunktionen sind die Anforderungen an die Zuverlässigkeit der Übertragung sowie eine kurze Verzögerungszeit weitaus relevanter. Das tatsächliche Verhalten des aus den verschiedenen Komponenten bestehenden Systems lässt sich für den Einsatzort jedoch nur schwer vorhersagen. Auch die Hersteller bieten in den seltensten Fällen detaillierte Planungswerkzeuge an. Auch hier offenbart sich der Bedarf an präzisen Methoden zur Vorhersage des Systemverhaltens für konkrete Anwendungen



Abbildung 1.2: Beispiele für Geräte zur Gebäudeautomation [20]

1.2 Einordnung

Die Veröffentlichung des Standards IEEE 802.11 (WLAN bzw. WiFi) 1997 markiert den Startpunkt einer flächendeckenden Nutzung von drahtlosen Kommunikationssystemen durch Endnutzer. Wie bereits in Kapitel 1 beschrieben, haben sich seither Hard- und Software stetig weiterentwickelt. Während die erste Gesamtspezifikation bereits 459 Seiten umfasste und die erste Erweiterung IEEE 802.11g-2003 78 Seiten [21], ist die letzte offiziell standardisierte Erweiterung IEEE 802.11ac-2013 allein bereits 425 Seiten lang [12]. Insgesamt ist also eine deutliche Komplexitätssteigerung zu verzeichnen, wobei anzumerken ist, dass in IEEE 802.11 nur Funktionen der OSI-Schichten „Bitübertragung“ und „Datensicherung“, d.h. die Schichten eins und zwei, spezifiziert sind.

Trotz des, durch die Vielzahl von Parametern, hochdimensionalen Entwurfsraums, besteht ein Bedarf an Möglichkeiten zur präzisen Vorhersage des Systemverhaltens unter Anwendungsbedingungen. In Folge der Komplexität beschränken sich bisherige wissenschaftliche Ansätze auf die Modellierung einzelner Effekte bzw. Funktionen einer OSI-Schicht. Das Aufkommen des Themenkomplexes drahtlose Sensornetzwerke sorgte beispielsweise für eine Vielzahl von Untersuchungen verschiedener Medienzugriffsverfahren. Das Projekt „Comprehensive MAC taxonomy database (comatose)“ [22] verzeichnet beispielsweise 327 verschiedene Protokolle, die in der Zeit von 1970 bis 2017 zusammengetragen wurden. Eine Übersicht 50 verschiedener Routingverfahren für drahtlose Sensornetzwerke, d.h. Umsetzungen der Vermittlungsschicht, findet sich in [23].

Wenige Arbeiten berücksichtigen bei der Analyse bzw. Optimierung mehr als eine Schicht des OSI-Modells. Eine Teildisziplin dieser lässt sich unter dem Oberbegriff „Cross-layer Optimization“ zusammenfassen. Entgegen der strikten Trennung in Schichten, ist es hier das Ziel, Probleme durch schichtübergreifende Maßnahmen zu lösen. Dabei können eigentlich schichtinterne Informationen, wie beispielsweise der aktuelle Kanalbelegungszustand, den benachbarten Schichten zur Verfügung gestellt

Tabelle 1.3: Für diese Arbeit relevante Forschungsarbeiten

Quelle	Jahr	Thema	Ansatz (Werkzeug)
[26]	2006	Drahtloses Sensornetzwerk mit schlafenden Knoten	Analytisch (Markov-Modell)
[27]	2009	Framework zur energiebedarfsbasierten Entwurfsraumexploration für WSNs	Analytisch
[28]	2010	Simulation eines optischen Kommunikationssystems	Analytisch(MATLAB)
[29]	2010	Flexible Plattform für die Entwurfsraumexploration von WSNs	Emulation & funkt. Simulation
[30]	2013	Analytische Modellierung von Verbindungsnetzwerken	Analytisch (Bedientheorie)
[31]	2013	Framework für die Entwurfsraumexploration für drahtlose Sensornetzwerke	Diskrete Simulation
[32]	2015	Modellbasierte Entwurfsraumexploration für drahtlose Chip-Verbindungsnetzwerke	Analytisch
[33]	2017	Iterative Entwurfsraumexploration für Netzwerke mit Bedarf für Leistungsgarantien	Analytisch (Network Calculus)

werden. In [24] wird das Zusammenspiel zwischen Medienzugriff und Vermittlung verbessert, in dem das quasistatische Routing von der aktuell geplanten Nutzung benachbarter Verbindungen beeinflusst wird. In [25] zeigen die Autoren generelle Ansätze zur „Cross-layer Optimization“ am Beispiel von Multimediaübertragungen in Mobilfunknetzen. Die beschriebene Methode kann als wirkungsvolles Mittel zum Erreichen von anwendungsbezogenen Entwurfszielen eingesetzt werden, hat allerdings nicht das Ziel ein Systemverständnis zu entwickeln oder eine allgemeingültige Analysemethoden darzustellen. Die vorhandenen Leistungsdefizite werden stattdessen durch Nutzung von Informationen abgemildert, die durch das teilweise Aufbrechen der OSI-Schichtgrenzen zur Verfügung stehen.

Um das Systemverhalten mit einem hohen Maß an Genauigkeit und gleichzeitig allgemeingültig untersuchen zu können, empfiehlt sich dessen Modellierung. Die Wahl des Modellierungsansatzes sowie die Gestaltung des Modells bestimmen den resultierenden Detailgrad, welcher insbesondere bei nichtlinearem Verhalten für die Modellgüte entscheidend ist. Tabelle 1.3 gibt einen Überblick über relevante Forschungsarbeiten, die mit verschiedenen Ansätzen versuchen, Kommunikationssysteme zu modellieren, beziehungsweise eine Entwurfsraumexploration durchzuführen.

In [26] wird sowohl das Verhalten eines einzelnen Knotens im Netzwerk als auch das Gesamtverhalten mittels eines Markov-Modells analytisch abgebildet und im Anschluss hinsichtlich Leistungsaufnahme, Datendurchsatzrate des Netzwerks sowie der resultierenden Übertragungslatenz bewertet. Dabei werden viele Eigenschaften nur abstrakt modelliert. So unterscheidet das Modell der Funkkomponente zwar Betriebszustände und eine damit verbundene Verlustleistungsaufnahme, vernachlässigt dabei allerdings die zum Wechsel der Zustände notwendigen Zeitdauern. Gemessen an den kurzen Zeiten im empfangsbereiten (Kanalprüfung) oder sendenden Zustand (ausgehende Datenübertragung) sind diese jedoch alles andere als unerheblich. Das in [27] vorgestellte Framework konzentriert sich ganz auf die energetischen Aspekte der Funkkommunikation, wie Fehlerkorrekturmaßnahmen sowie die Verlustleistungsaufnahme einzelner Komponenten im Sende- bzw. Empfangspfad, berücksichtigt allerdings keinerlei Mechanismen oberhalb der Bitübertragungsschicht. Die analytische Abbildung eines optischen Kommunikationssystems in [28] verbleibt ebenfalls auf dem Niveau der Bitübertragung, wodurch keine Aussagen im Anwendungskontext möglich sind. Mit der flexiblen Plattform in [29] können die innerhalb der zu simulierenden SoCs abzuarbeitenden Instruktionen inklusive der On-Chip-Kommunikation taktzyklengenau dargestellt werden, was einem sehr hohen Maß an Modellierungspräzision entspricht. Allerdings bleiben auch hier die Auswirkungen von relevanten Softwarekomponenten, wie Medienzugriffs- und Routingprotokollen, auf die Leistungsdaten im Anwendungsfall unberücksichtigt. Zudem kommt zur Abbildung des Kommunikationskanals ein so stark vereinfachtes Modell zum Einsatz, dass eine repräsentative Darstellung der Bitübertragung nur schwer möglich ist.

Ein anderer analytischer Modellierungsansatz findet sich in [30]. Hier werden bedientheoretische Modelle vorgestellt, um Networks-on-Chip (NoCs) als Übertragungskomponenten für Vielkernprozessoren analysieren zu können. Nachteilig wirkt sich hierbei jedoch aus, dass zum einen nur der stationäre, d.h. eingeschwingene Zustand eines Netzwerks dargestellt werden kann und keine Untersuchungen

von dynamischen Routingalgorithmen durchgeführt wurden. Besonders für dynamische Netzwerke, d.h. Netzwerke mit dynamischer Topologie oder Übertragungsbedingungen, ist das transiente Verhalten von großer Bedeutung, da durch die Kombination verschiedener Protokolle innerhalb der einzelnen Schichten unerwünschtes Verhalten, wie zum Beispiel oszillative Zustände oder spontan hohe Netzwerklast, entstehen kann.

Demgegenüber stehen die Ergebnisse in [31]. Hier wurden mit Hilfe eines diskreten Netzwerksimulators sowohl die hardware-spezifischen Eigenschaften der Übertragungskomponenten als auch verschiedene Medienzugriffsprotokolle abgebildet. Routingprotokolle und Anwendungen wurden nicht untersucht, obwohl der Simulator dies prinzipiell unterstützt.

In [32] wurden sowohl funkbasierte als auch optische Möglichkeiten zur On-Chip-Kommunikation analytisch modelliert und untersucht. Dabei bleiben jedoch Protokolle oberhalb der Bitübertragungsschicht vollständig unberücksichtigt.

Die Kommunikationsperformance kann auch mittels „Deterministic Network Calculus“ bestimmt werden. In [33] erfolgte mittels dieses Ansatzes eine Entwurfsraumexploration für drahtgebundene Netzwerke und es wurden Lösungen hinsichtlich des Optimierungskriteriums garantierter maximaler Latenz evaluiert. Eine Berücksichtigung verschiedener Medienzugriffsverfahren sowie konkreter Anwendungen erfolgte nicht.

Zusammenfassend lässt sich aussagen, dass bislang kein umfassendes Verfahren existiert, um komplexe drahtlose kommunizierende Netzwerke unter Einbeziehung von Hardware-Eigenschaften (z.B. Umschaltzeiten und Verlustleistungsaufnahme des Transceivers) sowie schichtübergreifend analysierbar und im Bezug auf eine konkrete Anwendung optimierbar zu machen.

1.3 Zielsetzung

Das Ziel dieser Arbeit ist die Entwicklung und Analyse einer Methode, die eine Verhaltensvorhersage und somit Untersuchung von drahtlosen Kommunikationssystemen im Kontext der Intragebäudekommunikation ermöglicht. Dadurch wird einerseits eine Entwurfsraumexploration ermöglicht, andererseits aber auch die anwendungsspezifische Auslegung ermöglicht. Insbesondere werden die folgenden Ziele realisiert:

Anwendungsorientierung

Die Analyse erfolgt anwendungsorientiert, d.h. die an das System gestellten Anforderungen erwachsen aus den Anforderungen auf Anwendungsebene. Leistungsdaten werden auf die Anwendungsebene bezogen angegeben, sofern dies möglich und sinnvoll ist.

Exakte Abbildung physikalischer Gegebenheiten

Ein großer Teil des Systemverhaltens resultiert aus den verwendeten Algorithmen/Protokollen sowie deren Parametrierung. Im Bereich der Datensicherungs- und der Bitübertragungsschicht werden diese aus dem Virtuellen ins Physische überführt, d.h. realisiert. Zum einen benötigen die verwendeten Sende- bzw. Empfangseinheiten Zeit und Energie, um ihre Funktion zu erfüllen. Zum anderen müssen die physikalischen Gesetzmäßigkeiten des Übertragungskanals ausreichend berücksichtigt werden, da sie die Kommunikation grundlegend bestimmen. Beispielsweise kann eine Nachricht, je nach Wahl

und Parametrierung des Kanalmodells, empfangbar sein oder nicht, was wiederum Aktionen in den anderen Schichten bewirkt.

Untersuchung des Entwurfsraums

Der zu erarbeitende Ansatz muss eine umfassende Untersuchung des Entwurfsraums ermöglichen. Um die Analyse weitestgehend zu automatisieren, sollten dazu Methoden wie Parameterstudien sowie automatisierte Datenauswertungsverfahren zum Einsatz kommen. Die Parameter mit dem größten Einfluss auf einzelne Leistungsdaten sollen identifiziert und Wechselwirkungen zwischen einzelnen Entwurfszielen aufgezeigt werden.

Bewertung auf Basis von Kostenfaktoren

Die Bewertung beruht auf einer feststehenden Auswahl von Kostenfaktoren, die applikationsspezifisch gewichtet werden. Dadurch wird den anwendungsspezifischen Anforderungen Rechnung getragen. Darüber hinaus können, numerische Vorgaben für bestimmte Leistungsparameter festgelegt und auf Einhaltung überprüft werden.

Der ganzheitliche Ansatz soll bezüglich der Leistungsfähigkeit, aber auch des Ressourcenbedarfs (Rechenzeit bzw. Speicherbedarf) sowie der Entwicklungs- bzw. Anwendungscomplexität bewertet werden. Zusätzlich wird die Anwendung der entwickelten Methode exemplarisch für zwei praxisrelevante Fallstudien mit unterschiedlichem Anforderungsprofil dargestellt.

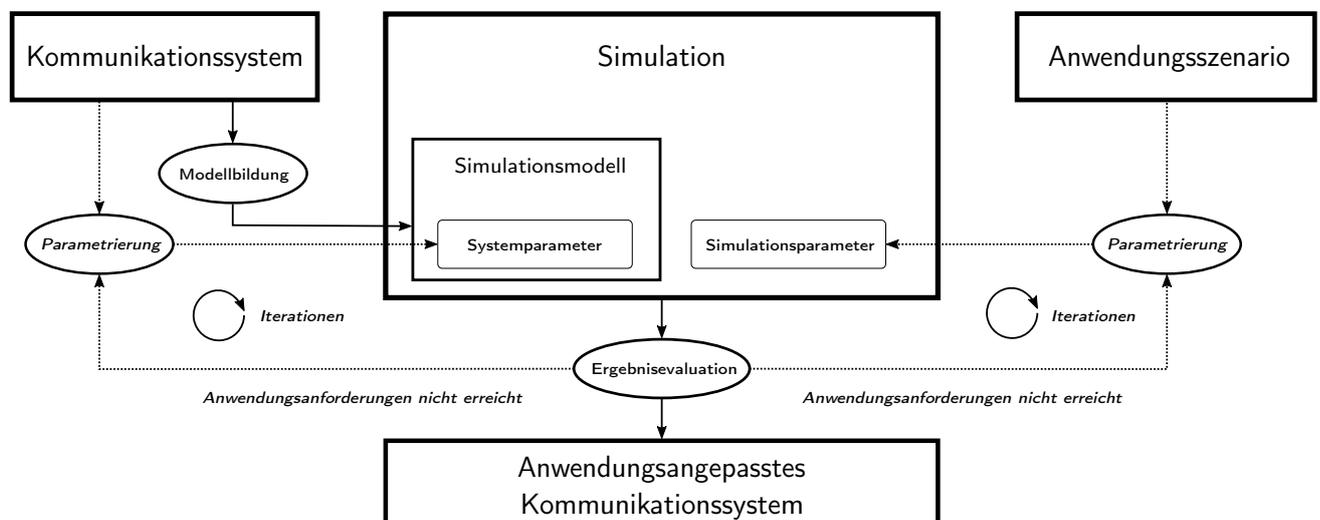


Abbildung 1.3: Darstellung des dieser Arbeit zugrundeliegenden Simulationsansatzes zur anwendungsorientierten Optimierung von Kommunikationssystemen

1.4 Aufbau der Arbeit

Die vorliegende Arbeit gliedert sich inhaltlich in insgesamt sechs Abschnitte. Zunächst werden in **Kapitel 2** Grundlagen der drahtlosen Datenübertragung erläutert. An den Schichten des OSI-Modells

orientiert, wird dabei zunächst auf die funkbasierte Kommunikation eingegangen. Im Anschluss erfolgt eine Ergänzung im Bezug auf die Unterschiede optischer Kommunikation.

Kapitel 3 befasst sich mit dem Themenkomplex der Entwurfsraumexploration und der dafür geeigneten Möglichkeiten zur Modellierung von Kommunikationssystemen.

Im Anschluss werden im **Kapitel 4** Softwarewerkzeuge vorgestellt, die eine diskrete event-basierte Simulation von Computernetzwerken und somit die vorangegangene erläuterte Entwurfsraumexploration ermöglichen. Dem Vergleich der Werkzeuge folgt eine Beschreibung des im Rahmen dieser Arbeit verwendeten Netzwerksimulationsframeworks OMNeT++.

Der Prozess der Modellierung eines drahtlosen Kommunikationssystems wird in **Kapitel 5** ausführlich erläutert. Dabei wird neben Anforderungen an die Implementierung des Simulationsmodells, insbesondere auf die notwendigen Schritte für dessen Validierung eingegangen. Die Beschreibung der im Rahmen dieser Arbeit umgesetzten Elemente/Module bildet den Abschluss des Kapitels.

Kapitel 6 verdeutlicht auf Basis zweier Fallstudien mit unterschiedlichen Anforderungsprofilen, wie die Modelle genutzt werden können, um den Entwurfsraum zu analysieren und das Systemverhalten mit den anwendungsbasierten Anforderungen in Einklang zu bringen. Dabei wird sowohl der für die Simulationen anfallende Rechenaufwand benannt, als auch die resultierende Möglichkeit zur statistischen Analyse der generierten Ergebnisse erläutert.

Kapitel 7 fasst die wesentlichen Erkenntnisse der Arbeit zusammen und gibt einen Überblick über mögliche weitere Herausforderungen.

Grundlagen drahtloser Datenübertragung

Im Rahmen dieses Kapitels sollen zunächst die notwendigen Grundlagen gelegt werden, um die in den nachfolgenden Kapiteln vorgestellte Implementierung sowie die Ergebnisse einordnen und bewerten zu können. Dabei werden zunächst funkbasierte Themen wie Pfadverlustmodelle, die zur Funkkommunikation notwendige Hardware sowie Medienzugriffs- und Routingprotokolle erläutert. Im Anschluss werden die spezifischen Eigenschaften von optischer Datenkommunikation dargelegt, wobei insbesondere Ausbreitungswege sowie die nachrichtentheoretischen Eigenschaften von Bedeutung sind.

2.1 Funkbasierte Datenübertragung

2.1.1 Beschreibung des Übertragungskanals mittels Pfadverlustmodellen

Der Pfadverlust bezeichnet das Verhältnis aus abgestrahlter und empfangener Leistung und wird üblicherweise in dB angegeben. Er beinhaltet alle auftretenden Verluste, die durch Interaktionen der sich ausbreitenden Welle mit Objekten innerhalb des Übertragungswegs auftreten können. Im Folgenden werden Modelle zur Vorhersage des zu erwartenden Pfadverlustes erläutert.

Tabelle 2.1: Auflistung berücksichtigter Pfadverlustmodelle

Name	Kategorie	Ansatz
Freiraumdämpfung	feldtheoretisch	Antenne als isotroper Kugelstrahler
ITU-R P.1238-9	empirisch	vereinfachte Formel inkl. tabellierter Werte
IEEE 802.15.4	empirisch	stark vereinfachte Formel
„Two-Ray Ground Reflection“	analytisch	LOS & bodenreflektierte Komponente
„Log-Normal Shadowing“	empirisch	stochastische Variation mittlerer Dämpfung
Raytracing	deterministisch	Berechnung „optischer“ Strahlenverläufe

Freiraumdämpfung

Geht man von einer Antenne als isotropem Kugelstrahler aus, verteilt sich die abgestrahlte Leistung P_T gleichmäßig auf eine Kugeloberfläche:

$$S = \frac{P_T}{A} = \frac{P_T}{4\pi r^2} \quad (2.1)$$

Da die Wirkfläche der Empfangsantenne über $A_W = \frac{\lambda^2}{4\pi}$ bestimmt ist, lässt sich die resultierende Empfangsleistung über den Zusammenhang $P_R = S \cdot A_W$ herleiten:

$$P_R = P_T \left(\frac{\lambda}{4\pi r} \right)^2 \quad (2.2)$$

Sind Antennenabstand d , Antennengewinne G_T und G_R sowie die Sendeleistung P_T bekannt, lässt sich die Empfangsleistung P_R mit Hilfe der Friis-Übertragungsgleichung (siehe Gleichung 2.3) errechnen:

$$\frac{P_R}{P_T} = G_T G_R \left(\frac{\lambda}{4\pi d} \right)^2 \quad (2.3)$$

Der Pfadverlust, d.h. das Verhältnis zwischen abgestrahlter Leistung P_T und empfangener Leistung P_R , ergibt sich zu:

$$PL_{dB} = 10 \log \left(\frac{P_T}{P_R} \right) = -10 \log_{10} \left[\frac{G_T G_R \lambda^2}{(4\pi)^2 d^2} \right] \quad (2.4)$$

Weiterhin gilt Gleichung 2.4 nur für Abstände $d > 0$, weshalb Pfadverlustmodelle oftmals eine eingeführte Referenzdistanz d_0 nutzen, für die die Empfangsleistung bekannt ist. Als Folge kann die Empfangsleistung $P_R(d)$ in jeder Entfernung $d > d_0$ auf die Empfangsleistung in Referenzdistanz $P_R(d_0)$ bezogen werden:

$$P_R(d) = P_R(d_0) \left(\frac{d_0}{d} \right)^2 \quad d \geq d_0 \geq d_f \quad (2.5)$$

Die Friis-Übertragungsgleichung stellt ein wertvolles Werkzeug zur Abschätzung der maximal empfangbaren Leistung dar. Die tatsächliche Empfangsleistung fällt aufgrund von Effekten wie umgebungsbedingter Dämpfung, Mehrwegeausbreitung und Abschattungen im Allgemeinen deutlich geringer aus.

Empirisches Pfadverlustmodell der ITU-R

Zur Abschätzung der innerhalb von Gebäuden auftretenden Pfadverluste hat die ITU-R eine Empfehlung namens ITU-R P.1238-9 veröffentlicht [34]. Sie beinhaltet Modelle und Modellparameter für den Frequenzbereich von 300 MHz bis 100 GHz zur Planung von funkbasierten System und Netzwerken im Innenbereich. Weiterhin werden Effekte wie Verzögerungsverteilung, Polarisation, Antennenabstrahlungscharakteristiken, Einfluss von sich bewegenden und statischen Objekten und sowie der Einfluss der im Gebäude verfügbaren Materialien (baulich oder als Mobiliar) berücksichtigt. Innerhalb der

Empfehlung werden zwei verschiedene Arten von Modellen unterschieden, generelles und gebäudespezifisches Modell. Die Grundlage des generellen Modells bildet die folgende Formel zur Berechnung des Pfadverlustes:

$$PL_{total} = PL(d_0) + N \log_{10} \left(\frac{d}{d_0} \right) + PL_f(n) \quad (2.6)$$

wobei N : = Verlustleistungskoeffizient
 f : = Frequenz in MHz
 d : = Distanz zwischen Sender und Empfänger in m
 d_0 : = Referenzdistanz in m
 $PL(d_0)$: = Pfadverlust bei Referenzdistanz von 1 m unter Annahme von Freiraumdämpfung
 $PL(d_0) = 20 \log_{10} f - 28$ (f in MHz)
 PL_f : = Verlustfaktor bei Etagenwechsel bzw. Boden-/Deckendurchdringung (dB)
 n : = Anzahl der Böden/Decken zwischen Sender und Empfänger ($PL_f = 0$ dB für $n = 0$)

Werte für den distanz-, umgebungs- und frequenzabhängigen Verlustleistungskoeffizienten N können dabei Tabelle A.1 und die Werte für den Verlustfaktor bei Etagenwechsel PL_f Tabelle A.2 entnommen werden. Es zeigt sich die große Variabilität aufgrund der verschiedenen Umgebungen sowie der, im Gegensatz zur Theorie, nichtlineare Zusammenhang zwischen Dämpfung und Frequenz.

Pfadverlustmodell gemäß IEEE 802.15.4

Innerhalb des Standards IEEE 802.15.4 wird ebenfalls ein Modell zur Abschätzung des Pfadverlusts angegeben, dass in seiner Form im Wesentlichen dem Modell der ITU-R entspricht. In diesem Modell wird allerdings zwischen einem Nah- und einem Fernbereich unterschieden, Im Nahbereich, d.h. bis zu einer Entfernung von 8 m wird die Freiraumdämpfung angenommen, im Fernbereich erfährt das übertragene Signal eine stärkere Dämpfung.

$$PL_d = \begin{cases} PL(1) + 10\gamma_1 \log_{10}(d) & d \leq 8 \text{ m} \\ PL(8) + 10\gamma_8 \log_{10}\left(\frac{d}{8}\right) & d > 8 \text{ m} \end{cases} \quad (2.7)$$

wobei γ_1 : = Verlustleistungskoeffizient im Nahbereich (lt. Standard 2,0)
 γ_8 : = Verlustleistungskoeffizient im Fernbereich (lt. Standard 3,3)
 d : = Distanz zwischen Sender und Empfänger in m

Der grundlegende Pfadverlust $PL(1)$ wird durch folgende Gleichung beschrieben und ist identisch mit der Freiraumdämpfung:

$$PL(1) = 10\gamma_1 \log_{10} \left(\frac{4\pi f}{c} \right) \quad (2.8)$$

wobei γ_1 : = Verlustleistungskoeffizient im Nahbereich (lt. Standard 2,0)
 f : = Trägerfrequenz
 c : = Lichtgeschwindigkeit $299\,792\,458 \text{ m s}^{-1}$

Log-Normal Shadowing

Aufgrund der Vielzahl der Variationsmöglichkeiten einer Gebäudestruktur bzw. deren Einrichtung sowie auftretender Abschattungseffekte kann der Pfadverlust nur näherungsweise genau bestimmt werden. Bei gleicher Entfernung zwischen Sender und Empfänger können sich signifikant unterschiedliche Empfangsleistungen einstellen, so dass die mit Hilfe von den Formeln 2.4, 2.6 bzw. 2.7 bestimmten Pfadverluste eher als Mittelwert verstanden werden können. Messungen haben gezeigt, dass der auftretende Pfadverlust $PL(d)$ bei beliebiger Distanz d zwischen Sender und Empfänger zufällig, aber um den abstandsabhängigen Pfadverlust $\overline{PL(d)}$ normalverteilt ist [35]. Daraus ergibt sich:

$$PL(d) = \overline{PL(d)} + X_\sigma = \overline{PL(d_0)} + 10n \log\left(\frac{d}{d_0}\right) + X_\sigma \quad (2.9)$$

X_σ bezeichnet hierbei eine mittelwertfreie gaussverteilte Zufallsvariable mit einer Standardabweichung von σ . Die Werte für n und σ sind vom Gebäude bzw. der Umgebung abhängig und können entweder durch lineare Regression aus Messwerten abgeleitet oder in der Literatur nachgeschlagen werden.

Two-Ray Ground Reflection/Plane Earth

Ein häufiges Problem bei der Vorhersage von zu erwartenden Empfangsleistungen bzw. dem zu erwartenden Pfadverlust sind Mehrwegeausbreitungen. Sie können je nach Phasenlage der auf die Empfangsantenne treffenden Wellen zu konstruktiver oder destruktiver Überlagerung führen. Das „Two-Ray Ground Reflection“-Modell berücksichtigt zwei Strahlengänge, die direkte Verbindung zwischen Sende- und Empfangsantenne und den Pfad nach einfacher Reflektion auf der als plan angenommenen Erdoberfläche (indirekter Pfad) [35]. Auf Basis von Distanz d sowie der Höhe von Sende- und Empfangsantenne h_t bzw. h_r lässt sich die Differenz der Pfadlängen d' (direkter Pfad) und d'' (indirekter Pfad) bestimmen:

$$\Delta = d'' - d' = \sqrt{(h_t + h_r)^2 + d^2} - \sqrt{(h_t - h_r)^2 + d^2} \quad (2.10)$$

Für den Fall, dass die Distanz d verglichen mit $h_t + h_r$ sehr groß ist, lässt sich Gleichung 2.10 durch Taylor-Entwicklung approximieren:

$$\Delta = d'' - d' = \frac{2h_t h_r}{d} \quad (2.11)$$

Mit Hilfe der Pfadlängendifferenz Δ lässt sich die Phasendifferenz der beiden eintreffenden Wellen berechnen:

$$\theta_\Delta = \frac{2\pi\Delta}{\lambda} = \frac{\Delta\omega_c}{c} \quad (2.12)$$

Der Betrag der elektrischen Feldstärke $|E_{Total}|$ in Distanz d zum Sender ist dann durch

$$|E_{Total}(d)| = 2 \frac{E_0 d_0}{d} \sin\left(\frac{\theta_\Delta}{2}\right) \quad (2.13)$$

bestimmt. Für genügend kleine Phasendifferenzen θ_Δ gilt, $\sin(\theta_\Delta/2) \approx \theta_\Delta/2$. Gleichung 2.13 kann für genügend große Distanzen d ($d > 20h_t h_r/\lambda$) durch

$$E_{Total}(d) \approx \frac{2E_0 d_0}{d} \frac{2\pi h_t h_r}{\lambda d} \quad (2.14)$$

approximiert werden, so dass die Empfangsleistung P_r mittels

$$P_r = P_t G_t G_r \frac{h_t^2 h_r^2}{d^4} \quad (2.15)$$

bestimmbar ist. Der Pfadverlust für das „Two-Ray Ground Reflection“-Modell kann in dB mittels

$$PL(d) = 40 \log d - (10 \log G_t + 10 \log G_r + 20 \log h_t + 20 \log h_r) \quad (2.16)$$

ausgedrückt werden.

Raytracing

Eine Methode zur Bestimmung der resultierenden Empfangsfeldstärke am Empfänger ist die Bestimmung und Verfolgung aller möglichen Übertragungspfade zum Sender. Dazu wird eine bestimmte Anzahl von Strahlen („rays“) in allen Raumrichtungen ausgesandt, die dann von auf dem Weg liegenden Hindernissen beeinflusst, d.h. teilabsorbiert, reflektiert, gebrochen, gestreut oder gebeugt, werden. Beim Empfänger werden pro Strahl Dämpfung und Verzögerung bestimmt und summiert, wodurch sich die insgesamt zu erwartende Dämpfung, die dominante Einfallsrichtung sowie richtungs- bzw. wegeabhängige Verzögerungen berechnen lassen. Um diese Berechnungen durchführen zu können, müssen die exakte Geometrie des Raumes inklusive aller sich darin befindlichen Objekte sowie die elektrischen Eigenschaften allen vorhandenen Materialien bekannt sein [36] [37]. Zudem ist der entstehende Rechenaufwand vor allen Dingen bei kleinen Winkelinkrementen für das Aussenden der Strahlen sehr hoch. Eine Möglichkeit, um diesen zu reduzieren, ist eine Raytracing-Berechnung mit groben Winkelinkrementen in Kombination mit stochastischer Modellierung, die eher in der Lage ist die zeitvariablen Eigenschaften des Übertragungskanals abzubilden [38]. Häufig werden Raytracing-Modelle auch zur Berechnung der Mobilfunkabdeckung in urbanen Szenarien eingesetzt [39] [40]. Abbildung 2.1 visualisiert die vorhergesagte Signalfeldstärke innerhalb eines Gebäudes für eine Empfängerhöhe von 2 m. Die Intensität ist in Falschfarben dargestellt. Es zeigen sich deutlich die Einflüsse der Türen bzw. Durchgangsbereiche sowie Abschattungseffekte. Weiterhin sind die Anteile der einfach und zweifach reflektierten Komponenten gesondert dargestellt. Sie tragen neben den nichtreflektierten Komponenten ebenfalls zur Gesamtfeldstärke bei, geben aber vor allem Auskunft über die eigentliche Signalausbreitung innerhalb der Räume.

2.1.2 Aufbau eines funkbasierten Datenübertragungssystems

Ein funkbasiertes Datenübertragungssystem besteht aus einzelnen Kommunikationsknoten sowie dem sie verbindenden Kanal. Die Kanaleigenschaften sind dabei durch die Ausbreitungseigenschaften, dessen insbesondere die Pfadverlustmodelle (siehe Abschnitt 2.2.1), bestimmt. Die Kommunikationsknoten unterscheiden sich aufgrund ihres Einsatzzwecks im Bezug auf die Leistungsfähigkeit beziehungsweise die maximale Verlustleistungsaufnahme. Letztere wird zum einen durch die stets enthaltene Transceiver-einheit verursacht, die in unterschiedlichen Betriebsmodi unterschiedlichen Leistungsbedarf entwickelt.

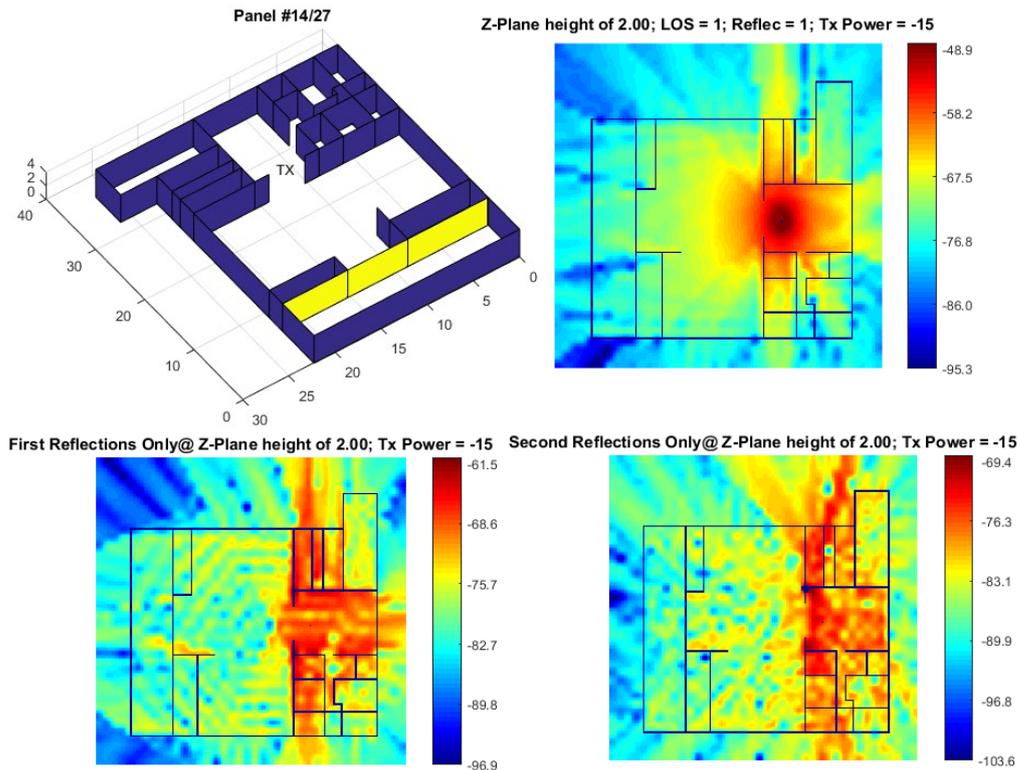


Abbildung 2.1: Beispiel für eine mittels Raytracing durchgeführte Analyse der resultierenden Signalfeldstärke innerhalb eines Gebäudes auf Basis der Reflektionen erster und zweiter Ordnung. In Falschfarben sind jeweils die verbleibenden Intensitäten der direktwirkenden sowie der einfach und zweifach reflektierten Komponenten für eine Empfängerhöhe von 2 m dargestellt. Der verwendete Algorithmus ist als Funktion für MATLAB frei verfügbar [41].

Ebenso tragen die Datenverarbeitungseinheit sowie aktivierte Peripherie zur Verlustleistungsaufnahme bei.

Ein Beispiel für den funktionalen Aufbau einer Plattform mit geringer Verlustleistungsaufnahme ist in Abbildung 2.2 am Beispiel des CC430 von Texas Instruments dargestellt. Neben den zum eigentlichen MSP430-Kern gehörigen Elementen (DMA-Controller, Bus Control Logic, Programm- bzw. Arbeitsspeicher, Power Management) und den Schnittstellen (JTAG, I/O Ports sowie „Embedded Emulation Module“), sind im SoC weitere Peripherieeinheiten untergebracht, wie beispielsweise Analog-Digital-Wandler, Echtzeituhr (RTC), Kryptocoprozessor (AES128) und Beschleuniger für CRC-Operationen (CRC16) oder Multiplikationen (MPY32). Die Anbindung der enthaltenen Transceivereinheit erfolgt dabei direkt über die internen Busstrukturen.

2.1.3 Aufbau und Funktion Sende- & Empfängereinheit

Der für die Bitübertragung via Funk zuständige Teil im Kommunikationsknoten gliedert sich in Analog- und Digitalteil. Im Analogteil sind üblicherweise ein- und ausgangsseitige Verstärker, Mischer, Bandpassfilter sowie der „Frequency Synthesizer“ untergebracht. Der notwendige Grundtakt wird dabei in den meisten Fällen aus einem extern platzierten Quarzoszillator bezogen, um die notwendige Frequenzstabilität sicherzustellen. Im Empfangspfad endet der Analogteil mit Analog-Digital-Konvertern,

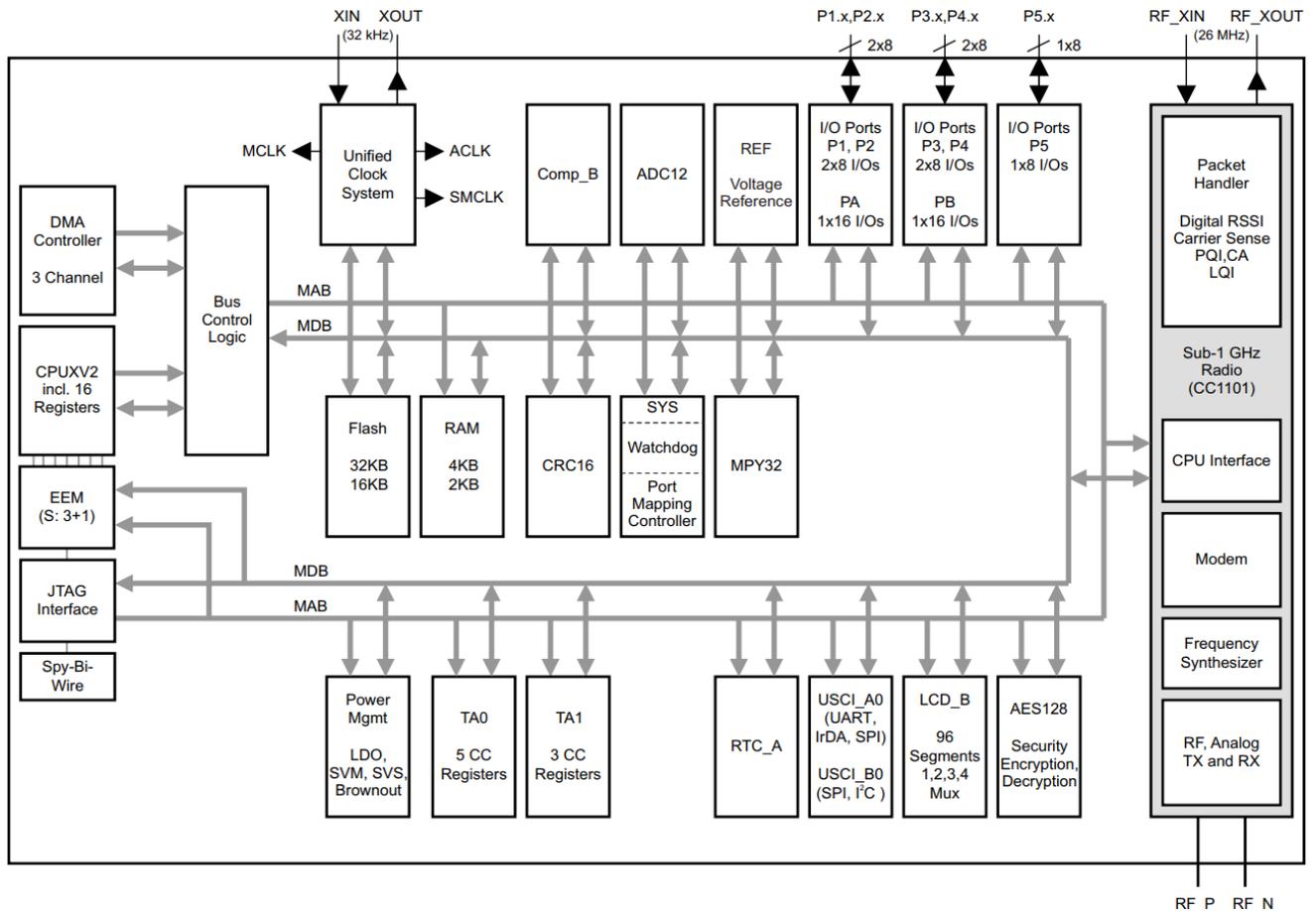


Abbildung 2.2: Funktionales Blockschaltbild des CC430F6137 von Texas Instruments [42]

die das In-Phase- und -Quadratur-Signal digitalisieren. Im Anschluss erfolgt eine automatische Verstärkungsanpassung, die Demodulation sowie die Bit- und Paketsynchronisierung. Das empfangene Paket wird in der Empfangswarteschlange zwischengespeichert und kann über die Schnittstelle zum MSP430-Kern gelesen werden. Zu versendende Pakete werden in der Sendewarteschlange abgelegt, nach erfolgter Modulation werden die I- und Q-Signale mit Hilfe eines spannungsgesteuerten Oszillators direkt synthetisiert.

Der Wechsel zwischen Bereitschafts- („IDLE“), Empfangs- („RX“) und Sendemodus („TX“) erfolgt dabei durch die Übermittlung von Kommandos in einer spezifischen Sequenz. Abbildung 2.4 zeigt alle Betriebszustände der im CC430-SoC enthaltenen Funkeinheit sowie alle möglichen Übergänge dazwischen. Insgesamt lässt sich das physikalische Verhalten der Funkeinheit als universell gültiges Schema durch notwendige Umschaltzeiten sowie zustandsabhängige Verlustleistungsaufnahmen beschreiben. Ist die Einheit als externes Modul ausgeführt, sind hier die Kommunikationsverzögerungen zur Übermittlung von Kommandos, Statusinformationen oder Daten zu berücksichtigen.

2.1.4 Medienzugriffsschicht

Die Medienzugriffsschicht erlaubt die gemeinsame Nutzung eines geteilten Kanals. Um Kollisionen zu vermeiden, kann der Zugriff entweder geregelt werden, d.h. jeder Teilnehmer erhält den Zugriff für eine bestimmte Zeit, oder er wird wahlfrei gewährt. Letzteres ist besonders dann sinnvoll, falls der Datenverkehr von spontaner Natur ist. Häufig kommt hier das CSMA/CA-Verfahren zum Einsatz.

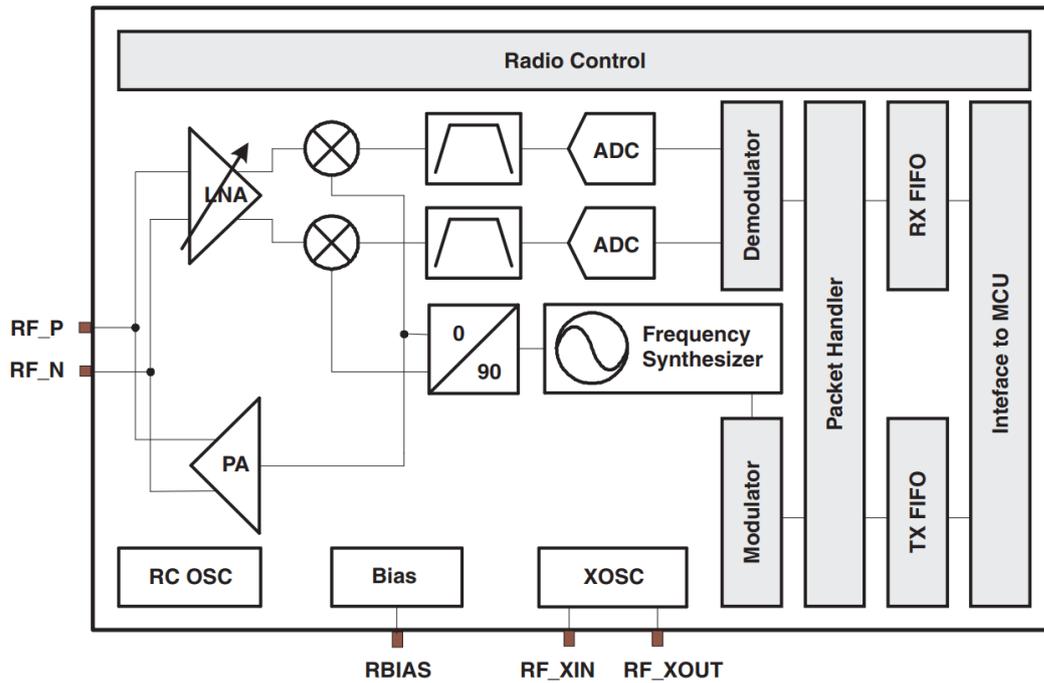


Abbildung 2.3: Aufbau einer typischen Transceivereinheit am Beispiel des im SoC CC430 integrierten Funkmoduls RFA1/CC1101 [43]

Tabelle 2.2: Auflistung berücksichtigter Protokolle für die Medienzugriffsschicht

Name	Kanalzugriff	Initiator	Zeitsynchronisation	Besonderheit
IEEE 802.15.4 RI-MAC	wahlfrei oder garantiert wahlfrei	Sender Empfänger	notwendig notwendig	Bandbreitengarantie möglich energetisch günstig für Empfänger

Die algorithmischen Eigenschaften des Medienzugriffs sind seit Jahren Gegenstand der Forschung, da viele verschiedene Optimierungskriterien existieren, beispielsweise die maximale Ausnutzung der verfügbaren Kanalkapazität, die faire Verteilung der zur Verfügung stehenden Bandbreite oder die unbedingte Einhaltung von Gütekriterien (Latenz, garantierte Datendurchsatzraten). Im Laufe der Jahre wurden deshalb zahlreiche MAC-Varianten veröffentlicht. Huang et al. geben in [44] eine Übersicht über relevante Protokolle sowie deren zeitliche Entwicklung. Zusätzlich sei hier auf das „comatose“-Projekt [22] verwiesen, welches gegenwärtig ca. 327 Beispiele für Medienzugriffsprotokolle auflistet.

Die im Rahmen der durchgeführten Untersuchungen implementierten und analysierten Medienzugriffsverfahren sind in Tabelle 2.2 aufgeführt. Verfahren mit fest zugeordneten Zugriff fanden keine Berücksichtigung.

IEEE 802.15.4

Da sich die Standards IEEE 802.11 sowie Bluetooth für die Ende der 1990er Jahre aufkommende Klasse von kleinen, im nahen Umfeld drahtlos kommunizierenden Geräten als zu komplex erwiesen, wurde 2006 die erste Version des Standards IEEE 802.15.4 veröffentlicht [45]. Dieser definiert sowohl die Bitübertragungsschicht als auch den Medienzugriff, wobei die logische Verbindungskontrolle (LLC) nicht vorgeschrieben ist.

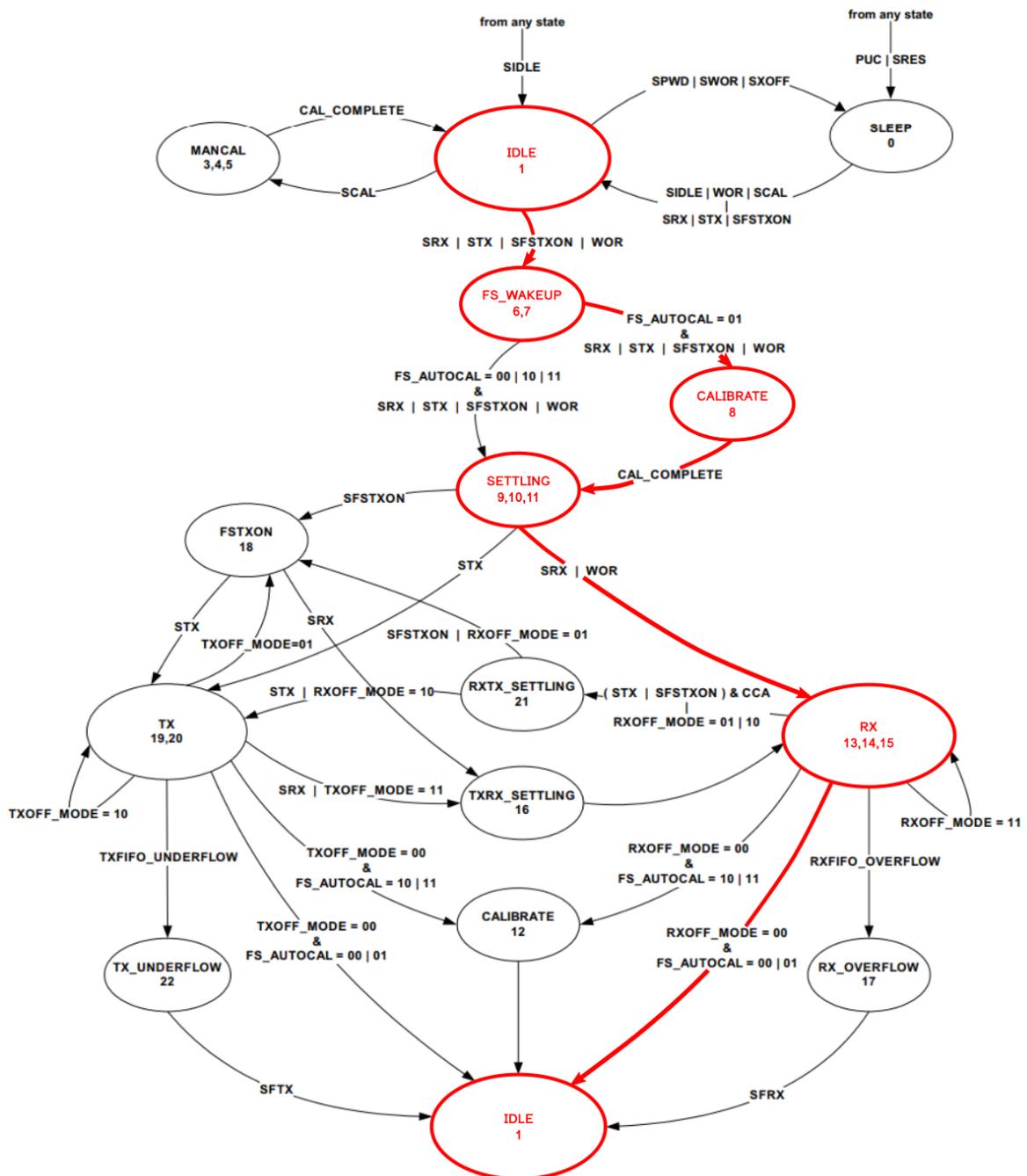


Abbildung 2.4: Darstellung der Betriebszustände sowie möglicher Zustandsübergänge der im SoC CC430 integrierten Funkeinheit RFA1/CC1101: Um Empfangsbereitschaft herzustellen, während sich die Funkeinheit im Bereitschaftszustand „IDLE“ befindet, muss zunächst der dafür nötige Frequenzsynthesizer gestartet werden, was durch einen Wechsel in den Zustand „FS_WAKEUP“ geschieht. Anschließend kann eine interne Kalibrierung des Basistakts erfolgen („CALIBRATE“) und danach muss ausreichend Zeit eingeräumt werden, damit die Schwingkreise stabil einschwingen können („SETTLING“). Erst danach kann in den Empfangsbetrieb („RX“) gewechselt werden. Nach Erhalt der erwarteten Nachricht oder um den Leistungsbedarf zu senken, kann im Anschluss ein direkter Wechsel zurück in den Bereitschaftszustand erfolgen [43].

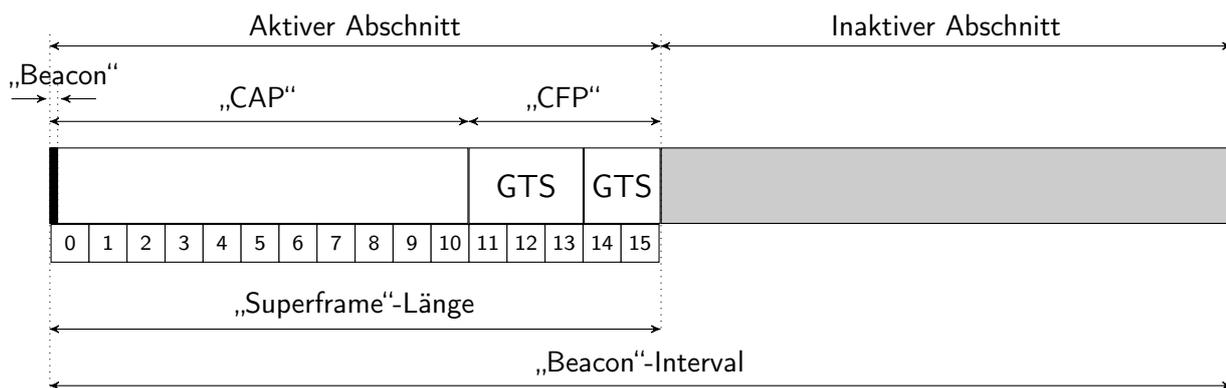


Abbildung 2.5: Struktur des in IEEE 802.15.4 definierten „Superframes“: Mit Hilfe des „Beacons“ wird der Beginn des „Superframes“ angezeigt und die Teilnehmer des Netzwerks werden synchronisiert. Danach beginnt der Zeitraum des konkurrierenden wahlfreien Zugriffs (CAP). Dieser ist in 16 Zeitschlitz („Slots“) unterteilt, was die Wahrscheinlichkeit von trotz des CSMA/CA-Verfahrens auftretenden Kollisionen reduziert. Teilnehmer können beim sog. PAN-Koordinator bis zu sieben Zeitschlitz mit exklusivem Zugriff beantragen, wodurch ein Teil der „CAP“, zur Verfügung gestellt wird, um die CFP in notwendiger Länge einzurichten. Das Verhältnis von „Superframe“-Länge zu „Beacon“-Intervall legt den netzwerkweiten Schlaf-Wach-Zyklus fest und hat somit einen erheblichen Einfluss auf den Leistungsbedarf der Teilnehmer.

Zur Kollisionsvermeidung wird ein CSMA/CA-Verfahren eingesetzt, d.h. vor dem Senden wird zunächst geprüft, ob der Kanal aktuell belegt ist. Als Indikator dient hierfür die vom Empfänger verzeichnete Empfangsfeldstärke. Falls der Kanal frei ist, kann unmittelbar gesendet werden, ist er belegt, wird die Kanalüberprüfung nach einer zufälligen Wartezeit wiederholt (Back-off). Sollte der Kanal später abermals belegt sein, wird die obere Grenze des Intervalls aus dem die Wartezeit zufällig ermittelt wird exponentiell vergrößert. Nach einer konfigurierbaren Maximalanzahl von erfolglosen Kanalüberprüfungen, wird der Übertragungsversuch abgebrochen.

Falls das Netzwerk durch einen sogenannten PAN-Koordinator zur Verfügung gestellt wird, erfolgt eine Synchronisation aller Teilnehmer auf Basis von „Beacon Frames“, die alle Netzwerkteilnehmer empfangen müssen. Die sich an die Übertragung des „Beacon Frames“ anschließende Phase des wahlfreien Kanalzugriffs („Contention Access Period/CAP“), kann zugunsten von Datendurchsatz garantierenden Zeitschlitz reduziert werden („Contention-free Period/CFP“). Ebenso kann eine verbindliche Inaktivitätszeit angezeigt werden, wodurch sich die Verlustleistungsaufnahme aller Teilnehmer deutlich reduzieren lässt, da sie während dieser Phase in einen energetisch günstigen Schlafzustand wechseln können.

Als ein insbesondere für drahtlose Sensornetze hochrelevanter Standard, finden sich zahlreiche Untersuchungen der Leistungsfähigkeit von IEEE 802.15.4 [46], [47], [48]. Es wird einerseits empfohlen, dass bei großen Netzwerken mit einem zu erwartenden Datendurchsatz nahe der Kanalkapazität, die obere Grenze des Wartezeitintervalls von Beginn an groß bzw. maximal gewählt wird. Ein anderes Ergebnis ist, dass der Mechanismus der Bereitstellung garantierter Bandbreite mittels fest zugewiesener Zeitschlitz besonders bei geringem Datenaufkommen sehr verschwenderisch agiert.

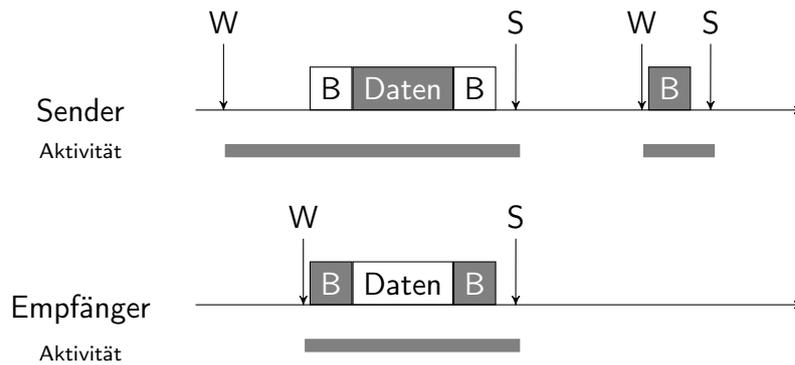


Abbildung 2.6: Funktionsweise von RI-MAC: Jeder Knoten erwacht periodisch aus dem Schlafzustand (W) und versendet ein sog. „Beacon“-Paket, welches die Empfangsbereitschaft signalisiert. Nach Empfang durch einen wartenden Knoten, beginnt dieser mit der Datenübertragung. Werden nach einer bestimmten Wartezeit keine Daten empfangen, erfolgt ein Wechsel zurück in den energetisch günstigen Schlafzustand (S).

RI-MAC

Der bereits 2008 von Sun, Gurewitz und Johnson in [49] vorgestellte „Receiver-initiated MAC“ ist ein asynchrones Medienzugriffsverfahren mit adaptierbarem Schlaf-Wach-Zyklus („Duty-Cycle“). Der zugrundeliegende Ansatz ist die Entkopplung der Schlaf-Wach-Zyklen von Sender und Empfänger. Der Sender zeigt dem Empfänger den Übertragungswunsch an und verbleibt anschließend empfangsbereit. Der Empfänger übermittelt nach dem Erwachen aus dem Schlafzustand ein kurzes „Beacon“-Paket zur Indikation der Empfangsbereitschaft, woraufhin der Sender unmittelbar mit der Übertragung beginnt. Insgesamt verschiebt sich die energetische Belastung so zu Lasten des Senders und die maximal ausnutzbare Kanalkapazität wird vergrößert, da das bei asynchronen Medienzugriffsverfahren sonst notwendige Versenden einer langen Präambel entfällt. Der sich ergebende Kommunikationsablauf ist in Abbildung 2.6 dargestellt. Das vom Empfänger versendete „Beacon“-Paket wird auch genutzt, um den fehlerfreien Empfang der Daten zu bestätigen, wodurch der Sender erneut aufgefordert wird Daten zu übermitteln. Sollte es zu einer Kollision kommen, da mehrere Sender gleichzeitig mit der Übertragung beginnen, wird dies vom Empfänger detektiert und mit der Übermittlung eines „Beacon“-Pakets mit vergrößertem angezeigtem Wartezeitintervall quittiert. Die Sender warten dann für eine aus dem Wartezeitintervall zufällig bestimmte Zeit und beginnen erneut mit der Übertragung.

2.1.5 Vermittlungsschicht

Befinden sich Absender und Empfänger nicht in gegenseitiger Empfangsreichweite, müssen Datenpakete weitergeleitet werden. Diese Aufgabe übernehmen Routingprotokolle, wobei je nach Verfahren unterschiedlich Metriken zur Bewertung gefundener Routen zum Einsatz kommen. Die Protokolle lassen sich je nach Ansatz in statische oder adaptive/dynamische Verfahren sortieren. Vor dem Hintergrund ausfallbedingter (z.B. durch Erschöpfung der Batterie) oder durch mobile Teilnehmer verursachter Topologieveränderungen sind adaptive Verfahren deutlich zu bevorzugen. Diese gliedern sich wiederum in proaktive und reaktive Verfahren. Während proaktive Protokolle versuchen, dauerhaft aktuelle Routen zu allen möglichen Empfängern vorzuhalten, was neben dem kontinuierlichen Datenverkehr insbesondere auch eine kontinuierliche energetische Belastung bedeutet, startet die Routenfindung bei reaktiven Verfahren erst im Augenblick des Übertragungswunsches. Die Routenfindung kann jedoch ergänzend auch auf Basis einer lokalen Routingtabelle erfolgen, wodurch vergrößerte Latenzen häufig

Tabelle 2.3: Auflistung berücksichtigter Routingalgorithmen

Name	Ansatz	Struktur	Routingmetrik	Anzahl Routen
„Ad-hoc On-Demand Distance Vector Routing“	Reaktiv	hierarchisch	kürzeste Distanz	1
Fluten	Reaktiv	–	–	n
„Minimal Energy Relay Routing“	Reaktiv	heterarchisch	geringste Energie	1
„Energy-efficient Multipath Routing“	Reaktiv	hierarchisch	geringste Energie	n

reduziert oder gänzlich eliminiert werden können. Im Kontext der vorliegenden Arbeit wurden aufgrund der genannten Vorteile nur reaktive Verfahren betrachtet. Eine Auflistung der implementierten und simulativ berücksichtigten Verfahren ist in Tabelle 2.3 gegeben.

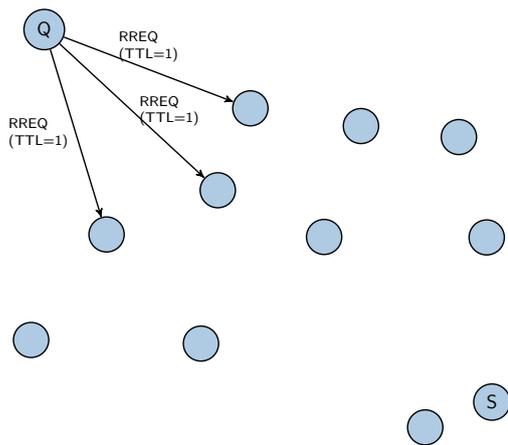
„Ad-hoc On Demand Distance Vector Routing“

Bei dem durch RFC 3561 [50] definierten „Ad-hoc On-Demand Distance Vector Routing“ (AODV) handelt es sich um ein dynamisches, topologiebasiertes und reaktives Routingverfahren, d.h. Routen werden nur nach Bedarf und nicht a priori ermittelt. Bereits ermittelte Routen werden in einer innerhalb jedes Knotens verwalteten Routingtabelle erfasst und mit einer Gültigkeitsdauer versehen nach deren Ablauf die Route als inaktiv bzw. invalidiert gekennzeichnet wird. Nur aktive Routen können zur Weiterleitung von Paketen genutzt werden.

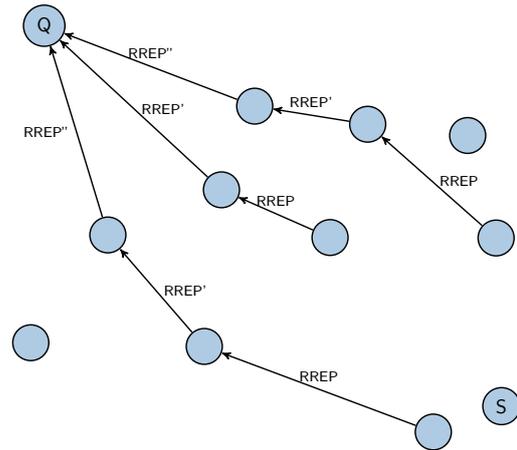
Ist die Route zu einem Zielknoten noch unbekannt oder bereits inaktiv, wird via Broadcast ein sog. „Route Request“-Paket (RREQ) an alle Knoten in Funkreichweite übermittelt, welches neben den Netzwerkadressen von Sender- und Empfängerknoten jeweils eine Sequenznummer für Ziel und Ursprung der Anfrage enthält. Mit Hilfe dieser Sequenznummern wird zum einen die Schleifenfreiheit sichergestellt, zum anderen erfolgt durch die Höhe der vor dem Absenden eines RREQs inkrementierten Zielsequenznummer eine zeitliche Ordnung. Falls zu einem Ziel alternative Routen existieren, ist der sendende Knoten verpflichtet die mit der höheren Sequenznummer zu nutzen. Erreicht das RREQ-Paket entweder den Zielknoten oder einen Knoten, welcher eine Route zum Zielknoten kennt, die aktiv und deren assoziierte Sequenznummer größer gleich der im RREQ-Paket enthaltenen ist, wird ein sog. „Route Reply“-Paket (RREP) als Antwort erzeugt und via Unicast an den Absenderknoten versendet. Um eine unnötig weite Verbreitung von RREQs zu verhindern, findet eine expandierende Suche statt, deren Reichweite über den Wert TTL gesteuert wird. Mit jedem Fehlschlag der Routenermittlung erhöhen sich TTL-Wert und Zeitbeschränkung und somit die Reichweite des ausgesendeten „RREQ“-Pakets, bis schließlich der Maximalwert von 35 erreicht ist. Da es durch die so vergrößerte Reichweite sein kann, dass der antwortende Knoten das „RREQ“-Paket in Folge einer Weiterleitung durch einen anderen Knoten erhalten hat, speichert jeder weiterleitende Knoten den Absender von dem er es empfangen hat, den sog. „last hop node“. Auf Basis dieser Information erfolgt die Beantwortung der Anfrage mittels „RREP“-Nachricht, falls der Zielknoten selbst oder ein Knoten in unmittelbarer Nachbarschaft des Zielknotens erreicht wurde. Nach Empfang des „RREP“-Pakets durch den Urheber der dazugehörigen Anfrage, quittiert dieser durch Aussenden eines „RREP-ACK“-Pakets entlang der nun bekannten Route.

Die fortwährende Aufrechterhaltung bzw. Auffrischung der lokalen Konnektivität kann durch eine optionale Mindestaktivität erreicht werden. Der Erhalt einer hierfür versendeten „HELLO“-Nachricht bewirkt eine Verlängerung der Lebensdauer im Fall einer bereits hinterlegten Route oder sorgt für einen neuen Eintrag in der Routingtabelle.

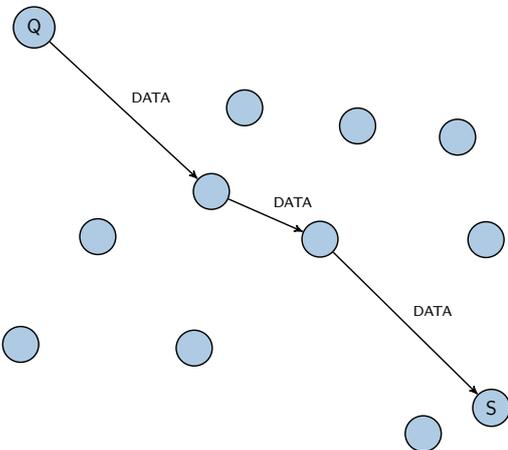
Sollte es im Laufe der Betriebsdauer des Netzwerks zu Unterbrechungen kommen, beispielsweise dem Ausbleiben des Empfangs von „HELLO“-Nachrichten eines Nachbarknotens, können betroffene



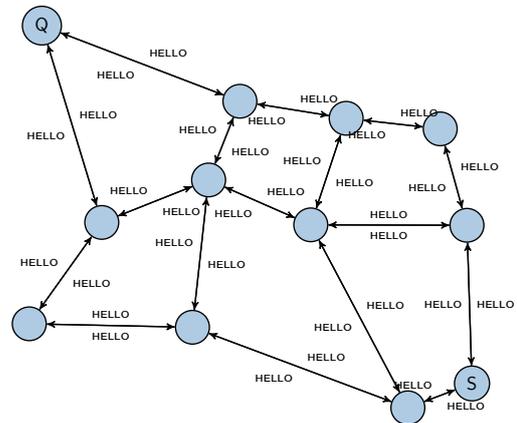
(a) Reichweitenbegrenzte Routenfindung mittels erster „RREQ“-Nachricht



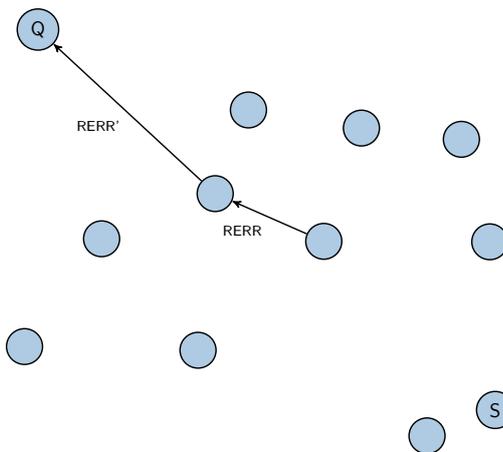
(b) Knoten in unmittelbarer Nachbarschaft zur gesuchten Senke S antworten mit „RREP“-Nachricht



(c) Übermittlung der Nutzdaten entlang der ermittelten Route



(d) Aktualisieren der Nachbarschaftsbeziehungen mittels „HELLO“-Nachrichten



(e) Signalisierung eines Verbindungsverlusts mit einem Nachbarn

Abbildung 2.7: Routenfindung mittels AODV

Routen durch Aussenden einer sog. „Route Error“-Nachricht (RERR) invalidiert werden. In diesem Fall werden folgende Schritte unternommen:

1. Betroffene Routen lokal invalidieren
2. Betroffene Ziele auflisten
3. Feststellen, welcher der Nachbarn betroffen sein könnten
4. Fehlerinformation mittels RERR an diese Nachbarn weiterleiten

Dieses Verhalten stellt sicher, dass beim Erhalt eines RREQs keine RREPs versendet werden, die später zum Scheitern der Übermittlung von Anwendungsdaten führen würden. Weiterhin kann ein Knoten, der eine solche Unterbrechung detektiert, versuchen die Route zu reparieren. Das hierfür festgelegte Verfahren ist im Abschnitt 6.12 „Local Repair“ in RFC 3561 [50] beschrieben und beinhaltet eine lokal beschränkte Suche nach Alternativrouten mittels reichweitenbegrenzter RREQs.

Insgesamt stellt AODV durch seine Eigenschaften ein Verfahren dar, welches sich insbesondere für Netzwerke mit nur geringem Datenaufkommen sowie variabler Topologie eignet. Energetische Aspekte, wie beispielsweise die verbleibende Batteriekapazität der bei der Datenweiterleitung involvierten Knoten, finden bei diesem Verfahren jedoch keine Berücksichtigung. In einer 2007 durchgeführten Untersuchung [51], bei der das Verhalten von AODV in Kombination mit IEEE 802.15.4 simulativ analysiert wurde, konnte jedoch gezeigt werden, dass ein Zusammenhang zwischen Verzögerung und Energiebedarf besteht und – vor allen Dingen in Gegenwart mobiler Knoten – der Energiebedarf pro übertragenem Bit durch Akzeptanz einer höheren Verzögerung sowie die Übertragung größerer Pakete reduziert werden kann. 2010 wurde AODV in einer weiteren Untersuchung im Vergleich mit den „Dynamic Source Routing“ (DSR) und „Temporary Ordered Routing Algorithm“ (TORA) die im Allgemeinen größte Leistungsfähigkeit attestiert [52].

Fluten

Das mit Abstand simpelste Verfahren um Daten netzwerkweit zu verteilen ist das Fluten. Hierbei werden Daten als Broadcast versendet und von jedem Knoten nach Erhalt wiederum als Broadcast verschickt. Zur Vermeidung oszillierenden Verhaltens, enthalten die in der Vermittlungsschicht versendeten Pakete einen Zähler, der vor der erneuten Übertragung dekrementiert wird. Diese sog. „Time-to-Live“, kurz TTL, sollte bei der Erzeugung des zu versendenden Pakets mit der größten im Netzwerk existierenden Routingdistanz in Hops, d.h. dem Netzwerkdurchmesser, initialisiert werden. Ist diese Information nicht verfügbar, muss als Wert die um eins verringerte Anzahl von Knoten, genutzt werden, was zu einer deutlich gesteigerten Netzwerkbelastung führen kann. Um die Effizienz des Flutens weiter zu verbessern, können Nachrichten identifiziert und in einer zeitlich beschränkten Liste gespeichert werden. Trifft eine bereits bekannte Nachricht ein, wird diese nicht erneut ausgesendet. Die Verbreitung innerhalb des Netzwerks erfolgt also in Form einer Art Wellenfront. Die Datenübertragung mittels des auf diese Weise modifizierten Flutens ist in Abbildung 2.8 dargestellt.

Der von Sasson, Cavin und Schiper 2003 veröffentlichte Ansatz [53] erweitert den Algorithmus um eine fest vorgegebene Weiterleitungswahrscheinlichkeit, wodurch sich die Erfolgsrate bei der Zustellung signifikant erhöhen lässt. Die Robustheit und Simplizität des Ansatzes machen das einfache Fluten zu einer geeigneten Rückfalllösung bzw. zu einer Standardreferenz, es ist dennoch vor allem energetisch ineffizient.

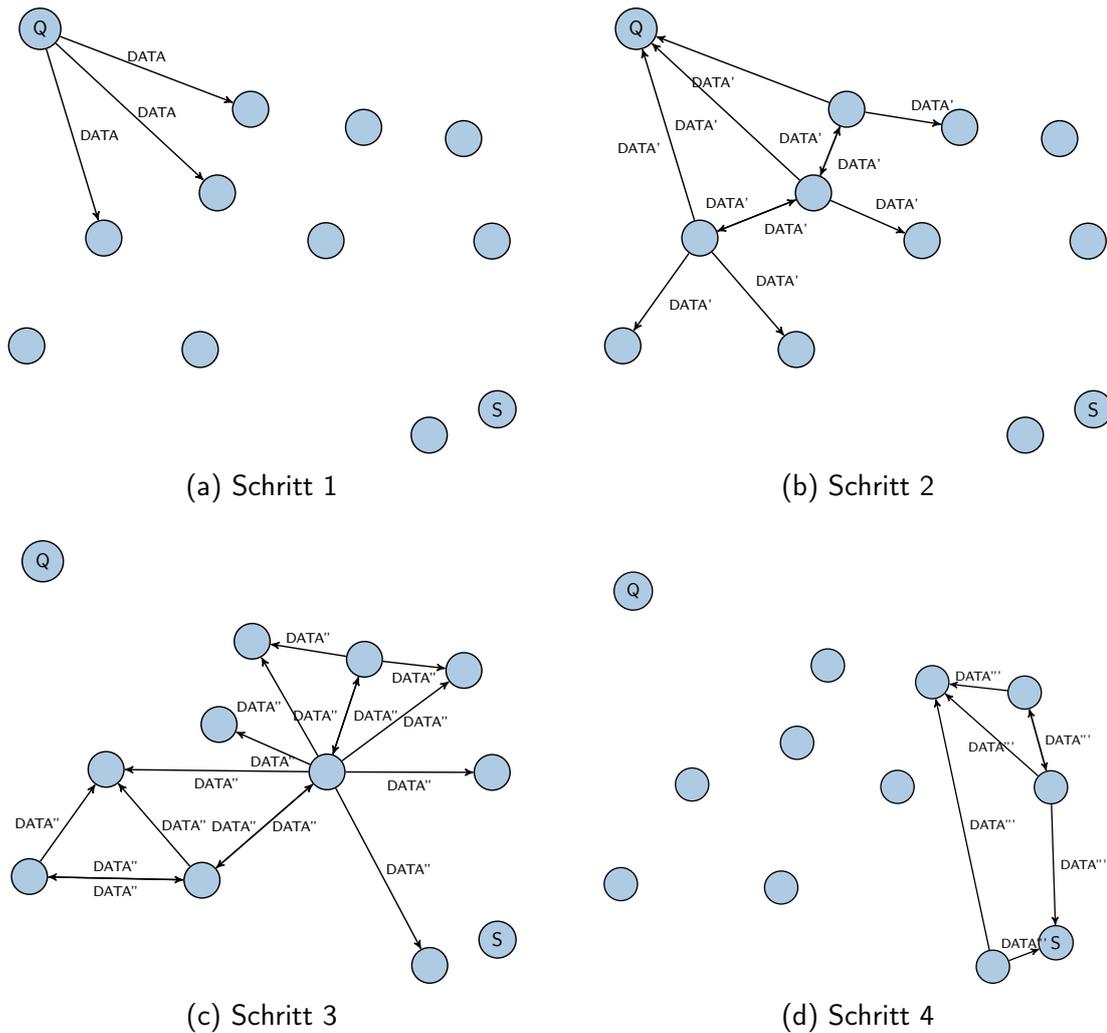


Abbildung 2.8: Verteilen einer Nachricht mit einer TTL von 4 mittels reichweitenbeschränkter Flutens, wobei keine erneute Weiterverbreitung bereits versendeter Nachrichten erfolgt

„Minimal Energy Relay Routing“

Im Bereich der drahtlosen Sensornetze kommt es häufig zu Anwendungsszenarien mit linearen Topologien. Mit Hilfe des „Minimal Energy Relay Routing“-Algorithmus (MERR), wird dabei der Ansatz verfolgt eine Nachricht, unter Kenntnis der Positionen aller Knoten, mit der insgesamt minimalen energetischen Belastung zu übermitteln. Der 2007 von Marco Zimmerling et al. vorgestellte Algorithmus [54], nutzt dazu Wissen über den zur Übertragung verwendeten Transceiver-Baustein, um eine optimale „charakteristische Distanz“ d_{char} zu berechnen, die wie folgt definiert ist

$$d_{char} = \sqrt[\lambda]{\frac{\alpha}{\epsilon(\lambda - 1)}}, \quad (2.17)$$

wobei für die für das Versenden bzw. Empfangen notwendige Energie gilt

$$E_{rx} = \alpha_{rx} \quad (2.18)$$

$$E_{tx} = \alpha_{tx} + \epsilon d^\lambda. \quad (2.19)$$

α bezeichnet hierbei die für die Weiterleitung eines Bits notwendige Energie, d.h. die Summe der pro Bit im Empfangsteil des Transceivers umgesetzten Energie α_{rx} , der im Sendeteil umgesetzten

Energie α_{tx} und der bei einer Übertragung über eine Entfernung d pro Bit im sendeseitigen Leistungsverstärker umgesetzten Energie ϵ (vgl. [55]). λ entspricht dem von den Übertragungsbedingungen abhängigen Pfadverlustexponenten, wie er beispielsweise auch in dem in Abschnitt 2.1.1 beschriebenen Pfadverlustmodell Verwendung findet. Für eine bekannte Gesamtübertragungsdistanz D lässt sich so die optimale Anzahl von Hops gemäß

$$K_{opt} = \left\lfloor \frac{D}{d_{char}} \right\rfloor \text{ or } \left\lceil \frac{D}{d_{char}} \right\rceil \quad (2.20)$$

bestimmen, wobei die optimale Anzahl von Relais $K_{opt} - 1$ beträgt.

Um mittels MERR den nächsten Hop bestimmen zu können, müssen zunächst die HW-abhängige charakteristische Distanz sowie alle Entfernungen zu potentiellen Relaisknoten in Zielrichtung innerhalb der Funkreichweite des Absenders bekannt sein. Anschließend bestimmt jeder Knoten den individuell geeignetsten Relaisknoten und sendet mit der für die Übertragungsdistanz minimal notwendigen Sendeleistung, wodurch die optimale Route angenähert wird. Abbildung 2.9 veranschaulicht die Routenfindung.

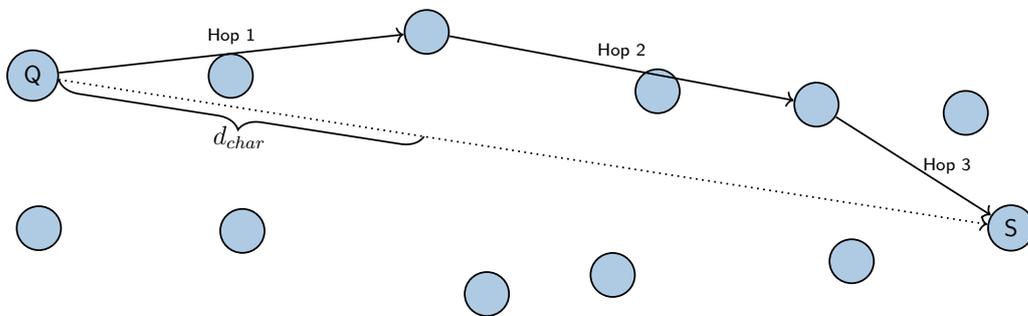


Abbildung 2.9: Routenfindung mittels MERR in einem Funknetzwerk mit vorwiegend linearer Topologie

Um MERR auch für nichtlineare Topologien sinnvoll anwendbar zu machen, muss das Auswahlkriterium für Folgeknoten erweitert werden. Besonders in dünn besetzten Netzwerken, existieren im radialen Abstand der charakteristischen Distanz oftmals nur wenige Knoten als Weiterleitungskandidaten, wodurch sich entweder die Anzahl der Hops erhöht oder ein Knoten gewählt wird, der sich nicht im optimalen Abstand befindet. Das angepasste Entscheidungskriterium erweitert die ursprünglich eindimensionale Abschätzung auf ein zweidimensionales Entfernungsmaß. Primäres Ziel ist es dabei, Knoten mit der geringsten Differenz zu einem Punkt zu bestimmen, welcher sich mit einem Abstand von d_{char} zur Quelle auf der Übertragungsstrecke zwischen Quelle und Senke befindet. Existieren mehrere gleich weit entfernte Knoten, wird derjenige Knoten gewählt, dessen letzte Nutzung als Weiterleitungsknoten am weitesten in der Vergangenheit liegt. So soll die energetische Belastung bei gleicher Eignung über mehrere Knoten verteilt werden. Alternativ könnte eine Optimierung im Bezug auf die für die Übertragung entlang der Route insgesamt aufgewendete Energie stattfinden. Dies würde jedoch globales Wissen über alle aktuellen Knotenpositionen erforderlich machen. Zusätzlich müssten die charakteristischen Distanzen aller Knoten bekannt sein, da diese sich aufgrund unterschiedlicher Sendereinheiten von Knoten zu Knoten unterscheiden können. Neben dem hierfür notwendigen Speicher, muss auch die Berechnung energieoptimaler Routen in den Knoten erfolgen, was besonders im Fall von Ad-Hoc-Routing, den Energiebedarf sowie die Latenz signifikant erhöhen würde.

„Energy-efficient Multipath Routing“

Der 2006 von Lu und Wong [56] vorgestellte Algorithmus EEMRP beruht auf dem Ansatz zur Datenübertragung mehrere unabhängige Pfade zu entdecken und auf Basis der verbleibenden Energiebudgets der involvierten Knoten sowie der resultierenden Anzahl von Hops einen dieser Pfade auszuwählen. Bei der Übertragung einer Nachricht vom Quellenknoten a zur Senke y , wählt a aus der Menge seiner Nachbarknoten N_a denjenigen als nächsten Hop aus, welcher folgende Bedingung erfüllt:

$$\text{nächster Hop} = \arg \min_{b \in N_a} \left\{ (1 - e_{b,\text{residual}}/e_{b,\text{init}})^{[\beta(1-(\Delta d+1)/d_{ay})]} \right\}. \quad (2.21)$$

Dabei bezeichnet d_{ay} die Anzahl der Hops (Routingdistanz) zwischen a und y , respektive d_{by} die Anzahl der Hops zwischen dem potentiellen nächsten Weiterleitungsknoten b und y . Δd ist die Differenz zwischen d_{ay} und d_{by} , $e_{b,\text{init}}$ das initiale Energiebudget des Knotens b und $e_{b,\text{residual}}$ das verbleibende. Der Gewichtungsfaktor β ermöglicht eine Priorisierung der verbleibenden Kapazität im Vergleich zur Hop-Distanz, wobei gilt ($\beta > 1$). Ein großer Wert für β führt zu einer stärkeren Gewichtung der verbleibenden Restkapazität, ein kleiner zu einer stärkeren Gewichtung der Routingdistanz.

Der Algorithmus unterscheidet drei Phasen: Initialisierungsphase, Pfadsuche sowie Datenübertragung und Wartung. Innerhalb der Initialisierungsphase werden sog. „HELLO“-Nachrichten mittels TTL-begrenztem Fluten lokal versendet, deren Format in Abbildung B.2 dargestellt ist. Danach beginnt das netzwerkweite Fluten mit „HELLO“-Nachrichten der Senken, um allen Knoten die verfügbaren Senken bekannt zu machen. Anhand der Verbindung von Sequenznummer und Absenderadresse lässt sich jede „HELLO“-Nachricht eindeutig zuordnen. Wird eine solche empfangen und der Absender ist noch unbekannt, wird dieser in die Liste benachbarter Knoten eingetragen. Dabei wird neben der Absenderadresse auch sein verbleibendes Energiebudget gespeichert. Handelt es sich laut Knotentyp um eine Datensenke und ist der Knoten noch nicht in der Liste bekannter Datensenken enthalten, wird ein neuer Eintrag angelegt. Andernfalls wird die aktuell eingetragene Hop-Distanz überprüft und aktualisiert, falls der in der Nachricht enthaltene Wert geringer ist. Anschließend werden die Felder „Hop Count“, „Forward Node ID“ und „Forward Node Energy Level“ aktualisiert und die Nachricht wird erneut via Broadcast ausgesendet. Mit Hilfe des selektiven Flutens kann jeder Knoten im Netzwerk die kürzeste Routingdistanz zu existenten Senken errechnen und erlangt Kenntnis von den Knoten in seiner Nachbarschaft.

Im Anschluss emittiert jeder Knoten sog. „CONNECTIVITY“-Nachrichten, deren Format in Abbildung B.3 dargestellt ist. Zusätzlich zu den bereits in der „HELLO“-Nachricht vorhandenen Feldern, wird in dem Feld „sink numbers“ die Anzahl bekannter Datensenken übersendet. Die nachfolgenden Felder enthalten jeweils die „Sink ID“ und die dazugehörige Distanz in Hops. Mit diesem Mechanismus werden Informationen über die verfügbaren Senken in der unmittelbaren Nachbarschaft verteilt. Eine netzwerkweite Verteilung durch wiederholte Broadcasts (Fluten) erfolgt nicht.

Müssen nun Datenpakete zu einer Senke übertragen werden, beginnt die Phase der Pfadsuche. Um mehrere möglichst disjunkte Pfade zu bestimmen, wird an jeden Nachbarn eine „REQUEST“-Message übermittelt. Handelt es sich beim Empfänger weder um die gesuchte Senke, noch ist diese ein Nachbarknoten, wird der nächste Hop nach (2.21) ermittelt und die Nachricht an diesen weitergeleitet, wobei die Felder „Path Cost“, „Forward Node ID“ und „Forward Node Energy Level“ aktualisiert werden. Als weitere Einschränkung kann nur ein Knoten als nächster Hop gewählt werden, der noch nicht Bestandteil eines bereits existenten Pfads zwischen Quelle und Senke sein. Quelle und Senke werden dabei mit den bereits in der Routingtabelle vorhandenen Einträgen verglichen und gegebenenfalls eingetragen. Falls keiner der verfügbaren Knoten die genannten Kriterien erfüllt, erfolgt keine Weiterleitung der „REQUEST“-Nachricht.

Empfängt eine Senke eine „REQUEST“-Nachricht, wird zunächst die Quelle identifiziert und falls unbekannt, in die Quellentabelle eingetragen. Daraufhin werden die Routingtabelle mit den Werten aus der „REQUEST“-Nachricht aktualisiert und ein Zeitmesser gestartet, welcher den Zeitraum festlegt innerhalb dessen „REQUEST“-Nachrichten von der gleichen Quelle noch ausgewertet werden. Nach Ablauf der Wartezeit wird jedem der ermittelten Pfade auf Basis seiner Pfadkosten eine Datenrate zugewiesen, die auf diesem Pfad mittels „ASSIGN“-Nachricht an die Datenquelle zurück übertragen wird. Ein Pfad ist dabei immer durch die eindeutige „Route ID“ identifizierbar. Die einem Pfad anteilig an der Gesamtdatenrate R zugewiesene Datenrate r_j errechnet sich dabei wie folgt:

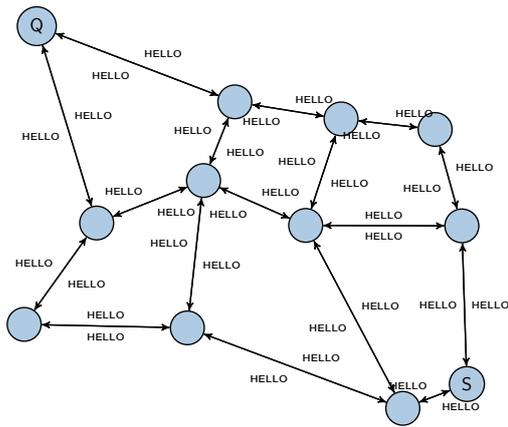
$$r_j = \frac{R}{p_j} \sum_{i=1}^N p_i \quad j = 1, 2, \dots, N \quad (2.22)$$

N entspricht dabei der Anzahl existenter unabhängiger Pfade und p_j bzw. p_i den Pfadkosten bei einer Übertragung entlang der Pfade j bzw. i . Für Hintergrundinformation zur Herleitung von (2.22) sei dabei an dieser Stelle an die Erstveröffentlichung [56] verwiesen.

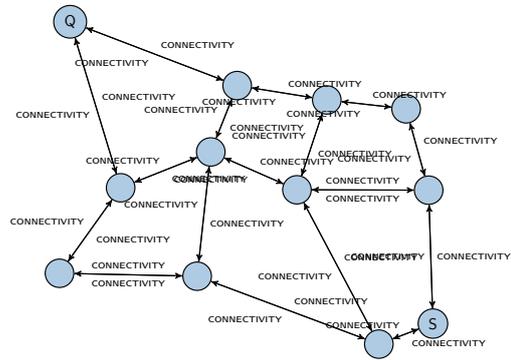
Falls ein weiterleitender Knoten eine „ASSIGN“-Nachricht erhält, wird die Routingtabelle nach Einträgen durchsucht, bei denen „Source ID“, „Sink ID“ und „Route ID“ übereinstimmen und ermittelt den Vorgängerknoten innerhalb des Pfades, an den die Nachricht weitergeleitet wird. Vor dem Absenden werden dabei die Felder „Forward Node ID“ und „Forward Node Energy Level“ aktualisiert. Der Quellknoten trägt in die Routingtabelle die zu Senke und Pfad gehörige Datenrate ein.

Nachdem auf diesem Weg mehrere Pfade ermittelt worden sind, kann die Quelle nun Daten mit der dazugehörigen Datenraten über die Pfade zur Senke übertragen. Dabei kommen sog. „DATA“-Nachrichten zum Einsatz, deren Format in Abbildung B.5 dargestellt ist. Durch die in der Nachricht vorhandenen Felder, können zum einen die noch ausstehende Menge von Daten angezeigt („Data Count“) als auch die bei der Übertragung auftretende Verzögerung errechnet werden („Timestamp“/TS). Weiterhin werden die bei der Übertragung akkumulierten Pfadkosten mitgeführt („Path Cost“). Die Weiterleitung der Nachricht erfolgt dann auf Basis der vorhandenen Routingtabelleneinträge, wobei stets die Pfadkosten stets aktualisiert werden. Dadurch ist es möglich, die sich durch die stärkere energetische Belastung für den aktuell genutzten Pfad ändernden Pfadkosten zu verfolgen und nach Überschreiten eines Schwellwerts die Route anzupassen. Dies wird durch die erneute Übermittlung einer „ASSIGN“-Nachricht von der Senke zur Quelle realisiert. Weiterhin wird die Verzögerung zwischen den eintreffenden Datenpaketen überwacht. Falls diese einen Schwellwert überschreitet, wird angenommen, dass entlang des Pfades Übertragungsprobleme auftreten. Hat sich dadurch die Anzahl verfügbarer Pfade auf höchstens zwei Pfade reduziert, wird die Pfadsuche durch Übermittlung einer sog. „RESET“-Nachricht entlang des unter den verbleibenden optimalen Pfades zur Quelle erneut ausgelöst.

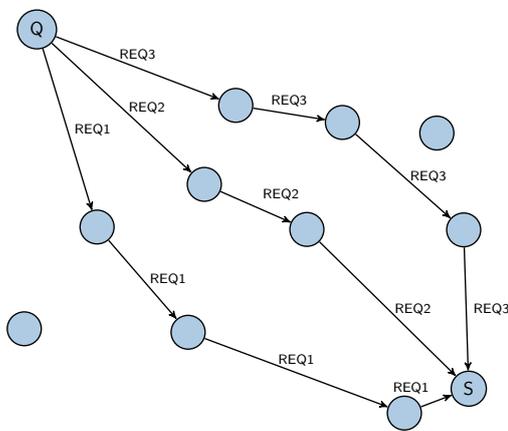
Die Autoren kommen in ihrer Veröffentlichung zu dem Ergebnis, dass sich die Nutzung des beschriebenen Verfahrens vor allen Dingen in größeren Netzwerken sowohl auf die Lebensdauer des Netzwerks als auch die durchschnittliche Leistungsaufnahme der Knoten positiv auswirkt. Es zeigte sich jedoch auch, dass mit zunehmender Netzwerkgröße, die durch Verteilung der routingrelevanten Informationen vor allem während der Initialisierungsphase, die energetische Belastung trotz einer größeren Anzahl von potentiellen Pfaden insgesamt steigt. Ein potentieller Nachteil ist auch die durch das gleichzeitige Nutzen mehrerer Pfade höhere Auslastung des geteilten Übertragungsmediums. Die gesteigerte Kanalbelegung sowie die höhere Interferenzleistung bzw. -wahrscheinlichkeit können das Übertragungsverhalten insgesamt stark negativ beeinflussen.



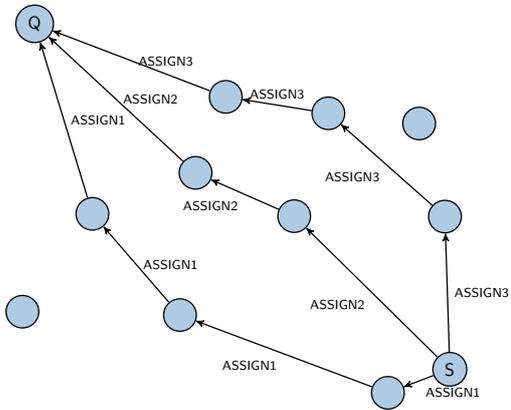
(a) Fluten mit „HELLO“-Nachrichten in der Initialisierungsphase



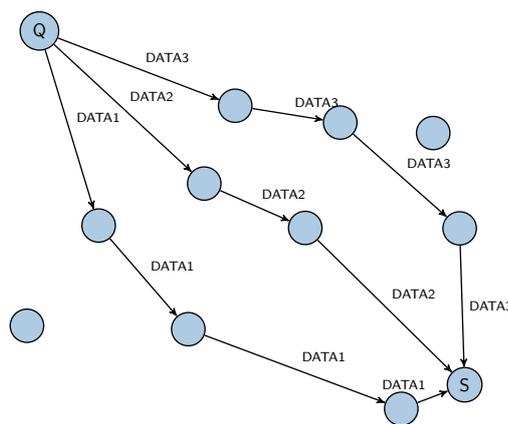
(b) Verbreiten der Kenntnis über Senken sowie deren Entfernung in Hops



(c) Aussenden von Routing-„REQUESTS“ zu ausgewählten Nachbarknoten



(d) Auswahl geeigneter Pfade durch die Senke und Festlegung der dazugehörigen Datenrate mittels „ASSIGN“-Nachrichten



(e) Übertragung der Daten gemäß der vorgegebenen pfadabhängigen Datenraten

Abbildung 2.10: Routenfindung mittels des in Abschnitt 2.1.5 beschriebenen EEMRP-Algorithmus

2.2 Optische Datenübertragung

Eine Alternative zur Übertragung mittels Funk ist die optische Übertragung. In Kommunikationssituationen mit direkter Sichtverbindung eingesetzt, stellt sie aufgrund des unterschiedlichen Spektralbereichs eine zur Funkübertragung alternative Technik dar. Weiterhin ist das genutzte Spektrum von 380 nm bis 750 nm im sichtbaren Bereich bzw. von 750 nm bis 2,5 µm im Nahinfrarotbereich lizenzfrei und bietet eine theoretische nutzbare Bandbreite von 670 THz.

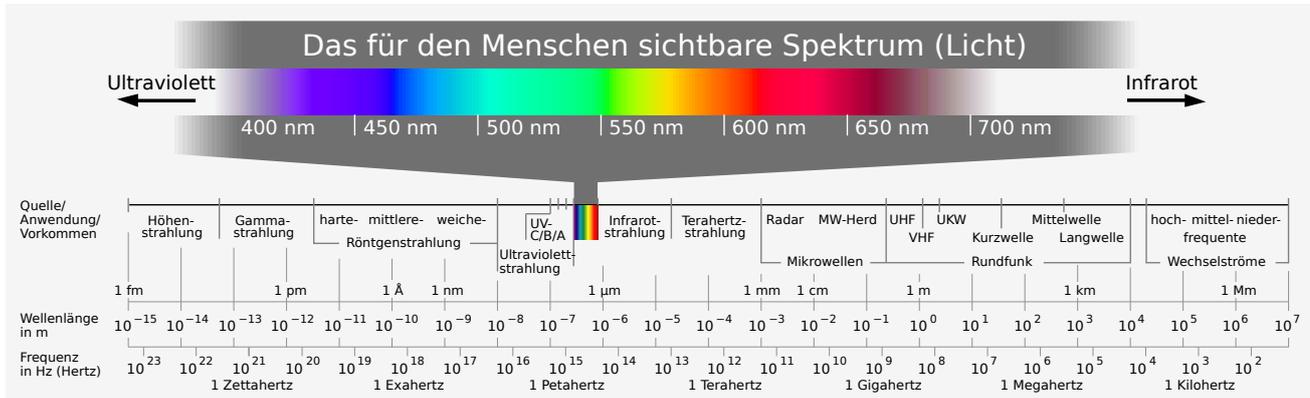


Abbildung 2.11: Elektromagnetisches Spektrum inkl. des für den Menschen sichtbaren Anteils des Lichts [57]

2.2.1 Aufbau eines optischen Datenübertragungssystems

Ein System zur optischen Datenübertragung beinhaltet drei Komponenten: Sender, Kanal und Empfänger. Der Sender nimmt einen Datenstrom entgegen, generiert daraus ein Nutzsignal und verwendet dieses zur Modulation eines Trägers. Das modulierte Signal wird dann an ein optisches Front-End weitergegeben, wo die elektrisch optische Wandlung erfolgt. Das optische Signal erfährt durch den Kanal, je nach Raumgeometrie und Anordnung von Sender und Empfänger, Dämpfung und Verzögerung bevor es auf den Empfänger trifft. Im Empfänger wird das optische Signale mit Hilfe einer Photodiode (oder eines bildgebenden Sensors) und eines Transimpedanzverstärkers (TIA) in ein analoges elektrisches Signal gewandelt. Dieses wird mit Hilfe eines Analog-Digital-Wandlers digitalisiert und demoduliert. Anschließend stellt der Empfänger den Datenstrom zur digitalen Weiterverarbeitung zur Verfügung. Abbildung 2.12 gibt einen Überblick über den beschriebenen Systemaufbau.

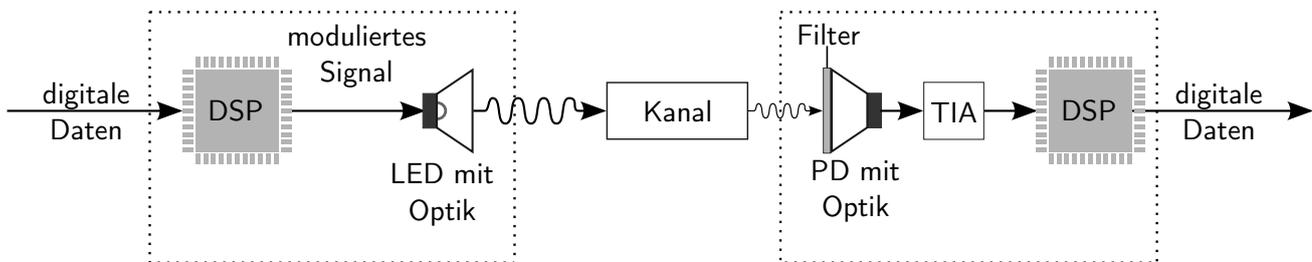


Abbildung 2.12: Aufbau eines Systems zur drahtlosen optischen Datenübertragung

Übertragungsszenarien

Für die optische Datenübertragung gibt es drei mögliche Übertragungsszenarien: direkt, d.h. mit Sichtverbindung, indirekt gerichtet und indirekt ungerichtet bzw. diffus. Diese Szenarien sind zur Verdeutlichung in Abbildung 2.13 dargestellt. Im Fall einer direkten Übertragung mit Sichtverbindung ist der optische Übertragungspfad hindernisfrei und Sender und Empfänger sind gemäß ihrer Abstrahlcharakteristiken bzw. ihres „Field of View“ (FOV) zu einander ausgerichtet. Versperrt ein Hindernis die direkte Sichtverbindung, kann die Übertragung auch indirekt und gerichtet erfolgen. Hat der Sender einen vergleichsweise kleinen Abstrahlwinkel, können Sender und Empfänger so ausgerichtet werden, dass eine starke Einfachreflektion an einer Decke oder Wand genutzt werden kann. Bei großem Abstrahlwinkel kann keine starke Einfachreflektion mehr erzielt werden. Stattdessen erreicht das Licht den Empfänger nach einer oder mehreren Reflektionen an den sich im geschlossenen Raum befindlichen Oberflächen. Zusätzlich kann eine ambiante Komponente zur Übertragung beitragen.

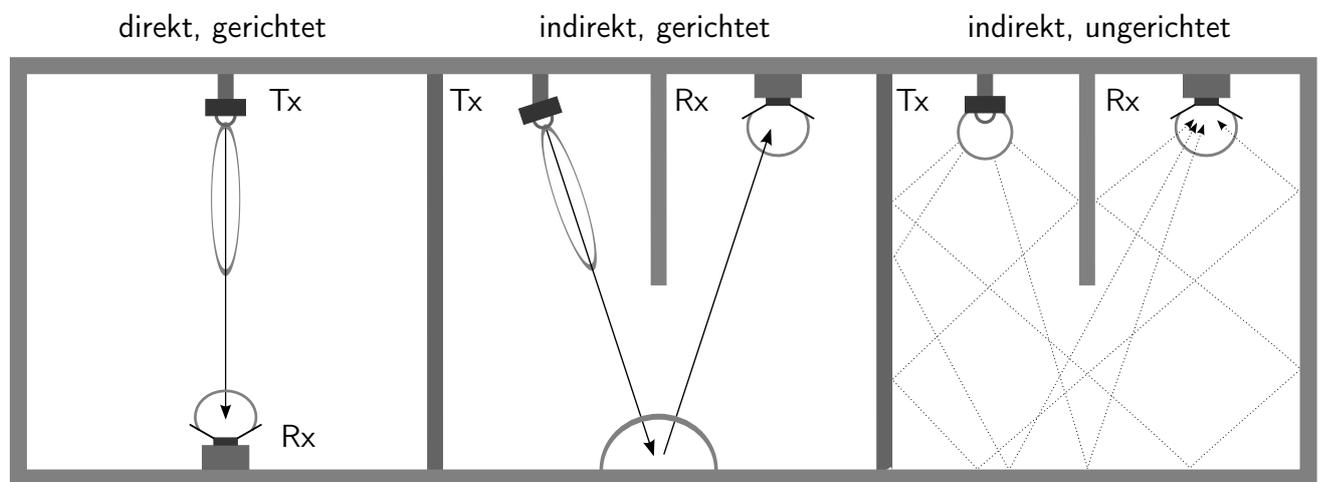


Abbildung 2.13: Mögliche Szenarien für die drahtlose optische Datenübertragung im Innenbereich

Sender zur optischen Datenübertragung

Der Sender beinhaltet neben einem Modulator ein sog. „Optical Front-End“. Es besteht aus elektrischen und optischen Komponenten. Zur Wandlung des modulierten Signals in ein Lichtsignal kommen zumeist LEDs [58], in bestimmten Anwendungsfällen aber auch Laser-Dioden [59], zum Einsatz. Diese lassen sich durch folgende Charakteristika beschreiben:

- Strahlungsintensität I_E (in mW sr^{-1})
- mittlere Wellenlänge λ_p
- spektrale Bandbreite $\Delta\lambda$
- FOV/Abstrahl- bzw. Öffnungswinkel θ_{FOV} oder $2\theta^{1/2}$
- Leistungsaufnahme P_V bzw. Wirkungsgrad η

Die Abstrahlungscharakteristik bzw. das Strahlungsmuster R lässt sich mit Hilfe des Lambertschen Gesetzes beschreiben:

$$R_{\text{LED}}(\theta, n) = \frac{n+1}{2\pi} \cos^n(\theta) P_{\text{LED}} \quad (2.23)$$

wobei $-\pi/2 \leq \theta \leq \pi/2$ (Zenithwinkel) und P_{LED} die insgesamt abgestrahlte optische Leistung angibt. n entspricht der Modenzahl der Abstrahlungskeule und ist in Abhängigkeit des Öffnungshalbwinkels β definiert

$$n = \frac{\log(1/2)}{\log(\cos(\beta))} \tag{2.24}$$

In Abbildung 2.14 sind die sich ergebenden Strahlungsmuster exemplarisch für drei LEDs mit einem Abstrahlwinkel von 10° , 20° respektive 60° dargestellt.

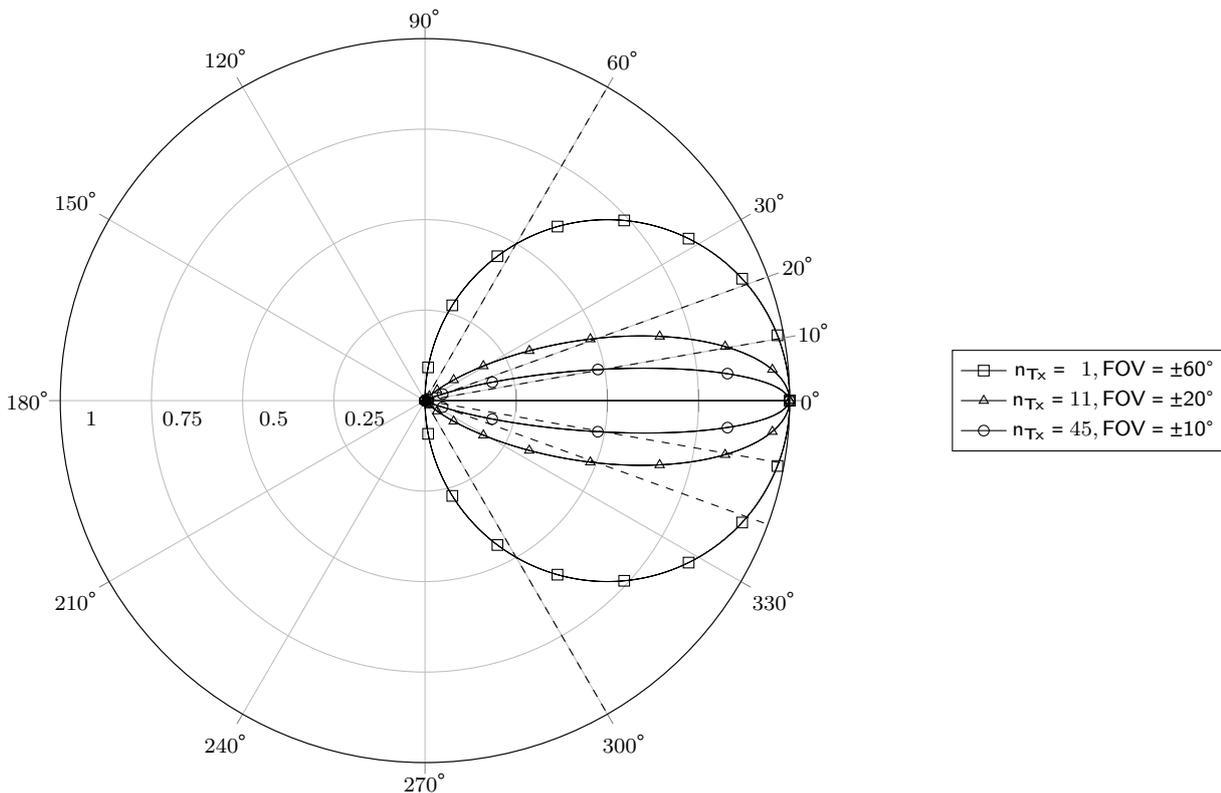


Abbildung 2.14: Normalisiertes Lambertsches Abstrahlverhalten für Öffnungswinkel von $\pm 10^\circ$, $\pm 20^\circ$ und $\pm 60^\circ$

Die Strahlungsintensität des eingesetzten „Optical Front-Ends“ ist durch die europäische Norm BS EN 62471:2008 „Photobiologische Sicherheit von Lampen und Lampensystemen“ auf 100 W m^{-2} begrenzt (Messentfernung 0,2 m; Expositionsdauer von 1000 s). Aktuelle LEDs erreichen Leistungsaufnahmewerte von ca. $12,5 \text{ W}$ bei einem Öffnungswinkel von 115° [60], so dass kritische Strahlungsintensitäten nur durch den Einsatz von stark bündelnden Optiken erreichbar sind.

Durch die große zur Verfügung stehende Bandbreite besteht zusätzlich die Möglichkeit mehrere LEDs mit überlappungsfreien Spektren parallel zu verwenden, um die zur Übertragung nutzbare Bandbreite zu vergrößern.

Kanalmodell

Die auf die Optik des Empfängers treffende Lichtleistung ist abhängig von der senderseitigen Lichtleistung und der Übertragungsgeometrie, d.h. dem Übertragungskanal. Barry und Kahn haben bereits 1994 gezeigt, dass es sich um ein lineares, zeitinvariantes und gedächtnisloses System handelt.[61] Der

sich ergebende Pfadverlust kann mit Hilfe eines von Gfeller und Bapst beschriebenen analytischen Modells für Szenarien mit direkter bzw. einfach reflektierter indirekter Sichtverbindung (vgl. Abbildung 2.13) nachempfunden werden [62].

Wie bereits erläutert, wird das Abstrahlverhalten des Senders mit Hilfe des Lambertischen Strahlungsgesetzes beschrieben. Das Sichtfeld des Empfängers lässt sich ebenso modellieren.

Die auf die PD des Empfängers entfallende optische Lichtleistung P_R kann für die genannten Szenarien wie folgt bestimmt werden:

$$\begin{aligned}
 P_R = & P_T \frac{n_{TX} + 1}{2\pi} (\theta_{Tx,d}) \frac{A}{d^2} \cos(\theta_{Rx,d}) \text{rect}(\theta_{Rx,d}) \\
 & + P_T \frac{n_{TX} + 1}{2\pi} \int_x \int_y \int_\theta \int_\phi \cos^{n_{TX}}(\theta_{Tx}) \frac{\rho \mathcal{R}(\theta, \phi)}{d_1^2} \\
 & \times \frac{A}{d_2^2} \cos(\theta_{Rx}) \text{rect}(\theta_{Rx}) dx dy d\theta d\phi.
 \end{aligned} \tag{2.25}$$

$$\text{rect}(\theta) = \begin{cases} 1, & \text{if } \theta \leq \theta_{FOV,Rx} \\ 0, & \text{if } \theta > \theta_{FOV,Rx} \end{cases} \tag{2.26}$$

$$\mathcal{R}(\theta_{inc}, \theta_{inc}) = \cos(\theta_{inc}) \frac{\cos(\theta_{obs})}{\pi}. \tag{2.27}$$

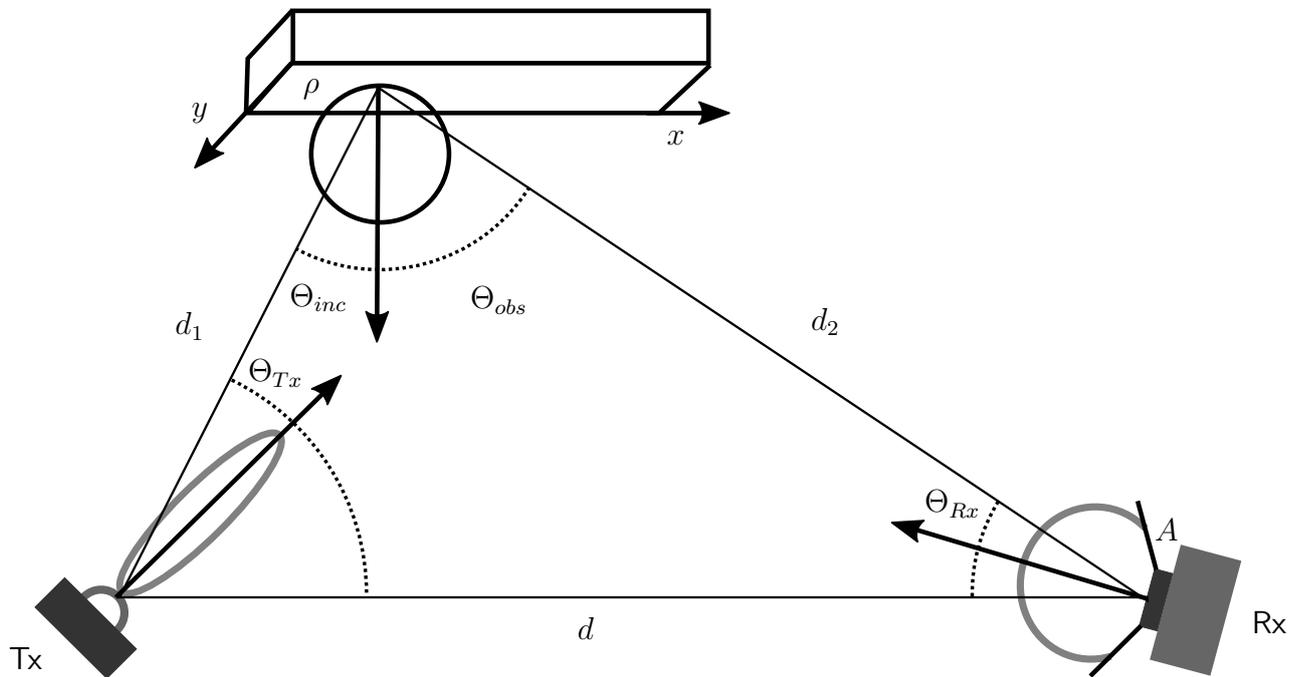


Abbildung 2.15: Geometrie des Übertragungswegs bei einer optisch drahtlosen Übertragung mit einfacher Reflektion (reproduziert aus [63])

P_T bezeichnet die insgesamt ausgestrahlte Lichtleistung, n_{TX} die Lambertische Modenzahl der Abstrahlungskeule der LED, d.h. ihren Öffnungswinkel. Die photoaktive Oberfläche der Photodiode A

steht im Verhältnis zum Abstand bei Direktübertragung mit Sichtverbindung d . Eine Übertragung kann weiterhin nur erfolgen, falls der Einfallswinkel $\theta_{Rx,d}$ kleiner oder gleich dem effektiven Öffnungswinkel der Photodiode ist ($rect(\theta_{Rx,d})$ ist in diesem Fall 1, andernfalls 0). Der zweite Summand beschreibt den Leistungseintrag einer einfach reflektierten Komponente. Ihr Betrag ist die Integration einer bidirektionalen Reflektionsverteilungsfunktion $\mathcal{R}(\theta, \phi)$ in x - sowie y -Richtung und in Abhängigkeit der Ein- und Ausfallswinkel θ und ϕ . ρ bezeichnet den Reflektionskoeffizient der reflektierenden Oberfläche, d_1 und d_2 die Abstände zwischen Oberfläche und LED bzw. PD wie in Abbildung 2.15 dargestellt.

Empfänger zur optischen Datenübertragung

Der Empfänger setzt sich aus Optik, Photodiode, einem Transimpedanzverstärker (TIA) sowie einem digitalen Signalprozessor inklusive Analog-Digital-Konverter zusammen (siehe Abbildung 2.12). Das einfallende Lichtsignal durchläuft zunächst den optischen Filter. Dieser ist optional, erlaubt es aber Licht bestimmter Wellenlängen am Einfallen auf die Photodiode zu hindern. Die nachgeschaltete Optik besitzt, wie die Optik am Sender, einen Öffnungswinkel, wodurch sich der Erfassungsbereich des Empfängers beispielsweise über den Öffnungswinkel der Photodiode hinaus erweitern lässt bzw. mehr Licht auf die PD trifft (Kollimator). Das Licht durchläuft die Optik und trifft auf eine Photodiode.

Durch den inneren photoelektrischen Effekt fließt ein Photostrom gegen die Durchlassrichtung des pn- oder PIN-Übergangs der proportional zur Intensität des einfallenden Lichts ist. Dieser Strom wird mit Hilfe eines Transimpedanzverstärkers verstärkt, anschließend spannungsgewandelt und durch den Digital-Analog-Konverter digitalisiert. Danach kann die Demodulation im Signalprozessor durchgeführt werden, so dass die ursprünglich zu versendenden Daten wieder zur Verfügung stehen.

Die am Empfänger verwertbare Signalbandbreite B wird durch die Bandbreite der Photodiode respektive des TIAs begrenzt, wobei sie im Allgemeinen dennoch größer ist als die Bandbreite des Senders. So erreichten Khalid et al. 2012 Bandbreiten von mehr als 100 MHz [64] was unter Nutzung einer diskreten Multiton-Modulation (DMT) zu einer Verbindungsgeschwindigkeit von ca. 1 Gbit/s führte.

Das ambiente Licht verursacht in der Photodiode intensives „Shot“-Noise (Schrotrauschen). Durch den TIA wird zusätzlich thermisches Rauschen eingebracht. Gemeinsam dominieren diese beiden Rauschquellen das additive Rauschen, welches als mittelwertfreier, reellwertiger gaußscher Zufallsprozess angenommen werden kann [65].

Bei bekanntem Frontend-Eigenschaften, kann die elektrische Leistung des AWGN in Abhängigkeit von der Wellenlänge λ ausgedrückt werden:

$$\sigma_{AWGN}^2(\lambda) = 2q(P_R(\lambda) + P_{bg}(\lambda))S_{PD}(\lambda)G_{TIA}T_{OF}(\lambda)G_{OC}B + 4k_BTB \quad (2.28)$$

Der erste Summand stellt die „Shot“-Noise-Komponente dar, wobei Anteile von der optischen Leistung des Signals sowie der Hintergrundbeleuchtung enthalten sind. Der zweite Summand entspricht der thermischen Rauschleistung. q ist die Elementarladung, $P_R(\lambda)$ entspricht der optischen Signalleistung, die auf den Empfänger einwirkt und P_{bg} ist die optische Leistung der Hintergrundbeleuchtung. Die Ansprechempfindlichkeit der Photodiode wird über S_{PD} ausgedrückt, G_{TIA} repräsentiert die Verstärkung des TIA und T_{OF} die Transmittanz des optischen Filters. Die Bündelungswirkung der eingesetzten Optik bzw. des optischen Konzentrators wird als Verstärkungsfaktor G_{OC} berücksichtigt. Die Leistung des thermischen Rauschens wird mit Hilfe der Boltzmann-Konstante k_B und der absoluten Temperatur T bestimmt.

Zur Rauschreduktion und somit der Verbesserung des Signalrauschverhältnisses (SNR) kann zunächst der optische Filter eingesetzt werden, um den Anteil der einfallenden optischen Hintergrundleistung zu

reduzieren. Weiterhin besteht die Möglichkeit eine Avalanche- bzw. Lawinen-Photodiode einzusetzen, da sie im Vergleich zu herkömmlichen PIN-Photodiode eine inhärente sperrspannungs- und temperaturunabhängige Verstärkung besitzt. Ein rauscharmer TIA mit hohem Verstärkungsfaktor trägt ebenso zur SNR-Verbesserung bei.

Vorwärtsfehlerkorrektur

Eine Möglichkeit, um den Datendurchsatz bei Übertragung über unzuverlässige und rauschbehaftete Kanäle zu erhöhen, ist die Anwendung einer Vorwärtsfehlerkorrektur. Eine typische Variante von FEC sind Reed-Solomon-Codes, die in die Klasse der nichtbinären zyklischen Codes fallen und bereits 1960 eingeführt wurden. Sie werden mit Hilfe der Parameter n und k parametrisiert, wobei n die Anzahl der pro Codewort übermittelten Symbole und k die Anzahl der Datensymbole mit jeweils m Bit angibt. Die verbleibenden $[n - k] = 2t$ Symbole enthalten Paritätsinformationen, welche die Korrektur von bis zu t Symbolen erlauben. Bei bekannter Anzahl von Codewörtern pro Paket sowie bekannter Bit- oder Symbolfehlerrate, kann die Wahrscheinlichkeit für einen nicht korrigierbaren Fehler innerhalb eines Pakets über

$$PER = 1 - s \cdot \sum_{i=0}^t \binom{2^m - 1}{i} p^i (1 - p)^{2^m - 1 - i} \quad (2.29)$$

berechnet werden, wobei s die Anzahl der Symbole pro Paket beschreibt und p der Symbolfehlerrate entspricht.

2.2.2 Genutzte Modulationsverfahren

Ein Vorteil bei der Nutzung von optisch übertragenden Systemen ist die Möglichkeit zur Integration in vorhandene LED-Beleuchtungselemente. Diese erzeugen inkohärentes Licht und eignen sich vornehmlich zur intensitätsbasierten Datenübertragung. Das Signal muss reellwertig, einpolig und nicht negativ sein. Zur Modulation können deshalb bekannte Verfahren wie PWM, PPM, OOK und PAM direkt eingesetzt werden. Sollen höhere Übertragungsraten erzielt werden, müssen aufgrund der durch den frequenzselektiven Kanal hervorgerufenen Intersymbolinterferenz (ISI) andere Modulationsverfahren zum Einsatz kommen. Ein Beispiel hierfür ist OFDM, welches eine individuelle Leistungsanpassung der Frequenzsubbänder erlaubt und so auch eine Anpassung an die Eigenschaften des Übertragungskanal ermöglicht, der sich beim Einsatz von kommerziell verfügbaren LED-Leuchten ergibt.

M-PAM ist als Verfahren für die optische Kommunikation sehr gut geeignet, da es einen hinreichend guten Kompromiss aus Übertragungsrate, Skalierbarkeit sowie einfacher Umsetzbarkeit darstellt. In seiner einfachsten Form, d.h. 2-PAM, entspricht es „On-off keying“ (OOK). Die spektrale Effizienz von M-PAM ist mit $\log_2(M)$ (bit/sec)/Hz [61] gegeben und der elektrische Signalrauschabstand definiert sich wie folgt

$$SNR_{b,elec} = SNR \cdot G_{eq} G_{dc} \quad (2.30)$$

G_{eq} und G_{dc} stellen hierbei den Equalizer- bzw. den DC-Vorspannungsgewinn dar. Um hieraus die Bitfehlerrate abzuleiten, sollten nur die Fehler der ähnlichsten Symbole betrachtet werden. Unter

dieser Maßgabe lässt sich die BER für M-PAM für Werte unterhalb von $BER \leq 10^{-2}$ sehr genau mit folgendem Ausdruck approximieren [66]

$$BER = \frac{N_s}{G_{GC} \log_2(M)} Q \left(d_s \sqrt{\frac{\log_2(M) SNR_{b,elec}}{2}} \right) \quad (2.31)$$

Über $G_{GC} = 1$ ließe sich der Einfluss einer zusätzlichen Gray-Codierung der Symbolbits ausdrücken, während $N_s = 2(M - 1)/M$ die Anzahl benachbarter Symbole ist. $Q(x)$ ist die mittelwertfreie komplementäre kumulative Verteilungsfunktion, die mittels

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp\left(\frac{-t^2}{2}\right) dt \quad (2.32)$$

bestimmt ist. Die Distanz zwischen einem Symbol und dem ähnlichsten störenden Symbol beträgt $d_s = \sqrt{12/(M^2 - 1)}$.

Methoden zur Entwurfsraumexploration für Kommunikationssysteme

Kommunikationssysteme stellen aufgrund ihrer Vielschichtigkeit sowie der zahlreichen vorhandenen konfigurierbaren Parameter komplexe Systeme dar. Um die verschiedenen Konfigurationen im Kontext der Anwendung bewerten zu können, ist es notwendig, diese im Rahmen einer Entwurfsraumexploration zu analysieren. In diesem Kapitel sollen daher die Entwurfsraumexploration sowie einige dazu geeignete Methoden vorgestellt werden.

3.1 Entwurfsraumexploration

Unter Entwurfsraumexploration (engl. „Design Space Exploration“) wird die systematische Analyse des Verhaltens eines komplexen Systems \mathbb{S} in Verbindung mit der Wirkung relevanter Auslegungs-, respektive Betriebsparameterwerte, \mathbb{P} verstanden. Sind diese Beziehungen sowie die Anwendungen \mathbb{A} bekannt, kann eine anwendungsangepasste Auslegung erfolgen [67], [68]. Der Entwurfsraum \mathbb{E} ergibt sich als Tripel:

$$\mathbb{E}(\mathbb{S}, \mathbb{P}, \mathbb{A}) \quad (3.1)$$

Dabei sind \mathbb{S} und \mathbb{A} im Allgemeinen mehrdimensionale Unterräume innerhalb von \mathbb{E} , woraus eine Vielzahl von Realisierungsmöglichkeiten resultiert, die nicht alle sinnvoll sind. Ziel der Entwurfsraumexploration ist es, eine Kombination von Auslegungsparameterwerten zu finden, die allen Randbedingungen (z.B. effektiver Datendurchsatz oder Batterielebensdauer) gerecht wird und im Bezug auf das Optimierungskriterium die beste oder einer zur besten gleichwertige Lösung darstellt [69]. Angewendet auf ein Kommunikationssystem, kann es sich beim Optimierungskriterium beispielsweise um einen möglichst hohe Datenübertragungsrate, eine möglichst geringe Latenz oder um eine besonders niedrige Verlustleistungsaufnahme handeln. Auch die effiziente Nutzung des zur Verfügung stehenden Kanals, respektive Spektrums, kann, besonders in Szenarien mit vielen Kommunikationsteilnehmern, ein Auslegungsziel sein. Beeinflussen lässt sich das Systemverhalten zum einen über die Parametrierung der eingesetzten Protokolle, aber auch durch die Wahl der Protokolle selbst. Darüber

hinaus sind die Eigenschaften der Kommunikationshardware zu berücksichtigen, beispielsweise die maximale Sendeleistung, Aufwachzeiten oder verfügbare Modulationsverfahren.

Existieren mehrere gleichwertige Lösungen, können diese nach Erfüllung der Randbedingungen in lexikalischer Ordnung sortiert werden, d.h. es lässt sich eine Priorisierung mittels Gewichtung der einzelnen Randbedingungen vornehmen, um so zur geeignetsten Systemkonfiguration zu gelangen. Dabei ist es hilfreich die möglichen Lösungen durch Abbildung in einen Bildraum \mathbb{B} mit reduzierter Dimensionsanzahl hinsichtlich ausgewählter Bewertungsmetriken leichter vergleichbar zu machen (Reichweite vs. Durchsatz oder durchschnittliche Verlustleistungsaufnahme vs. Latenz). Darüber hinaus wird mit Hilfe der Entwurfsraumexploration der Einfluss einzelner Parameter auf die Systemperformance untersuchbar. So lassen sich einflussreiche Parameter benennen und allgemeinere Auslegungsregeln definieren.

3.2 Modellierung des Verhaltens von Kommunikationssystemen

Um das Verhalten eines Kommunikationssystems in Abhängigkeit von der gewählten Parametrierung untersuchen zu können, muss dieses modellhaft beschrieben werden. Dabei findet im Allgemeinen ein Abtausch zwischen Modellierungsniveau bzw. Präzision, sowie des Rechenzeit- und Speicherplatzbedarfs, d.h. der Analyseperformance statt. Letztere ist von Bedeutung, da sie erst ab einem bestimmten Maß eine praktikable Exploration des Entwurfsraums ermöglicht. Zur Modellierung des Systemverhaltens lassen sich verschiedene Ansätze nutzen, die im Folgenden näher beschrieben werden sollen.

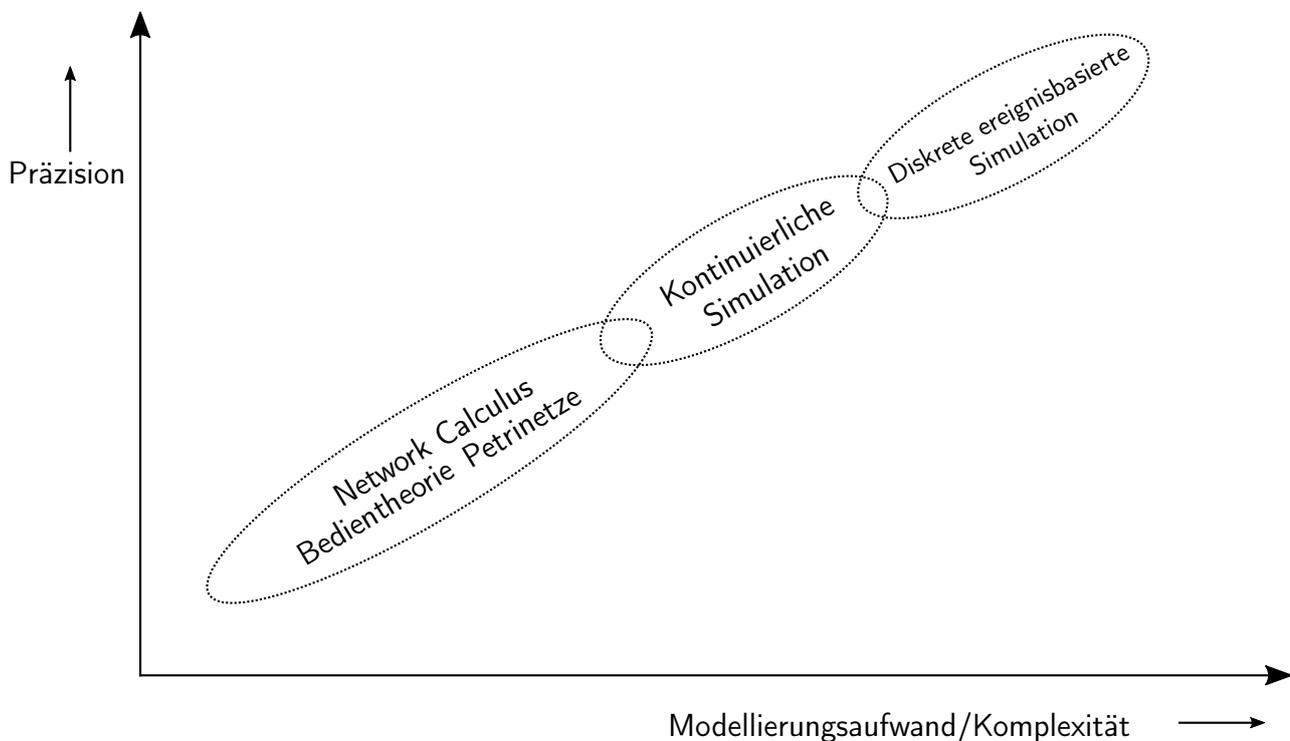


Abbildung 3.1: Vergleichende Einordnung der vorgestellten Modellierungsmethoden

3.2.1 Analytische Modelle

Analytische Modelle bilden das Systemverhalten auf Basis mathematischer Zusammenhänge nach, deren Variable die Kennwerte des Kommunikationssystems darstellen [26]. Zunächst erfordert dies globales Wissen über die systeminhärenten Zusammenhänge, die im Rahmen einer Exploration des Entwurfsraums oder aufgrund der Komplexität des Systems nicht notwendigerweise bekannt sind. Weiterhin müssen – vor allem bei komplexen Systemen – vereinfachende Annahmen getroffen werden, um die Modellierung überhaupt zu ermöglichen, da sich die Vielzahl der Parameter sowie deren Wechselwirkungen und Abhängigkeiten in solchen Fällen nicht mehr in handhabbaren Modellen abbilden lassen. Der Einsatz solcher Modelle resultiert, insbesondere im Vergleich mit diskreten Netzwerksimulationen, in einer deutlich verkürzten Berechnungszeit, jedoch werden Aussagekraft sowie Allgemeingültigkeit der Ergebnisse ebenfalls reduziert. Weiterhin lassen sich zwar Extrem-, d.h. Minimal- und Maximalwerte für die zu analysierenden Variablen bestimmen, jedoch nicht deren typische Verteilung [70].

Bedien- oder Warteschlangentheorie

Als Teilgebiet der Wahrscheinlichkeitstheorie beschäftigt sich die Bedien- oder Warteschlangentheorie primär mit der Bestimmung von Bedienzeiten [71]. Sie erlaubt sowohl die Abbildung von statistischen Verteilungen der gesuchten Größen als auch von Extrem- und Mittelwerten. Nachteilig wirkt sich die Modellierung auf Basis von Zufallsverteilung, z.B. Exponential- oder Poissonverteilung, insofern aus, als das nicht alle Prozesse im Kontext von Kommunikationssystemen diesen Verteilungen folgen (z.B. spontane Netzwerklasten). Weiterhin ist die Genauigkeit der Aussagen in Folge der abstrahierten Modellierung oftmals gering. Dennoch stellt die Bedientheorie ein probates Mittel dar, um beispielsweise schnell die durchschnittlich zu erwartende Latenz einer Kommunikationsverbindung abschätzen zu können.

Petrinetze

Petrinetze sind bi-partite Graphen und erlauben eine präzise Modellierung von diskreten, ereignisbasierten Prozessen auf Basis von Zuständen und bedingten Zustandsübergängen. Durch die Wahl des Abstraktionsniveaus der Zustände, z.B. „Bit wird gesendet“ vs. „Benutzeraktion hat Aktion ausgelöst“, lassen sich der Detailgrad der Modellierung sowie der damit verbundenene Rechenaufwand flexibel anpassen. Dennoch ist der Aufwand der Abbildung des Systemverhaltens mittels Petrinetzen vergleichsweise hoch, da die Basiselemente nur eine primitive Funktionalität aufweisen. Hierarchische Petrinetze können hier Abhilfe schaffen, wodurch sich die Modellierungskomplexität aber weiter erhöht. Weiterhin ist es nicht möglich das Verhalten mathematisch zu analysieren, falls gleichzeitig zwei oder mehrere deterministische Übergänge aktiviert sind [72][73].

„Network Calculus“

Network Calculus ist ein Hilfsmittel zur Bestimmung von Performancegarantien innerhalb von Kommunikationsnetzwerken oder Computerprogrammen als eine Theorie deterministischer Warteschlangensysteme [74]. Darauf basierende Modelle eignen sich nicht nur zur Bestimmung stochastischer Erwartungswerte, sondern geben beispielsweise auch Auskunft über maximal auftretenden Latenzen oder erzielbaren Durchsatzraten, wobei auch Bedingungen wie Verbindungskapazität, Überlastkontrolle, Grundlast oder auch „Traffic Shaping“ berücksichtigt werden. Den mathematische Hintergrund

bilden Faltungsoperationen und die sog. *Min-Plus-Algebra*, die kombiniert werden, um insbesondere Verkettungseffekte in Netzwerkpfaden darzustellen [75] [33]. Nachteilig wirkt sich bei der Nutzung von Network Calculus die mit der Netzwerkgröße bzw. der Anzahl der Knoten entlang eines Netzwerkpfades steigende Komplexität aus, so dass dieser Ansatz nur schlecht skaliert.

3.2.2 Simulation

Granino A. Korn definierte die Bedeutung des Begriffs Simulation folgendermaßen „A simulation is an experiment on a model.“ [76]. Können bspw. Systeme, bzw. ihr dynamisches Verhalten, nicht mehr analytisch beschrieben werden, kann man sich diesem mit Hilfe von Modellen und Simulationen annähern. Dabei können sowohl kontinuierliche wie auch diskrete Modelle zum Einsatz kommen [77], die in den folgenden Abschnitten näher beschrieben werden.

Kontinuierliche Simulation

Sollen stetige Prozesse nachempfunden werden, bietet sich die kontinuierliche Simulation als Verfahren an. Hierbei wird ein kontinuierliches Modell geschaffen, welches das abzubildende System mit einer oder mehrerer Differentialgleichungen oder einem Differentialgleichungssystem zu beschreiben versucht. Besondere Bedeutung nimmt diese Simulationsmethode im Bereich der theoretischen oder Systembiologie, der Chemie, aber auch der Elektronik und Mechanik ein.

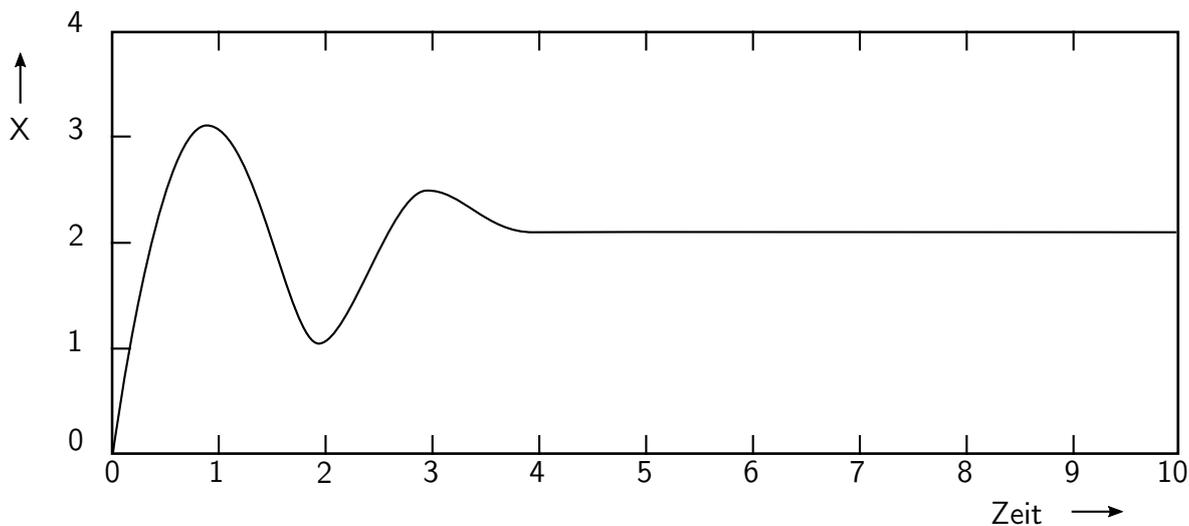


Abbildung 3.2: Kontinuierliche Modellierung des zeitlichen Verhaltens einer Variable X

Diskrete Simulation

Im Gegensatz zur kontinuierlichen wird bei der diskreten Simulation das Systemverhalten in fest definierten Zeitschritten untersucht. Eine besondere Rolle spielt dabei die ereignisbasierte oder ereignisorientierte Simulation. Das Verhalten des zu simulierenden Systems wird dabei über eine endliche Menge von Zuständen sowie den Übergängen zwischen diesen im zeitlichen Kontext abgebildet. Simuliert wird also weiterhin im Zeitbereich, jedoch nicht notwendigerweise in äquidistanten Zeitschritten. Die Wahl des Abstraktionsniveaus ermöglicht den Abtausch zwischen Detailgrad und Simulationsaufwand, wobei

theoretisch keine Präzisionsgrenzen existieren. Um eine Entwurfsraumanalyse sinnvoll durchführen zu können, sollte das Abstraktionsniveau jedoch so hoch wie möglich gewählt werden, ohne dabei die für die Analyse notwendige Präzision zu opfern.

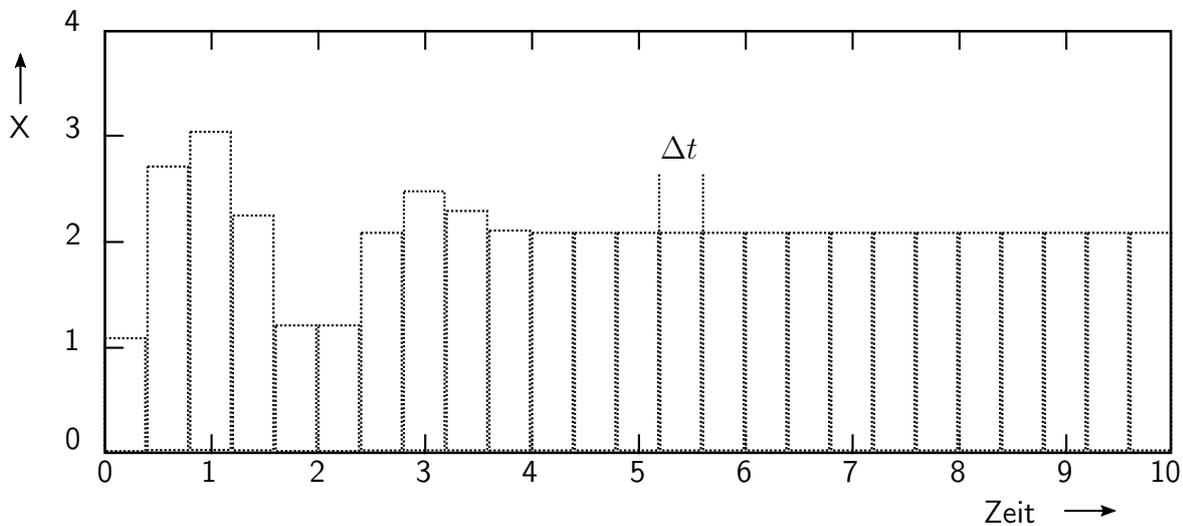


Abbildung 3.3: Diskrete Simulation des zeitlichen Verhaltens einer Variable X mit einem Zeitintervall t

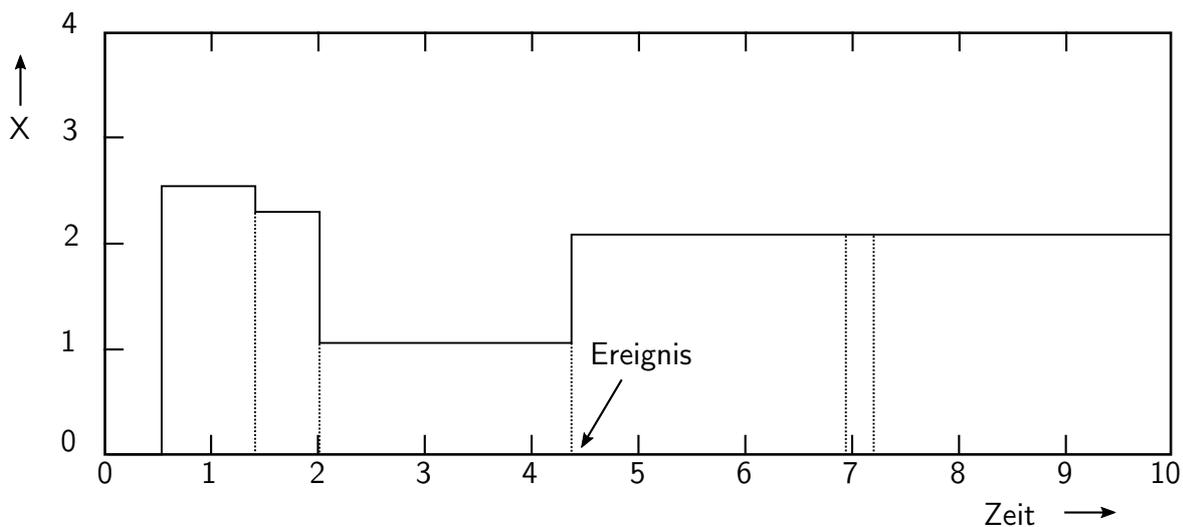


Abbildung 3.4: Ereignisbasierte diskrete Simulation des zeitlichen Verhaltens einer Variable X

Die Simulation im Zeitbereich stellt in Verbindung mit dem gewählten Abstraktionsniveau zugleich einen der Nachteile dieser Analysemethoden dar. Es entsteht im Vergleich zu analytischen Ansätzen ein größerer Aufwand bei der Modellierung, da alle notwendigen Systemzustände inkl. ihrer Beziehungen zu einander beschrieben und mit Zeitdauerinformationen versehen werden müssen [78]. Dadurch wird es jedoch möglich, nichttriviale zeitlich verknüpfte Zusammenhänge darzustellen, beispielsweise oszillatives Verhalten aufgrund von ungünstigen Verzögerungsbedingungen. Darüber hinaus lassen sich im Vergleich zu den anderen vorgestellten Methoden, physisch vorhandene Kommunikationssysteme mit einer diskreten Simulation koppeln, um sog. „Hardware-in-the-loop“-Analysen durchzuführen [79].

3.2.3 Simulationsbasierte Entwurfsraumexploration

Wie zu Beginn von Kapitel 3 beschrieben, müssen Punkte aus dem Entwurfsraum, d.h. konkrete Systemparameterkonstellationen, gewählt und hinsichtlich der Erfüllung der Anwendungsanforderungen bewertet werden. Da eine vollständige Exploration aufgrund der Vielzahl der Parameter und der erforderlichen kleinen Schrittweiten sowie der anfallenden Simulationsdauer vor dem Hintergrund nur begrenzt zur Verfügung stehender Rechenressourcen schwierig bis unmöglich umzusetzen ist, muss eine Explorationsstrategie angewendet werden. Diese kann in Studien einzelner, für das Systemverhalten maßgeblicher, Parameter oder sogar der Analyse zufällig gewählter Parameterkombinationen bestehen. Nach Wahl der Parameterwerte wird das Simulationsmodell parametrisiert und die Simulation durchgeführt. Die Randbedingungen des Simulationsszenarios entstammen dabei dem Anwendungsfall und sollen diesen möglichst realistisch nachbilden. Nach Abschluss der Simulation, bzw. der Simulationen, kann das Systemverhalten in durch die Anwendung vorgegebenen Leistungsmetriken bewertet und mit den gestellten Anforderungen verglichen werden. Neben der Aussage, ob das System diese mit der Parametrierung erfüllt, lassen sich im Rahmen der Exploration auch grundlegende Erkenntnisse über auftretende Wechselwirkungen sowie den Einfluss einzelner Parameter erlangen.

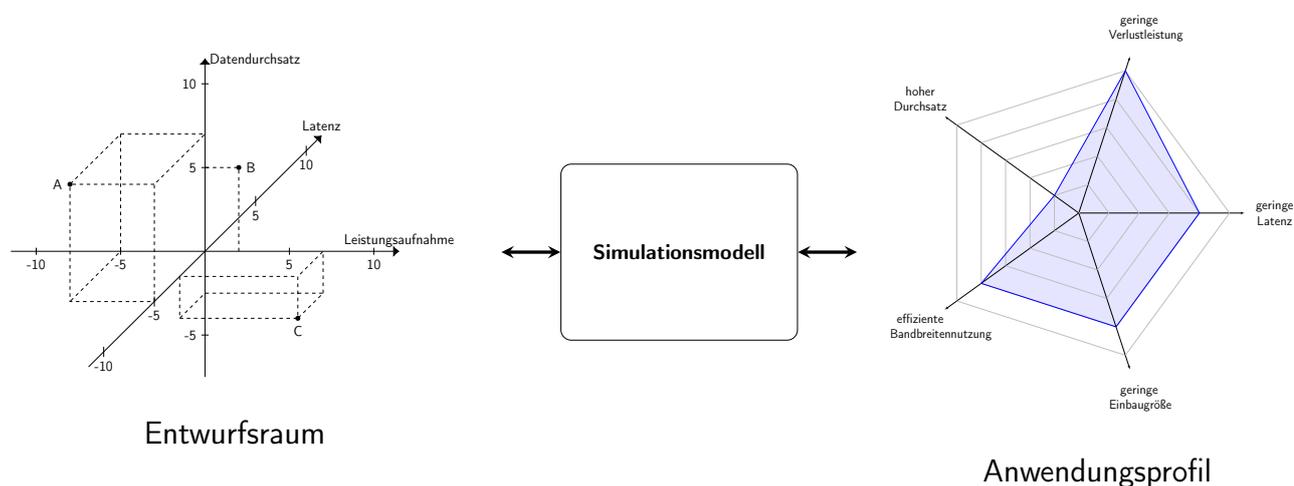


Abbildung 3.5: Simulationsbasierte Entwurfsraumexploration: Die Simulation auf Basis des angefertigten Modells erlaubt die Einordnung einer bestimmten Parameterkonstellation in den anwendungsspezifischen Kontext. Dadurch lässt sich die Erfüllung der durch das Anwendungsprofil vorgegebenen Ziele überprüfen.

Werkzeuge für die simulationsgestützte Analyse von Netzwerkkommunikation

Eines der wichtigsten Elemente bei der Simulation von Kommunikationssystemen ist das dafür verwendete Werkzeug, in diesem Fall der Netzwerksimulator. Mit seiner Hilfe wird das Verhalten des zu untersuchenden Netzwerks auf Basis von modellhaften Annahmen vorhergesagt. Das folgende Kapitel gibt einen Überblick über die gängigsten diskreten ereignisbasierten Simulationswerkzeuge. Werkzeuge für die analytische Modellierung finden aufgrund der in Kapitel 3.2.1 genannten Nachteile keine Berücksichtigung. Die aufgeführten Netzwerksimulatoren werden anschließend hinsichtlich ihrer Eigenschaften und Einsetzbarkeit verglichen. Aus diesem Vergleich geht OMNeT++ als geeignetstes Werkzeug hervor, welches dann in Abschnitt 4.2 im Bezug auf Modellierungsparadigma und Struktur ausführlich erläutert wird.

4.1 Überblick über vorhandene Simulationswerkzeuge

Seitdem der Bedarf an Werkzeugen für die Netzwerksimulation Mitte der 90er Jahre stark gewachsen ist, entstanden zahlreiche Projekte und Simulatoren. Im Folgenden wird ein Überblick über die wichtigsten Vertreter gegeben. Im Anschluss erfolgen eine kriterienbasierte Bewertung sowie ein Vergleich der Vertreter im Hinblick auf ihre Eignung zur Durchführung der Untersuchungen.

4.1.1 Network Simulator/ns

Bereits 1995, unter Förderung der DARPA entwickelt, etablierten sich die verschiedenen Varianten des quelloffenen und kostenfreien „network simulator“ im akademischen Bereich. Vor allen ns-2, das seit 2010 keine Unterstützung mehr erhält und auch nicht weiterentwickelt wird, war lange Zeit der De-facto-Standard bei der Analyse von Kommunikationsnetzwerken und ist weiterhin sehr weit verbreitet [80]. Ein großer Nachteil bei der Modellierung von und mit ns-2 ist die Vielzahl der einzusetzenden Werkzeuge. Der Quellcode des Modells wird in C++ abgefasst, die Steuerung der Simulation erfolgt mittels der Skriptsprache *tcl* und die Analyse der erzeugten Dateien erfolgt mittels der Skriptsprache

awk. Darüber hinaus existierten keine vorgefertigten Modelle für typische Basiskomponenten eines Netzwerks wie beispielsweise Switches oder Router. Mit der aktuellen Generation ns-3 wurden einige Limitierungen von ns-2 durch einen kompletten Neuentwurf aufgehoben. Aktuell befindet sich das Projekt jedoch immer noch in einer Phase der Portierung der Inhalte von ns-2 [81].

4.1.2 OPNET Modeler/Riverbed Modeler

Das Simulationswerkzeug OPNET Modeler, seit der Übernahme durch Riverbed 2012 „Riverbed Modeler Suite“ genannt, ist ein kommerzielles Werkzeug zur Performanceanalyse von drahtgebundenen oder drahtlosen Kommunikationsnetzwerken [82]. Wie von Prokkola [83] gezeigt, läuft die Simulation dabei auf Paketebene ab, d.h. das Abstraktionsniveau ist als vergleichsweise hoch zu bewerten. Weiterhin bietet OPNET eine umfangreiche Modellbibliothek, die vom Hersteller gepflegt und erweitert wird und viele kommerziell verfügbare Geräte nachbildet. Für akademische Zwecke existiert eine kostenfreie „Academic Edition“, welche jedoch im Funktionsumfang deutlich reduziert ist. Dies schränkt auch die Wiederverwendbarkeit von im Rahmen von Forschungsaktivitäten entstandenen Modellen deutlich ein. Darüber hinaus lässt sich die Korrektheit der vorhandenen Modelle nicht bzw. nur schwer verifizieren, da kein Quellcode vorliegt. Weiterhin ist die Topologie eines Netzwerks immer fest vorgegeben, d.h. es können keine parametrischen Topologien angelegt werden, was im Fall von Studien bezüglich der Teilnehmerzahl in Netzwerken von Nachteil ist.

4.1.3 NetSim

NetSim ist ebenfalls ein kommerzieller Netzwerksimulator der Firma Tetcos. Seit der Version 7 sind darin auch Modelle für drahtlose Kommunikationsprotokolle mit niedriger Verlustleistung enthalten. Wie bei OPNET Modeler (s. Abschnitt 4.1.2) ist eine spezielle Version für den akademischen Bereich verfügbar, diese ist jedoch eher für den Einsatz in Laboren, d.h. den Unterricht gedacht, nicht für die Forschung [84]. Darüber hinaus ist diese Variante nicht kostenfrei verfügbar, wodurch der Nutzerkreis von erstellten Modellen beschränkt ist.

4.1.4 OMNeT++

Die bereits 1997 zum ersten Mal veröffentlichte Umgebung zur Entwicklung von Netzwerksimulatoren ist quelloffen und zur akademischen Nutzung kostenfrei verfügbar. Für den Industrieinsatz existiert eine kommerzielle Variante [85]. Bei der Entwicklung standen die Möglichkeit zur Hierarchisierung sowie der komponentenweisen Wiederverwendung von Modellen im Vordergrund. Im Vergleich zu ns beinhaltet OMNeT++ eine graphische Oberfläche zur Programmierung der Modelle, der Definition von Simulationen sowie deren Durchführung. Weiterhin kann ein graphisches Debugging stattfinden und es existiert ein integriertes Werkzeug zur graphischen Auswertung der erzeugten Simulationsergebnisse. Mit der vom Entwicklerteam ebenfalls zur Verfügung gestellten Modellbibliothek *inet* existiert ein umfangreiches Spektrum von Modellen von drahtlos oder drahtbehaftet kommunizierenden Elementen [86]. Im Laufe der Jahre sind darüber hinaus zahlreiche Erweiterungen aus dem Umfeld der aktiven Entwicklungsgemeinschaft um OMNeT++ entstanden.

4.1.5 Vergleich und Auswahl der Simulationsumgebung

Die Auswahl der genutzten Simulationsumgebung erfolgte nach den Kriterien Nutzbarkeit, dessen insbesondere die Verfügbarkeit von integrierten Debugging-Möglichkeiten, Umfang der verfügbaren Simulationsbibliotheken sowie Geschwindigkeit und Speicherbedarf bei der Simulation. Nicht einsehbarer Quellcode stellte ein Ausschlusskriterium dar, da so weder die genaue Funktionsweise noch die Korrektheit der vorhandenen Modelle untersucht werden kann. Die qualitativen Bewertungen in den verschiedenen Kategorien sind Tabelle 4.1 zu entnehmen. Die Daten zur Simulationsperformance, d.h. Geschwindigkeit und Speicherbedarf stammen aus der Untersuchung von Weingärtner, Lehn und Wehrle [87], erweitert um die Ergebnisse von Khan, Bilal und Othmann [88]. In der Gesamtbewertung zeigte sich OMNeT++ 2010 als am vielversprechendsten, da bei der Entwicklung häufig genutzte Funktionen vollständig integriert waren und umfangreiche Visualisierungsmöglichkeiten existierten. Weiterhin mangelte es ns-3 zum Zeitpunkt der Erstellung dieser Arbeit an Anwendungsbreite sowie Dokumentation, da das Projekt erst im Juli 2008 gestartet worden war. Ebenso war OMNeT++ nach der Neuveröffentlichung 2006 deutlich präsenter als ns-3, was sich auch in einer aktiveren „Community“ zeigte. Aus diesem Grund fiel die Wahl der Simulationsumgebung auf OMNeT++, welche in den folgenden Kapiteln näher beschrieben werden soll.

Tabelle 4.1: Qualitativer Vergleich der Eignung verschiedener Netzwerksimulationsumgebungen (Stand: Juli 2010)

	Kategorie	ns-2	ns-3	OMNeT++	OPNET	NetSim
Verfügbarkeit	Quelloffenheit	✓	✓	✓	--	--
	Kostenfreiheit	✓	✓	✓	--	--
Nutzbarkeit	IDE	--	--	++	++	++
	integr. Debugger	-	-	++	-	-
	integr. Profiling	-	-	++	--	--
	Visualisierung Simulation	+	+	++	++	+
Performance	Rechenbedarf	o	++	+	?	?
	Speicherbedarf	o	++	+	?	?
Gesamtwertung		o	+	++	(-)	(-)

Mittlerweile hat sich die Vielfalt vorhandener Modelle sowie der Umfang verfügbarer Dokumentation für ns-3 deutlich verbessert. Dennoch bietet der Simulator weiterhin keine direkte Möglichkeit zur Visualisierung, so dass OMNeT++ auch 2020 noch als für die Zwecke dieser Arbeit am geeignetsten angesehen werden kann.

Aktuelle Veröffentlichungen im Kontext von ns-3 widmen sich deshalb zum einen der Entwicklung einer graphischen Benutzeroberfläche [89], aber natürlich auch der stetigen Verbesserung der physikalischen Exaktheit, beispielsweise der korrekten Modellierung des gemeinsam zur Funkkommunikation genutzten Kanals [90].

Auch OMNeT++ wird weiterhin äußerst aktiv genutzt, um Fragen im Zusammenhang mit neuen Technologien wissenschaftlich zu beantworten. Dabei handelt es sich beispielsweise um die Sicherheit zukünftiger 5G-Netze [91] oder auch um lichtbasierte Kommunikationstechnologien [92], wie sie auch im Rahmen dieser Arbeit bereits untersucht wurden. Weiterhin wird aber auch nach Wegen gesucht, um groß angelegte Simulationsstudien weitestgehend zu automatisieren und so die Prozesse der Simulationskonfiguration und Auswertung zu vereinfachen und zu beschleunigen [93].

4.2 Netzwerksimulationsframework OMNeT++

Bei OMNeT++ handelt es sich, wie in Abschnitt 4.1.4 bereits erläutert, um ein Framework zur Entwicklung von C++-basierten Netzwerksimulatoren sowie gleichzeitig um eine Simulationsbibliothek. Es existieren Erweiterungen für die Echtzeitsimulation, die Emulation von Netzwerken, die Kopplung einer Simulation an real existierende Netzwerke („Hardware-in-the-loop“) und zur Integration von SystemC-basierten Modellen. Die angebotene graphische Entwicklungsumgebung basiert auf dem weitverbreiteten Programmierwerkzeug „Eclipse“ und bietet dadurch viele komfortable Funktionen zur Exploration von vorhandenem, aber auch der Entwicklung oder Überarbeitung von neuem Quelltext (z.B. automatische Formatierung, Vervollständigung von Methoden, Parameterlisten, Variablennamen und Elementen strukturierter Datentypen, Sprung zur Definition von Methoden). Der komponentenbasierte Modellierungsansatz sowie die Dreiteilung in Simulationsmodell, Modellbeschreibung und Simulationskonfiguration werden in den folgenden Abschnitten erläutert.

4.2.1 Komponenten und Struktur

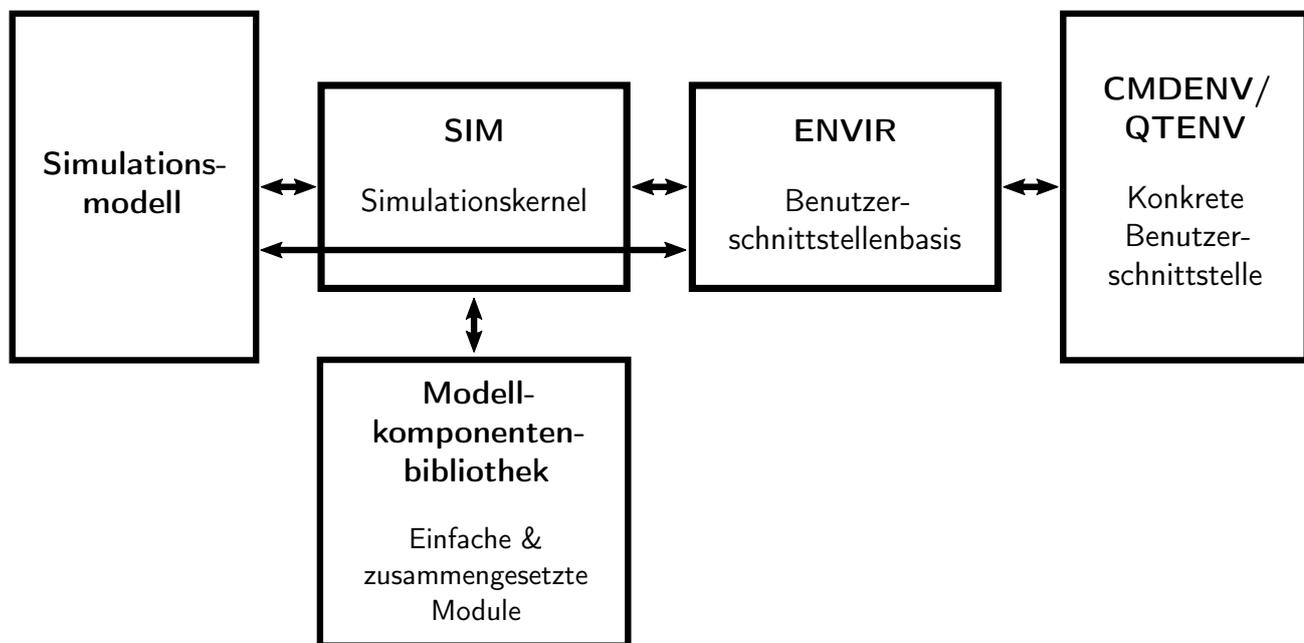


Abbildung 4.1: Übersicht über die Komponenten eines mittels OMNeT++ erstellten Simulationsprogramms

Ein mit Hilfe des Simulationsframeworks OMNeT++ erstelltes ausführbares Simulationsprogramm besteht aus fünf Komponenten, die in Abbildung 4.1 inklusive der zwischen ihnen stattfindenden Informationsflüsse dargestellt sind. Das durch den Benutzer erstellte Simulationsmodell interagiert mit dem Simulationskernel, um beispielsweise Nachrichten abzusetzen oder zu empfangen. Ebenfalls kann es über diesen auf bereits bestehende Modellkomponenten zugreifen, die zur Laufzeit dynamisch eingebunden werden. Weiterhin ist der Simulationskernel mit der Benutzerschnittstellenbasis *ENVIR* verbunden, um zum einen Informationen über den aktuellen Status des Simulationslaufes anzeigen zu können, aber auch, um bei Nutzung der interaktiven graphischen Benutzerschnittstelle den Simulationslauf beeinflussen zu können. Die graphische Benutzerschnittstelle ist in Abbildung 4.3 dargestellt.

Die Entwicklung und Anwendung eines Simulationsmodells mit OMNeT++ lässt sich in fünf Schritte unterteilen:

1. Erstellen des Simulationscodes für eine Komponente in C++
2. Beschreibung der Modellkomponente inkl. Parameter in Network Description Language (NED)
3. Konfiguration einer Simulation
4. Durchführung der Simulation
5. Ergebnisanalyse

Im Rahmen der Modellierung können einfache Module, d.h. so genannte „simple modules“ definiert werden, welche sich zu zusammengesetzten Modulen, sog. „compound modules“, kombinieren lassen (siehe Abbildung 4.2). Um die Kommunikation zwischen diesen Modulen zu ermöglichen, werden Ein- und Ausgangsschnittstellen in Form sog. „gates“ definiert. Dadurch lassen sich die Module im Sinne des OSI-Modells in Form von Schichten organisieren, wodurch die Funktionalität des einfachen Moduls auf die schichtspezifische Zuständigkeit fokussiert bleibt. Die Verbindung zwischen zwei „gates“ wird mittels des Schlüsselworts „connection“ in der das zusammengesetzte Modul beschreibenden NED-Datei festgelegt. Die Anzahl der Modulhierarchieebenen ist unbegrenzt, jedoch ist das zusammengesetzte Modul mit der höchsten Hierarchiestufe stets das zu simulierende Netzwerk. Speziell in diesem Rahmen lassen sich die sog. „connections“ mit Eigenschaften versehen, wodurch sich dann Kanäle („channels“) ergeben.

Fast alle Module benötigen Verzögerungszeiten, um beispielsweise Wartezeiten oder sich periodisch wiederholende Vorgänge wie konstant generierten Datenverkehr, abzubilden. Dazu dienen sog. „self-messages“, d.h. Nachrichten, die von einem Modul an sich selbst gerichtet sind und vom Simulationskernel nach der angegebenen Verzögerungszeit, respektive zu einem festgelegten Zeitpunkt, übermittelt werden.

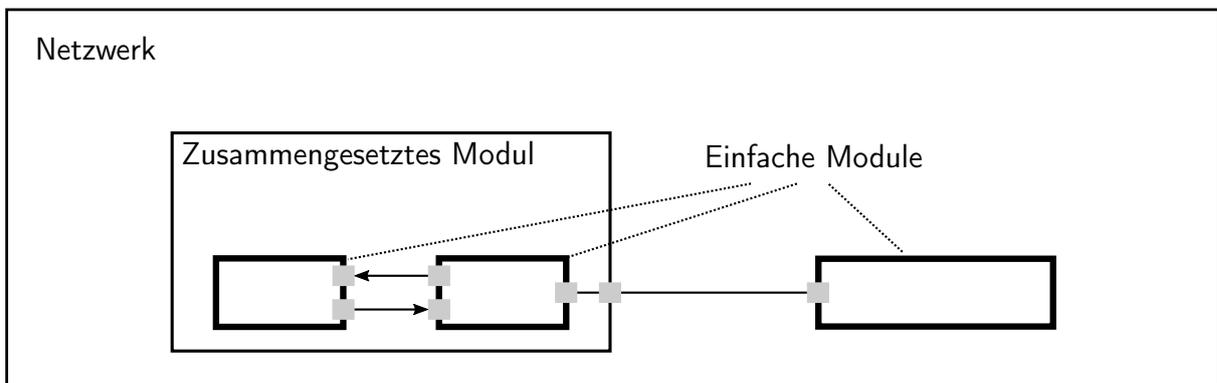


Abbildung 4.2: Exemplarische Darstellung des Aufbaus eines zu simulierenden Netzwerks: Einfache Module können entweder alleinstehend oder zu einem zusammengesetzten Modul kombiniert innerhalb des Netzwerks wirken. Dabei erfolgt die Kommunikation über sog. „gates“ (Pfeile) oder Kanäle (Linie). (Reproduziert aus [94]).

Codebeispiel 4.1: Einfaches Modul, welches einlaufende Datenpakete nach einer Verzögerung wieder aussendet (basierend auf TicToc-Tutorial Teil 7 [95])

```

1  #include <stdio.h>
   #include <string.h>
   #include <omnetpp.h>

   using namespace omnetpp;

6  class Txc7 : public cSimpleModule           /* Objekt basiert auf einem einfachen Modul*/
   {
   private:
       cMessage *event;                       /* Referenz auf Ereignisnachricht */
11      cMessage *tictocMsg;                  /* Referenz auf übertragene Nachricht */

   public:
       Txc7();
       virtual ~Txc7();

16      protected:
           virtual void initialize() override;
           virtual void handleMessage(cMessage *msg) override;
   };

21  Define_Module(Txc7);

Txc7::Txc7()
   {
26      event = tictocMsg = nullptr;          /* Bei Instanziierung keine gültigen Nachrichten*/
   }

Txc7::~Txc7()
   {
31      cancelAndDelete(event);              /* Ereignisnachricht stornieren und löschen */
       delete tictocMsg;
   }

   void Txc7::initialize()
36   {
       event = new cMessage("event");        /* Neue Nachricht mit dem Namen "event" anlegen */
       tictocMsg = nullptr;

       if (strcmp("tic", getName()) == 0) {  /* Überprüfung des Nachrichtennamens */
41         EV << "Scheduling first send to t=5.0s\n"; /* Ausgabe einer Debug-Meldung */
           tictocMsg = new cMessage("tictocMsg"); /* Anlegen einer neuen Nachricht "tictocMsg" */
           scheduleAt(5.0, event);           /* Eintragen in Ereignisliste für t=5.0s*/
       }
   }

46   void Txc7::handleMessage(cMessage *msg)
   {

       if (msg == event) {                  /* Überprüfung ob Ereignisnachricht */
51         EV << "Wait period is over, sending back message\n"; /* Debug-Meldung*/
           send(tictocMsg, "out");          /* Nachricht über Port "out" versenden */
           tictocMsg = nullptr;             /* Löschen des Bezugs zum Nachrichtenobjekt */
       }
       else {
56         simtime_t delay = par("delayTime"); /* Abfrage der konfigurierten Verzögerung */
           EV << "Message arrived, starting to wait " << delay << " secs...\n"; /* Debug-Meldung */
           tictocMsg = msg;                 /* eingegangene Nachricht für Versand speichern */
           scheduleAt(simTime()+delay, event); /* Ereignis mit Verzögerung "delay" vormerken */
       }zum
61   }

```

Codebeispiel 4.2: Beschreibung des einfachen Moduls aus dem Code-Beispiel 4.1 sowie eines Netzwerks bestehend aus zwei Instanzen des einfachen Moduls innerhalb einer NED-Datei (basierend auf TicToc-Tutorial Teil 7 [95])

```

1  simple Txc7                                     # Modul vom Typ Txc7
  {
    parameters:
      volatile double delayTime @unit(s);         # konfigurierbarer Parameter
5     @display("i=block/routing");
    gates:
      input in;                                   # Eingangskanal
      output out;                                # Ausgangskanal
  }
10 network Tictoc7                                # zu simulierendes Netzwerk
  {
    submodules:                                 # Netzwerkelemente
      tic: Txc7 {                                  # erster Knoten
15         parameters:
           @display("i=,cyan");
        }
      toc: Txc7 {                                  # zweiter Knoten
20         parameters:
           @display("i=,gold");
        }
    connections:                               # Verbindungen zw. Gates
      tic.out --> { delay = 100ms; } --> toc.in;  # Verzögerungskanal
      tic.in <-- { delay = 100ms; } <-- toc.out;
25 }

```

Codebeispiel 4.3: Beispiel für den Inhalt der Konfigurationsdatei omnetpp.ini: Es wird eine Studie für den Parameter *tic.delayTime* durchgeführt, wodurch sich insgesamt sieben einzelne Simulationsläufe ergeben. Der Wert für *toc.delayTime* wird für jeden Simulationslauf zufällig aus einer Exponentialverteilung mit $\lambda = 1/3\text{s}$ bestimmt. (basierend auf TicToc-Tutorial Teil 7 [95])

```

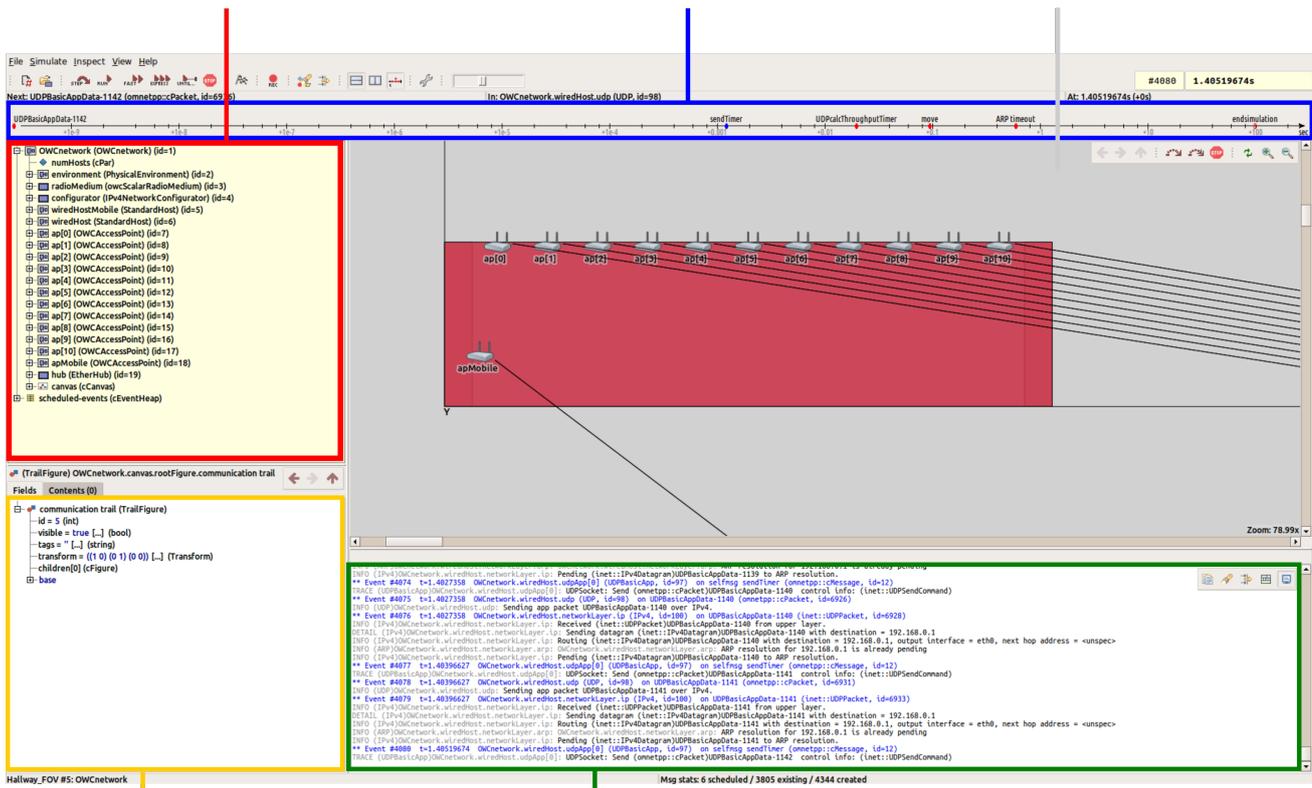
1  [Config Tictoc7]
  network = Tictoc7                               # Netzwerkname wie in NED-Datei
  Tictoc7.tic.delayTime = ${0.0..3.0 step 0.5}s    # Parameterstudie der Verzögerung
  Tictoc7.toc.delayTime = exponential(3s)          # zufällige Verzögerungszeit

```

Objektnavigator
zeigt die Objekthierarchie
aktueller und zukünftiger Objekte

Zeitlinie
zeigt zukünftige Ereignisse auf
einer logarithmischen Skala

Netzwerkzeuge
graphische Visualisierung aller Module
& Zeichenfläche für Animationen



Objektinspektor
zeigt Inhalt und Eigenschaften des
ausgewählten Objekts an

Protokollierungsanzeige
Protokoll aller übertragenen Nachrichten sowie aller Status- bzw. Debug-
Meldungen, die von den Modulen ausgegeben werden

Abbildung 4.3: Überblick über die graphische Benutzerschnittstelle QTEnv bei Durchführung einer Simulation mit OMNeT++

Modellierung von drahtlosen Kommunikationssystemen zur diskreten Netzwerksimulation

Zur Durchführung von simulationsbasierten Analysen wurden im Rahmen dieser Arbeit zwei umfangreiche Simulationsmodelle für diskrete, ereignisbasierte Simulation mit OMNeT++ entwickelt: eines für ein funkbasiertes Kommunikationssystem mit dem primären Ziel einer niedrigen Verlustleistung im Betrieb, das zweite zur Abbildung eines optischen Kommunikationssystems zur Bereitstellung mittlerer bis hoher Datenraten. Wie in Kapitel 4.2.1 beschrieben, wird der während der Simulation ausgeführte Code in der Programmiersprache C++ abgefasst. Eine anschließende Parametrierung erfolgt mittels der „Network Description Language“, kurz NED.

Der Einsatz von ereignisbasierter Simulation mittels OMNeT++ als Mittel zur Exploration des Entwurfsraums zukünftig zu verwendender sowie bestehender Systeme, zieht die Forderung nach einem hohen Maß an Exaktheit nach sich. Alle Vorgänge innerhalb des Realsystems, d.h. sowohl die Intraschicht- respektive Interschichtkommunikation innerhalb eines Knotens als auch die Interknotenkommunikation müssen funktional und zeitlich äquivalent abgebildet werden. Da das Abstraktionsniveau jedoch einen signifikanten Einfluss auf die Simulationsgeschwindigkeit hat, ist dieses so hoch wie möglich zu wählen, ohne die vorangegangene Bedingung zu verletzen. Ausnahmen stellen hierbei Vorgänge dar die, die aufgrund ihrer Komplexität nur mit sehr großen Aufwand abbildbar sind und die Simulationsgeschwindigkeit signifikant negativ beeinflussen würden. So lässt sich beispielsweise das Verhalten eines SoC-integrierten Transceivers im Bezug auf das energetische, logische und zeitliche Verhalten sehr gut beschreiben, das Laufzeitverhalten eines softwarebasierten MAC, welches auf dem im SoC ebenfalls enthaltenen Mikrocontroller ausgeführt wird, jedoch nur sehr schwer. Bei ersterem sorgt die Hardwarenähe für eine vergleichsweise geringe Komplexität im Bezug auf den Raum möglicher Zustände sowie die Übergänge dazwischen (siehe auch Abbildung 2.4 in Unterabschnitt 2.1.4). Bei letzterem ist die Komplexität ungemein höher, da der Raum der Zustände, welche die Software einnehmen kann, um einige Größenordnungen größer ist. Zudem müsste die Ausführung, der in einer Hochsprache abgefassten Programmanweisungen vollständig modelliert werden.

Weiterhin sollte die Ausführung des Modells, z.B. zu Verifikations- oder Wartungszwecken, in der Simulation noch leicht nachvollziehbar sein. Dies erfordert unter Umständen das Einfügen einzelner zusätzlicher Ereignisse, wodurch sich die Simulationsgeschwindigkeit insgesamt geringfügig reduzieren

kann. Im Gegenzug erhöhen sich die Geschwindigkeit bei der Fehlersuche und der Verifikation sowie die Wartbarkeit, was die verlängerte Simulationsdauer mehr als aufwiegt.

Alle entwickelten Module folgen den, entweder durch Standards (in diesem Fall IEEE 802.15.4) oder durch Vorgaben von OMNeT++, definierten Schnittstellen und sind modular aufgebaut. Dadurch wird eine schnelle Austauschbarkeit und somit Vergleichbarkeit gewährleistet. Zusätzlich sind alle das Verhalten beeinflussenden Parameter via NED-Beschreibung verfügbar, wodurch Parameterstudien ohne erneute Kompilierung durchgeführt werden können.

Da speziell im Bereich der Simulation von Funkkommunikation bereits eine Vielzahl von Modellen für einzelne Protokolle/Algorithmen existiert, wurden einige davon nach vorheriger Verifikation und Anpassung zur Erstellung der Simulationsmodelle genutzt. Im Folgenden wird deshalb neben den existenten Implementierungsalternativen auch auf den Ursprung der genutzten Module eingegangen sowie deren Aufbau erläutert. Die Betrachtung orientiert sich dabei am OSI-Modell.

5.1 Entwurfsprozess

Zunächst soll der Entwurfsprozess zur Erstellung eines Simulationsmodells für die diskrete Netzwerksimulation exemplarisch vorgestellt werden. Da das Simulationsmodell Software darstellt, können die bekannten Softwareentwurfsansätze angewandt werden. Relevant sind hierbei vor allem Ansätze im objektorientierten Kontext, da die Objektorientierung das von OMNeT++ vorgegebene und in allen dafür verfügbaren Erweiterungen angewendete Programmierparadigma darstellt. Die einzelnen Phasen und ihre Berücksichtigung bei der Umsetzung eines Simulationsmodells für OMNeT++ werden in den folgenden Abschnitten erläutert. Sie folgen dem in Abbildung 5.1 dargestellten V-Modell nach Boehm [96], welches die einzelnen Phasen der Softwareentwicklung, ihre zeitliche Abfolge und die damit verbundenen Abhängigkeiten beschreibt. Es ist allerdings anzumerken, dass der dem Entwicklungsprozess inhärente dynamische Charakter durch das V-Modell nur unzureichend abgebildet wird. Nach Abschluss des Systementwurfs- oder der Softwarearchitekturphase lassen sich die daraus resultierenden Spezifikationen trotz neuer Erkenntnisse aus der Softwareentwurf- oder auch einer der Testphasen formal nicht anpassen, ohne alle folgenden Phasen erneut zu durchlaufen. Diese fehlende Dynamik wird in agilen Entwicklungsansätzen, z.B. SCRUM [97] oder „Extreme Programming“ [98], berücksichtigt, bei denen das iterative und inkrementelle Vorgehen im Vordergrund stehen. Bei der Umsetzung der Simulationsmodelle wurde deshalb schlussendlich ein hybrider Ansatz verfolgt. Zunächst erfolgten die Anforderungsanalyse sowie Betrachtungen zur Strukturierung auf Systemebene. Im Anschluss wurden unter Berücksichtigung vorhandener Module sowie der späteren Austauschbarkeit alle notwendigen Klassen und Schnittstellen als Platzhalter, d.h. ohne Funktionsinhalt erstellt. Die Umsetzung der einzelnen Funktionalitäten orientierte sich dann an den agilen Ansätzen, um schnell erste lauffähige Minimalsimulationen zu erhalten.

5.1.1 Anforderungen

Bevor die Implementierung eines Simulationsmodells erfolgen kann, ist es notwendig festzulegen, welcher Funktionsumfang, respektive Detailgrad, realisiert werden soll. Als Startpunkt für die Definition können dabei Standards, Referenzimplementierungen oder zu untersuchende Phänomene dienen. Im Fall von IEEE 802.15.4 wird beispielsweise ein Satz von Funktionen, sog. „Primitives“ inklusive einer Liste von Übergabeparametern (Signatur) sowie deren Bedeutung definiert. Ein Beispiel für eine Funktion befindet sich im Code-Auszug 5.1. Die Abbildung der Aufrufe dieser Funktionen

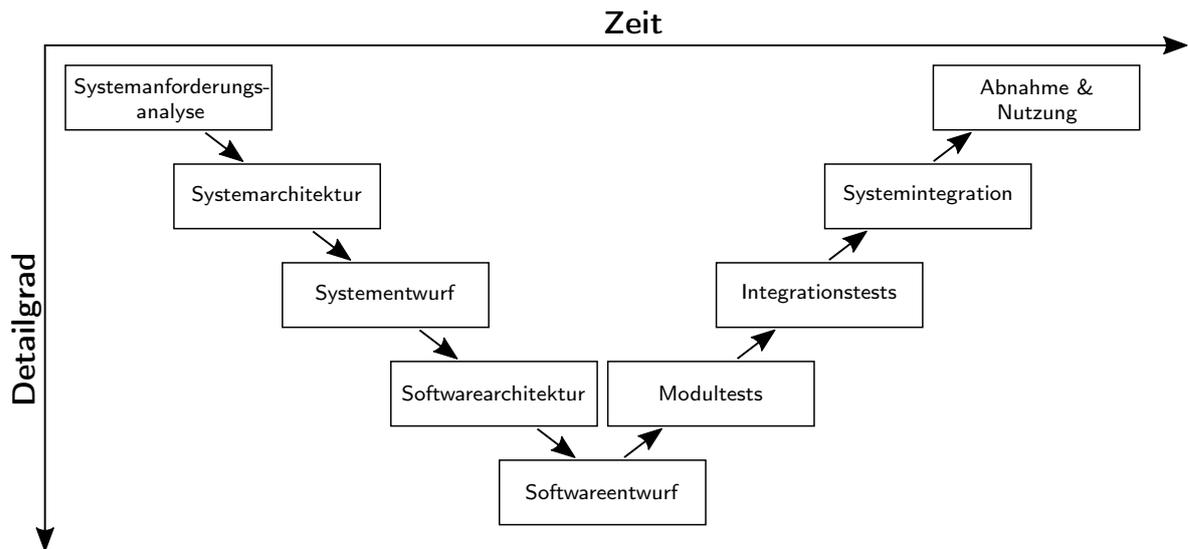


Abbildung 5.1: Entwurfsprozess gemäß V-Modell [96]

Codebeispiel 5.1: Beispiel für eine Verwaltungsanweisung zum passiven oder aktiven Abhören (ScanType) eines oder mehrerer Kanäle (ScanChannels) für eine definierte Zeit (ScanDuration) von der nächsthöheren Schicht an eine Instanz einer IEEE 802.15.4-konformen Datensicherungsschicht

```

MLME - SCAN_request
(
    ScanType ,
    ScanChannels ,
    ScanDuration ,
    ChannelPage ,
    SecurityLevel ,
    KeyIdMode ,
    KeySource ,
    KeyIndex
)
  
```

stellt das Mindestmaß an Modellierungsgranularität dar. Wie dem in Abbildung 5.2 dargestellten Sequenzdiagramm der gezeigten Funktion *MLME-SCAN_request()* zu entnehmen ist, kann der Funktionsaufruf eine Reihe von Einzelaktionen auslösen. So muss beispielsweise für das passive Abhören des Kanals die Bitübertragungsschicht angewiesen werden den Empfänger zu aktivieren und auf den richtigen Kanal zu wechseln. Zu Zwecken der Verifizier- und Wartbarkeit sollten dennoch alle Einzelaktionen sowie die dazu nötigen Teilschritte (z.B. das Erstellen und Versenden eines Datenpakets) als Ereignisse modelliert werden. Der damit einhergehende Performanceverlust ist, falls überhaupt vorhanden, im Vergleich zum Gewinn an Nachvollziehbarkeit zu vernachlässigen.

Weiterhin können analytische Voruntersuchungen stattfinden, um eine Referenz für die quantitativen Ergebnisse der diskreten Simulationen zu etablieren. So lässt sich beispielsweise die erzielbare Kommunikationsreichweite in Abhängigkeit eines festgelegten Kanalmodells (Signaldämpfung und -störungen), mit geringem Aufwand für festgelegte Parameterwerte oder -konstellationen berechnen.

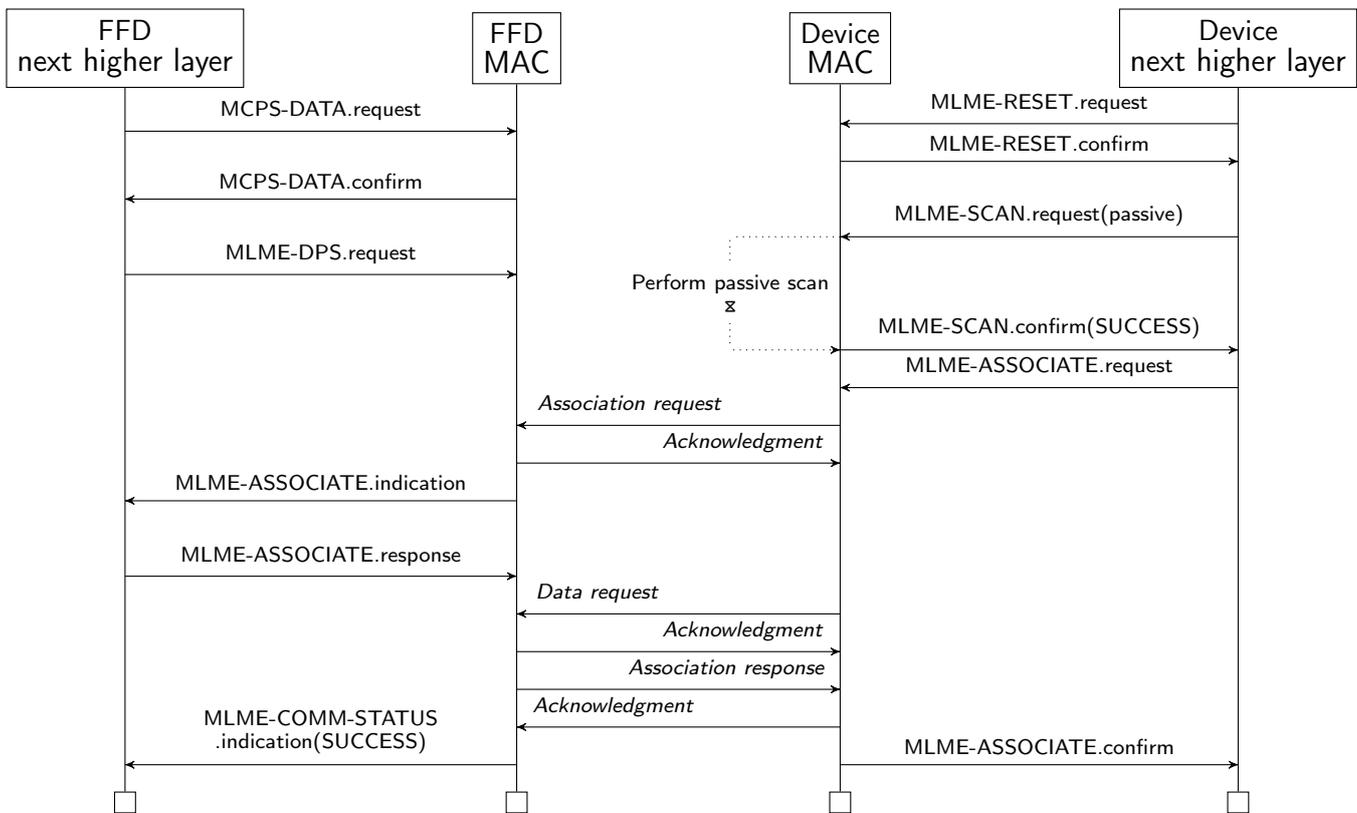


Abbildung 5.2: Sequenzdiagramm für die Assoziation eines Geräts („Device“) mit einem bestehenden PAN gemäß IEEE 802.15.4 als Beispiel für Verhaltensbeschreibungen als Vorlage für die Modellierung: Der sog. PAN-Koordinator ist ein „full function device“ (FFD). Im Rahmen von IEEE 802.15.4 wird zwischen Datenzugriffen (MCPS) und Verwaltungsoperationen (MLME) unterschieden. Nachdem eine höhere Schicht, die Eigenschaften des PANs der Medienzugriffsschicht MAC per bestätigtem Schreibzugriff mitgeteilt hat, wird dieses eröffnet und der Koordinator beginnt mit dem regelmäßigen Aussenden sog. „Beacons“. Das Gerät, welches sich verbinden möchte, hört passiv nach ausgesandten „Beacons“, nachdem die Medienzugriffsschicht auf Standardeinstellungen zurückgesetzt worden ist. Die Ergebnisse dieses „Scans“ werden der nächsthöheren Schicht mitgeteilt, die daraufhin das Netzwerk auswählt, dem beigetreten werden soll. Diese Beitrittsanfrage („Association request“) erreicht die nächsthöhere Schicht des PAN-Koordinators, die wiederum darüber entscheidet, ob das anfragende Gerät Teil des Netzwerks werden darf. Das Ergebnis dieser Entscheidung wird vom Gerät, welches beitreten möchte aktiv angefragt und der Erhalt der entsprechenden Antwort vom Gerät bestätigt. Mit Erhalt dieser Empfangsbestätigung betrachtet der PAN-Koordinator das Gerät als zum Netzwerk gehörig. Parallel dazu bestätigt die Medienzugriffsschicht der nächsthöheren Schicht im beitretenden Gerät, dass die Assoziation erfolgreich war und auch diese betrachtet das Gerät als Teilnehmer des PANs

Dieser Schritt stellte ebenfalls eine wichtige Grundlage für die Implementierung der Funktionsmodule sowie deren initiale Parametrierung dar.

5.1.2 Implementierung

Wie bereits erläutert, liegt der Implementierung der Simulationsmodelle eine objektorientierte Betrachtungsweise zu Grunde. Um die Wiederverwendbarkeit zu verbessern und im Entwurfsprozess leichter Anpassungen durchführen zu können, wurden alle Grundstrukturen in Form möglichst weniger Klassen abgebildet. Spätere Differenzierungen, respektive spezialisierte Strukturen, lassen sich über den Mechanismus der Vererbung mit vergleichsweise geringem Aufwand umsetzen.

Dem V-Modell folgend, führen Tests unter Umständen zu funktionalen Anpassungen, die wiederum verifiziert werden müssen. Gegen Ende des Entwurfsprozesses erfolgen jedoch keine die Funktionalität betreffenden Änderungen mehr. Es können allerdings Refaktorisierungen durchgeführt werden, um die Lesbarkeit, die Wartbarkeit und die Erweiterbarkeit des erstellten Quelltexts zu verbessern. Gleichzeitig kann so Redundanz ggfs. vermieden oder zumindest reduziert werden. Auch nach dieser Maßnahme muss das Verhalten über alle Stufen hinweg auf Korrektheit überprüft werden.

Verwendete Bibliotheken

Im Umfeld der Forschung und Entwicklung mit OMNeT++ sind einige Protokollbibliotheken bzw. Protokollsammlungen entstanden, von denen die für Modellierung im Rahmen dieser Arbeit relevanten kurz vorgestellt werden sollen.

INET Bei INET [86] handelt es sich um eine umfangreiche, quelloffene Sammlung von Protokoll- und Agentenimplementierungen sowie Modellen für drahtbehaftete und drahtlose Netzwerke für die Simulationsumgebung OMNeT++. Der Modellkatalog enthält Anwendungen (z.B. VOIP, BitTorrent, Video Streaming), Transportprotokolle (z.B. TCP, UDP, RTP), Protokolle aus der Vermittlungsschicht (IPv4/v6, ICMP, ARP) sowie Medienzugriffsprotokolle für drahtgebundene und drahtlose Übertragungen (Ethernet, IEEE 802.11(p), 802.16e, LTE). Einige der Entwickler sind auch an der Entwicklung von OMNeT++ beteiligt, so dass eine kontinuierliche Synchronisation mit dessen Entwicklungsfortschritten erfolgt.

MiXiM MiXiM [99] wurde entwickelt, um Vorgänge in mobilen und ortsfesten drahtlose Netzwerke besser abbilden zu können. Im Fokus stehen hierbei insbesondere drahtlose Sensornetzwerke, „Body Area Networks“ (BANs), Ad-Hoc- und Fahrzeugnetzwerke. Es beinhaltet detaillierte Ausbreitungs- und Interferenzschätzungsmodelle sowie Modelle für die Schätzung der Verlustleistungsaufnahme des Transceivers und eine Reihe von MAC-Protokollen. Die Entwicklung von MiXiM wurde 2013 mit der Version 2.3 eingestellt. Anschließend erfolgte eine Integration in das INET-Framework.

5.1.3 Verifikation

Ein wesentlicher Bestandteil des Softwareentwurfs ist die Verifikation. Insbesondere im Bereich der Simulation bzw. der Modellierung von Systemen fällt diesem Schritt eine große Bedeutung zu, da sich hieraus direkt die Qualität der Ergebnisse ableitet. Gleichzeitig hat sie praktische Relevanz, da

nur unter bestimmten Randbedingungen auftretende Laufzeitfehler zum spontanen Abbruch eines Simulationslaufs und somit dem Verlust der generierten Informationen bzw. Aussagen führen kann.

Im Rahmen des V-Modells, und auch in anderen Modellen (z.B. Spiralmodell), wird implizit iterativ und auf verschiedenen Ebenen, d.h. mit unterschiedlicher Granularität, verifiziert. Im Rahmen der erstellten Simulationsmodelle konzentriert sich ein großer Teil des Aufwands auf die Verifikation der Kernmodule, da aus diesen auch der wesentliche Teil der Funktionalität resultiert. Mit Hilfe anschließender Integrationstests wurde dann die Gesamtfunktionalität des Simulationsmodells sichergestellt.

Zunächst wurde der erzeugte C++-Code mit Hilfe des Diensts *Coverity Scan* [100] statisch analysiert. Dieses Vorgehen erlaubt die frühe Detektion von grundlegenden Fehlern, wie zum Beispiel sog. „Race Conditions“, möglichen Pufferüberläufen, Zugriffen außerhalb gültiger Grenzen bei Arrays oder potentiellen Speicherlecks. Im Anschluss erfolgte eine funktionale Verifikation im Rahmen von Testsimulationen, bei denen mit Hilfe spezifischer Stimuli oder Stimulisequenzen das korrekte Verhalten der ausgeführten Module manuell überprüft wird. Die Referenz für die Bewertung bildet hierbei das gewünschte oder bspw. im Rahmen eines Standards definierte Verhalten.

Zeitgleich wurde mit Hilfe der Werkzeugsammlung *Valgrind* [101] das Laufzeitverhalten der implementierten Softwarekomponenten untersucht. Im Vergleich zur statischen Codeanalyse, finden hier die tatsächlichen Aufrufe von Methoden und Operatoren Berücksichtigung. Dadurch lassen sich beispielsweise nichtinitialisierte Variablen, nicht freigegebene Speicherbereiche (Speicherlecks) identifizieren, die bei der statischen Analyse nicht zweifelsfrei als solche bestimmt werden können. Weiterhin ist es so möglich Speicherzugriffsverletzungen, zu detektieren, die statisch ebenfalls nicht zwingend erkennbar sind und bei Ausführung zum sofortigen Abbruch der Simulation führen, wodurch die Lokalisierung des Problems erschwert oder sogar verhindert wird. Insbesondere Speicherlecks stellen für die potentiell lang laufenden Simulation ein schwerwiegendes Problem dar, da hierdurch der zur Verfügung stehende Arbeitsspeicher komplett belegt werden kann und anschließend eine Auslagerung auf den Massenspeicher für eine extrem verlangsamte Ausführung sorgt. Weiterhin lässt sich das implementierte Programm mit den Werkzeugen von Valgrind einem Profiling unterziehen, wodurch performancekritische Quelltextabschnitte identifiziert und schließlich verbessert werden können.

Dieser, in Abbildung 5.3 dargestellte, Ablauf fand sowohl bei den Modul- als auch den Integrationstest wiederholt Anwendung.

5.2 Modellierung eines funkbasierten Kommunikationssystems mit geringer Verlustleistungsaufnahme

Das Simulationsmodell des funkbasierten Kommunikationssystems beinhaltet Implementierungen der Bitübertragungs-, der Datensicherungs-, der Vermittlungs- und der Anwendungsschicht. Funktionen von Transport-, Sitzungs- und Darstellungsschicht sind nicht integriert (siehe Abbildung 5.4).

5.2.1 Bitübertragungsschicht

Zu den Aufgaben der Module, die dieser Schicht des OSI-Modells zuzuordnen sind, gehören die exakte Nachbildung der Eigenschaften des Übertragungskanal, der zur Datenübertragung erzeugten Signale und des Transceivers, welcher diese Signale erzeugt sowie empfängt. Zur Modellierung konnten einige Elemente des Modellierungs-Frameworks MiXiM [99] genutzt werden, jedoch nur wenige unverändert.

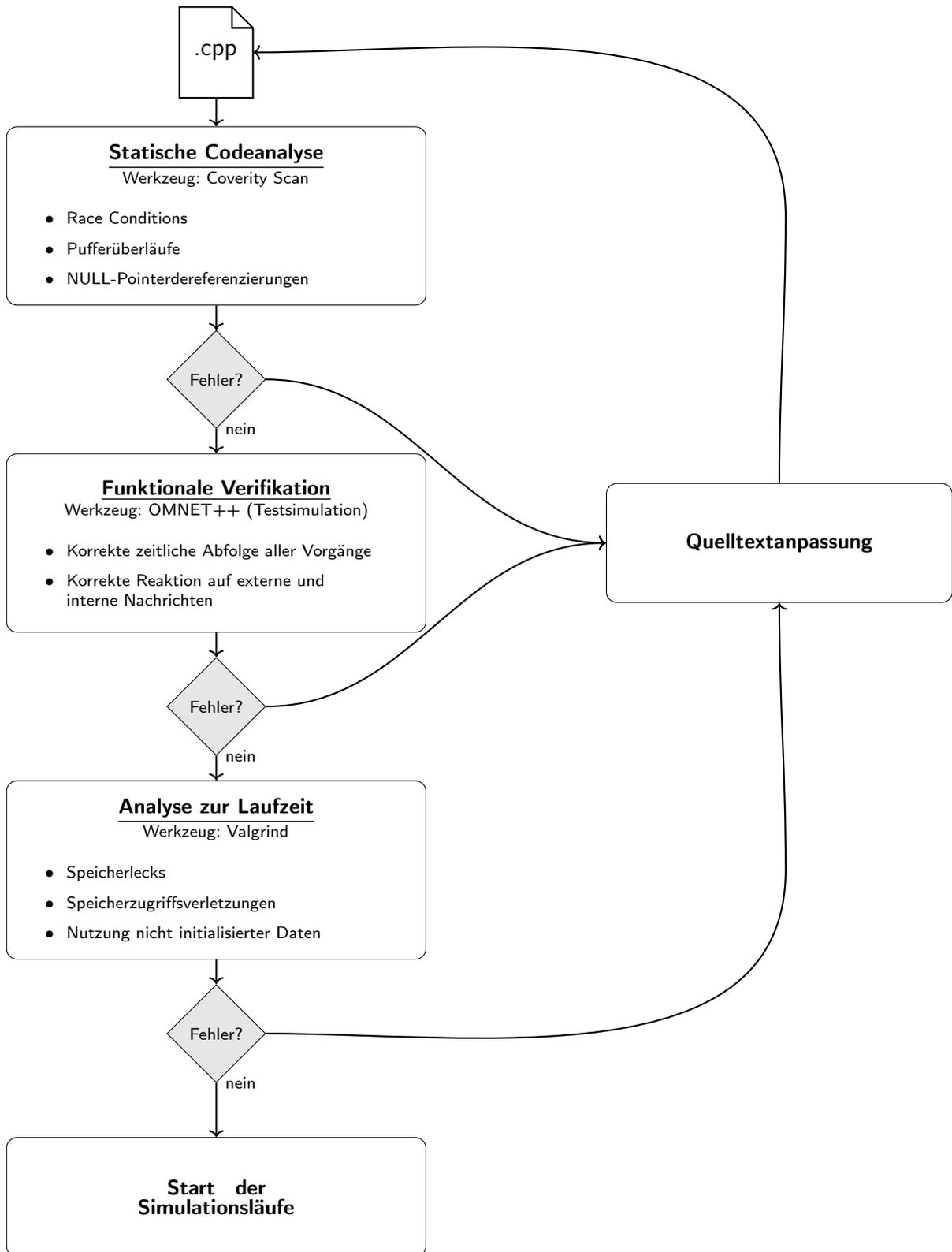


Abbildung 5.3: Ablauf der Verifikation bei der Erstellung der Simulationsmodelle

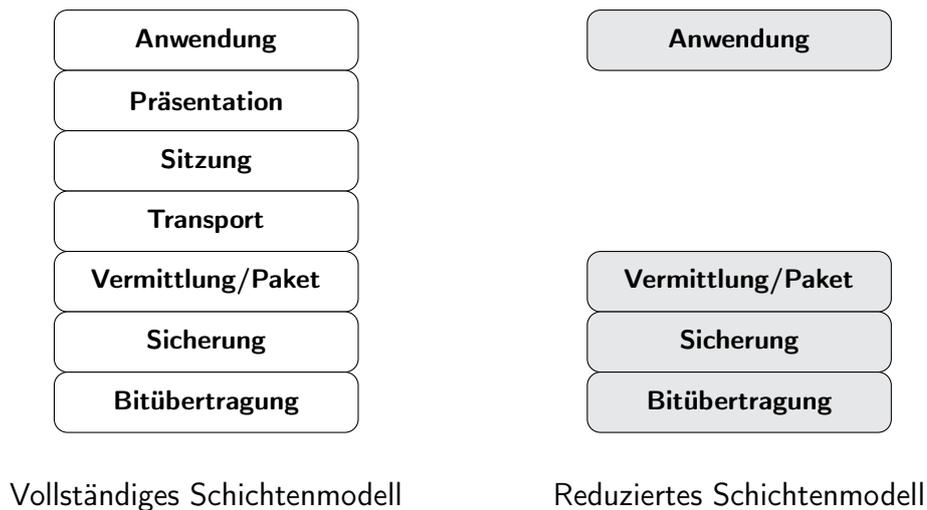


Abbildung 5.4: Gegenüberstellung der im Rahmen des OSI-Modells definierten und der bei der Modellierung des Funkkommunikationssystems genutzten Schichten

Die Änderungen waren zum einen durch die exakte Abbildung des in IEEE 802.15.4 definierten Kommunikationsschemas zwischen MAC und PHY notwendig, zum anderen, um die für das Routing mittels „Minimal Energy Relay Routing“ MERR notwendigen Informationen über Sendeleistung und damit assoziierte Stromaufnahme verfügbar zu machen. Dabei ist die Codebasis thematisch in mehrere Verzeichnisse unterteilt innerhalb derer die verschiedenen Quelltexte für die genutzten Klassen abgelegt sind. Eine Übersicht über die Verzeichnisse sowie die Anzahl der Abhängigkeiten zwischen ihnen gibt Abbildung 5.5.

5.2.2 Sicherungsschicht

Aufgabe der Sicherungsschicht ist es, die Datenintegrität sowie einen geregelten Zugriff auf das geteilte Medium zu gewährleisten (MAC). Letzteres ist hierbei die deutlich komplexere Aufgabe, da die Maximierung der über den Kanal realisierbaren Übertragungsbandbreite mit einer fairen Verteilung dieser in Einklang gebracht werden muss. Die Rahmen des erstellten Simulationsmodells implementierte Sicherungsschicht basiert auf dem Standard IEEE 802.15.4, welcher einen wahlfreien, über Zeitscheiben/„Slots“ synchronisierten Zugriff auf das Übertragungsmedium definiert. Auch hier konnten durch MiXiM zur Verfügung stehende Klassen genutzt werden, die Mehrheit davon jedoch nur nach vorheriger Modifikation. Dies war bspw. notwendig, um die im Standard definierten Zugriffs- bzw. Kommunikationsmethoden vollständig abzubilden. Ebenso wurde der „Guaranteed Time Slot“-Mechanismus (GTS) integriert, der es erlaubt, einzelnen wenigen Teilnehmern einen konstanten Bandbreitenanteil zu garantieren. Da gemäß IEEE 802.15.4 eintreffende Datenrahmen oder Kommandos aus der nächsthöheren Schicht zustandsabhängig interpretiert werden, erfolgte eine Umsetzung in der Struktur von hierarchischen endlichen Zustandsautomaten. Einer von diesen bildet den globalen Zustand der Instanz der Sicherungsschicht ab, während Hilfsautomaten die Abarbeitung von Sequenzen bewerkstelligen, die durch den Empfang bestimmten Kommandos, respektive Datenrahmen, initiiert werden. Beispiele hierfür sind der Start eines neuen PANs, wodurch die Instanz der Sicherungsschicht die Rolle des PAN-Koordinators einnimmt, oder die Assoziation mit einem solchen.

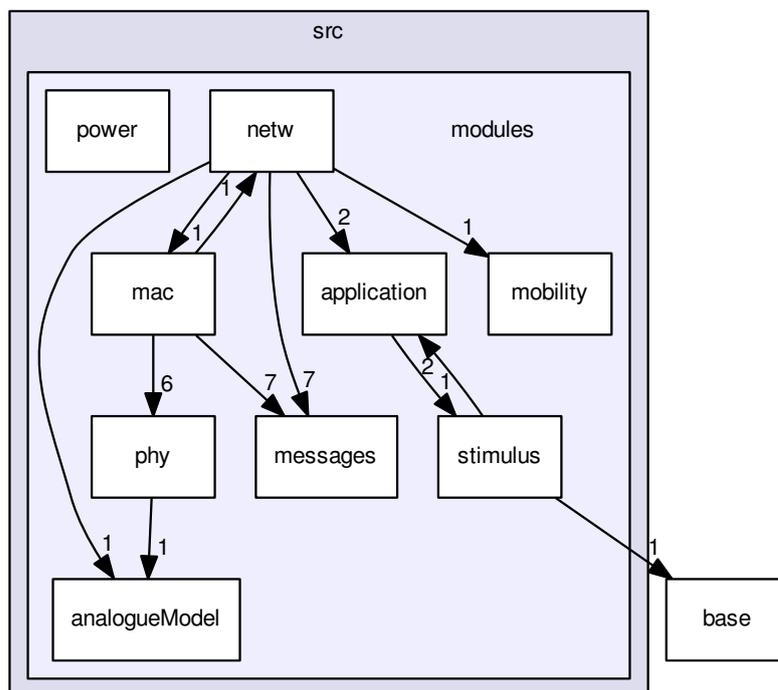


Abbildung 5.5: Übersicht über die Quelltextverzeichnisse für die zur Simulation des Funkkommunikationssystemen genutzten Klassen, wobei die Anzahl der Abhängigkeiten zwischen diesen angegeben ist.

5.2.3 Vermittlungsschicht

Ist es unmöglich Daten direkt an den Empfänger zu übertragen, da sich dieser beispielsweise nicht in Reichweite der Funkübertragung befindet, müssen diese weitergeleitet werden. Die Auswahl zielführender Weiterleitungsknoten erfolgt über die innerhalb der Vermittlungsschicht operierenden Routingalgorithmen. Zum Zeitpunkt der Erstellung des Simulationsmodells war keiner der genutzten Algorithmen nutzbar implementiert. Später wurde eine aus dem Jahr 2002 stammende Implementierungsvariante von AODV [102] unter dem Namen *aodvuu* verfügbar gemacht.

Die Umsetzung der Routingalgorithmen AODV, Fluten, MERR und EEMRP erfolgte monolithisch, wobei die Klasse Fluten die geringste und die Klasse EEMRP die größte Komplexität aufweist. Damit das Routing mittels MERR erfolgen kann, muss indirekt über die Sicherungsschicht hinweg, ein Zugriff auf die als Eigenschaft in der Instanz der in der Simulation genutzten Transceiver-Klasse hinterlegten diskreten Sendeleistungsstufen sowie die damit verbundene Stromaufnahme erfolgen. Zusätzlich werden die geographischen Positionen aller verfügbaren Netzwerkknoten in Form einer statischen Liste benötigt. Diese sich aus dem Einsatz von MERR ergebenden Anforderungen an die darunter liegenden Schichten stellen einen Sonderfall dar. Oberhalb der Vermittlungsschicht existieren derartige Abhängigkeiten nicht. Zur Kommunikation mit der Medienzugriffs-/Datensicherungsschicht, kommt das durch IEEE 802.15.4 vorgeschriebene Kommunikationsschema zum Einsatz, welches Verwaltungs- und Datennachrichten

beinhaltet. Die Kommunikation mit der Anwendungsschicht erfolgt konsequent auf identische Art und Weise und entspricht so auch dem von den OMNeT++-Entwicklern empfohlenen Schema.

5.2.4 Anwendungsschicht

Da der Anwendungsschicht im Rahmen der Simulation nur reaktive Funktionen zukommen und die meisten durch das zentrale Stimulusmodul ausgelöst werden, weist die Anwendungsschicht lediglich eine Klasse auf. Ihr wesentlicher Zweck ist die Anregung von Netzwerkübertragungen, d.h. einer Netzwerklast. Ein großer Anteil der Funktionalität entfällt zudem auf die Auswertung von Antworten und Antwortzeiten sowie deren Protokollierung zur späteren Analyse.

5.3 Modellierung eines optischen Kommunikationssystems

Da die wesentlichen Elemente der Modellierung einzelner Schichten bereits in den vorangegangenen Abschnitten erläutert worden sind, werden im Folgenden nur die für das optische Kommunikationssystem spezifischen diskutiert.

Die Modellierung erfolgte mit dem Fokus auf der Erstellung eines detailgenauen Modells der Bitübertragungsschicht. Zum einen existierten zum Zeitpunkt der Erstellung keine bekannten alternativen Implementierungen. Zum anderen können für die Umsetzung der höheren Schichten bereits im Kontext des Funkkommunikationssystems erstellte Module genutzt werden.

5.3.1 Bitübertragungsschicht

Die physikalisch korrekte Modellierung von optischen Punkt-zu-Punkt-Verbindungen stellt den Hauptteil des Implementierungsaufwands dar. Sie beinhaltet die positions- und orientierungsrichtige Berechnung des auftretenden Pfadverlusts, wobei die Öffnungswinkel von Emitter-LED und Empfänger-Photodiode berücksichtigt werden müssen. Im Kontext der Modellierung werden nur Übertragungen bei direkter Sichtverbindung berücksichtigt, jedoch keine ein- oder mehrfach reflektierten. Das Pfadverlustmodell berücksichtigt dabei die Opakheit von Objekten, welche sich innerhalb der Sichtlinie befinden. So ist es beispielsweise möglich, Innenwände zu spezifizieren, um den Einfluss des Grundrisses auf das Kommunikationsverhalten des simulierten Systems zu untersuchen, respektive dieses an den Grundriss anzupassen.

Pfadverlustmodelle

Das in Kapitel 2.2.1 beschriebene Kanalmodell wurde als Pfadverlustmodell umgesetzt. In Verbindung mit der konfigurierten Abstrahlleistung sowie der gewählten Modulationsart, lässt sich die auf die Photodiode einfallende optische Leistung errechnen. Mittels dieser lässt sich eine Bitfehlerwahrscheinlichkeit bzw. -rate errechnen, die wiederum zur Bestimmung einer Paketfehlerrate, respektive des Durchsatzes auf Anwendungsebene genutzt werden kann.

Transceivermodell

Eine Schwierigkeit bei der Umsetzung des optischen Transceivers bestand in der Aufteilung der Signalwege für den Empfangs- bzw. den Sendepfad. Bei den üblicherweise mittels OMNeT++ simulierten drahtlosen Netzwerkadaptern werden Signale über die gleiche oder die gleichen Antennen empfangen und ausgesendet. Dies ist bei der modellierten optischen Kommunikation nicht der Fall. Die optische Leistung wird über eine LED, mit oder ohne nachgeschaltete Optik, emittiert und von einer Photodiode, ebenfalls mit oder ohne vorgeschaltete Optik, sensiert. Diese lassen sich zwar ähnlich einer Antenne modellieren, aber die Trennung der Signalwege bewirkt, dass keine der in INET vorhandenen Klassen, die zur Beschreibung des Verhalten eines Netzwerkadapters genutzt werden, im Rahmen der Implementierung verwendet werden konnte.

Vorwärtsfehlerkorrektur

Die als ein Beispiel für ein Vorwärtsfehlerkorrekturverfahren umgesetzte RS-Kodierung wird bei jedem Empfang eines Datenrahmens zur Überprüfung der nach Dekodierung resultierenden Bitfehlerwahrscheinlichkeit genutzt. Die Berechnung erfolgt dabei statistisch korrekt, d.h. die Bitfehlerwahrscheinlichkeit ergibt sich aus dem Quotienten aller möglichen fehlerhaften Bitstellen und aller korrigierbaren Sequenzen von Bitstellen (siehe Formel 2.31). Die Berechnung des Binomialkoeffizienten erfolgte unter Nutzung des Ansatzes der fallenden Fakultäten. Andernfalls führt die gemäß IEEE 754 auf 11 Bit beschränkte Länge des zur Gleitkommadarstellung genutzten Exponenten in Verbindung mit der großen Anzahl von Bitstellen zu einem Überlauf.

5.3.2 Höhere Schichten

Um die für das Modell notwendigen höheren Schichten umzusetzen, wurden keine vollständig neuen Klassen implementiert. Es erfolgten stattdessen Anpassungen an den existenten Klassen sowie die Kombination von Ethernet-Modul, Warteschlange und optischer Bitübertragungsschicht zu einem Netzwerkadapter.

Modellgestützte Parameteroptimierung von Drahtloskommunikationssystemen

All models are wrong, but some are useful. (– George E.P. Box, 1976, brit. Statistiker)

Die Anwendung der in den Kapiteln 5.2 und 5.3 beschriebenen Simulationsmodellen zur Analyse und Optimierung von Kommunikationssystemen wird innerhalb dieses Kapitels anhand von zwei Anwendungsbeispielen illustriert. Die beiden Anwendungsbeispiele stellen dabei gegensätzliche Anforderungen. Im ersten Anwendungsbeispiel, einer funkvernetzten Schließanlage, stehen vor allem eine lange Batterielebensdauer, d.h. eine geringe Leistungsaufnahme, und gegebenenfalls eine geringe Übertragungsverzögerung im Vordergrund. Im zweiten Anwendungsbeispiel, einer beleuchtungsintegrierten optischen Kommunikationslösung, stellen vor allem der erzielbare Durchsatz und die Mobilität der Nutzer die maßgeblichen Bewertungs- bzw. Optimierungskriterien dar. Um trotz der Vielzahl der gegenseitigen schichtübergreifenden Abhängigkeiten die Individualbeiträge einzelner Veränderungen innerhalb der involvierten Netzwerkschichten aufzeigen zu können, erfolgt eine schichtorientierte Einzelbetrachtung.

6.1 Anwendungsbeispiel funkvernetzte Schließanlage

Ein Beispiel für funkvernetzte Infrastruktur im Innenbereich sind elektronische Schließanlagen. Hierbei erfolgt die Authentifikation des Schließberechtigten nicht durch einen mechanischen Schlüssel, sondern bei den meisten verfügbaren Systemen kontaktlos durch einen drahtlos sendenden aktiven Transponder (s. Abbildung 6.2) oder ein passives Identifikationsmedium (z.B. Karte gemäß ISO/IEC 7816). Kontaktbehaftete Lösungen haben sich aufgrund der damit verbundenen Verschleißerscheinungen nicht durchgesetzt. Die Information über die tatsächliche Schließberechtigung des Nutzers liegt dabei entweder lokal, d.h. im Schloss selbst (Offline-System) oder an zentraler Stelle vor (Online-System). Weiterhin können die Schließzylinder zum Austausch von Informationen untereinander ein Netzwerk aufbauen (Peer-to-Peer/P2P) oder Teil eines Infrastrukturnetzwerks werden, das entweder durch

eigens dafür vorgesehene und netzspannungsversorgte „Access Points“ (AP) oder bereits vorhandene Geräte (z.B. APs für IEEE 802.11) bereitgestellt wird.

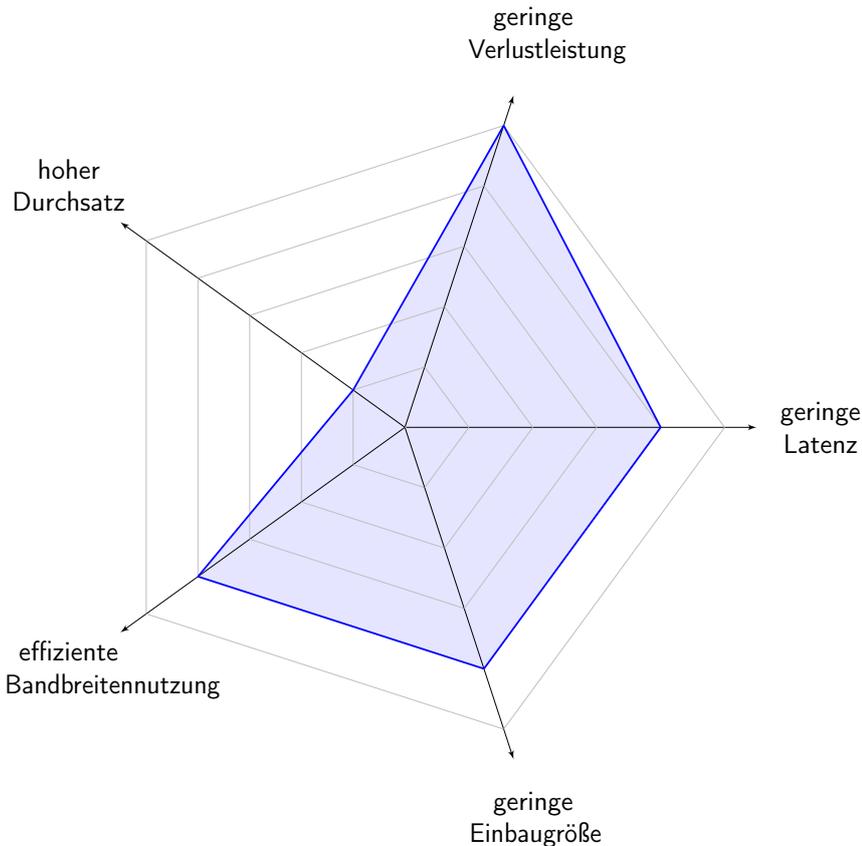


Abbildung 6.1: Anforderungsprofil des Anwendungsbeispiels „funkvernetzte Schließanlage“

Da die in den Schließzylinder bzw. das Schloss eingebrachten elektronischen Komponenten eine Energieversorgung benötigen, muss zusätzlich eine Spannungsquelle integriert werden. Üblicherweise kommen hier Batterien zum Einsatz, deren Kapazität in Verbindung mit der durch die Elektronik aufgenommenen Verlustleistung das Batteriewechselintervall vorgibt. Deshalb stellt die Verlustleistungsaufnahme im Folgenden ein wesentliches Bewertungskriterium dar (siehe Anforderungsprofil in Abbildung 6.1). Wird die Schließberechtigung nicht lokal, das heißt vom Schließzylinder/Schloss selbst überprüft, muss die Latenz für die Übermittlung der Schließanfrage an eine zentrale Stelle sowie deren Antwort gering sein. Andernfalls erfährt der Benutzer vor der zu schließenden Tür eine deutliche Verzögerung, welche die Akzeptanz des elektronischen Schließsystems erheblich verringert. Weiterhin ist der für die Installation eines solchen Systems notwendige Aufwand zu berücksichtigen, das heißt beispielsweise, ob und wie viele APs installiert werden müssen, um die Funktionalität zu garantieren und gegebenenfalls auftretende Verzögerungszeiten zu reduzieren. Im Folgenden werden daher die verschiedenen Systemrealisierungen bzw. Optimierungsmaßnahmen im Bezug auf Verlustleistungsaufnahme respektive Batterielebensdauer, Latenz sowie die Nützlichkeit, respektive Notwendigkeit, zusätzlicher APs bewertet.

6.1.1 Systemaufbau

Ein elektronisches und funkvernetztes Schließanlagensystem besteht grundsätzlich aus den elektronischen Schließeinheiten (Schließzylinder oder Schloss) und einem „Access Point“ (AP), der je

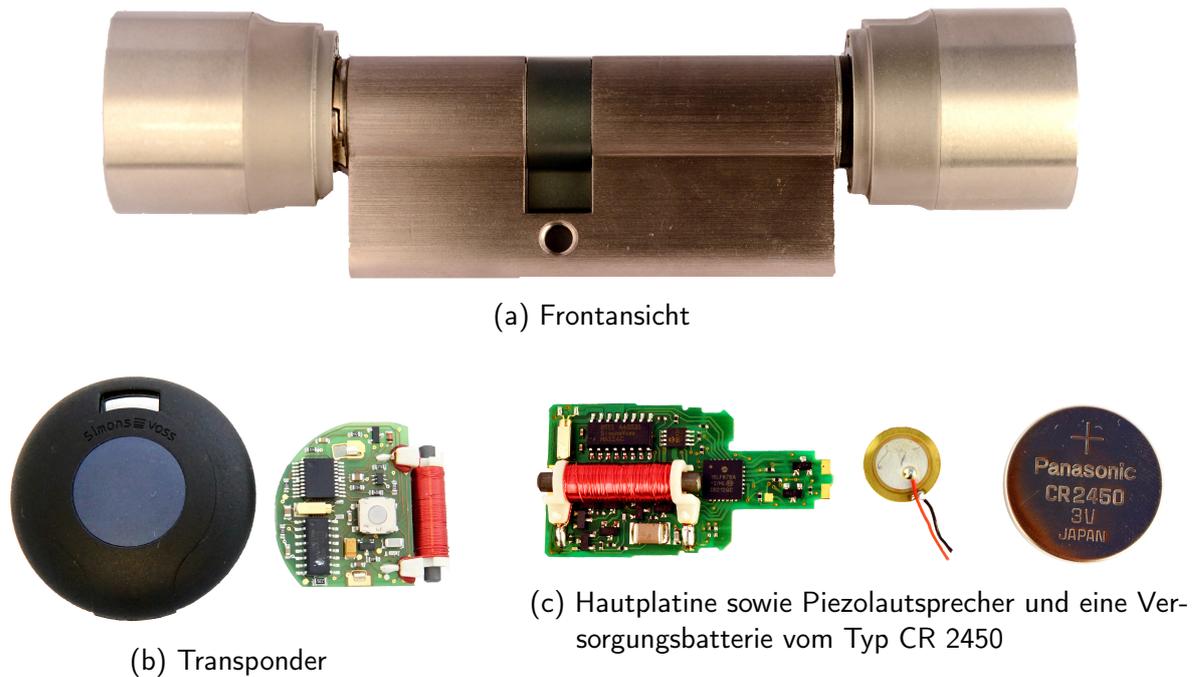


Abbildung 6.2: Beispiel für einen elektronischen Schließzylinder inklusive Transponder (Doppelknäufzylinder 3061-FD und Transponder 3064 der Firma SimonsVoss)

nach Auslegung entweder als protokollkonvertierendes „Gateway“ (OSI-Schichten 1 – 4) oder als medienkonvertierende „Bridge“ (OSI-Schichten 1 – 2) umgesetzt sein kann.

Der beispielhafte Aufbau eines elektronischen Schließzylinders ist in Abbildung 6.3 dargestellt und entspricht der im Rahmen der durchgeführten Netzwerksimulationen verwendeten Schließzylinderkonfiguration. Sie besteht aus einer Kommunikationseinheit für die Benutzerinteraktion, einer Einheit für die allgemeine Funkkommunikation, einer Datenverarbeitungseinheit inklusive optionalem externen Speicher, einer Kupplung mit dem Schließmechanismus oder einem elektronisch aktuierten Sperrstift, sowie einer Energieversorgung. Alternativ oder ergänzend kann ein Generator („Energy Harvesting“) verbaut werden, um beispielsweise die zum Schließen notwendige elektrische Energie aus der Umwandlung beim Schließvorgang verfügbarer kinetischer Energie (Drücken des Türgriffs, Drehen von Zylinder oder Schlüssel, Einstecken des Schlüssels) zu beziehen. Weiterhin kann eine elektrische Fremdversorgung durch das Schließmedium stattfinden (bspw. batterieversorgter Schlüssel mit Kontakten). Zur Benutzerinteraktion werden üblicherweise optische oder akustische Signalgeber eingesetzt, um Hinweise zum Erfolg des Schließversuchs, Batteriewarnungen oder Störungsmeldungen anzuzeigen.

Der für die Funkkommunikation notwendige Teil der Bitübertragungsschicht wird durch das Modul „Nic802154sync“ abgebildet, welches nach den Spezifikationen des CC430F6137 von Texas Instruments parametrisiert wurde. Hier sind vor allen Dingen die Parameter Empfindlichkeit und maximale Sendeleistung von Bedeutung, da sie das zur Verfügung stehende „Link Budget“ und somit die Reichweite der Funkverbindung vorgeben. Weiterhin wirken sich die für den Zustandswechsel des Transceivers vorgegebenen Umschaltzeiten aus, wobei jeder Betriebszustand durch eine individuelle Verlustleistungsbzw. Stromaufnahme gekennzeichnet ist (siehe Tabelle 6.1). Bei jedem Zustandswechsel wird dem Batteriemodul der aktuell simuliert fließende Strom mitgeteilt und zusätzlich als Last für die spätere Auswertung verbucht.

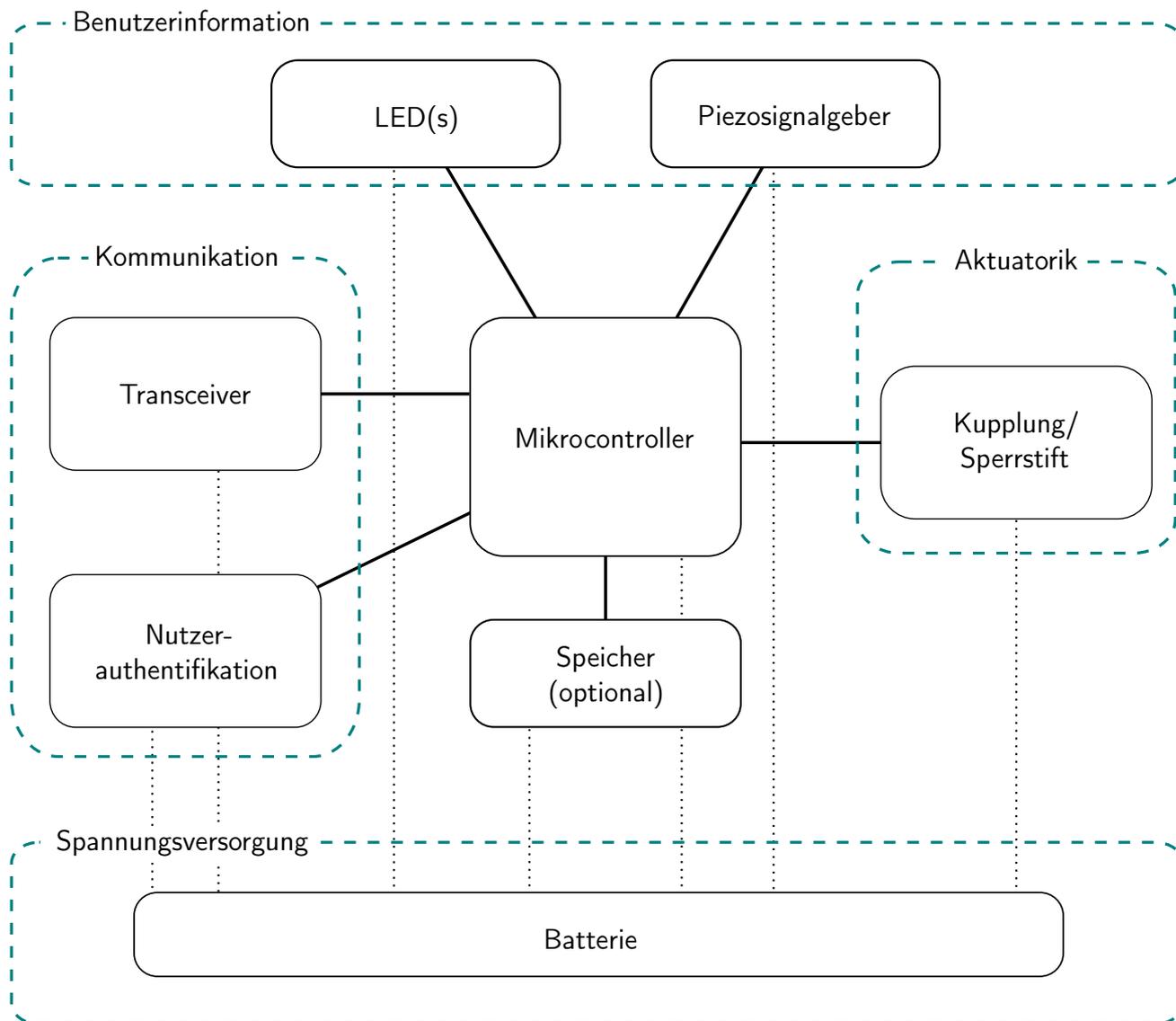


Abbildung 6.3: Grundlegender Aufbau eines digitalen Schließzylinders

Der Medienzugriff erfolgt wie in Kapitel 5.2.2 beschrieben nach dem von IEEE 802.15.4 vorgegebenen Schema. Dazu erwachen alle Schließzylinder innerhalb eines festen Zeitintervalls von 0,1 s aus einem nonkommunikativen, aber energetisch vorteilhaften Schlafzustand und sind dann für eine vorgegebene Zeit in der Lage, mit anderen Schließzylindern oder einem AP zu kommunizieren. Wie in den späteren Analysen gezeigt, beeinflusst die Wahl dieser Wachzeit t_{wach} erheblich die Funktionsfähigkeit des gesamten Netzwerks.

Falls keine direkte Funkverbindung zwischen Sender und Empfänger hergestellt werden kann, muss eine Route für die zielgerichtete Weiterleitung der Pakete gefunden werden. Diese Aufgabe übernimmt ein in jedem Knoten vorhandenes Routing- bzw. Vermittlungsmodul, wodurch verschiedene Routingalgorithmen eingesetzt und miteinander verglichen werden können.

Die zu versendenden Nachrichtenpakete werden vom Anwendungsmodul generiert, welches die in einem Schließzylinder auftretenden Ereignisse simuliert. Dabei handelt es sich um folgende Aktionen:

- Schließenanfrage durch Benutzer
- Aktualisierung der Schließberechtigungen (nur im Falle eines Offline-Systems)

- Abfrage des Türstatus
- Anzeige eines Alarms

Zusätzlich bietet jedes in einem Knoten existierende Modul der Anwendungsschicht eine Schnittstelle, um die betreffenden Aktionen auszulösen. Dies erfolgt durch das in jeder Simulation nur einmalig instanziierte Modul „Stimulus“, welches die angeführten Ereignisse zentralisiert in den jeweiligen Modulen initiiert. Dies erlaubt eine exakte Kontrolle über die zur Simulation verwendeten Stimuli und erhöht die Vergleichbarkeit, respektive Reproduzierbarkeit, der einzelnen Simulationsläufe.

6.1.2 Simulationsszenario

Die im Anwendungsszenario simulierte Schließanlage besteht aus 28 Türen, die jeweils in einem Abstand von 5 m beidseitig entlang eines virtuellen Flures mit einer Länge von 66 m verteilt sind. Jede Tür trägt dabei einen elektronischen Schließzylinder. Zusätzlich werden in Abhängigkeit von der Anzahl bzw. Position verfügbarer APs vier verschiedene Szenarien unterschieden. Beim Szenario „0AP“ existiert zwar ein AP, allerdings befindet sich dieser an einem Ende des simulierten Flures. Dadurch findet die Funkkommunikation fast ausschließlich zwischen den Schließzylinderknoten statt, wodurch ein sog. P2P-Netzwerk entsteht. Auf diese Weise wird ein Anwendungsfall dargestellt, bei dem weitestgehend auf das zur Verfügung stellen von zusätzlicher Infrastruktur für das Schließsystem verzichtet wurde. Bei den folgenden Szenarien „1AP“, „2AP“ und „3AP“ sind die „Access Points“ jeweils entlang der Längsachse symmetrisch um den Flurmittelpunkt verteilt, die Position der Schließzylinderknoten bleibt jeweils unverändert. Zur Verdeutlichung sind die Szenarien in Abbildung 6.5 dargestellt.

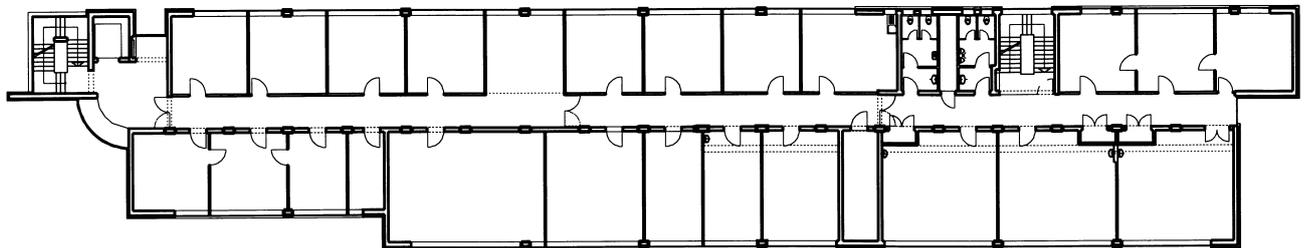


Abbildung 6.4: Grundriss des als Beispiel dienenden Gebäudeflurs mit einer Gesamtlänge von 66 m, einer Flurbreite von 2 m sowie insgesamt 28 Türen

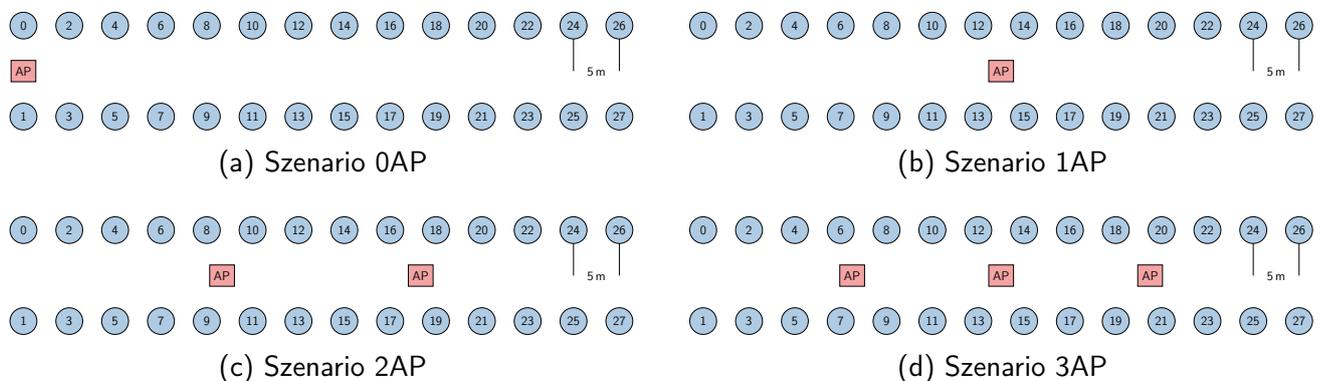


Abbildung 6.5: Anordnung der 28 Schließzylinderknoten und variierter Anzahl und Position der „Access Points“ in den Simulationsszenarien „0AP“ bis „3AP“

Ablauf der Simulation

Nach Starten des Simulationsdurchgangs, beginnen die in jedem Knoten vorhandenen Module mit der Selbstinitialisierung, um in einen betriebsbereiten Zustand zu wechseln. Je nach spezifiziertem Routingalgorithmus können hier bereits Pakete generiert werden, die der proaktiven Routenfindung dienen. Im Anschluss beginnt das Stimulus-Modul damit, in Abhängigkeit der definierten Intervalle, die jeweiligen Ereignisse in den Anwendungsmodulen ausgewählter Schließzylinderknoten auszulösen. Bei einem simulierten Online-System wird durch jede dieser Aktionen mindestens eine Nachricht innerhalb des Schließsystemnetzwerks übertragen, die jeden an der Übertragung beteiligten Knoten sowohl energetisch belastet, als auch das gemeinsam genutzte Medium belegt. Wird ein Offline-System simuliert, werden Schließenfragen lokal beantwortet und nicht durch das Netzwerk weitergeleitet. Es erfolgt dann jedoch eine tägliche Aktualisierung des lokal gespeicherten Schließplans.

Schließenfrage Handelt es sich bei der simulierten Schließenanlage um ein Online-System, wird durch das Auslösen einer Türöffnungs- bzw. Türschließenfrage durch einen virtuellen Nutzer zunächst die vom Batteriemodul verwaltete Restenergiemenge um den für die Benutzerinteraktion festgelegten Energiebetrag reduziert. Danach generiert das Anwendungsmodul ein Paket mit festgelegter Größe, welches die Identifikatoren von Benutzer und zu schließender Tür erhält. Dieses Paket muss nun einen der verfügbaren APs erreichen, wodurch der Übergang in ein nicht simuliertes kabelgebundenes Kommunikationsnetz erfolgt. Nachdem es die zentrale Einrichtung erreicht hat, welche über die Schließenfrage entscheidet, wird dort eine passende Antwort generiert. Dieses Antwortpaket wird nun von dem AP mittels Funkübertragung emittiert, welcher auch die dazugehörige Anfrage empfangen hat. Die Übertragung des Antwortpakets an den ursprünglich anfragenden Schließzylinderknoten belastet die involvierten Knoten erneut energetisch und belegt das zur Übertragung notwendige Medium. Nach dem Eintreffen der Antwort beim anfragenden Schließzylinderknoten, kann die vom Benutzer wahrgenommene Verzögerung protokolliert werden. Weiterhin wird der virtuellen Batterie der für den Schließvorgang notwendige Energiebetrag entzogen. Sollte entweder die Übermittlung von Anfrage oder Antwort scheitern, wird von der Anwendungsschicht im anfragenden Schließzylinder explizit keine erneute Anfrage übermittelt. Stattdessen wird die Schließenfrage als gescheitert protokolliert. Dieses Vorgehen erlaubt es, Aussagen über die allgemeine Funktionsfähigkeit des Schließenanlagensystems mit der gewählten Parametrierung zu treffen. Wiederholungen auf der Ebene von Vermittlungs- oder Medienzugriffsschicht sind davon unabhängig weiterhin möglich.

Schließplanaktualisierung Zusätzlich zur Schließenfrage kann im Falle eines Offline-Systems die Aktualisierung der lokal gespeicherten Schließberechtigungen als Ereignis simuliert werden. Dabei wird in festgelegten Zeitintervallen die Übertragung einer Datenmenge von 2048 B an einen sequentiell ausgewählten Schließzylinderknoten gestartet. Anschließend kann überprüft werden, wie viele der Aktualisierungen initiiert und wie viele erfolgreich beendet wurden. Wird ein Online-System simuliert, findet keine Schließplanaktualisierung statt, d.h. dieses Ereignis wird seitens des Stimulusmoduls nie ausgelöst.

Statusmeldung Um fortlaufend über den Batteriezustand der elektronischen Schließzylinder informiert zu sein und so einerseits den Betrieb gewährleisten, als auch den Batterieaustausch planen zu können, kann dieser durch einen AP abgefragt werden. Zusätzlich lässt sich der aktuelle Verschlusszustand nach dem gleichen Prinzip ermitteln. In beiden Fällen werden auf Anwendungsebene 8 B große Pakete als Anfrage und Antwort versendet.

Alarm Die letzte übertragene Art von Nachrichten ist die Alarmmeldung. Hierbei handelt es sich um eine 8 B lange Nachricht, deren Übertragung vom Stimulus-Modul in einem zufällig ausgewählten Schließzylinder ausgelöst wird. Anschließend wird diese zu einem der im Netzwerk verfügbaren APs weitergeleitet und die übertragungsbedingte Verzögerung wird protokolliert. Die Verzögerung der kleinen und nur in eine Richtung übertragenen Nachrichten spiegelt die Kommunikationsfähigkeit bzw. die aktuelle Auslastung des Funknetzwerks wider. Die praktische Entsprechung dieses Ereignisses ist beispielsweise eine Zweitnutzung des Schließzylinders als Alarmanlagensensor oder die Meldung einer akuten Störung bzw. eines Manipulationsversuchs.

Alle Simulationsdurchgänge enden sobald der erste Knoten sein vorgegebenes Energiebudget erschöpft hat. Dabei werden neben der simulierten Zeit auch die verbleibende Restkapazität in den restlichen Schließzylinderknoten sowie die Anzahl aller eintreffenden und abgehenden Nachrichten/Pakete der einzelnen Module protokolliert. Die für alle Simulationen als Basis fungierenden Parameterwerte sind in Tabelle 6.1 zusammengefasst.

Um den Entwurfsraum nahezu vollständig abzubilden, wurden Simulationsläufe für alle Kombinationen von eingesetzten Medienzugriffsverfahren, Routingalgorithmen sowie Umgebungsszenarien durchgeführt. Dabei wurde für das Medienzugriffsverfahren gemäß IEEE 802.15.4 (siehe Abschnitt 2.1.4) jeweils eine Studie für den Parameter *Wachzeit* t_{wach} bzw. den resultierenden *Duty-Cycle* durchgeführt, um den vermuteten Zusammenhang zu Operabilität des Systems, Lebensdauer und Verzögerung des Schließvorgangs im Online-Schließsystem zu untersuchen. Es erfolgte eine Variation in 29 Schritten, d.h. von 0,001 s bzw. 1 % Duty-Cycle bis 0,1 s bzw. 100 % Duty-Cycle. In Verbindung mit Sonderuntersuchungen für ein alternatives SoC, das Aufwachintervall, den Systementwurf (offline oder online) und zwei Möglichkeiten zur Nutzerauthentifikation ergab sich so insgesamt eine Anzahl von 1392 Simulationsläufen, deren Zusammensetzung in Tabelle 6.2 aufgelistet ist.

Im Folgenden werden Auswirkungen der einzelnen Parameter auf die Bewertungsmetriken Zuverlässigkeit/Operabilität, Lebensdauer und kommunikationsbedingte Verzögerung des Schließvorgangs dargestellt.

6.1.3 Optimierungen in der Bitübertragungsschicht

Pfadverlust

Eine möglichst exakte Repräsentation der physikalischen Übertragungseigenschaften einer drahtlosen Kommunikationsstrecke stellt die Basis für aussagekräftige Simulationsergebnisse dar. Zur Bewertung verschiedener Pfadverlustmodelle bezüglich deren Übereinstimmung mit den auf einem Flur eines Bürogebäudes vorherrschenden Übertragungsbedingungen, wurde eine Messkampagne durchgeführt. Es stellte sich heraus, dass bei Übertragungen mit ununterbrochener Sichtlinie auch über lange Strecken die Freiraumdämpfung dominiert. Weiterhin zeigten sich Effekte destruktiver und konstruktiver Überlagerung, die besonders in Bereichen mit irregulärer Raumgeometrie, z.B. einem angrenzendem Treppenhaus mit Trennung durch eine Glasfront, stark ausgeprägt waren. Bei Übertragungen ohne Sichtlinie wurde der Pfadverlust von diesen Mehrwegeausbreitungseffekten dominiert, so dass sich positionierungs- und höhenabhängige Schwankungen abzeichneten. Im nachgestellten Alltagsbetrieb, d.h. mit mehreren auf dem Flur verkehrenden Menschen, kam es ebenfalls zu statistischen Schwankungen. Aus diesem Grund wurde für alle Simulationsläufe das Pfadverlustmodell „Log-Normal Shadowing“ gewählt, da es den dynamischen Charakter der veränderlichen Übertragungsbedingungen am besten widerspiegelt. Eine Erläuterung findet sich in Abschnitt 2.1.1.

Tabelle 6.1: Grundlegende Simulationsparameterwerte

Bereich	Parameter	Wert	Kommentar
Stimuli	Intervall Schließenfragen	99,31 s	30 Türöffnungen pro Tag
	Intervall Schließplanaktualisierung	86 400 s	1 Aktualisierung pro Tür pro Tag
	Intervall Statusabfrage	3600 s	1 Statusabfrage pro Tür pro Stunde
	Intervall Batteriestatusabfrage	86 400 s	1 Statusabfrage pro Tür pro Tag
	Intervall Alarm	86 400 s	1 Alarm pro Tür pro Tag
	Verteilung Schließenfragen	Gleichverteilung	
	Verteilung Schließplanaktualisierungen	Gleichverteilung	
	Verteilung Statusabfragen	Sequentiell	
	Verteilung Batteriestatusabfragen	Sequentiell	
	Verteilung Alarme	Gleichverteilung	
Anwendung	Paketgröße Schließenfrage	20 B	
	Paketgröße Schließplanaktualisierung	2048 B	
	Paketgröße Statusabfrage	8 B	
	Paketgröße Batteriestatusabfrage	8 B	
	Paketgröße Alarmmeldung	8 B	
Vermittlung	Routing-Algorithmus	Fluten, AODV, MP, MERR	
	Time-to-Live	4	
Medienzugriff	Aufwachintervall/Zeitrahmen	0,1 s	
	Wachzeit t_{wach}	0,0 s bis 0,1 s	
	Empfangsbestätigungen	ja	
	Maximal Back-off Exponent (maxBE)	5	Standardwert lt. IEEE 802.15.4
	Minimal Back-off Exponent (minBE)	3	Standardwert lt. IEEE 802.15.4
	Größe Paketpuffer	10 Pakete	
Bitübertragung	Länge MAC-Header	24 bit	
	Datenrate	250 kbit/s	
	Trägerfrequenz	868,3 MHz	
	maximale Sendeleistung	10 dBm	
	Empfängerempfindlichkeit	-90 dBm	
	therm. Hintergrundrauschen	-110 dBm	
	Wechsel „Aus“ zu „Bereit“ ¹	809,4 µs	Inkl. PLL-Kalibrierungszeit von 721 µs
	Wechsel „Bereit“ zu „Empfang“ ¹	809,0 µs	Inkl. PLL-Kalibrierungszeit von 721 µs
	Wechsel „Empfang“ zu „Bereit“ ¹	721,0 µs	Inkl. PLL-Kalibrierungszeit von 721 µs
	Wechsel „Bereit“ zu „Senden“ ¹	809,0 µs	Inkl. PLL-Kalibrierungszeit von 721 µs
Wechsel „Senden“ zu „Bereit“ ¹	721,0 µs	Inkl. PLL-Kalibrierungszeit von 721 µs	
Energiequelle	Kapazität	850 mAh	
	Nennspannung	3,3 V	Spannungsverlauf nicht simuliert
	Energiebedarf RFID-Kommunikation (UID) ²	0,5 mW s	Abfrage Transponder-ID
	Energiebedarf RFID-Kommunikation (sicher) ²	409,2 mW s	Abfrage verschlüsseltes Geheimnis
	Energiebedarf Kupplungsbetrieb ²	19,2 mW s	
	Leistungsaufnahme „Aus“ ¹	330 µW	Transceiver abgeschaltet ³
	Leistungsaufnahme „Bereit“ ¹	5,6 mW	
	Leistungsaufnahme „Empfang“ ¹	52,8 mW	bei maximaler Empfindlichkeit
Leistungsaufnahme „Senden“ ¹	108,9 mW	Sendeleistung 10 dBm	
zeitl. Auflösung	1 s		
Pfadverlustmodell	Typ	Log-Normal Shadowing	
	Pfadverlustkoeffizient n	2,6	s. Erläuterung in Abschnitt 2.1.1
	Zufallskomponente X_{σ}	5 dB	

¹ Werte für CC430F6137 von Texas Instruments

² Experimentell ermittelt

³ Die CPU befindet sich dabei im tiefstmöglichen Schlafzustand.

Tabelle 6.2: Übersicht der durchgeführten Simulationsläufe

SoC	Medienzugriffs- verfahren	Aufwach- intervall	Routing- verfahren	System	Nutzer- authentifikation	Szenario				Summe
						0AP	1AP	2AP	3AP	
TI CC430	CSMA/CA (IEEE 802.15.4)	0,1 s	AODV Fluten MERR EEMPR	Online	RFID: UID	29	29	29	29	116
						29	29	29	29	116
						29	29	29	29	116
						29	29	29	29	116
NXP KW0x	CSMA/CA (IEEE 802.15.4)	0,1 s	AODV Fluten MERR EEMPR	Online	RFID: UID	29	29	29	29	116
						29	29	29	29	116
						29	29	29	29	116
						29	29	29	29	116
TI CC430	CSMA/CA (IEEE 802.15.4)	0,1 s	AODV	Offline	RFID: UID	29	29	29	29	116
TI CC430	CSMA/CA (IEEE 802.15.4)	0,1 s	AODV	Offline	RFID: Schlüssel	29	29	29	29	116
TI CC430	CSMA/CA (IEEE 802.15.4)	1 s	AODV	Offline	RFID: UID	29	29	29	29	116
TI CC430	CSMA/CA (IEEE 802.15.4)	10 s	AODV	Offline	RFID: UID	29	29	29	29	116
Gesamtanzahl									1392	

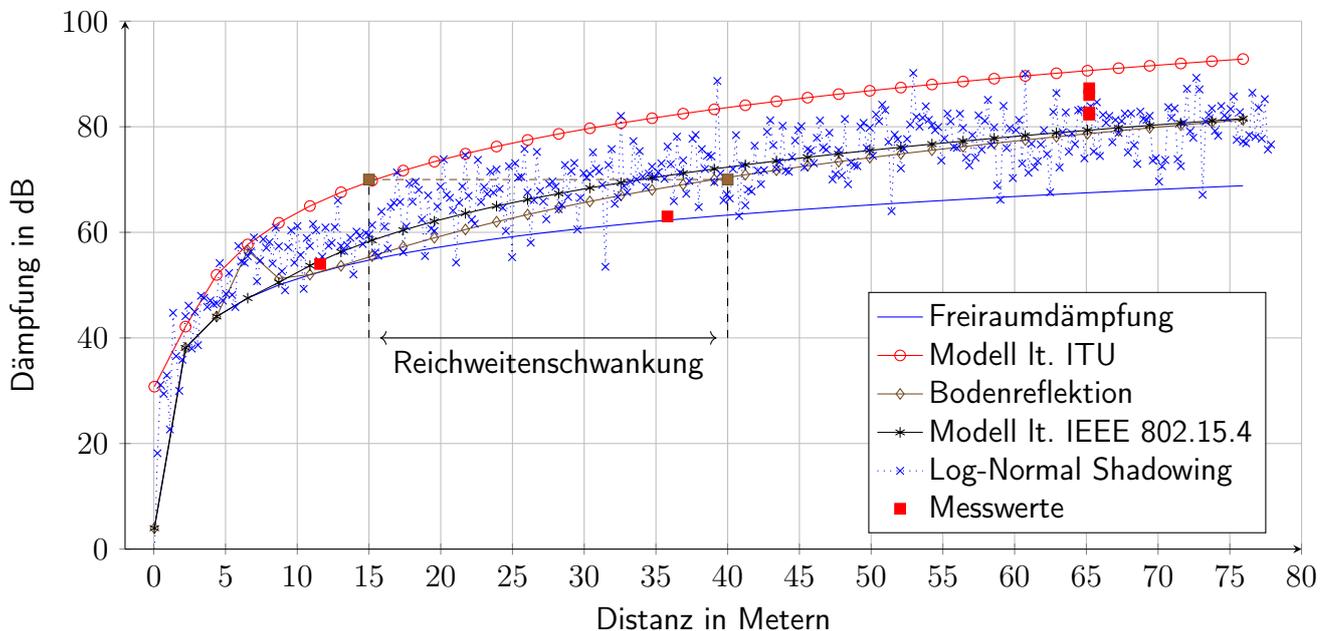


Abbildung 6.6: Vergleich des simulierten Pfadverlusts in Abhängigkeit vom verwendeten Pfadverlustmodell und sowie der Distanz zwischen Sender und Empfänger (Bodenhöhe jeweils 1 m) bei einer Übertragungsfrequenz von 868,3 MHz. (Parameter ITU: $N = 33$, Parameter Log-Normal Shadowing: $n = 2,6$, $X_{\sigma} = 5$ dB)

Hardware-Modell

Das erstellte Simulationsmodell bildet, wie im Abschnitt 5.2.1 beschrieben, das Verhalten existierender Transceiver bzw. SoCs nach. Dabei werden insbesondere die für den Wechsel der Betriebszustände notwendigen Umschaltzeiten, aber auch die aus dem Betriebszustand resultierende Stromaufnahme, respektive die Verlustleistung, berücksichtigt (siehe Tabelle 6.1). Das ermöglicht einen anwendungsspezifischen Vergleich am Markt verfügbarer Lösungen bzw. die Bewertung ihrer Eignung für den Einsatz als Plattform für einen elektronischen Schließzylinder.

Im Rahmen der durchgeführten Untersuchungen, wurden exemplarisch zwei aktuelle SoCs für den Sub-GHz-Bereich verglichen, der CC430F6137 von Texas Instruments und der KW0x von NXP. Beide haben im Bezug auf Sendeleistung, Empfindlichkeit ähnliche Kennwerte, unterscheiden sich aber geringfügig bei der Leistungsaufnahme im Empfangsmodus sowie der Dauer für die Zustandswechsel des Transceivers. Die genauen Werte können Tabelle 6.3 entnommen werden. Es handelt sich um Datenblattwerte, wobei die Kennwerte für die Leistungsaufnahme des CC430F6137 durch Messungen bestätigt werden konnten.

Bei der Auswahl der HW-Plattformen ist anzumerken, dass der Standard IEEE 802.15.4 bei der hier genutzten Übertragungsfrequenz von 868,3 MHz lediglich eine Bitübertragungsschicht mit BPSK-Modulation und effektiver Datenrate von 20 kbit/s definiert wird. Beide HW-Plattformen erlauben allerdings auch die Übertragung mittels QPSK-Modulation und einer Datenrate von 250 kbit/s. Da es das Ziel der Simulationen ist möglichst optimale Umsetzungsvarianten aufzuzeigen und die genannten Parameter aufgrund der kürzeren Übertragungs- und somit Wartezeiten im Anwendungsszenario vorteilhafter sind, wurden diese durchgehend für alle Simulationsläufe verwendet.

Tabelle 6.3: Vergleich geeigneter SoCs im Bezug auf Transceiverumschaltzeiten sowie betriebszustandsabhängige Stromaufnahme bei einer Trägerfrequenz von 868,3 MHz

	Parameter	CC430F6137	NXP KW0x
Umschaltzeit	„Aus“ zu „Betriebsbereit“	809,4 μs ¹	650,0 μs ²
	„Betriebsbereit“ zu „Empfangen“	809,0 μs ¹	213,9 μs
	„Empfangen“ zu „Betriebsbereit“	721,0 μs ¹	–
	„Betriebsbereit“ zu „Senden“	809,0 μs ¹	120,0 μs
	„Senden“ zu „Betriebsbereit“	721,0 μs ¹	–
	„Empfangen“ zu „Senden“	9,6 μs	–
	„Senden“ zu „Empfangen“	21,5 μs	–
	„Aus“ zu „Empfangen“ ³	897,4 μs	863,9 μs
	„Aus“ zu „Senden“ ³	897,4 μs	770,0 μs
Stromaufnahme	„Aus“	0,1 mA	0,1 mA
	„Betriebsbereit“	1,7 mA	1,3 mA
	„Empfangen“	16,0 mA	17,0 mA
	„Senden“ ⁴	33,0 mA	33,0 mA

¹ Beinhaltet PLL-Kalibrierungszeit von 721 μs

² Beinhaltet PLL-Kalibrierungszeit von 151 μs

³ Akkumulation aller Umschaltzeiten, um in den angegebenen Betriebszustand zu wechseln.

⁴ Sendeleistung 10 dBm

Der in Abbildung 6.8 dargestellte Vergleich der resultierenden Lebensdauern für beide SoCs zeigt, dass für Duty-Cycle von mehr als 10 % nahezu kein Unterschied mehr besteht. Bei kleinerem Duty-Cycle,

und somit auch insgesamt längerer Lebensdauer, wirkt sich die geringfügig größere Leistungsaufnahme des KW0x im Empfangsmodus negativ aus. Die um 3,7 % kleinere Umschaltzeit für den Wechsel in den empfangsbereiten Zustand des KW0x, bewirkt zwar einen längeren Verbleib im energetisch günstigen Schlafzustand, verlängert aber auch das zur Verfügung stehende Übertragungsfenster bei grenzwertig kleinen Duty-Cycle-Werten. Infolgedessen können mehr Nachrichten gesendet und empfangen werden, wodurch die energetische Belastung insgesamt steigt und sich in einem Einbruch der Lebensdauer bei ca. 3 % Duty-Cycle ausdrückt. Hier versucht nahezu jeder Knoten nach jedem Erwachen sendend auf den Kanal zuzugreifen. Im operablen Bereich, mit einem Duty-Cycle von 5 %, fällt beim Einsatz des KW0x der erste Knoten bereits nach ca. 41 d aus, mit dem CC430F6137 erst nach rund 47 d.

Weiterhin offenbart der Graph eine Besonderheit des im CC430-SoC enthaltenen Zustandsautomaten, der für die Steuerung des integrierten Transceivers zuständig ist (vgl. Abbildung 2.4). Falls der aktuelle Zustandsübergang noch nicht abgeschlossen ist, werden einlaufende Anfragen für einen erneuten Zustandswechsel ignoriert. Dieses Verhalten ist im Simulationsmodell konsequent identisch implementiert und führt dazu, dass bei zu geringem Duty-Cycle von unter 1,09 %, d.h. weniger als 1,09 ms bei einer Grundintervalllänge von 100 ms, nur jede zweite Zustandswechselanfrage akzeptiert wird. Effektiv stellen sich also ein Duty-Cycle von ungefähr 50 % und somit eine resultierende Lebensdauer von ca. 4,3 Tagen ein.

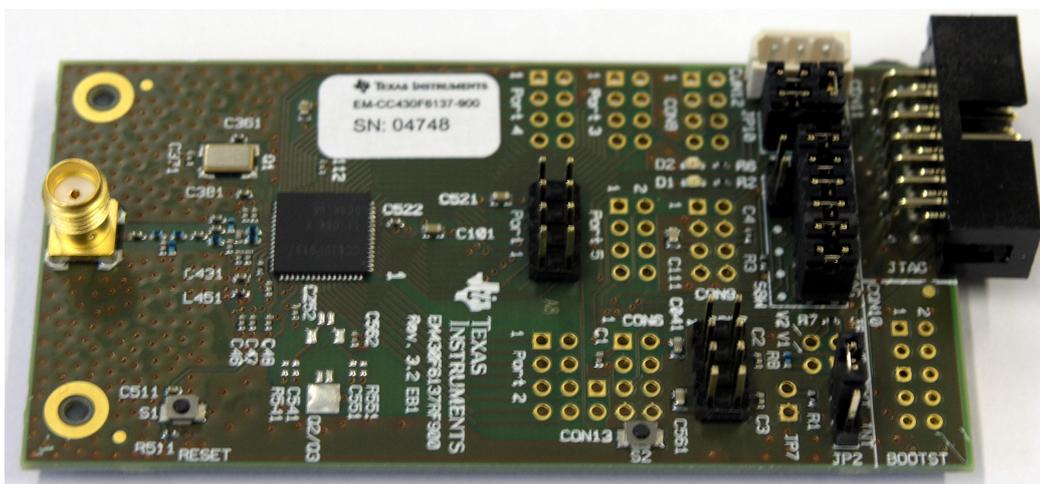


Abbildung 6.7: Die für die Messungen verwendete Entwicklungsplatine EM430F6137 mit CC430F6137-SoC von Texas Instruments (Darstellung ohne $\lambda/2$ -Dipolantenne)

6.1.4 Optimierungen in der Vermittlungsschicht

Wie in Abbildung 6.6 ersichtlich fällt die simulierte Signaldämpfung so gering aus, dass fast über die ganze Länge des simulierten Flures kommuniziert werden kann. Es wäre also möglich das System ohne funktionale Vermittlungsschicht zu betreiben bzw. zu simulieren. Unter geringfügig anderen Ausbreitungsbedingungen kann es jedoch bereits notwendig sein, dass Pakete weitergeleitet werden müssen. Um eine allgemeine Einsetzbarkeit des Simulationsansatzes unter Beweis zu stellen, wurde die Vermittlungsschicht im Rahmen der Simulationen folglich dennoch integriert. Innerhalb des Anwendungsbeispiels kamen dabei die vier in Abschnitt 2.1.5 erläuterten Routingalgorithmen zum Einsatz. Durch vergleichende Simulationen in den Szenarien „0AP“ bis „3AP“, wurde dabei der Einfluss der Algorithmen auf die energetische Belastung der Knoten sowie die Verteilung dieser untersucht. Weiterhin sollte geklärt werden, inwiefern das verwendete Routingverfahren einen Einfluss auf Erfolg und Verzögerung der vom Benutzer ausgelösten Schließenanfragen hat.

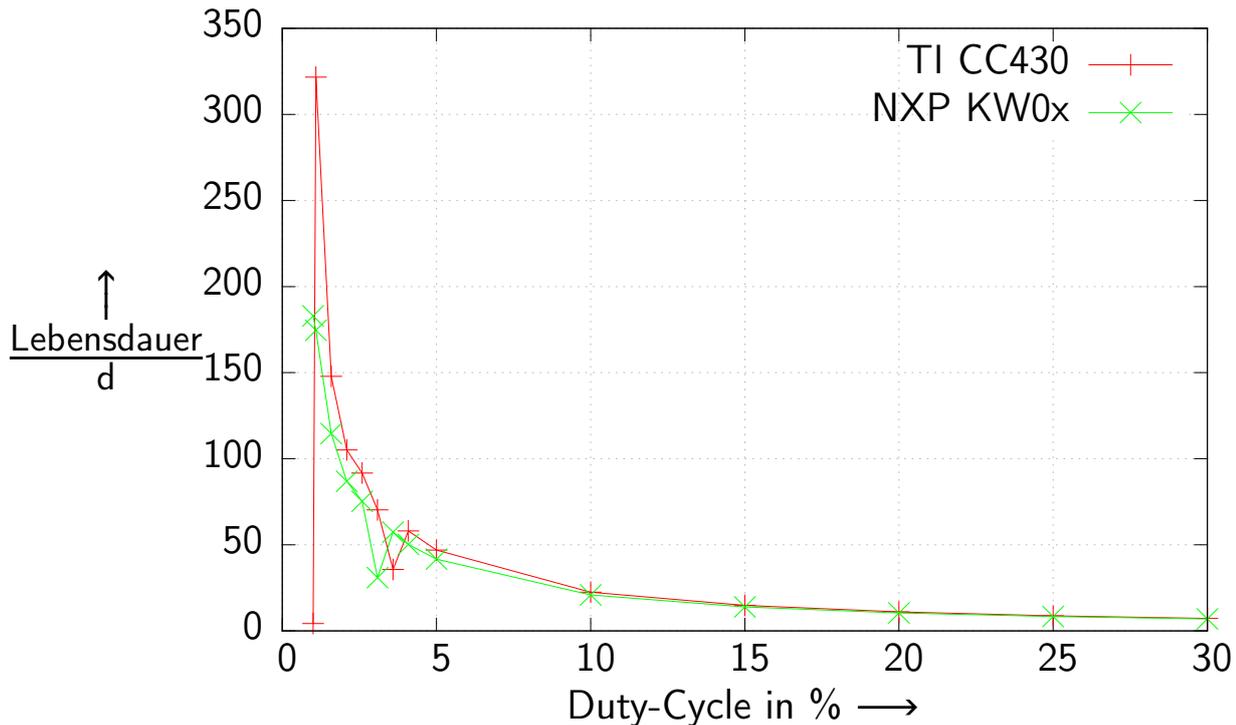


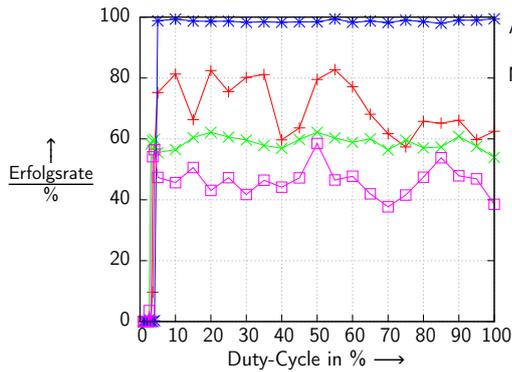
Abbildung 6.8: Vergleich der Auswirkungen des Einsatzes zweier verschiedener SoCs (TI CC430- und NXP KW0x-Familie) auf die Lebensdauer im Szenario „0AP“ bei ansonsten unveränderten Simulationsparametern

Einfluss auf den erfolgreichen Systembetrieb

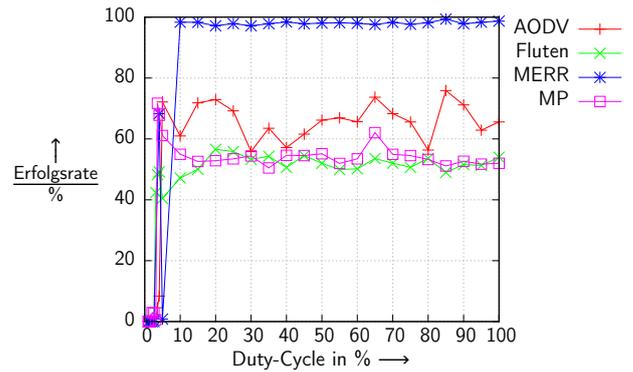
Zunächst gilt es die Operabilität des Schließanlagensystems sicher zu stellen. Diese ist gegeben, wenn die Mehrheit aller Schließvorgänge im Online-System erfolgreich beendet wird. Der Einfluss der zur Weiterleitung der Schließanfragen eingesetzten Routingalgorithmen auf diesen Erfolg ist in Abbildung 6.9 dargestellt.

Es lässt sich erkennen, dass das Schließanlagensystem erst mit einer Wartezeit von mindestens 5 ms, d.h. einem Duty-Cycle von 5 %, stabil operabel ist. Wie bereits in Abschnitt 6.1.2 erläutert, werden anwendungsseitig im Schließzylinderknoten generierte Schließanfragen (das Äquivalent zum Schließwunsch eines Benutzers) explizit nicht nach einer voreingestellten Zeit wiederholt. In Verbindung mit hoher Kommunikationslast und begrenzten Warteschlangen kann es deshalb zu Paketverlusten kommen. Trifft die zur Anfrage passende Antwort jedoch ein, wird die resultierende Verzögerung (siehe Abschnitt 6.1.4) bestimmt und protokolliert. Im Szenario „0AP“ mit linearer Topologie kann bei Nutzung von MERR durch das vorhandene geographische Wissen über alle Schließzylinderpositionen, eine hohe Erfolgsrate erzielt werden. Eine Phase der Routenfindung nach der Initialisierung entfällt. Weiterhin, ergibt sich in diesem Szenario für MERR die niedrigste durchschnittliche Anzahl an Hops pro Nachricht, was die Erfolgsrate ebenfalls erhöht.

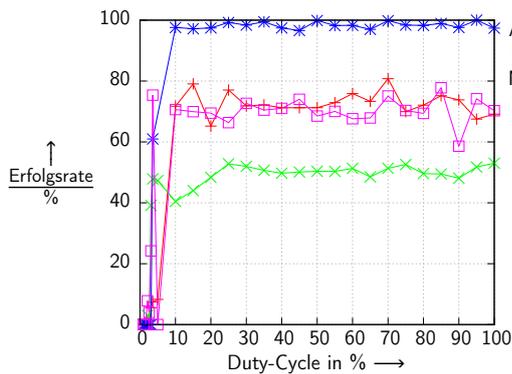
Beim Einsatz von AODV werden Erfolgsraten zwischen 56 % und 83 % erzielt. Wird zur Weiterleitung der Pakete das ineffiziente, aber robuste Fluten eingesetzt, werden ca. 58 % der Schließanfragen erfolgreich beantwortet, allerdings auf Kosten der Batteriebensdauer (siehe Abbildung 6.1.4). Mit einer Erfolgsrate zwischen 42 % und 58 % führt MultiPath-Routing am seltensten zu einem erfolgreichen Schließvorgang. Aufgrund der, im Vergleich zu den anderen Verfahren, komplexesten Routenfindungsphase, steigt insbesondere über lange Strecken die Wahrscheinlichkeit für deren Scheitern und somit



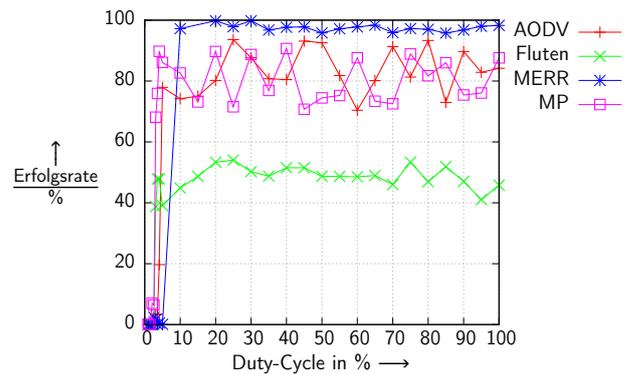
(a) Szenario mit endständigem Access Point (0AP)



(b) Szenario mit Access Point in der Flurmitte (1AP)



(c) Szenario mit zwei Access Points (2AP)



(d) Szenario mit drei Access Points (3AP)

Abbildung 6.9: Erfolgsrate für die Beantwortung einer Schließanfrage für ein Online-Schließsystem in Abhängigkeit vom gewählten Routingverfahren für die Infrastrukturszenarien „0AP“ bis „3AP“

auch des Scheiterns der erfolgreichen Übermittlung oder Beantwortung einer Schließanfrage (siehe Abbildungen 6.14a und 6.15a).

Betrachtet man stattdessen das Szenario „1AP“, ändern sich die Ergebnisse. Der einzige vorhandene „Access Point“ befindet sich nun in der Mitte statt am Ende des virtuellen Flures, wodurch bei identischer Funkreichweite die Anzahl der durchschnittlichen Hops sinkt. Hiervon profitiert vor allem das Multi-Path-Routing. Werden die Datenpakete mittels Fluten weitergeleitet, sinkt die Erfolgsrate stattdessen. Während sich im Szenario „0AP“ die Pakete ähnlich einer Wellenfront durch das Netzwerk ausbreiten, kommt es in der Mitte des Flures im Gebiet rund um den „Access Point“ zu Paketfehlern aufgrund der Überlagerung von Übertragungen.

Im Szenario „3AP“ erhöht sich die Erfolgsrate im operablen Bereich des Duty-Cycles für alle Routingalgorithmen außer Fluten auf über 70%. Wird MERR eingesetzt, werden durchschnittlich sogar 99,6% der Schließanfragen erfolgreich beantwortet. Erneut profitiert das Multi-Path-Routing deutlich von der gestiegenen Anzahl verfügbarer APs, erzielt aber im direkten Vergleich mit AODV fast immer geringere Erfolgsraten.

Die Erhöhung der Anzahl von APs trägt also, außer beim Fluten, wie erwartet zu einer Verbesserung des Systemverhaltens bei. Zusätzlich sollten versandte Schließanfragen in einer tatsächlichen Online-Schließanlage nach Überschreiten einer maximalen systemabhängigen Beantwortungszeit erneut ausgesendet werden, um so die Erfolgsrate noch weiter zu erhöhen und gleichzeitig die Verzögerung zu reduzieren bzw. einzuschränken. Weiterhin sollte ein Routingverfahren eingesetzt werden, welches eine

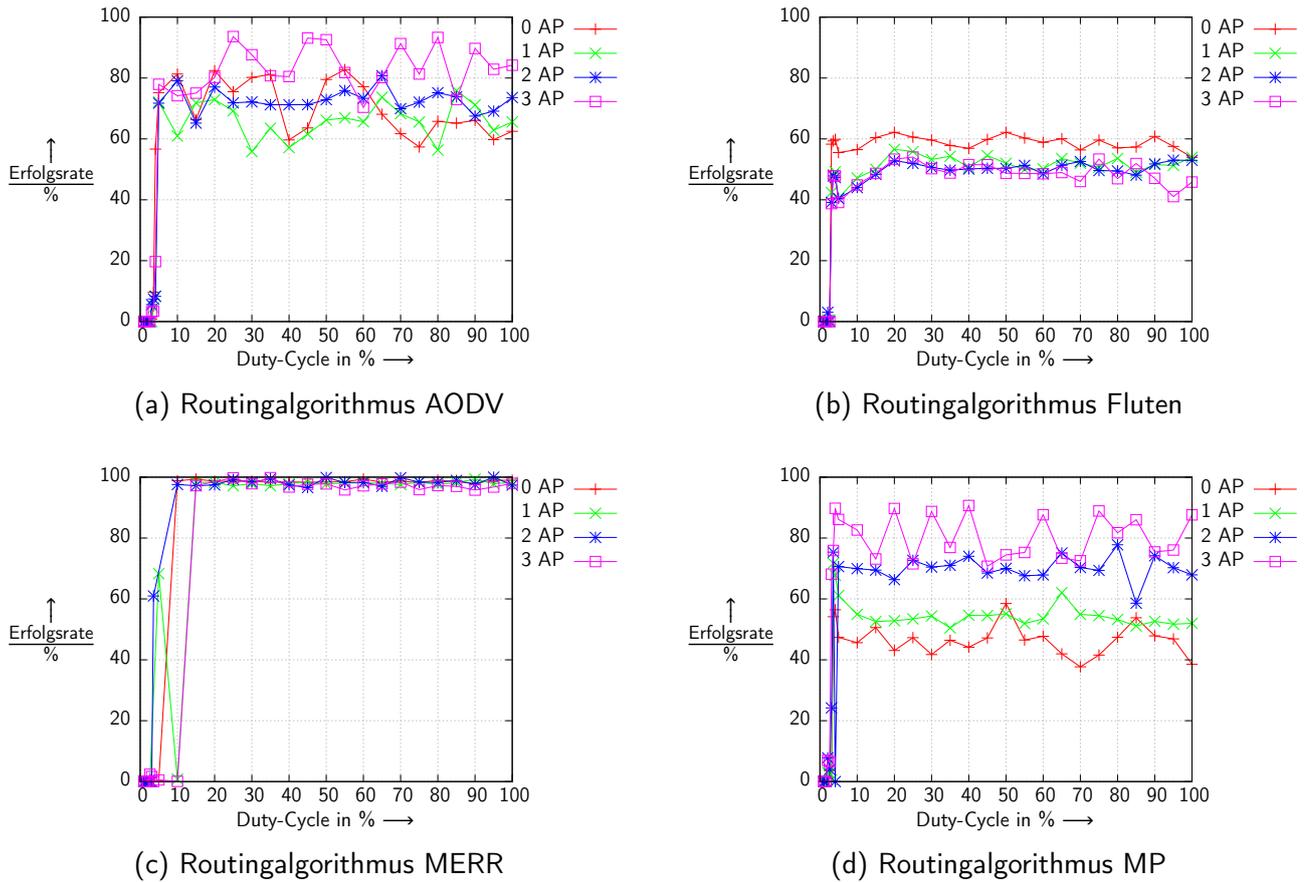


Abbildung 6.10: Erfolgsrate für die Beantwortung einer Schließenfrage bei einer Online-Schließenanlage in Abhängigkeit vom gewählten Routingverfahren für die Infrastrukturszenarien „0AP“ bis „4AP“

wenig komplexe Routenfindungsphase aufweist, um im Fall einer angestrebten infrastrukturlosen Vernetzung auch über lange Distanzen und mit geringem Duty-Cycle die Funktionalität des Schließsystems zu ermöglichen.

Einfluss auf die Lebensdauer

Neben den hardwarebedingten Einflüssen auf die Lebensdauer der Schließzylinder, kann auch ein Einfluss der eingesetzten Routingverfahren identifiziert werden. Dazu wurden die resultierenden Lebensdauern für die Szenarien „0AP“ bis „3AP“ verglichen.

Die Zeit bis zum energiebedingten Ausfall des ersten Knotens in Tagen ist in Abbildung 6.11 dargestellt. Es zeigt sich, dass MERR vor allem in einem Wachzeitbereich zwischen 4,09 ms und 25 ms zu deutlich verringerten Batterielevensdauern führt. Die Ursache hierfür ist die im Kapitel 2.1.5 beschriebene Anpassung der Sendeleistung. Mit dem Ziel ein globales Minimum der für die Übertragung notwendigen Energie zu erreichen, wird die Sendeleistung in einzelnen Knoten zwischenzeitlich erhöht. Zusätzlich werden durch die diskreten Türabstände in Verbindung mit der charakteristischen Distanz häufig die gleichen Knoten zur Weiterleitung von Paketen genutzt. Dies führt insgesamt zu einer energetischen Mehrbelastung bzw. einer inhomogenen Belastungsverteilung und somit zu batteriebedingten Frühausfällen einzelner Knoten. Dieses Verhalten ist vor allem im Szenario „0AP“ mit seiner linearen Topologie auffallend, zeigt sich allerdings ebenfalls im Szenario „3AP“, in dem im

Durchschnitt nur eine geringe Anzahl von Hops zur Übermittlung der Nachricht notwendig ist (siehe Abbildung 6.15a bzw. 6.15d).

Die resultierende inhomogene Verteilung offenbart sich anhand der pro Knoten verbleibenden Restenergie. Da die Simulation beendet wird, sobald der erste Knoten sein Energiebudget erschöpft hat, hätten bei optimaler Verteilung die verbleibenden Knoten ebenfalls ein Restenergiebudget nahe Null. Dies ist insbesondere beim Einsatz von MERR nicht der Fall, wie in Abbildung 6.17 ersichtlich ist, welche die Verteilung der Restenergie aller Schließzylinderknoten bei einem Duty-Cycle von 5 % darstellt. Außer im Szenario „2AP“, zeigt der Median beim Einsatz von MERR die durchschnittlich größte verbleibende Restenergiemenge. Fluten führt durch die hohe und nahezu identische Aktivität aller Knoten erwartungsgemäß zur geringsten Spreizung der Verteilung. Die große Spreizung beim Einsatz des Multi-Path-Routings offenbart, dass die Verteilung der Energiebelastung mit den gewählten Einstellungen in diesem Szenario nicht so gut funktioniert wie angenommen oder die Routen zwangsweise häufiger neu ermittelt werden müssten, um eine gleichmäßigere Verteilung zu bewirken.

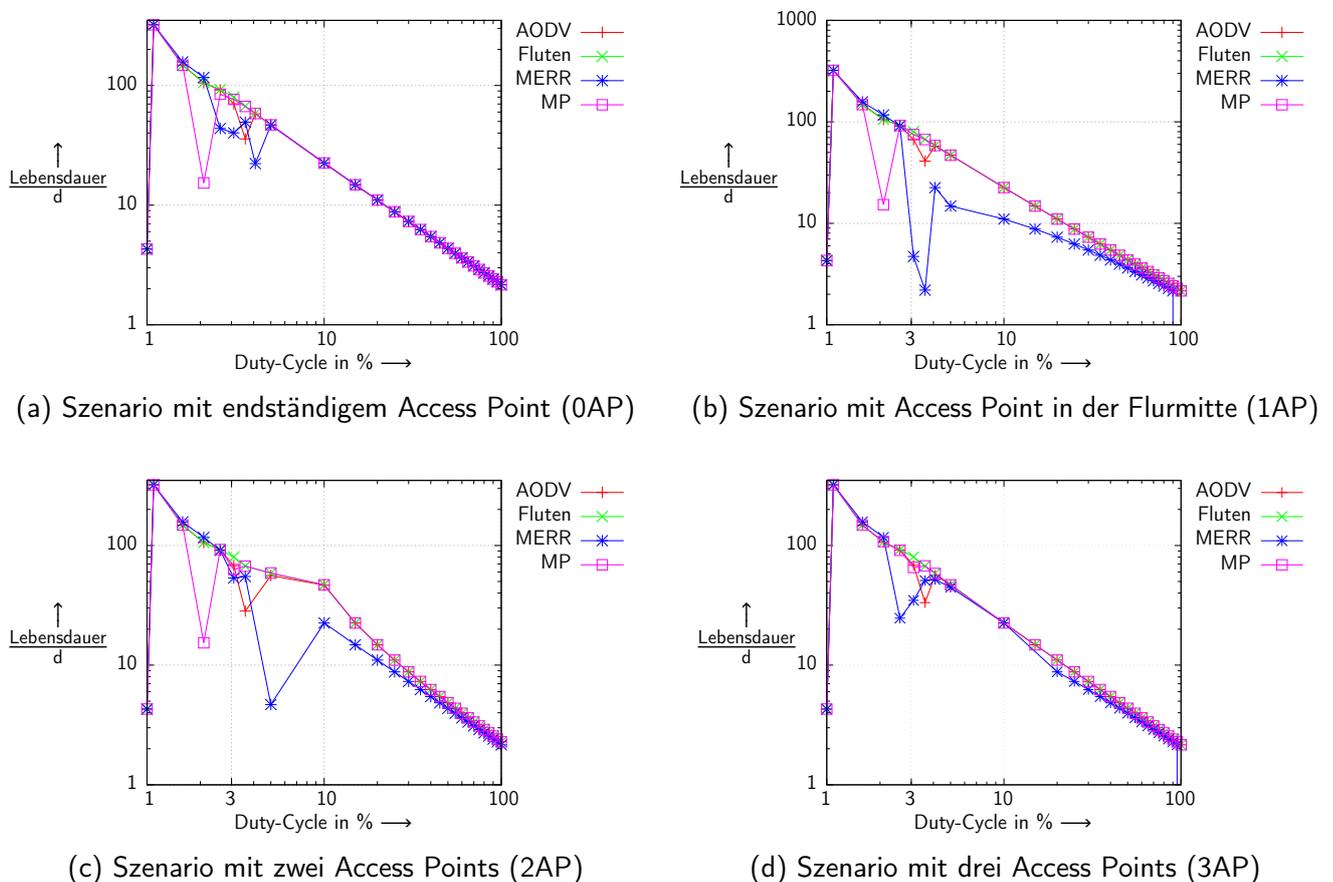


Abbildung 6.11: Lebensdauer in Abhängigkeit vom gewählten Routingverfahren für die Infrastrukturszenarien „0AP“ bis „3AP“ in doppelt logarithmischer Darstellung

Insgesamt lässt sich auf Basis der durchgeführten Simulationen in Verbindung mit der gewählten Batteriekapazität von 850 mAh und einem Duty-Cycle von 5 % eine zu erwartende Lebensdauer von ca. 47 Tagen vorhersagen.

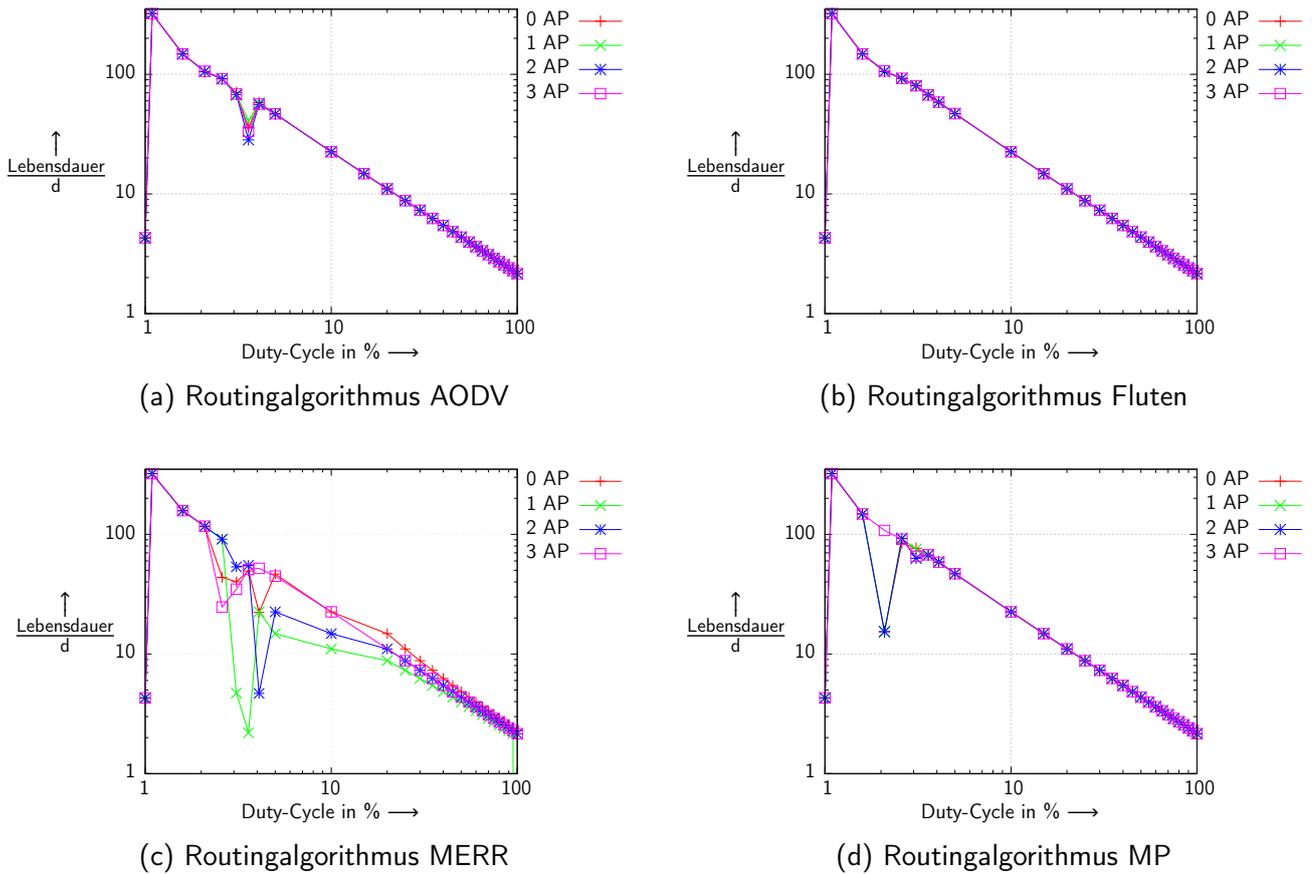


Abbildung 6.12: Zeit bis zum batteriebedingten Ausfall des ersten Schließzylinders (Lebensdauer) in Abhängigkeit vom gewählten Routingverfahren für die Infrastrukturszenarien „0AP“ bis „4AP“ in doppelt logarithmischer Darstellung

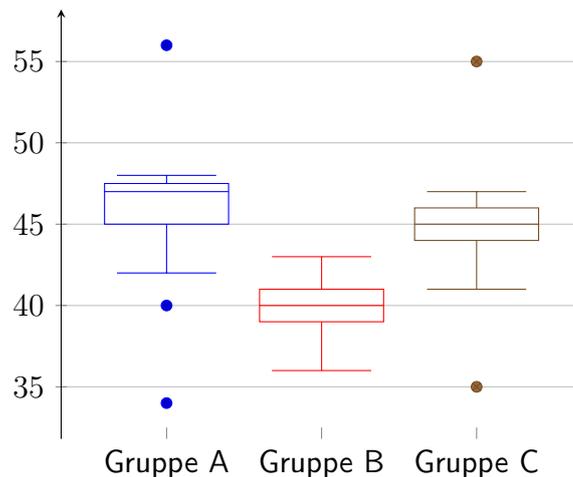


Abbildung 6.13: Beispiel für die Darstellung einer eindimensionalen Verteilung mittels Boxplots. Die Mittellinie kennzeichnet den Median, die beiden Enden der Box jeweils das untere bzw. das obere Quartil. Der Abstand der Antennen beträgt das 1,5-fache des Interquartilsabstands. Alle Werte, die außerhalb dessen auftreten, werden einzeln als Datenpunkte dargestellt, wodurch sich Minimum und Maximum ablesen lassen.

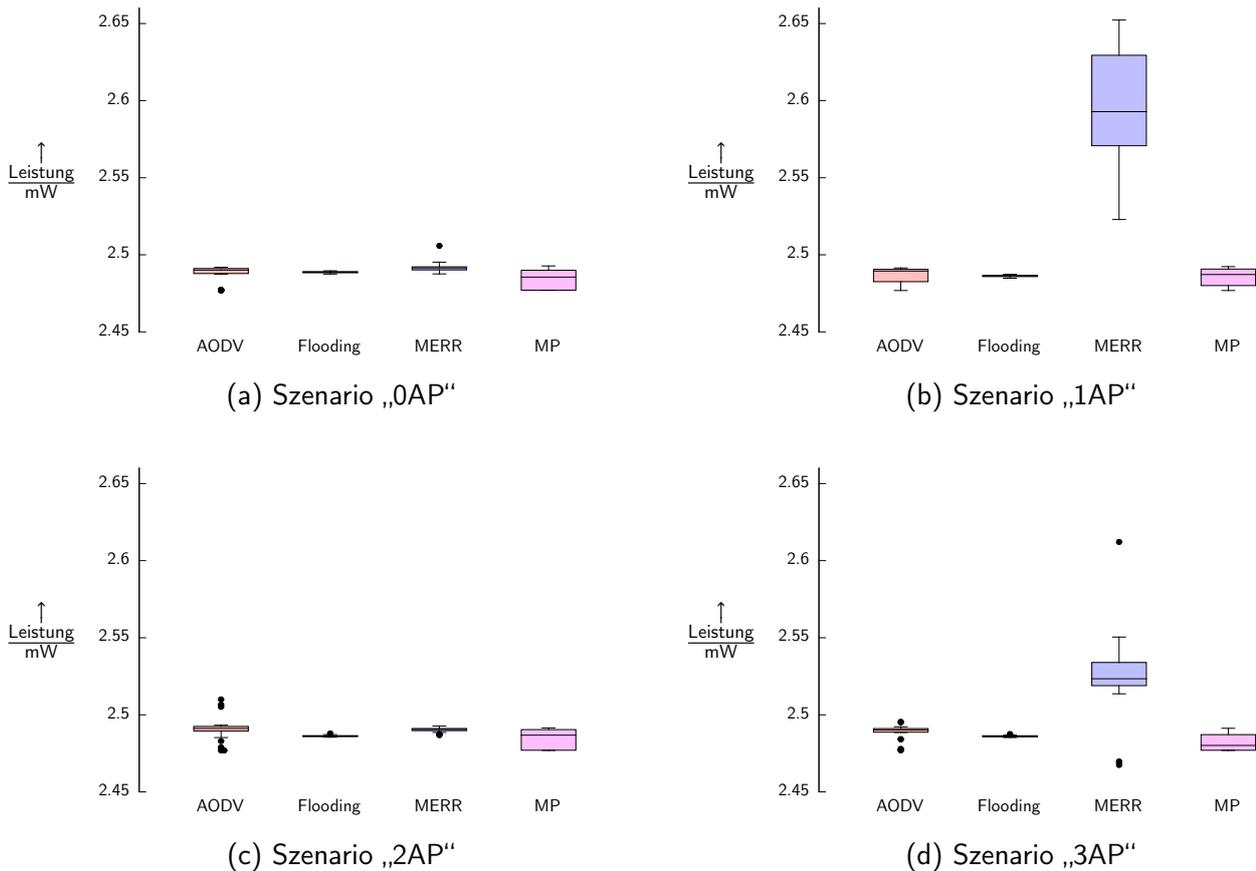


Abbildung 6.14: Vergleich der Verteilungen der durchschnittlichen Leistungsaufnahme für alle Schließzylinderknoten in Abhängigkeit des genutzten Routingalgorithmus sowie des Szenarios für den einen Betriebsfall mit 5% Duty-Cycle.

Einfluss auf die Verzögerung beim Schließvorgang

Neben der Erfolgsrate ist im Online-System auch die Verzögerung bei der Beantwortung einer Schließanfrage von großer Bedeutung für die Akzeptanz des Systems. Es besteht dabei ein direkter Zusammenhang zwischen erzeugtem Netzwerkverkehr, der Anzahl von Hops pro Nachricht und der resultierenden Latenz. Eine Nachricht gilt erst dann als erfolgreich zugestellt, falls eine Empfangsbestätigung empfangen worden ist. Ist dies nicht der Fall, werden erneut Übermittlungsversuche vorgenommen, deren maximale Anzahl vom Routingverfahren abhängig ist.

Im operablen Duty-Cycle-Bereich zwischen 5% und 100% Wachzeit im Szenario „0AP“, stellen sich in Abhängigkeit vom gewählten Routingalgorithmus übertragungsbedingte Verzögerungszeiten von 50 ms bis 490 ms ein. Mit Ausnahme des Szenarios 3AP, führt dabei der Einsatz von MERR zur geringsten durchschnittlichen Anzahl von Hops pro Nachricht (s. Abbildung 6.15). Allerdings ist die erzielte durchschnittliche Verzögerung, in allen Szenarien die höchste, falls dieses Routingverfahren eingesetzt wird. Die Sendeleistungsanpassung bewirkt ein geringeres Signal-Rauschverhältnis, so dass selbst schwach störende Übertragungen in einiger Entfernung Bitfehler erzeugen, wodurch Neuübertragungen eingeleitet werden und sich die Beantwortung der Schließanfrage verzögert.

Der Einsatz von AODV oder Flooding führt im Vergleich zu insgesamt besseren Ergebnissen. Die durchschnittlich geringste Latenz wird im Szenario „0AP“ bei der Nutzung des Multi-Path-Routings erzielt, obwohl die durchschnittliche Anzahl von Hops bei der Übermittlung von Schließanfragen mit

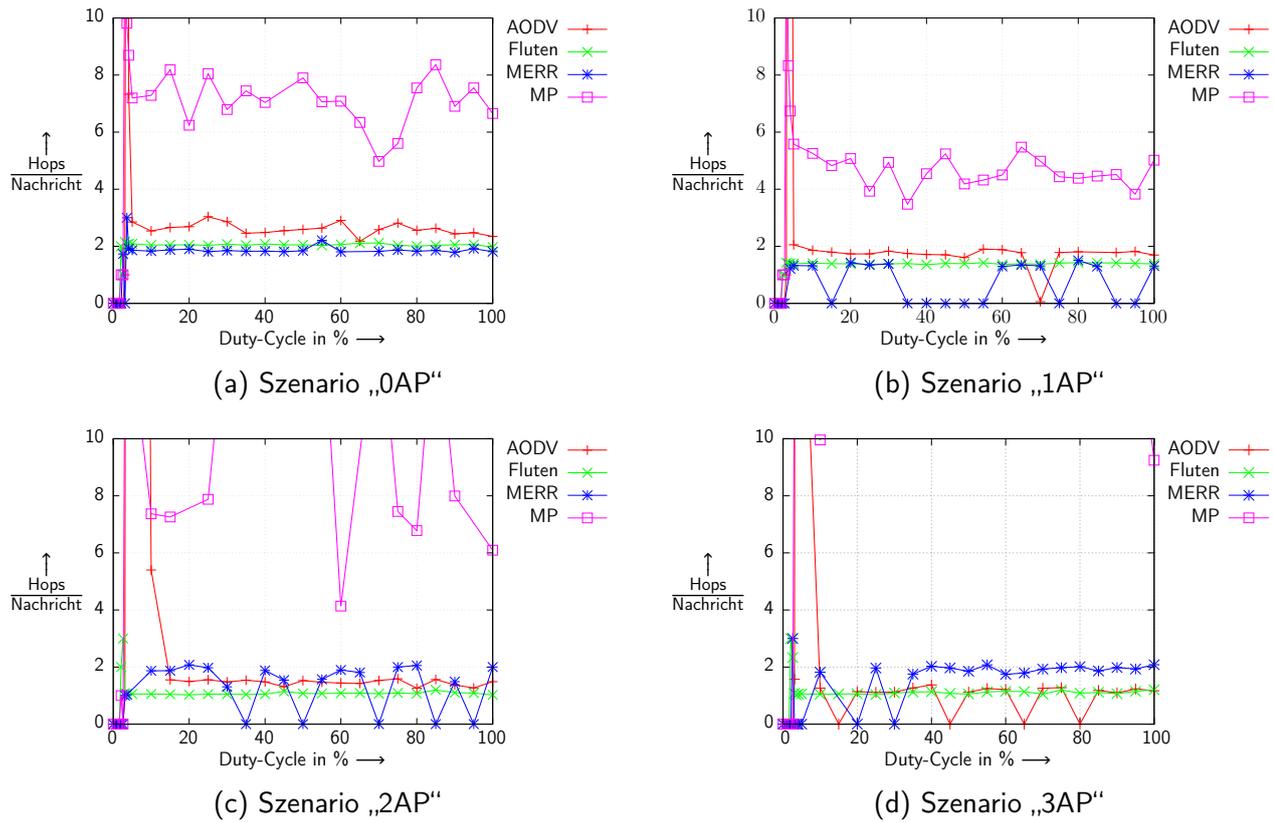


Abbildung 6.15: Durchschnittliche Anzahl von Hops pro Nachricht in Abhängigkeit von Anzahl und Position verfügbarer Access Points: Ein Vergleich für verschiedene Routingalgorithmen bei einem Duty-Cycle von 5% (in den Szenarien 2AP und 3AP erfolgt eine gleichgewichtete Mittelung der Einzelwerte der APs)

ca. 7 Hops deutlich größer ist als bei den anderen Routingverfahren (siehe Abbildung 6.15a). Die Ergebnisse werden hier durch die distanzabhängige Erfolgsrate verzerrt. Da es sich bei AODV und MP um reaktive Routingverfahren handelt, geht der Übertragung des Pakets der Anwendungsschicht, in diesem Fall der Schließenfrage, zunächst eine Phase der Routenfindung voraus. Während dieser können Pakete durch Übertragungskollisionen unlesbar werden, so dass die Datensicherungsschicht eine Neuübertragung initiiert. Unter ungünstigen Umständen, beispielsweise hoher Netzwerklast oder sehr klein gewähltem Duty-Cycle, kann die Routenfindung jedoch scheitern, so dass die Schließenfrage nicht erfolgreich beantwortet und folglich auch keine Beantwortungslatenz berechnet werden kann.

Weiterhin kann mit Hilfe des Simulationsmodells die positionsabhängige Durchschnittslatenz in den vier Szenarien ermittelt werden. Diese spiegelt – wie zu erwarten – deutlich die Position des bzw. der verfügbaren Access Points wider (siehe Abbildung 6.20). Im Falle von MERR ist ebenfalls die Grenze zwischen einem oder zwei festgelegten Hops, d.h. die charakteristische Distanz deutlich zu erkennen (vgl. Abbildung 6.20c bzw. 6.21c). Die Auswirkungen der gescheiterten Routenfindung zeigen sich besonders im Fall des EEMRP-Routings. Im Szenario „0AP“ wäre keiner der Schließzylinder im vom AP am weitesten entfernten Drittel des Flures betriebsfähig.

Durch den Einsatz des Simulationsmodells ist es möglich solche Bereiche und auch Bereiche mit zu hoher zu erwartender Latenz ausfindig zu machen und gegebenenfalls durch Platzierung eines oder mehrerer APs aufzulösen.

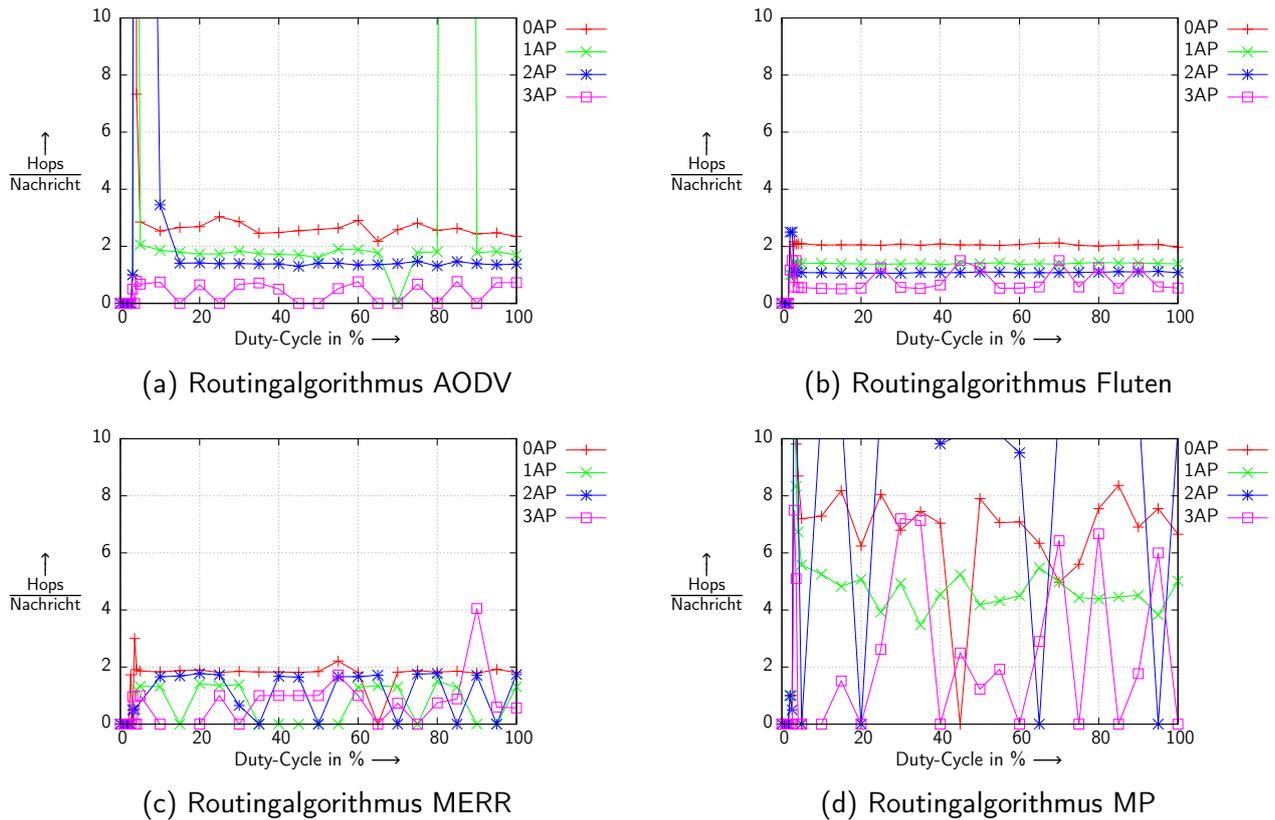


Abbildung 6.16: Durchschnittliche Anzahl von Hops pro Nachricht für verschiedene Routingalgorithmen: Ein Vergleich der Szenarien 0AP bis 3AP (in den Szenarien 2AP und 3AP erfolgt eine gleichgewichtete Mittelung der Einzelwerte der APs) bei einem Duty-Cycle von 5 %

Zusammenfassung

Folgende Erkenntnisse lassen sich bezüglich der Auswirkungen des Einsatzes der verschiedenen Routingverfahren aus den Untersuchungsergebnissen ableiten:

- Die Routingprotokolle haben einen signifikanten Einfluss auf die Lebensdauer des SchließzylinderNetzwerks. Für das Szenario 3AP und einen Duty-Cycle von 5 % beträgt dieser beispielsweise 4,84 % (Verhältnis der Differenz zwischen maximaler und minimaler Lebensdauer relativ zum Durchschnitt). Im Vergleich zum Einfluss des Verhältnisses zwischen Schlaf- und Wachzeit, d.h. dem Duty-Cycle, ist der Einfluss jedoch gering.
- Die Verfahren AODV und MP eignen sich gut für die untersuchten Anwendungsszenarien. Beide Verfahren verteilen die energetische Belastung auf die Schließzylinder gleichmäßiger als die anderen Protokolle.
- Das Verfahren MERR profitiert von der linearen Topologie im Szenario „0AP“, da hier die energetische Belastung ähnlich einer Relaiskette gleichmäßig über mehrere Knoten verteilt wird. In anderen Szenarien befinden sich die APs oftmals in Funkreichweite, so dass fast ausschließlich mit maximaler Sendeleistung übertragen wird. Dadurch erhöht sich zwar die Erfolgsrate, allerdings findet keine Belastungsverteilung mehr statt, was zu Frühausfällen und somit einer – per Definition – geringeren Lebensdauer führt.
- Die erfolgreiche Beantwortung einer vom Zylinder ausgesendeten Schließanfrage, d.h. die Erfolgsrate, wird maßgeblich durch das Routingverfahren und die Anzahl verfügbarer APs beeinflusst.

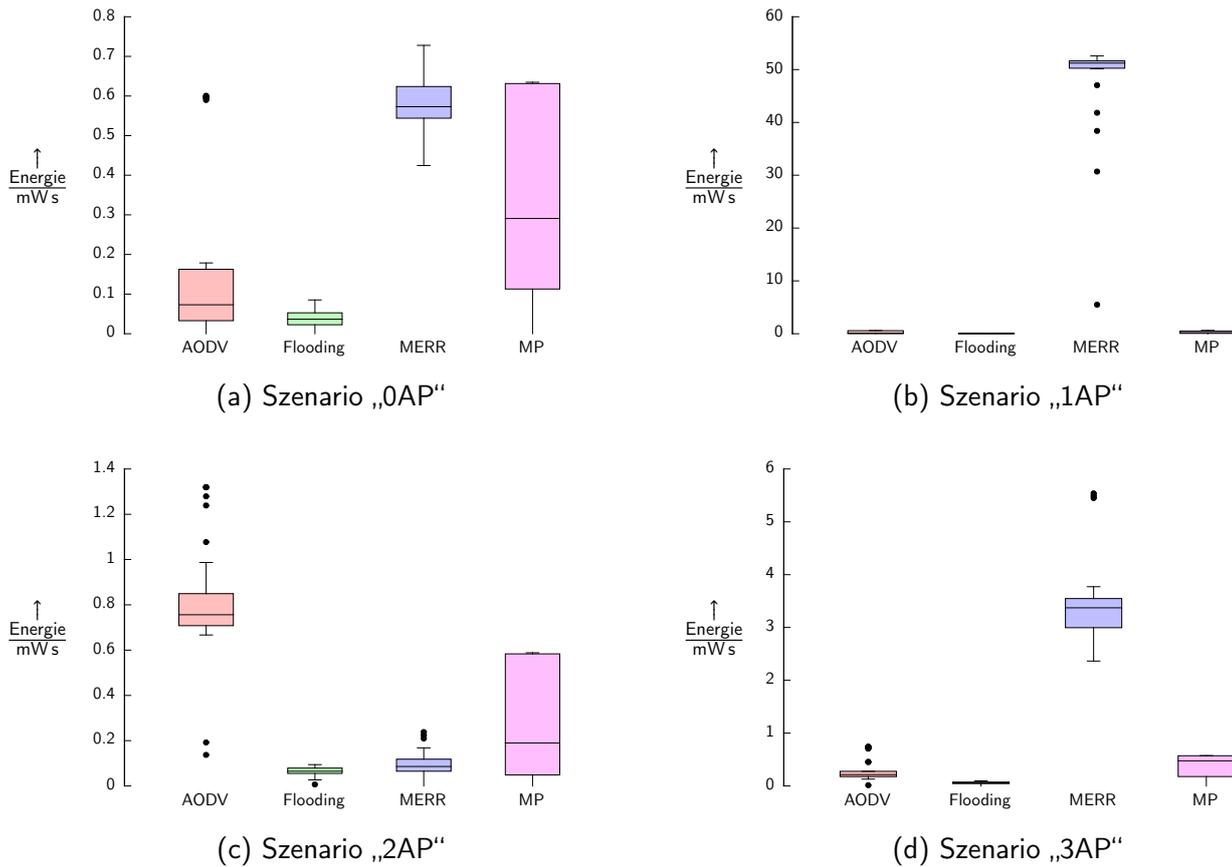


Abbildung 6.17: Vergleich der Verteilung der verbleibenden Restenergie für alle Schließzylinderknoten in Abhängigkeit des genutzten Routingalgorithmus sowie des Szenarios für den einen Betriebsfall mit 5% Duty-Cycle

Der Einsatz von MERR führt in allen Szenarien zur höchsten Erfolgsrate, da durch das Wissen über die geographische Verteilung aller Knoten keine Routenfindung erfolgen muss. Die Paketvermittlung unter Nutzung von AODV bzw. MP, führt zu einer Erfolgsrate die abhängig von der Anzahl der zur Routenfindung notwendigen Übertragungen ist. Existiert eine valide Route, ist die Erfolgsrate hoch. Zusätzlich steigt diese für alle Verfahren mit der Anzahl verfügbarer „Access Points“.

- Die vom Benutzer erfahrene Latenz im Online-System wird maßgeblich vom verwendeten Routingverfahren beeinflusst. Die niedrigsten Latenzen werden mit den Verfahren AODV und MP erzielt. Wird MERR eingesetzt, resultieren, aufgrund der durch die Sendeleistungsanpassung bzw. -reduzierung erhöhten Bitfehlerrate und den damit verbundenen Übertragungswiederholungen, die höchsten Latenzwerte.

6.1.5 Optimierungen in der Anwendungsschicht

Online- bzw. Offline-System

Neben des Einsatzes verschiedener Medienzugriffs- und Routingalgorithmen können weiterhin die Auswirkungen verschiedener Systementwürfe untersucht werden. Beispielsweise kann die Schließanlage neben der zentral administrierten Online-Variante auch offline betrieben werden. Die Schließberechtigung eines Benutzers wird dann auf Basis einer lokalen Kopie des aktuellen Schließplans im

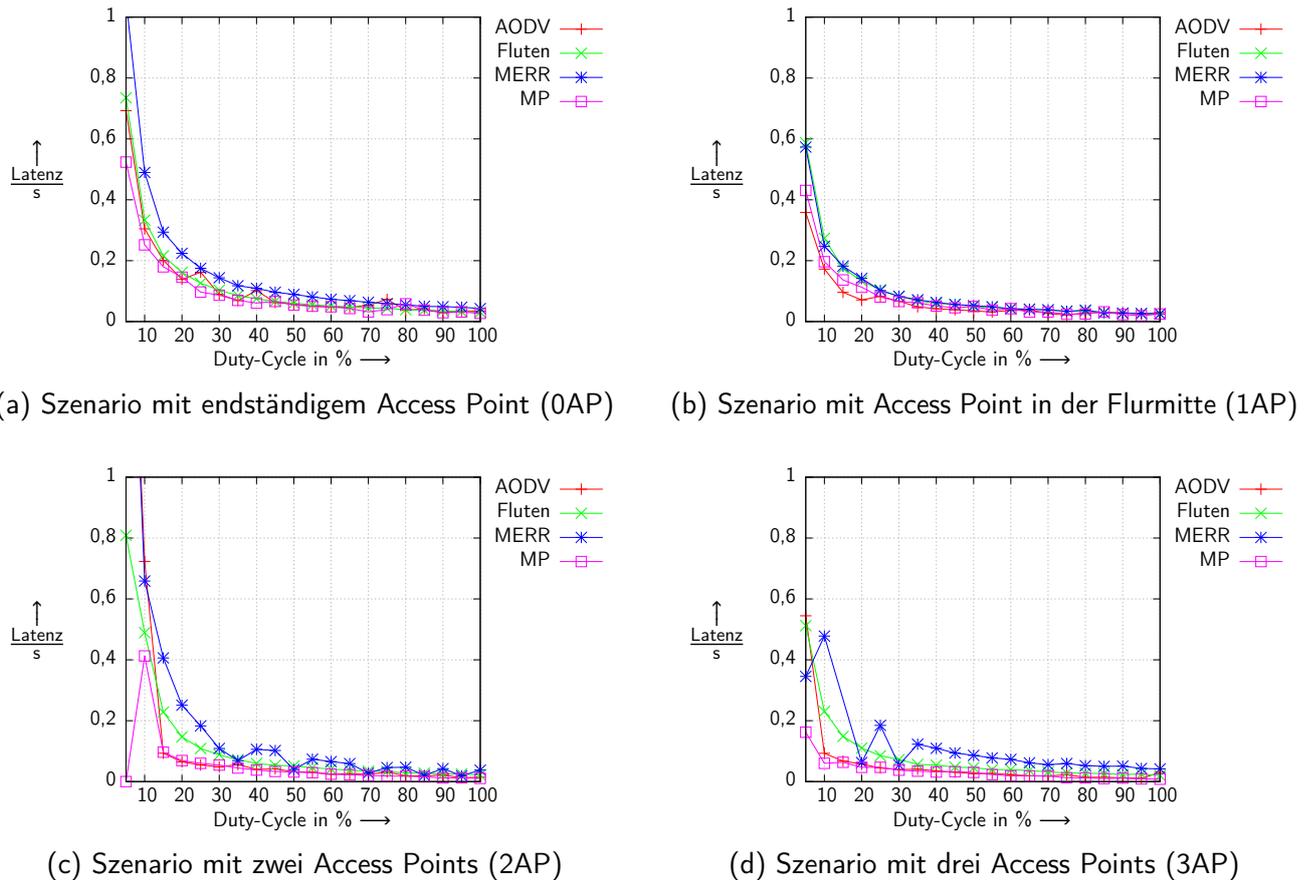


Abbildung 6.18: Latenz für die Beantwortung einer Schließanfrage für ein Online-Schließsystem in Abhängigkeit vom gewählten Routingverfahren für die Infrastrukturszenarien „0AP“ bis „4AP“

Schließzylinder geprüft. Dem Nachteil der verzögerten Aktualisierung des Schließplans stehen die Vorteile der deutlich schnelleren Entscheidung und der potentiell signifikant größeren Lebensdauer entgegen. Um den sich so ergebenden Betriebsparameter quantifizieren zu können, wurden erneut Simulationsläufe durchgeführt. Die Auswertung zeigt, dass die durchschnittlich zu erwartende Lebensdauer eines Schließzylinders bei einem Duty-Cycle von 5% nicht steigt, sondern weiterhin ca. 47 d beträgt. Dabei wurde eine tägliche Aktualisierung des Schließplans mit einer zu übertragenden Größe von 2048 KiB für jeden Schließzylinder zu Grunde gelegt. Das Aufwachintervall wurde bei 100 ms belassen. Dadurch ergibt sich eine geringe Antwortlatenz, falls Kontakt mit dem Schließzylinder aufgenommen werden muss, um beispielsweise eine Status- oder Batteriezustandsabfrage durchzuführen. Die Lebensdauer der Schließzylinder kann jedoch nur signifikant gesteigert werden, falls das Schlaf-Wach-Verhältnis vergrößert bzw. der Duty-Cycle verkleinert wird. Abbildung 6.22 zeigt die resultierende Lebensdauer, falls die Aufwachperiode auf 1 s, respektive 10 s, verlängert wird. Die Wachzeiten t_{wach} werden dabei nicht verändert, so dass insgesamt ein um den Faktor 10 bzw. 100 kleinerer Duty-Cycle und somit eine deutlich gesteigerte Lebensdauer resultieren. Es fällt allerdings auf, dass sich die Lebensdauer bei einer Aufwachperiode von 10 s im Duty-Cycle-Bereich zwischen 0,3% und 1,5% deutlich reduziert. Hier sammeln sich innerhalb der Schlafperiode der Schließzylinder so viele Pakete an, dass diese in der kurzen Wachzeit nicht versendet bzw. aufgrund von Kollisionen auch nicht fehlerfrei empfangen werden können. Obwohl es sich um ein Offline-System handelt, finden Übertragungen von Schließplanaktualisierungen und Status- bzw. Batterieabfragen statt, die im dargestellten Szenario „0AP“ fast immer von benachbarten Schließzylindern weitergeleitet werden

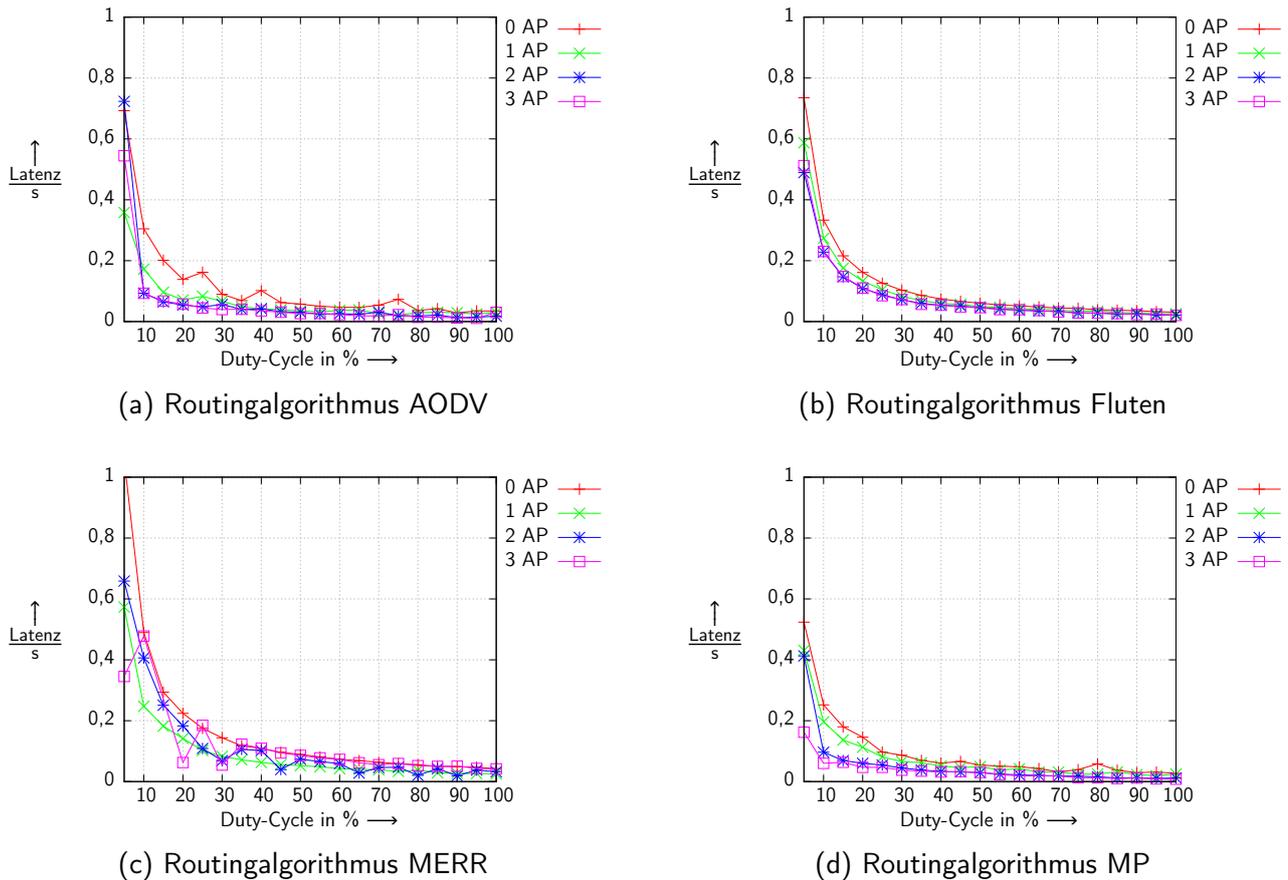


Abbildung 6.19: Latenz für die Beantwortung einer Schließanfrage bei einer Online-Schließanlage in Abhängigkeit vom gewählten Routingverfahren für die Infrastrukturszenarien „0AP“ bis „4AP“

müssen. Die so entstandene Überlastsituation führt zu einem erheblichen Anstieg des Energiebedarfs, da beim laufenden Empfang von Paketen der Eintritt in den Schlafzustand verschoben wird, bis das Paket empfangen worden ist. Zusätzlich wirkt sich, aufgrund des geringen Duty-Cycles und der relativ höheren Anzahl von zu versendenden Nachrichten, der Unterschied in der Leistungsaufnahme zwischen Sendebetrieb (108,9 mW) und Empfangsbetrieb (52,8 mW) deutlicher aus.

Aufgrund der größeren regelmäßig anfallenden Datenmengen, d.h. der Grundlast, welche das Netzwerk übertragen können muss, lässt sich die Lebensdauer nur auf ca. 645 d steigern (Duty-Cycle 1,5 % bei einer Aufwachperiode von 10 s). Im Vergleich zu einer Lebensdauer von ca. 47 d bei einer Aufwachperiode von 0,1 s, entspricht dies nur einer Steigerung um den Faktor 13,7, jedoch nicht um den, durch die Verlängerung der Aufwachperiode auf 10 s, suggerierten Faktor 100. Wird ein geringerer Duty-Cycle gewählt, schlagen die regelmäßig an die Schließzylinder ausgesendeten Statusanfragen fehl, denen jedoch in einem Offline-System eine hohe Bedeutung zufällt.

Methode der Nutzerauthentifikation

Simulativ kann ebenfalls untersucht werden, welchen Einfluss der für die Nutzerauthentifikation anfallende Energiebedarf auf die Lebensdauer der Schließzylinder hat. Dazu lassen sich zwei Energiemengen unterscheiden: der Bedarf beim Einsatz eines passiven RFID-Mediums, dessen leicht zu

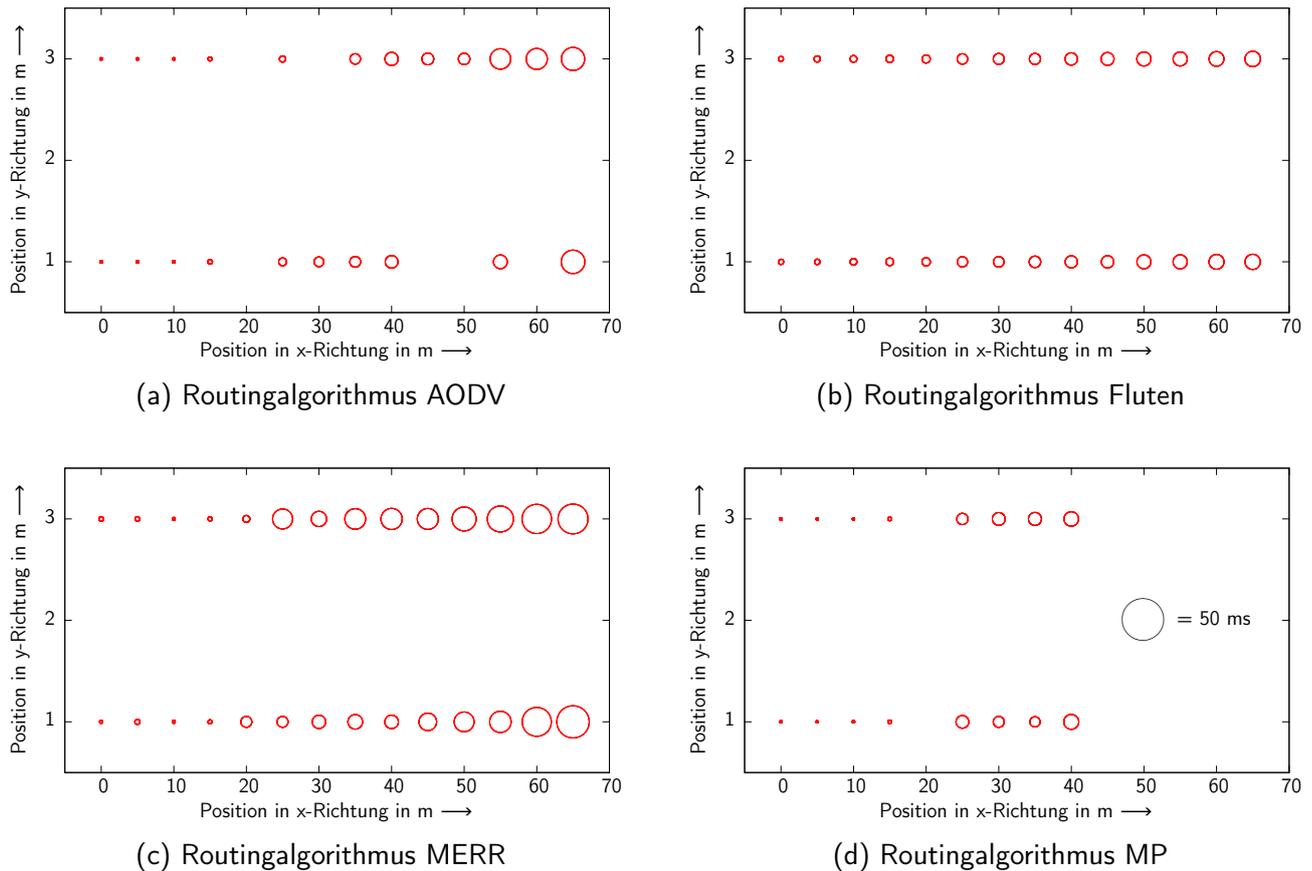


Abbildung 6.20: Qualitative Darstellung der positionsabhängigen Latenz für die Beantwortung einer Schließanfrage in Abhängigkeit vom gewählten Routingverfahren im Szenario „0AP“ mit einem Duty-Cycle von 5%, wobei die Latenz dem Durchmesser des Kreises an einer Schließzylinderposition entspricht

fälschende Hardware-Kennung/UID ausgelesen wird, und der Bedarf beim Auslesen einer verschlüsselt auf einem RFID-Medium abgelegten Kennung. Die genauen Werte sind Tabelle 6.1 zu entnehmen. Es zeigt sich, dass der durch die sichere Form der der Nutzerauthentifizierung verursachte Anteil am insgesamten Energiebedarf mit maximal 10,58% im Fall eines Online-Systems mit 5% Duty-Cycle deutlich hinter dem durch die periodisch hergestellte Empfangsbereitschaft verursachten Anteil von 76,36% zurücksteht. Die maximal erreichbare Lebensdauer reduziert sich von 46,9 d auf 41,2 d, d.h. um 12,2%.

6.1.6 Simulationsperformance

Ein wichtiger Aspekt für die Entwurfsraumexploration ist die dafür benötigte Zeit. Das erstellte Simulationsmodell soll bei Wahrung der Exaktheit in möglichst kurzer Zeit Ergebnisse generieren, um so Parameterstudien oder vollständige Suchen innerhalb des Entwurfsraums zu ermöglichen. Die zur vollständigen Durchführung der Simulation nötige Zeit richtet sich dabei nach der Anzahl von Ereignissen, die simuliert werden müssen. Diese steigt mit der Häufigkeit der Übertragung von Daten, respektive der Paketrage, sowie der Anzahl zu simulierender Knoten. Zusätzlich besteht eine direkte Abhängigkeit zu den verwendeten paketgenerierenden Protokollen und des Betriebs- oder Lastzustands in dem sich das Netzwerk befindet.

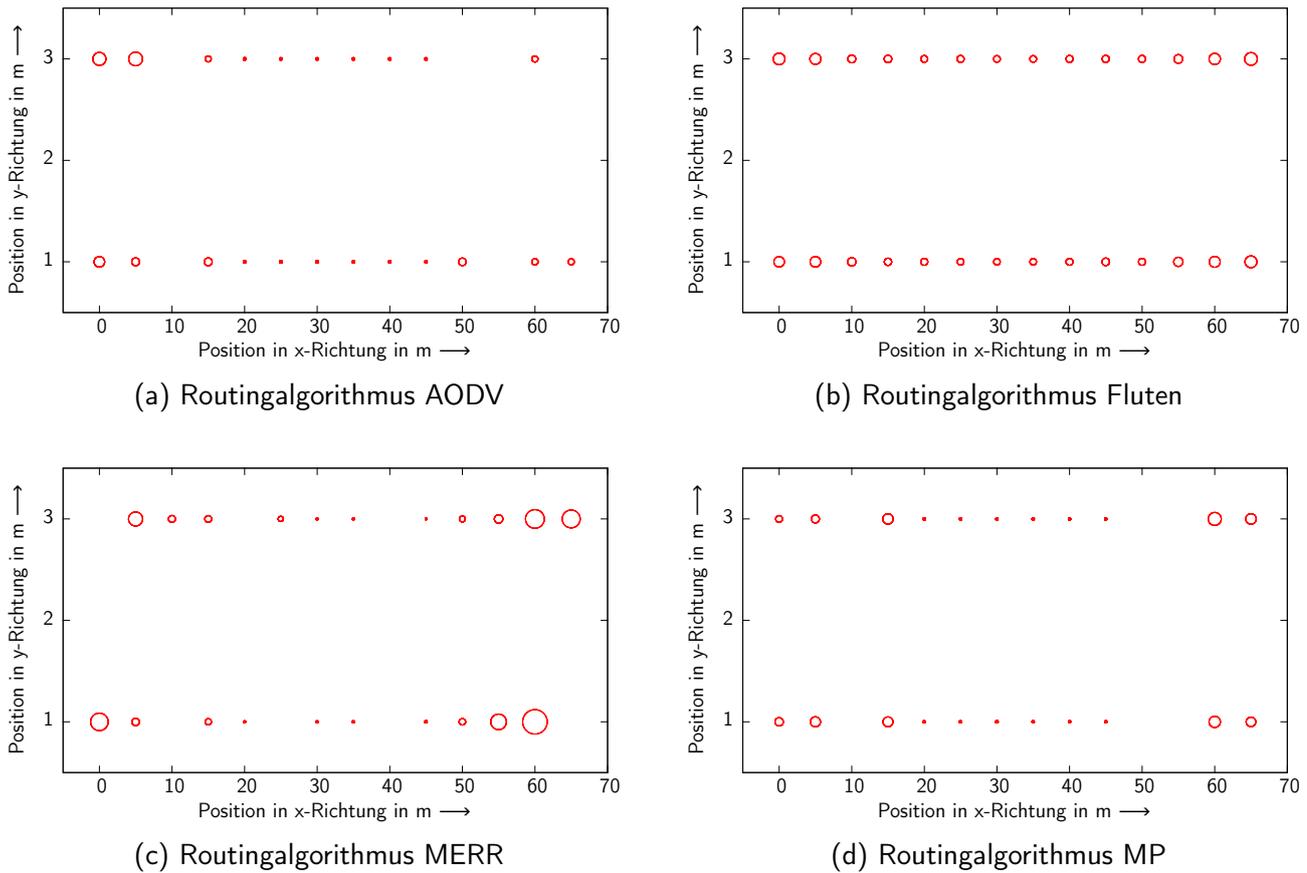


Abbildung 6.21: Qualitative Darstellung der durchschnittlichen positionsabhängigen Latenz für die Beantwortung einer Schließenfrage in Abhängigkeit vom gewählten Routingverfahren im Szenario „1AP“ mit einem Duty-Cycle von 5%

Um das Auftreten der vermuteten Effekte bestätigen zu können, wurden einzelne Simulationsläufe als eigener Single-Thread-Prozess gestartet und hinsichtlich simulierter Zeit und dafür benötigter CPU-Zeit ausgewertet. Eine Auflistung der Zeiten findet sich in Tabelle 6.4.

Zunächst lässt sich feststellen, dass zwischen der kürzesten und der längsten aufgetretenen Simulationsdauer von 2 s bzw. 8 h eine große Spreizung auftritt. Deutlich ist auch der Zusammenhang zwischen Simulationszeit, simulierter Zeit und dem Nachrichtenaufkommen. Während im Online-System beim Einsatz von AODV und einem Duty-Cycle von 5 % pro Sekunde rund 133 s Systemzeit simuliert werden, sind es beim Einsatz von Fluten als Routingverfahren nur 100 s. Ähnlich verhält es sich beim Vergleich zwischen Online- und Offline-System, was besonders am Vergleich der Simulationszeiten bei einem Duty-Cycle von 100 % und dem Einsatz des Routingverfahrens AODV deutlich wird (28 s gegenüber 2 s Simulationszeit). Betrachtet man hingegen die Simulationszeiten für einen Duty-Cycle von 5 % präsentiert sich ein geringfügig anderes Bild. Der Grund hierfür ist, dass im Offline-Szenario jedem Knoten ein Mal pro Tag ein aktueller Schließplan mit einer Größe von 2048 B übermittelt wird, was mit einer Datenrate von 250 kbit/s mindestens 0,0655 s dauert und somit mit einem Aufwachintervall von 0,1 s en bloc erst ab einem Duty-Cycle von mindestens 65,5 % möglich ist. Kommt es hingegen zu Unterbrechungen, sind Neuübertragungen und somit ein deutlich gesteigertes Nachrichtenaufkommen die Folge, wodurch die Simulationsperformance sinkt.

Da die Mehrheit der insgesamt 1392 Einzelsimulationen innerhalb von weniger als einer Stunde durchgeführt werden kann und sehr gut parallelisierbar sind, erscheint das erstellte Simulationsmodell als

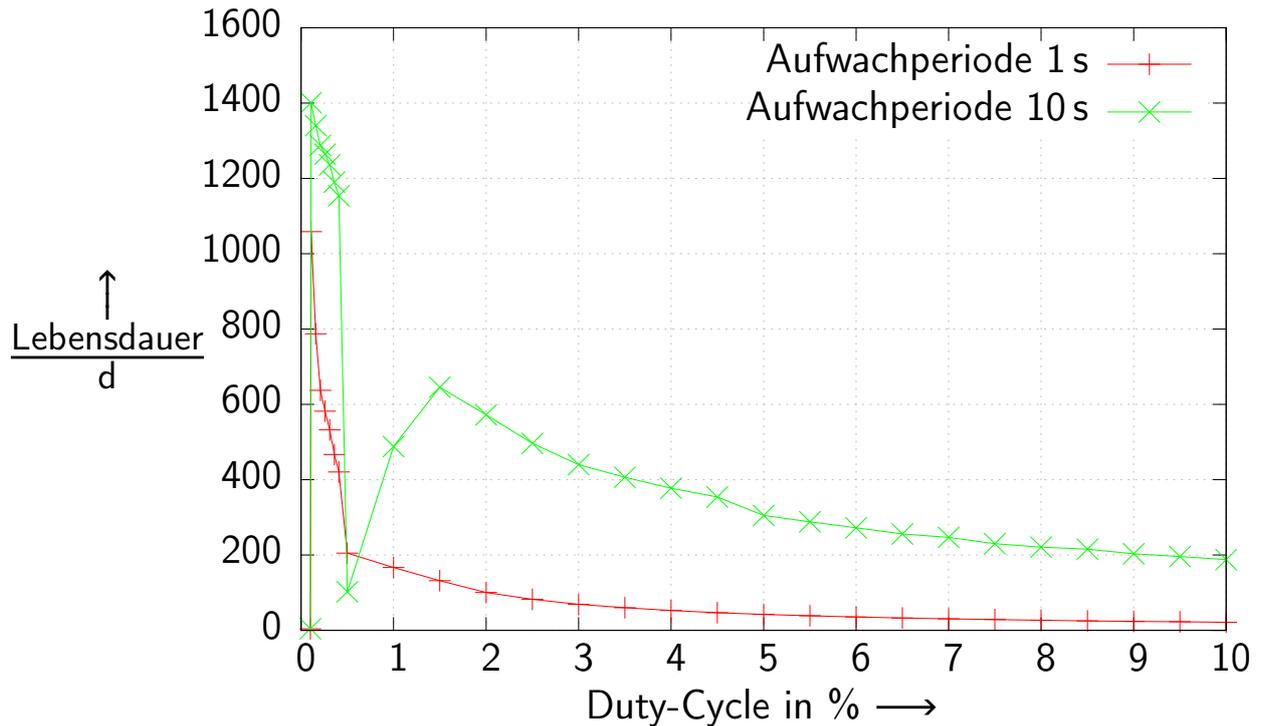


Abbildung 6.22: Vergleich der resultierenden Lebensdauern eines Offline-Systems mit auf 1 s, respektive 10 s, verlängerter Aufwachperiode beim Einsatz von AODV im Szenario „0AP“

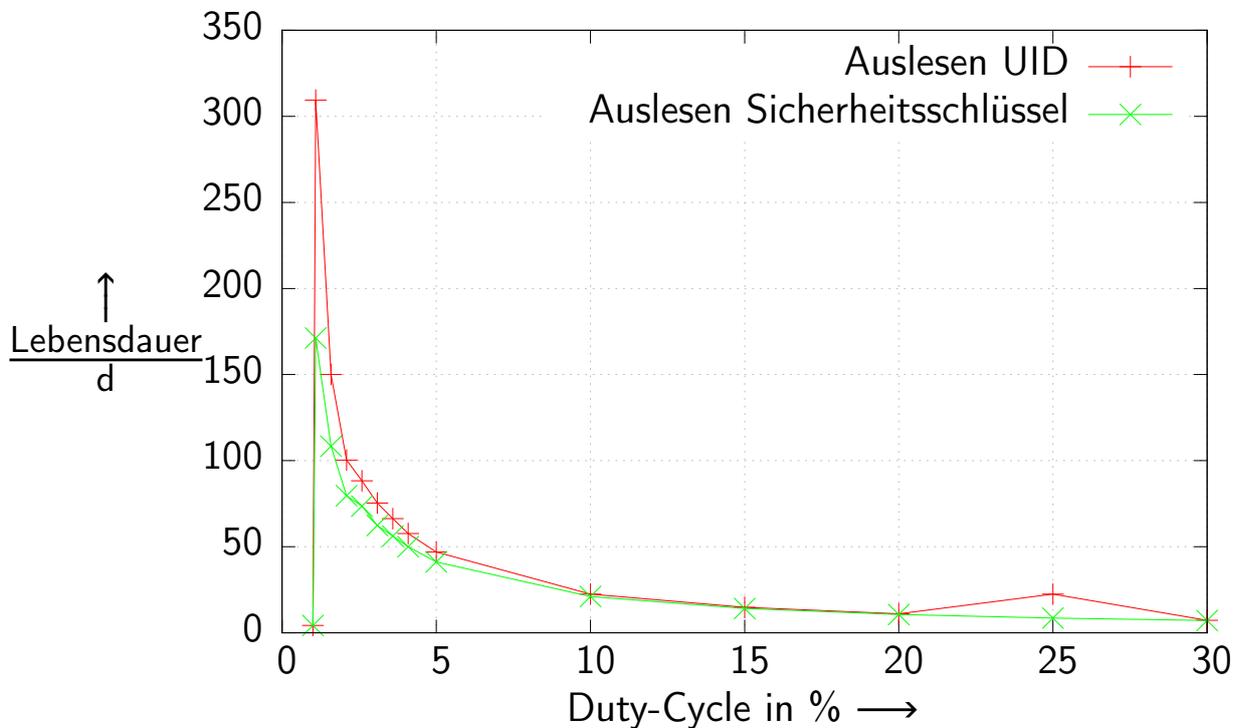


Abbildung 6.23: Vergleich der resultierenden Lebensdauern eines Offline-Systems im Szenario „0AP“, falls die HW-Kennung (UID) eines RFID-Identifikationsmediums oder ein auf dem Identifikationsmedium abgelegter Sicherheitsschlüssel ausgelesen wird

Tabelle 6.4: Vergleich der Simulationszeiten für ausgewählte Simulationsläufe in Abhängigkeit von Auslegungs- und Betriebsparametern bei Ausführung auf einem Server mit vier Prozessoren vom Intel Xeon X7550 sowie 128 GB Arbeitsspeicher (Batteriekapazität zur Verkürzung der Simulationszeiten um den Faktor 10 auf 85 mA h reduziert).

System	Szenario	Routing-verfahren	Aufwach-intervall	Duty-Cycle	simulierte Zeit	Simulations-zeit	simul. Zeit CPU-Zeit
Online	0AP	AODV	0,1 s	5,00 %	112,6 h	50,92 min	132,65
Online	0AP	AODV	0,1 s	100,00 %	5,18 h	0,47 min	666,14
Online	0AP	Fluten	0,1 s	5,00 %	112,7 h	67,25 min	100,1
Online	0AP	Fluten	0,1 s	100,00 %	5,18 h	1,15 min	270,35
Offline	0AP	AODV	0,1 s	1,09 %	742,6 h	400,87 min	111,15
Offline	0AP	AODV	0,1 s	5,00 %	112,5 h	55,28 min	122,12
Offline	0AP	AODV	0,1 s	100,00 %	5,18 h	0,03 min	9325,50
Offline	0AP	AODV	1,0 s	1,09 %	2541,2 h	108,9 min	1400,13
Offline	0AP	AODV	1,0 s	5,00 %	491,5 h	484,9 min	60,82
Offline	0AP	AODV	10,0 s	1,09 %	3362,2 h	31,27 min	6451,91
Offline	0AP	AODV	10,0 s	1,50 %	3216,3 h	200,17 min	964,07
Offline	0AP	AODV	10,0 s	5,00 %	245,0 h	21,68 min	677,82

gut geeignet, um eine vernetzte elektronische Schließenanlage untersuchen zu können. Sollen Netzwerke mit deutlich mehr Knoten simuliert werden, empfiehlt sich das Erreichen einer festgelegten simulierten Zeit und nicht den batteriebedingten Ausfall des ersten Knotens als Abbruchkriterium zu wählen, um die Simulationszeiten zu reduzieren. Eine Betrachtung der energetischen Belastung kann dann auf Basis der durchschnittlichen Leistungsaufnahme erfolgen.

6.1.7 Diskussion der Ergebnisse

Anhand des Anwendungsbeispiels „funkvernetzte Schließenanlage“ lässt sich deutlich aufzeigen, dass Veränderungen einer Schicht systemweite Auswirkungen haben können und somit der Einfluss einzelner Maßnahmen nur differentiell bestimmt werden kann. So beeinflusst beispielsweise die verwendete Sendeleistung nicht nur den Energiebedarf für die Übermittlung der Pakete, sondern bestimmt auch wesentlich das beim Empfänger resultierende Signal-Rausch-Verhältnis und das Potential benachbarte Übertragungen zu stören. Dies hat wiederum Auswirkungen auf die Anzahl der für eine erfolgreichen Empfang notwendigen Übertragungsversuche sowie die Möglichkeit zur gleichzeitigen Kommunikation in enger räumlicher Nähe.

Gleichzeitig konnte verdeutlicht werden, dass viele nichtlineare Effekte bei der Parametrierung des untersuchten Schließenanlagenetzwerks existieren. So sorgt beispielsweise die Verringerung des Duty-Cycles, vor allem im Online-System, für eine lineare Steigerung der Lebensdauer. Ab einem bestimmten Grenzwert verringert sich diese jedoch wieder, da bei konstantem Aufkommen von Anwendungsdaten eine Überlastsituation entstanden ist. Ein weiteres Beispiel für nichtlineare Skalierungseffekte ist die Vergrößerung der Aufwachperiode von 0,1s auf 10s im Offline-Betrieb. Dominiert bei kurzer Aufwachperiode der durch die Empfangsbereitschaft verursachte Anteil die durchschnittliche Leistungsaufnahme, ist es bei langer Aufwachperiode der durch die Nutzerauthentifizierung via RFID verursachte Anteil.

Für ein Schließenanlagensystem spezifizierte Anforderungen an Batterielebensdauer, Erfolgsraten und gegebenenfalls maximale Verzögerungszeiten können mit Hilfe des Simulationsmodells und einer vorgesehenen Parametrierung/Auslegung auf Einhaltung untersucht werden. Ein Beispiel hierfür kann auch die ortsabhängige Latenz sein, wie in Kapitel 6.1.4 beschrieben. Ebenso lassen sich Abtauscheffekte

zwischen den einzelnen Anforderungsdimensionen untersuchen. Das Routingverfahren arbeitet aufgrund des vorhandenen geographischen Vorwissens sehr zuverlässig, sorgt allerdings für eine ungünstige Verteilung der energetischen Belastung (siehe Kapitel 6.1.4).

Die im Rahmen dieses Anwendungsbeispiels maximal auftretenden Simulationszeiten von fast 18 h sind zu berücksichtigen, erlauben aber dennoch auch ausführliche Parameterstudien, da die große Mehrheit der Simulationsläufe bereits innerhalb einer Stunde erfolgreich abschließt. Um die Simulationszeiten zu verkürzen kann aber beispielsweise die Batteriekapazität skaliert oder nur eine vorgegebene Zeit simuliert werden. Dabei ist zu beachten, dass sich das simulierte System für eine ausreichend lange Zeit im eingeschwungenen Betriebszustand befinden muss, um belastbare Aussagen zu erzielen. Mit insgesamt 31 simulierten Knoten, handelt es sich bei der simulierten elektronischen Schließanlage um ein Netzwerk von kleiner bis mittlerer Größe. Eine Steigerung der Knotenanzahl hätte aufgrund der steigenden Nachrichtenanzahl eine nahezu lineare Steigerung der Simulationszeit zur Folge, so dass ab einer bestimmten Größe komplexitätsreduzierende Maßnahmen ergriffen werden müssen. So kann beispielsweise die Anzahl generierter Nachrichten reduziert werden, indem innerhalb eines Knotens ablaufende Teilschritte zusammengefasst werden, wodurch sich im Gegenzug die Möglichkeiten zur detailgenauen Analyse oder Fehlersuche verringern.

6.2 Anwendungsbeispiel beleuchtungsintegriertes Drahtloskommunikationssystem

Das folgende Kapitel erläutert die Anwendung des in Kapitel 5.3 beschriebenen Simulationsmodells für drahtlose optische Kommunikation. Es wird dabei besonders auf die Bitübertragungsschicht eingegangen, um die im Grundlagenkapitel 2.2 beschriebenen Effekte mit Hilfe des entwickelten Simulationsmodells im Rahmen von konkreten Anwendungsszenarien aufzuzeigen. Das Verhalten des optischen Kommunikationssystems wird dabei maßgeblich anhand der Kriterien Durchsatz und effektive Abdeckung beschrieben, wobei auch implementierungsrelevante Eigenschaften, insbesondere die resultierende HW-Komplexität, Berücksichtigung finden.

Die Abbildung 6.1 veranschaulicht das dem Anwendungsbeispiel zu Grunde liegende Anforderungsprofil. Anders als im Anwendungsszenario der Schließanlage, ist das Profil maßgeblich durch einen hohen Datendurchsatz sowie eine effiziente Bandbreitennutzung definiert. Das ist durch die zu erwartende Asymmetrie des Datenverkehrs bzw. des Datenaufkommens zu begründen. Es wird angenommen, dass das mobile Endgerät erheblich mehr Daten empfangen wird als zu senden (z.B. Videostreaming, standortabhängige Dienste). Ebenso erklärt sich die nur geringen Anforderungen an die sich einstellende Latenz der Datenübertragung. Da gleichzeitig mehrere Benutzer vom gleichen System versorgt werden sollen, ist die effiziente Nutzung der zur Verfügung stehenden Bandbreite hingegen stark gewichtet. Eine geringe Verlustleistungsaufnahme ist, im Vergleich zur batteriebetriebenen Schließanlage, hier von geringer Bedeutung, da eine Integration in bestehende Beleuchtungselemente oder eine dauerhafte Spannungsversorgung angenommen wird. Ebenfalls werden nur geringe Anforderungen an die Einbaugröße gestellt.

Die im Fokus stehenden Beleuchtungseinheiten weisen eine typische Verlustleistungsaufnahme zwischen 10 W und 200 W auf. Da die, für die Kommunikation notwendigen, elektronischen Komponenten im Bezug auf die Verlustleistung lediglich im mW-Bereich einzuordnen sind, wird auf eine quantitative energetische Analyse und Bewertung verzichtet. Im Zusammenspiel mit mobilen Endgeräten ergibt sich dennoch eine Asymmetrie im Bezug auf das zur Verfügung stehende Energiebudget und somit auch im Bezug auf die erreichbare optische Sendeleistung. Es bietet sich deshalb beispielsweise an, den

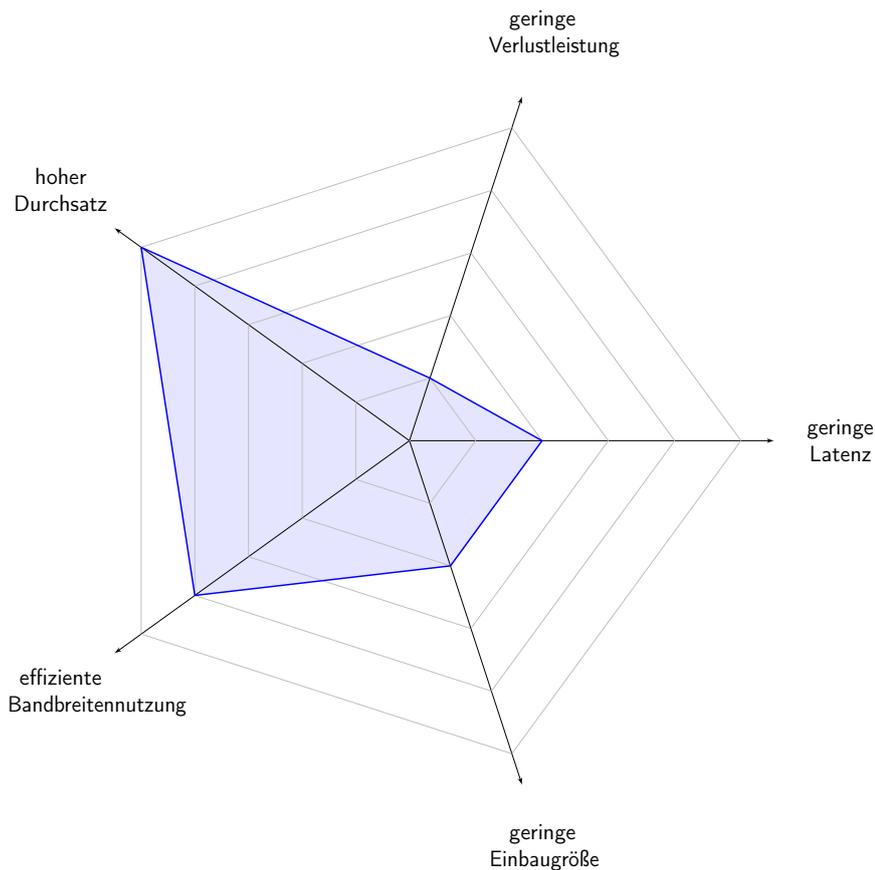


Abbildung 6.24: Anforderungsprofil des Anwendungsbeispiels „beleuchtungsintegriertes Drahtloskommunikationssystem“

Uplink mittels Funkkommunikation herzustellen, wodurch sich ein hybrides Kommunikationskonzept ergibt [103]. Alternativ zum sichtbaren Lichtspektrum, kann im infraroten Bereich übertragen werden. Hier ist die Hintergrundrauschleistung erheblich geringer, da sowohl Leuchtstoffröhren als auch weiß leuchtende LEDs nur gering im infraroten Spektrum emittieren [104]. In Folge dessen, könnte die benötigte optische Sendeleistung auf ein für mobile Endgeräte geeignetes Niveau reduziert werden.

Um aussagekräftige Ergebnisse zu gewährleisten, wurde das erstellte Simulationsmodell zunächst experimentell validiert. Wie in Abschnitt 6.2.3 beschrieben, diente der „Li-Fi Hotspot“ des Fraunhofer IPMS in Dresden [105] als Referenzsystem. Nach der erfolgreichen Validierung konnte das Simulationsmodell eingesetzt werden, um das Verhalten eines virtuellen optischen Kommunikationssystems in mehreren konkreten Szenarien zu untersuchen bzw. Zusammenhänge zwischen Betriebsparametern und dem sich einstellenden Durchsatz, respektive des effektiven Abdeckungsbereichs, quantitativ zu veranschaulichen. Da das erstellte Modell hauptsächlich Effekte der Bitübertragungsebene (OSI-Schicht 1) abbildet, stellen diese einen elementaren Bestandteil der durchgeführten Analysen dar. Hierbei können Auswirkungen verschiedener effektiver Öffnungswinkel (Abschnitt 6.2.5), unterschiedlicher Hintergrundbeleuchtungsintensitäten (Abschnitt 6.2.4) oder Modulationsverfahren, respektive -graden (Abschnitt 6.2.5) ebenso aufgezeigt werden, wie der szenarioabhängige Nutzen eingesetzter Vorwärtsfehlerkorrekturmaßnahmen (Abschnitt 6.2.5).

Neben der Analyse der Bitübertragungsschicht, ermöglicht das Simulationsmodell auch eine Untersuchung des Verhaltens der auf den OSI-Schichten 3 - 4 verwendeten Protokolle. In den Abschnitten 6.2.7 und 6.2.8 wird aufgezeigt, dass diese unterschiedlich auf die stark variablen Kommunikationsbe-

dingungen reagieren. Darüber hinaus lässt sich das Protokollverhalten durch gezielte Wahl geeigneter Werte der Konfigurationsparameter verbessern. Diese sind mit Hilfe des erstellten Simulationsmodells durch Parameterstudien bestimmbar. Bei Ausführungszeiten von wenigen Minuten bis zu maximal 3 Stunden ist dies sogar in Form einer vollständigen Suche im Entwurfsraum möglich, was am Beispiel der Parametrierung von TCP im Abschnitt 6.2.8 illustriert wird. Eine Diskussion der Ergebnisse dieser Untersuchungen, daraus abgeleiteter Hinweise zur Auslegung eines drahtlosen optischen Kommunikationssystems sowie möglicher Maßnahmen zur Verbesserung der Qualität der Ergebnisse findet abschließend im Abschnitt 6.2.10 statt.

6.2.1 Systemaufbau

Der Aufbau des in diesem Anwendungsbeispiel angenommenen Systems ist in Abbildung 6.25 dargestellt. Es besteht aus zwei kommunizierenden Partnern, den sog. *Hosts*. Diese Geräte können alle Schichten des OSI-Modells abbilden, enthalten also auch die die Kommunikation bedingenden Anwendungen (siehe Kapitel 5.3). Die Anwendungsdaten können mittels der Transportprotokolle TCP oder UDP übertragen werden, wobei auf der Vermittlungsschicht IPv4 zum Einsatz kommt. Beide *Hosts* sind jeweils via 100 Mbps-Full-Duplex-Ethernet-Link mit einem optischen „Access Point“ (*OAP*) verbunden, welcher die eintreffenden Ethernet-Frames optisch überträgt. Die Medienzugriffssteuerung erfolgt hierbei gemäß IEEE 802.3, wobei der OAP als optischer Repeater eine Konvertierung auf der Bitübertragungsschicht durchführt, d.h. die elektrisch eintreffenden Ethernet-Frames optisch abstrahlt und umgekehrt.

Der optische Übertragungskanal (*owcScalarMedium*) berücksichtigt die Signaldämpfung sowie die optische Hintergrundrauschleistung wie in Kapitel 2.2.1 beschrieben. Weiterhin wird das lichtleistungsabhängige Schrotrauschen innerhalb der Photodiode (PD) im Empfänger (*owcReceiver*) bestimmt.

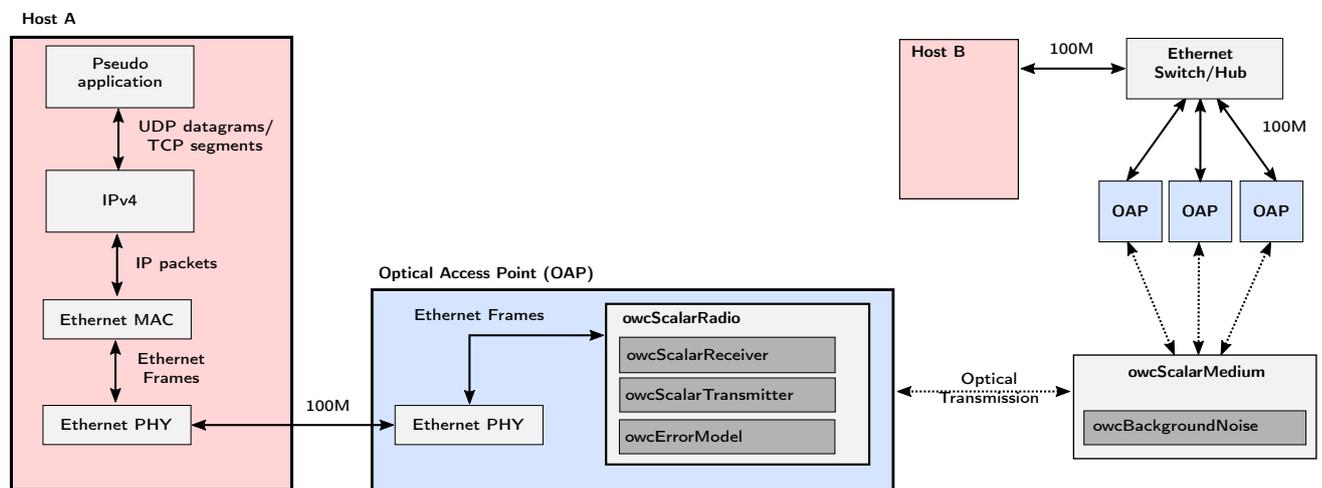


Abbildung 6.25: Abstrahierte Darstellung der im Simulationsmodell verwendeten Module und ihrer Verbindungen untereinander

6.2.2 Simulationsszenario

Das Simulationsszenario besteht in seiner einfachsten Variante aus zwei OAPs, einem mobilen OAP, der das Gerät des mobilen Endnutzers repräsentiert und einem decken- bzw. beleuchtungsintegrierten Infrastruktur-OAP. Diese werden innerhalb eines virtuellen Flures mit den Abmaßen $(10 \times 3 \times 3) \text{ m}^3$

platziert. Dabei ist der Infrastruktur-OAP zentriert an der Decke positioniert und nach unten orientiert (negative Z-Richtung). Der mobile OAP befindet sich initial an einer Seitenwand angrenzend ($x = 0,0$ m) in einer Höhe von 1 m und ist nach oben orientiert (positive Z-Richtung). Nach Start der Simulation bewegt sich der mobile OAP mit einer Geschwindigkeit von 0,1 m/s in positiver x-Richtung und erreicht die gegenüberliegende Wand somit nach 100 s, woraufhin die Simulation beendet wird. Zur Veranschaulichung ist dieses Basisszenario in Abbildung 6.26 dargestellt. Die OAPs sind dabei, wie in Abschnitt 6.2.1 beschrieben, via Ethernet über einen Switch oder Hub mit den Host-Modulen verbunden, die aus Gründen der Übersichtlichkeit nicht abgebildet sind.

Zur Durchführung eines Simulationslaufes muss eine Konfiguration angelegt werden, die Werte für alle die Parameter enthält, die nicht mit etwaigen Standardwerten belegt werden sollen. Da innerhalb von OMNeT++ Konfigurationen von anderen Konfigurationen erben können, ist eine Aufteilung in „Grundkonfiguration“ und „Spezialkonfiguration“ möglich, was die Lesbarkeit und Wartbarkeit deutlich erhöht. Ein vollständiger Konfigurationsabschnitt für das hier beschriebene Simulationsszenario „1 OAP“ ist im Codebeispiel 6.1 zu finden. Hierbei erbt die Konfiguration `Hallway_SingleAP` (Zeile 67) vom grundlegenden Konfigurationsabschnitt `General` (Zeile 1), wodurch sich in diesem Fall die Anzahl der Infrastruktur-OAPs auf einen reduziert und somit das Szenario „1 OAP“ entsteht.

Mit Start des Simulationslaufes, d.h. bei $t = 0,0$ s, beginnt die im Infrastruktur-Host vorhandene Pseudoapplikation mit dem Versuch Daten an die Pseudoapplikation im mobilen Host zu übermitteln. Dabei kommen als Transportprotokolle UDP oder TCP in verschiedenen Konfigurationen sowie IPv4 und ARP als Netzwerk- respektive Adressauflösungsprotokoll zum Einsatz.

Erst nachdem mobiler und Infrastruktur-OAP in das gegenseitige Sichtfeld eintreten, kommt eine Kommunikationsverbindung zustande. Mit den in Tabelle 6.5 angegebenen Parameterwerten ist dies ab $x = 4,50$ m der Fall. Nachdem die IPv4-Schicht im Infrastruktur-Host die Ziel-IPv4-Adresse des mobilen Hosts nach Erhalt einer ARP-Antwort erfolgreich einer physikalischen Adresse zuordnen kann, beginnt die eigentliche Übertragung der Pseudoanwendungsdaten und der auf Anwendungsebene erzielte Durchsatz kann errechnet und aufgezeichnet werden. Das sich so ergebende Positions-Durchsatz-Profil ist in Abbildung 6.27 dargestellt.

Ein standardkonformer Ethernet-Datenrahmen kann bei einer Größe von 1538 B maximal 1472 B (UDP) bzw. 1460 B (TCP) Anwendungs- oder Nutzdaten enthalten (siehe Abbildung 6.28). Bei einer physikalischen Datenübertragungsrate von 10 Mbit/s ergibt sich so eine maximale Nutz- bzw. Anwendungsdatenrate von 9,571 Mbit/s (UDP) bzw. 9,492 Mbit/s (TCP). Dieses Maximum wird nach Aufbau einer optischen Verbindung und der darauf folgenden Adressauflösung nahezu unmittelbar erreicht. Nach Erreichen einer Position von $x = 5,54$ m, überlagern sich die Sichtfelder beider OAPs nicht mehr ausreichend und eine Kommunikation kann nicht mehr stattfinden. Eine vollständige Übersicht aller für die Simulation verwendeter Simulationsparameter in Grundkonfiguration ist in der Tabelle 6.5 gegeben.

Die im folgenden für die Simulationsszenarien verwendeten Bezeichnungen orientieren sich an der Anzahl vorhandener Infrastruktur-OAPs, d.h. im Szenario „10 OAP“ existieren zehn Infrastruktur-OAPs, die in x-Richtung äquidistant an der Mittelachse des simulierten Flures entlang verteilt sind. Sollen mehrere OAPs mit dem Infrastruktur-Host verbunden werden, muss ein Ethernet-Switch bzw. -Hub zum Einsatz kommen. Beim Einsatz eines Hubs wird jedes einlaufende Paket an alle Anschlüsse weitergeleitet, weshalb alle Infrastruktur-OAPs optische Übertragungen beginnen. Dies bedeutet zum einen, dass der Störpegel in den sich überlappenden Zonen der Abdeckungsgebiete zunimmt, hat aber auch einen deutlichen Einfluss auf die Laufzeit der durchgeführten Simulationen, da jede optische Übertragung individuell berechnet wird. Alternativ kann ein Ethernet-Switch eingesetzt werden, welcher einlaufende Pakete nur an den Anschluss weiterleitet, an dem sich ein Gerät mit passender physikalischer Adresse befindet. Das setzt allerdings voraus, dass die Weiterleitungstabelle bei einer sich verändernden

Codebeispiel 6.1: Konfiguration in der Datei omnetpp.ini

```

[General]
network = OWCnetwork

4 # medium
**.backgroundNoise.power = 15 dBm
**.obstacleLossType = "OWCTracingObstacleLoss"
**.pathLossType = "OpticalPathLoss"

9 # number, room & positions
*.numHosts = ${x=ceil(${hallwayLength}/(2*tan(${angleMax}*${pi}/180) * (${ceilingHeight}-${mobileHeight})))}
**.config = xmldoc("hallway.xml")
**.wallWidth = ${wallWidth = 0.5m}
**.hallwayLength = ${hallwayLength = 10m}
14 **.hallwayLength = ${hallwayLength2 = 10}
**.mobileHeight = ${mobileHeight = 1m}
**.ceilingHeight = ${ceilingHeight = 3m}
**.constraintAreaMinX = 0m
**.constraintAreaMinY = 0m
19 **.constraintAreaMinZ = 0m
**.constraintAreaMaxX = 50m
**.constraintAreaMaxY = 20m
**.constraintAreaMaxZ = 5m
**.mobility.acceleration = 0
24 **.mobility.rotationSpeed = 0
**.apMobile.mobilityType = "OwcRotatingMobility"
**.apMobile.mobility.initialX = ${wallWidth}
**.apMobile.mobility.initialY = ${wallWidth} + 1.5m
**.apMobile.mobility.initialZ = ${mobileHeight}
29 **.apMobile.mobility.initialAlpha = 0deg
**.apMobile.mobility.initialBeta = 270deg
**.apMobile.mobility.initialGamma = 0deg
**.apMobile.mobility.speed = 0.1mps
**.apMobile.mobility.angle = 0deg
34 **.ap[*].mobilityType = "OwcRotatingMobility"
**.ap[*].mobility.initialX = (parentIndex() + 0.5) * (${hallwayLength}/${x}) + ${wallWidth}
**.ap[*].mobility.initialY = ${wallWidth} + 1.5m
**.ap[*].mobility.initialZ = ${ceilingHeight}
**.ap[*].mobility.initialAlpha = 0deg
39 **.ap[*].mobility.initialBeta = 90deg
**.ap[*].mobility.initialGamma = 0deg

# optics
**.radio.txAntenna.txBeamWidth = ${txBeamWidth = 16.92}deg
44 **.radio.rxAntenna.rxBeamWidth = ${rxBeamWidth = 14.93}deg

# NIC settings
**.errorModel.modlevel = 2
**.mac.address = "auto"
49 **.transmitter.power = 21mW
**.transmitter.bitrate = 10Mbps
**.radio.maxQueueSize = 100000
**.queueType = "DropTailQueue"

54 # udp
**.Host.numUdpApps = 1
**.Host.udpApp[*].typename = "UDPBasicApp"
**.wiredHost.udpApp[0].destAddresses = "wiredHostMobile"
**.Host.udpApp[0].destPort = 1000
59 **.Host.udpApp[0].messageLength = 1472B
**.wiredHost.udpApp[0].sendInterval = 0.0012304663467454s
**.Host.udpApp[0].stopTime = 100s
**.HostMobile.numUdpApps = 1
**.HostMobile.udpApp[0].typename = "UDPSinkThroughput"
64 **.HostMobile.udpApp[0].throughputWindowLength = 0.025s
**.HostMobile.udpApp[0].localPort = 1000

[Hallway_SingleAP]
*.numHosts = 1

```

Anschlusssituation fortwährend aktualisiert wird. Da die über die OAPs aufgebaute optische Verbindung transparent ist, haben die einzelnen OAPs keine eigene physikalische Adresse, sondern nur die jeweiligen Hosts. Die Aktualisierung der Weiterleitungstabelle kann in diesem Fall nur durch eine häufige Wiederholung von neu ausgelösten ARP-Anfragen erfolgen, da andernfalls fortwährend der falsche Switch-Ausgang zur Weiterleitung verwendet werden würde. Realisieren lässt sich solch eine häufige Aktualisierung durch Reduzierung der Gültigkeitsdauer gefundener ARP-Auflösungen. Dabei wirken sich allerdings die durch den Auflösungsprozess kurzfristig auftretenden Verzögerungszeiten nachteilig aus, da während dieser Zeit keine Ende-zu-Ende-Kommunikation stattfinden kann.

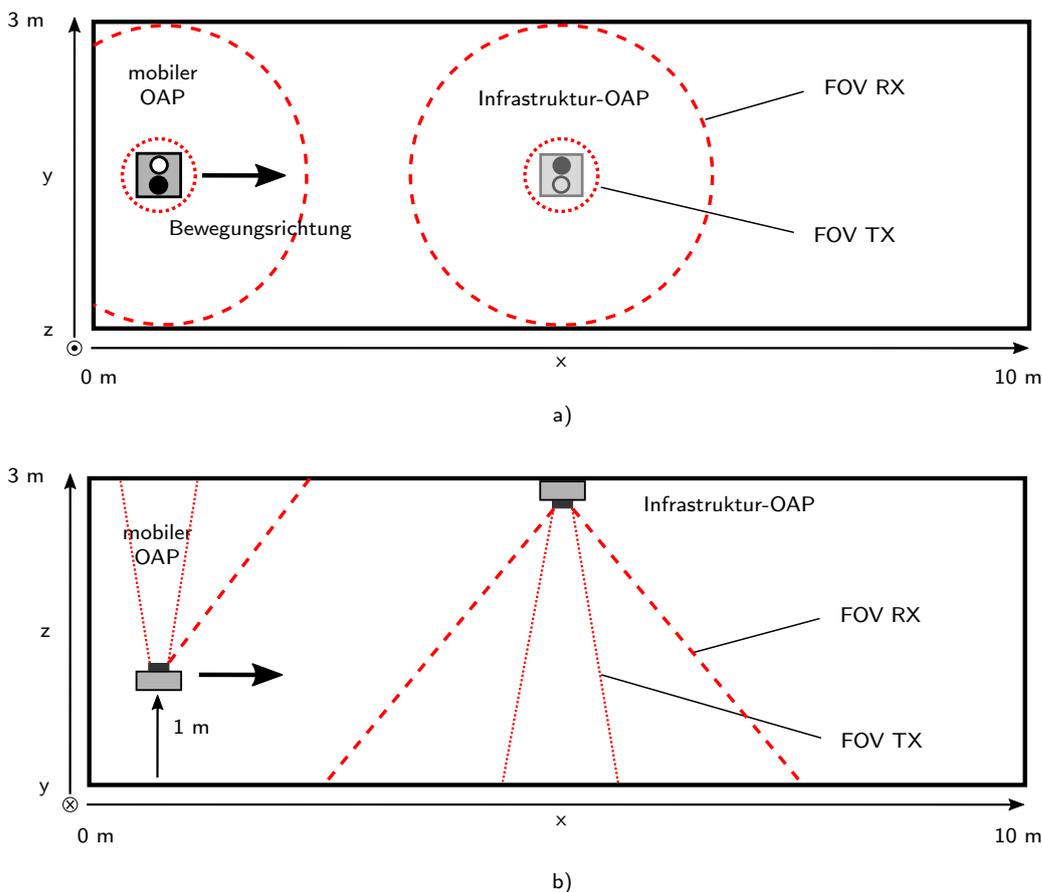


Abbildung 6.26: Darstellung des grundlegenden Simulationsszenarios „1 OAP“ a) in xy-Ebene und b) in xz-Ebene mit insgesamt zwei OAPs, wobei sich der mobile OAP im Laufe der simulierten Zeit entlang der x-Richtung fortbewegt.

6.2.3 Experimentelle Verifikation des Simulationsmodells

Um die mittels Simulationsmodell gewonnen Aussagen hinsichtlich ihrer Gültigkeit bewerten zu können, wurde das in Kapitel 6.2.2 Szenario „1 OAP“ experimentell nachgestellt. Der Aufbau ist in Abbildung 6.30 dargestellt.

Als Hardware-Referenz diente hierbei ein optisches Übertragungssystem des Fraunhofer-Instituts für Photonische Mikrosysteme (IPMS) namens „Li-Fi Hotspot“ [105], welches in Abbildung 6.29 zu sehen ist. Abbildung 6.29b zeigt dabei die optische aktive Frontseite bei demontierter Optik, d.h. das Array aus Transmitter-LEDs sowie die Photodiode, Abbildung 6.29c die dazugehörige bestückte Rückseite. Der „Li-Fi Hotspot“ stellt einen Konverter für die Bitübertragungsschicht, d.h. eine sog. Bridge, dar.

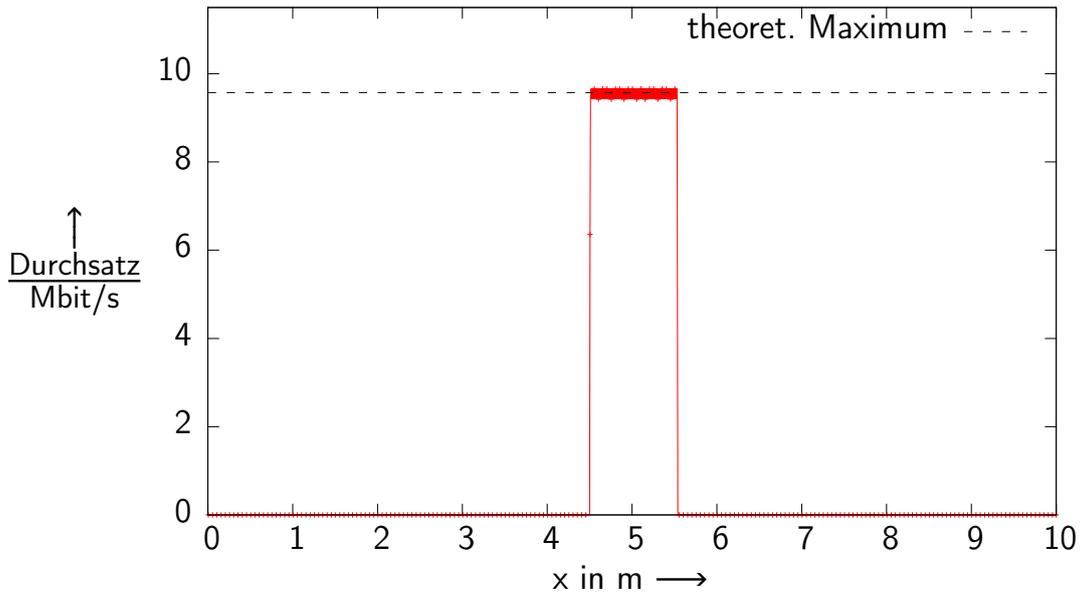


Abbildung 6.27: Zeitlicher und ortsabhängiger Verlauf des auf Anwendungsebene erreichten Durchsatzes im Basisszenario „1 OAP“. Der Empfänger bewegt sich mit konstanter Geschwindigkeit von $0,1 \text{ m s}^{-1}$ in positiver x -Richtung. Mit Eintritt in den Empfangsbereich zwischen $x = 4,50 \text{ m}$ und $x = 5,54 \text{ m}$, wird der theoretisch maximale Datendurchsatz von $9,571 \text{ Mbit/s}$ erreicht.

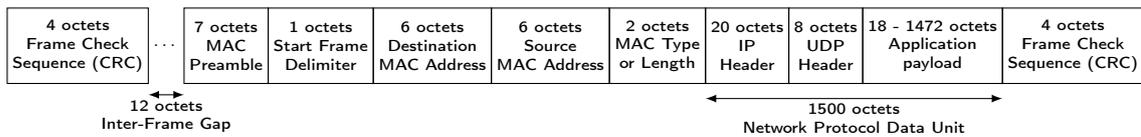


Abbildung 6.28: Aufbau eines Datenrahmens nach IEEE 802.3u, welcher ein UDP-Datagramm transportiert

Kabelgebunden erfolgt die Bitübertragung gemäß des im IEEE-Standard 802.3i beschriebenen Prinzips, welches auch als 10BASE-T bezeichnet wird (10 Mbit/s via „Twisted Pair“). Die Konvertierung erfolgt dann von, respektive nach, 10BASE-FL (10 Mbit/s via Glasfaser). In der vom IPMS zur Verfügung gestellten Bridge wird jedoch nicht in eine Glasfaser eingekoppelt, sondern mittels des Transmitter-LED-Arrays kabellos gerichtet emittiert. Empfängerseitig erfolgt die Signaldetektion über die vorhandene Photodiode. Der Öffnungswinkel der Photodiode Θ_{Rx} ist mit $\pm 35^\circ$, der des LED-Arrays Θ_{Tx} mit $\pm 10^\circ$ angegeben, wobei die mittlere optisch Übertragungsleistung 21 mW beträgt.

Zur Untersuchung des auf Anwendungsebene erzielbaren Durchsatzes, wird das Programm „iPerf“ in der Version 3.0 [106] verwendet, welches Datenübertragungen mittels UDP und TCP sowie das Einstellen zahlreicher Protokollparameter erlaubt. Auf dem Computer, welcher Daten empfangen soll (in Abbildung 6.30 als iPerf-Server bezeichnet) wird „iPerf“ im Server-Modus gestartet, auf dem zweiten Computer (in Abbildung 6.30 als iPerf-Client bezeichnet), erfolgt der Start des Verbindungstests im Client-Modus mit dem folgenden Aufruf:

```
iperf3 -c 10.0.0.2 -b10M -O5 -t65 -M1460 -w65535 --logfile out.txt
```

Als Parameter werden hier die Zielbandbreite von 10 Mbit/s ($-b10M$), eine um fünf Sekunden verzögerte Aufzeichnung zur Vermeidung der Erfassung von Einschwingvorgängen ($-O5$), eine Erfas-

Tabelle 6.5: Allgemeine Simulationsparameter

Bereich	Parameter	Wert
Front-End	FOV Öffnungswinkel Tx LED, Θ_{Tx}	16,92°
	FOV Öffnungswinkel Rx PD, Θ_{Rx}	14,93°
	Optische Sendeleistung, P_{Tx}	21 mW
	Empfindlichkeit der PD, S_{PD}	0,5 A W ⁻¹
	Fläche der PD, A	120 mm ²
	Bandbreite optischer Filter	20 nm
	Wellenlänge	1080 nm
	Verstärkung des TIAs, G_{TIA}	20 kΩ
	Systembandbreite, B	20 MHz
PHY	FEC	RS-Code (64,32)
	Modulationsart	2-PAM (OOK)
	Datenrate PHY	10 Mbit/s
Umgebung	Hintergrundhelligkeit, P_{bg}	6 mW/(cm ² nm)
	Temperatur, T	293 K
	Höhe Infrastruktur-OAP, $h_{fixed} = h_{ceil}$	3 m
	Höhe mobiler OAP, h_{mobile}	1 m
	Geschwindigkeit mobiler OAP, v_{mobile}	0,1 m s ⁻¹
Anwendung	UDP Paketgröße	1472 B
	UDP Paketintervall	1,23047 ms
	TCP Datenvolumen	125 MB
	TCP Maximale Segmentgröße (MSS)	1460 B
	TCP Maximale Fenstergröße	65536
	TCP Initial Soft Start Threshold	65536
	TCP Verzögerte Empfangsbestätigungen	nein
	TCP Selektive Empfangsbestätigungen	nein
	TCP Nagle-Algorithmus	nein
	TCP Fensterskalierung	nein
TCP Zeitstempelunterstützung	nein	

sungsdauer von 65 s (-t65), eine maximale TCP-Segmentgröße (MSS) von 1460 B (-M1460), eine TCP-Übertragungsfenstergröße von 65 535 B (-w65535) sowie ein Dateiname zur Aufzeichnung der Erfassung angegeben (--logfile out.txt). Gleichzeitig wurde jede Datenverbindung mit dem Programm „Wireshark“ [107] mitgeschnitten, was eine spätere Analyse ermöglicht.

Angefangen mit einer Positionierung des mobilen OAPs direkt unterhalb des deckenmontierten OAPs, wird der Versatz in x-Richtung in Schritten von 5 cm erhöht. Die Durchsatzmessung erfolgt dabei stationär an den jeweiligen Messpunkten. Die Messergebnisse sind in Abbildung 6.31 gemeinsam mit den aus der Simulation für das Experimentalszenario gewonnenen Werten dargestellt. Es zeigt sich, dass die winkelabhängige Entwicklung des Durchsatzes mit einer relativen Abweichung von ca. 3 % durch das Simulationsmodell dargestellt werden kann und dabei unterschätzt wird. Die Abweichung im Bezug auf die effektiv abgedeckte Fläche beträgt dabei 9,1 %. Als wahrscheinlichste Ursache für die Abweichung kommt hierfür die Modellierung der LED als Punktlichtquelle im Gegensatz zum im Experimentalsystem verbauten LED-Array in Frage, da die Überlagerung einzelner Öffnungswinkel zu

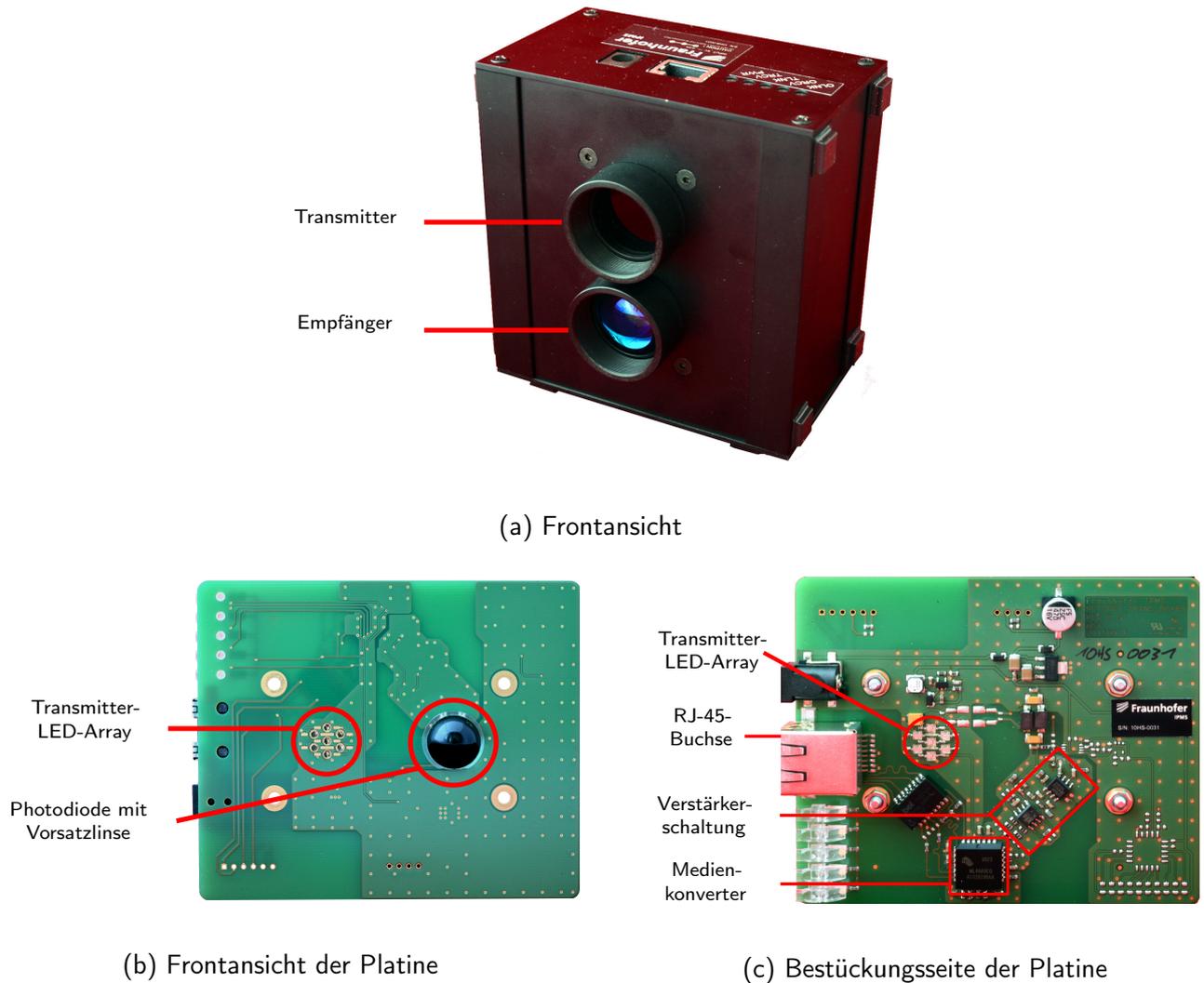


Abbildung 6.29: Zur experimentellen Verifikation verwendetes System „Li-Fi Hotspot“ des Fraunhofer IPMS in Dresden [105]

einer Aufweitung des effektiven Öffnungswinkels führt. Da keine Datenblattwerte für die optischen Eigenschaften der vor der PD liegenden Linse verfügbar sind, konnte ihre konzentrierende Wirkung nicht quantitativ berücksichtigt werden. Weiterhin lassen sich Mehrwegeausbreitungen sowie Imperfektionen des Experimentalsystems nicht ausschließen.

Die Datenblattwerte für die Öffnungswinkel bzw. die FOVs müssen aufgrund der großen Abweichung von ca. 63,93 % zu den durch die Gehäusegeometrie vorgegebenen Werten kritisch betrachtet werden. Leider liegen darüber hinaus keine weiteren unabhängigen Messungen des Systems vor, die als Referenz dienen können.

Der im Experiment maximal erzielte Durchsatz liegt mit 8,18 Mbit/s reproduzierbar unterhalb des in den Simulationen ermittelten Werts von ca. 9,571 Mbit/s (UDP). Als Ursache für die Abweichung von ca. 14,53 % kamen zunächst Bit- respektive Paketfehler in Frage. Deshalb wurde ein weiteres Experiment durchgeführt, um den mit dem verfügbaren „Li-Fi Hotspot“-System maximal erzielbaren Durchsatz bestimmen zu können. Zwei optische Einheiten wurden dabei in einem Abstand von 1,2 m auf einer eben Fläche positioniert und ohne Versatz zu einander ausgerichtet. Zur Gewährleistung optimaler Übertragungsbedingungen wurde der Raum verdunkelt und anschließend erneut eine 65 s dauernde Übertragung mit Hilfe des Programms „iPerf“ gestartet (5 s Vorlauf). Durch zeitgleiches

Mitschneiden der Datenübertragung mit Hilfe von „Wireshark“, ließen sich die stattfindenden Ereignisse im Nachgang analysieren. Dabei zeigte sich, dass der Anteil an wiederholten Paketen (10 von insgesamt 73.859) mit 0,014 % zu vernachlässigen ist und die signifikante Abweichung des erzielten Durchsatzes nicht erklären können. Abbildung 6.32 zeigt den Verlauf des mittels Wireshark ermittelten Durchsatzes.

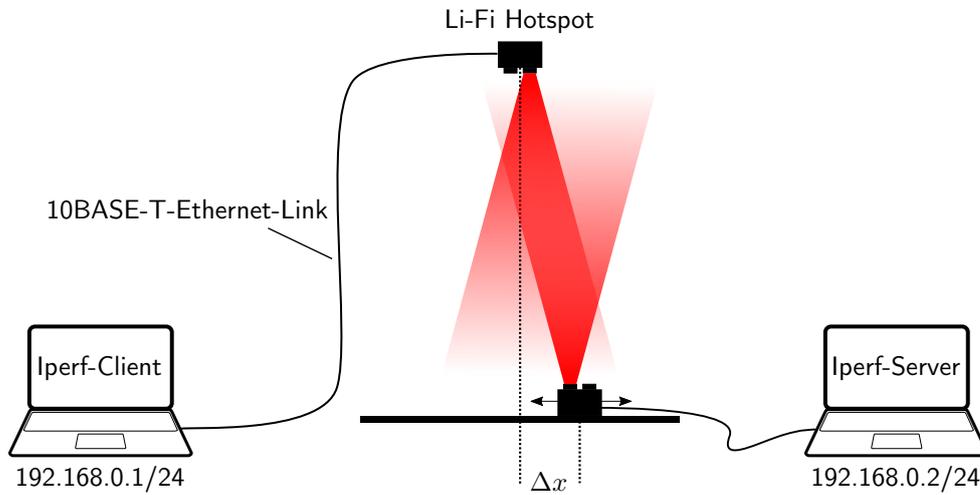


Abbildung 6.30: Experimenteller Aufbau zur Ermittlung des positions- und abstandsabhängigen Durchsatzes auf Anwendungsebene

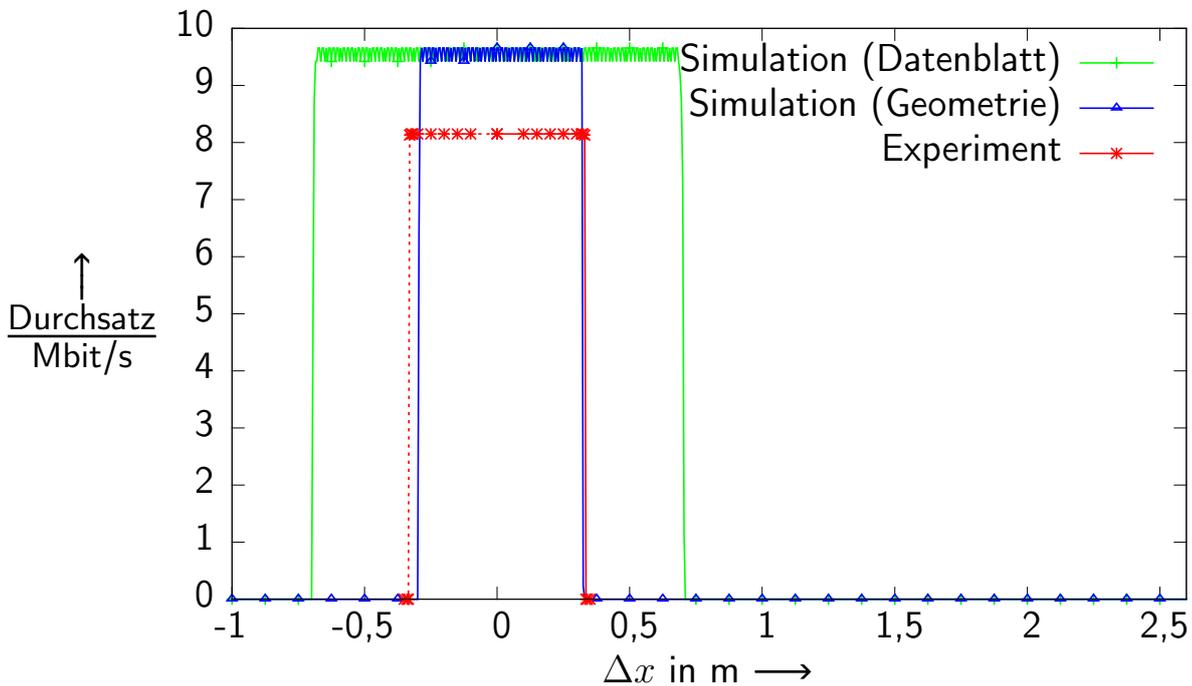


Abbildung 6.31: Vergleich des auf Basis der Datenblattwerte bzw. der Gehäusegeometrie simulierten mit dem experimentell bestimmten Durchsatz in Abhängigkeit der Positionierung in x-Richtung

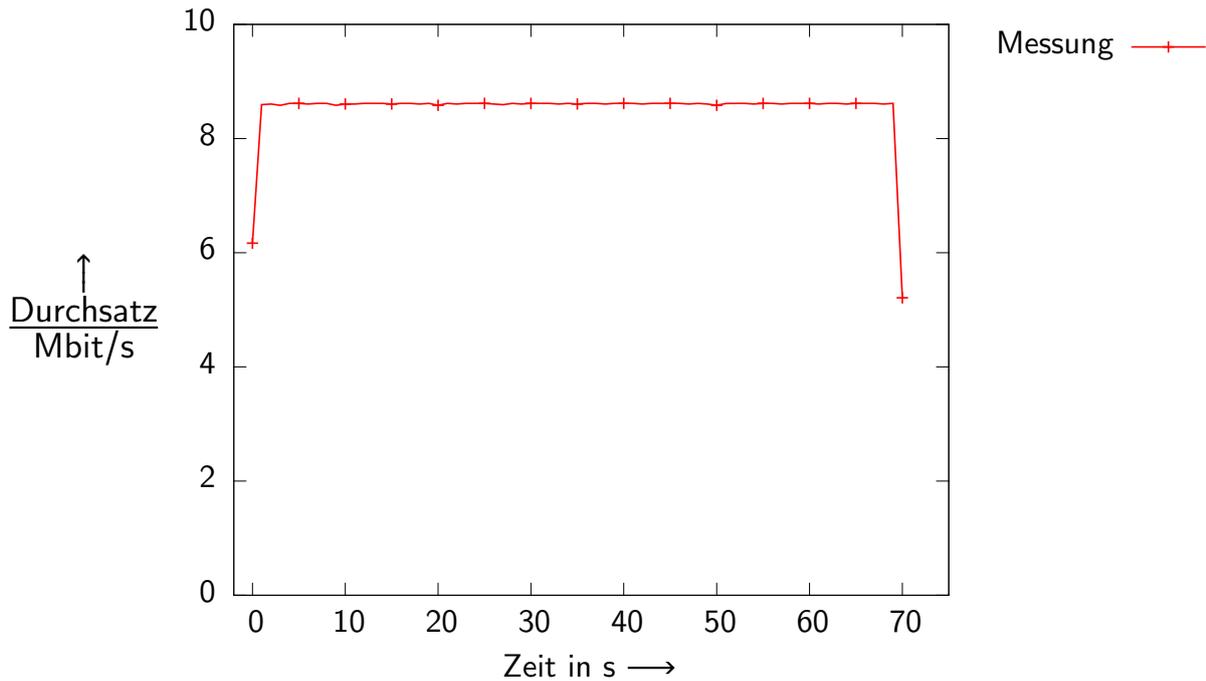


Abbildung 6.32: Mittels „Wireshark“ ermittelter Anwendungsdurchsatz im verdunkelten Raum bei 1,2 m optischer Distanz zwischen zwei „Li-Fi Hotspots“

6.2.4 Optimierungen der Umgebungsbedingungen

Störungen

Die ambiante Beleuchtung sowie Störungen durch andere optische Strahler beeinflussen maßgeblich das Rauschverhalten am Empfänger. In Abhängigkeit der Installationsumgebung (Hintergrundbeleuchtung, Sonneneinstrahlung, andere optisch aktive Einheiten) ist die Komponente P_{bg} in (2.28) mehr oder weniger stark ausgeprägt. Infolgedessen ergibt sich beim Empfang einer optischen Übertragung ein entsprechendes Signalrauschverhältnis. In Abbildung 6.33 ist die SNIR für unterschiedliche Hintergrundbeleuchtungsintensitäten bei konstanter optischer Sendeleistung dargestellt. Das thermische Rauschen stellt hierbei, als eine von der optischen Strahlungsleistung unabhängige additive Konstante, die untere Grenze der Rauschleistung dar (vgl. (2.28)). Mit zunehmend größerem optischen Leistungseintrag wirkt sich das Schrotrauschen der PD stärker aus, bis es letztlich die Rauschleistung dominiert. Daraus resultiert das ungewöhnlich konstante Verhalten nach dem sich Sende-FOV des Infrastruktur-OAP und Empfangs-FOV des mobilen OAPs zwischen $x = 3,5$ m und $x = 6,5$ m überlagern.

6.2.5 Optimierungen in der Bitübertragungsschicht

Öffnungswinkel Sender/Empfänger

Die Öffnungswinkel von Sender und Empfänger bestimmen unmittelbar die effektiv erzielbare Abdeckung. Durch eine geeignete Wahl der Optiken von Sender und Empfänger lassen sich zum einen die abgedeckten Bereiche stark beschränken, was durch die größere Anzahl von realisierbaren Femto- bzw. Attozellen zu einer verbesserten Nutzung des zur Verfügung stehenden Frequenz- bzw. Spektralbereiches ermöglicht. Weiterhin lässt sich so die Präzision der Nutzerlokalisierung erhöhen. Im Gegenzug

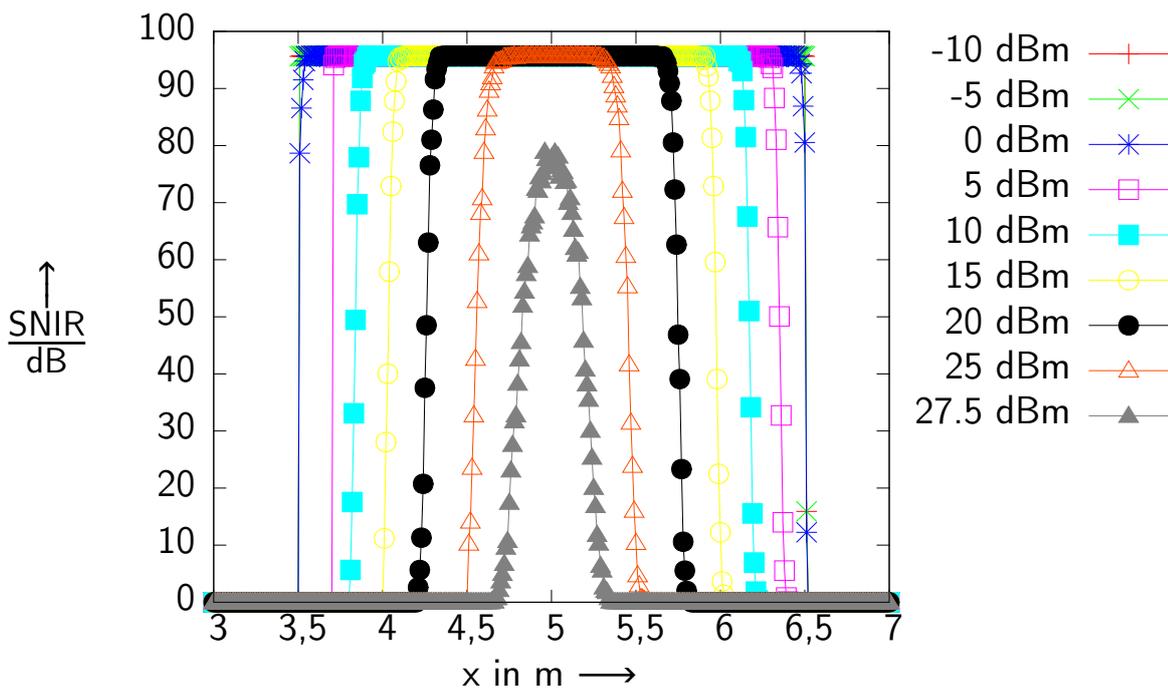


Abbildung 6.33: Signalrauschabstand in Abhängigkeit von Abstand, Orientierung und Hintergrundbeleuchtungsintensität mit einer 1024-PAM im Szenario „1 OAP“

erhöht sich die Anzahl der zu installierenden OAPs, bzw. kommunikationsfähiger Beleuchtungseinheiten, um eine gleichbleibende Abdeckung zu gewährleisten.

Soll die Abdeckung innerhalb eines Raumes mit möglichst geringer Anzahl von OAPs erfolgen, empfiehlt es sich andererseits die Öffnungswinkel von Sender und Empfänger möglichst groß zu wählen. Erfolgt eine zu dichte Platzierung bzw. werden die Öffnungswinkel zu groß gewählt, sind zur Vermeidung von Zellinterferenzen entweder ausreichend große Abstände der Trägerwellenlängen und stark selektive optische Filter (WDMA) oder mehrbenutzerorientierte Übertragungsverfahren wie bspw. DSSS/PSSS oder OFDM-A zu verwenden.

Abbildung 6.34 veranschaulicht den im mobilen Host auf Anwendungsebene resultierenden durchschnittlichen Durchsatz für verschiedene Kombinationen von Öffnungswinkeln in den Szenarien „1 OAP“ bis „15 OAP“. Es zeigt sich, dass der empfängerseitige Öffnungswinkel Θ_{Rx} einen größeren Einfluss auf die Übertragung hat, jedoch, wie zu erwarten, eine Wechselbeziehung mit dem senderseitigen Öffnungswinkel Θ_{Tx} besteht. Zusätzlich offenbart sich die Unterabdeckung im Szenario „1 OAP“. Hier wird selbst in der günstigsten Öffnungswinkelkombination nur ein durchschnittlicher Durchsatz von 7,8 Mbit/s erreicht, da hier der für die Simulationen gewählte maximale senderseitige Öffnungswinkel Θ_{Tx} von 25° limitierend wirkt.

Mit steigender Anzahl vorhandener Infrastruktur-OAPs vergrößert sich der Abdeckungsbereich und der durchschnittlich erzielte Durchsatz steigt. In Abhängigkeit der Kombination von Θ_{Rx} und Θ_{Tx} lassen sich dabei Sättigungseffekte beobachten. So wird beispielsweise mit einer Anzahl von 15 Infrastruktur-OAPs für alle Öffnungswinkelkombinationen der maximal erzielbare durchschnittliche Durchsatz von 9,571 Mbit/s erreicht (siehe Abbildung 6.34h). In Abhängigkeit der gewählten Öffnungswinkel lässt sich mit Hilfe des Simulationsmodells also die geringstmögliche Anzahl von OAPs bestimmen. Hierbei ist anzumerken, dass der simulierte Bewegungspfad des mobilen OAPs entlang der Mittelachse des Raumes erfolgt.

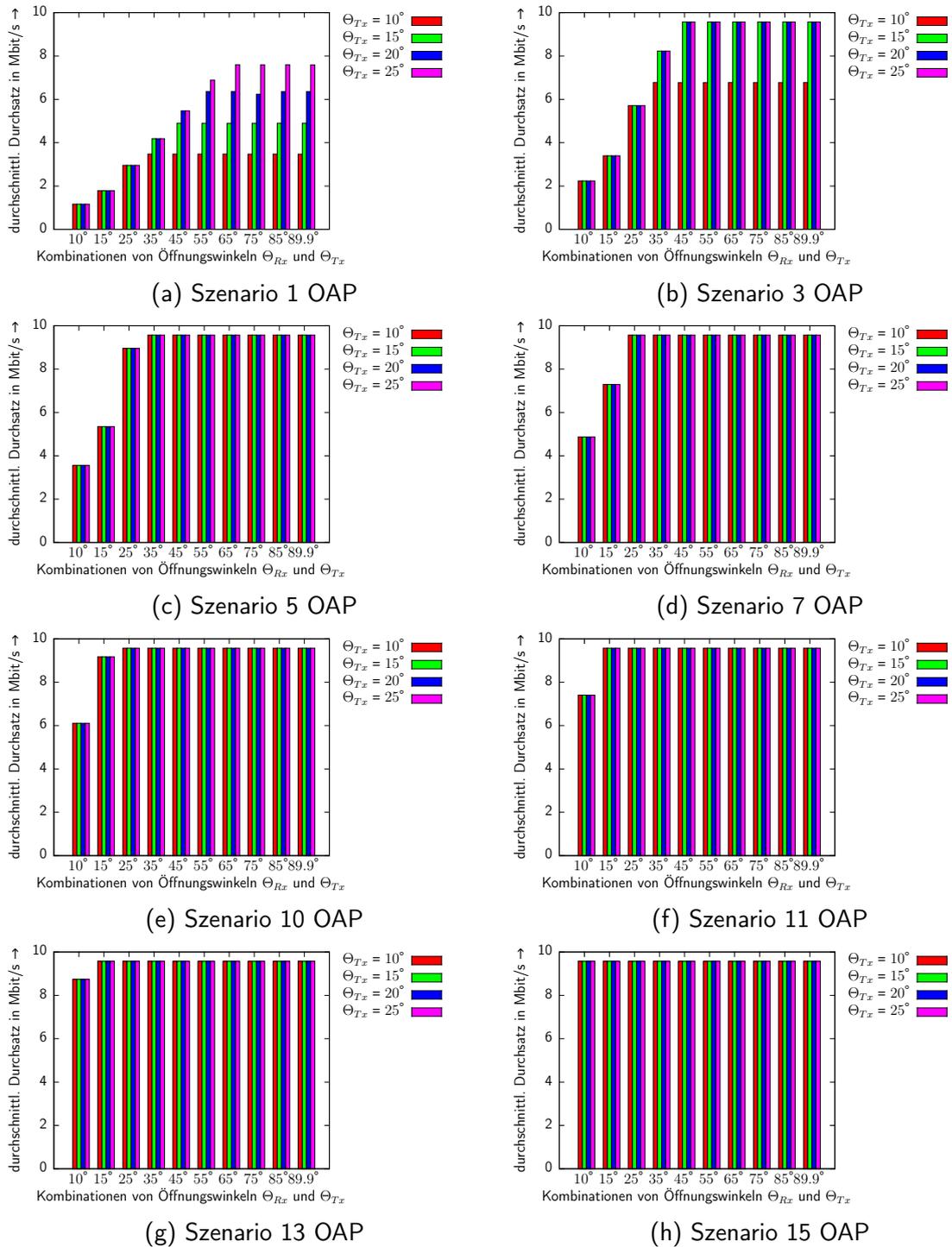


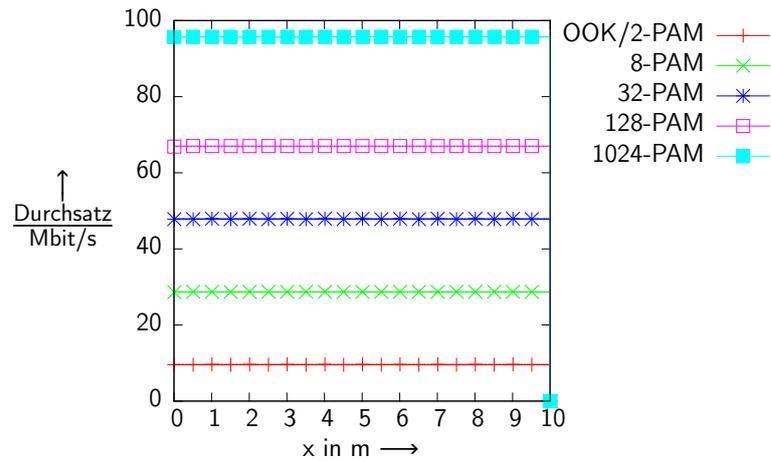
Abbildung 6.34: Durchschnittlich erreichter Durchsatz auf Anwendungsebene in Abhängigkeit der Öffnungswinkels Θ_{Rx} und Θ_{Tx} sowie der Anzahl verfügbarer Infrastruktur-OAPs

Modulationsverfahren

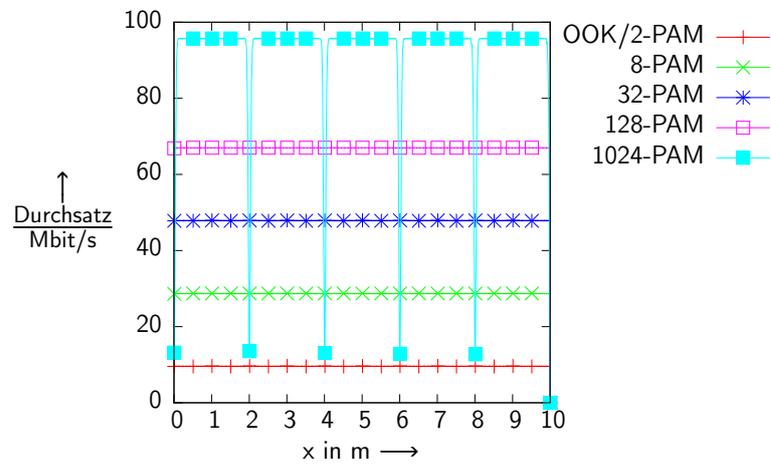
Der theoretisch maximal erzielbare Durchsatz auf Anwendungsebene ist unmittelbar vom gewählten Modulationsverfahren abhängig, da dieses den Durchsatz der Bitübertragungsschicht bestimmt. Wie bereits in Kapitel 2.2.2 erläutert, lassen sich Ein- oder Mehrträgerverfahren unterscheiden. Beispiele für Einträgerverfahren sind *Pulsamplitudenmodulation* (PAM) oder *Pulsphasenmodulation* (PPM). Bei beiden Verfahren lässt sich die Anzahl der möglichen diskreten Amplituden bzw. Pulspositionen M festlegen (M -PAM bzw. M -PPM). Bei gleicher Symbolrate erhöht sich so die Datenrate, wobei die Störempfindlichkeit (M -PAM) bzw. der Bandbreitenbedarf (M -PPM) ebenfalls zunimmt.

Um den Einfluss des eingesetzten Modulationsverfahrens auf den Anwendungsdurchsatz und die effektiv erzielbare Abdeckung quantitativ bestimmen zu können, wurden vergleichende Simulationen mit variiertem Parameter M für verschiedene Szenarien durchgeführt. Die Ergebnisse sind in Abbildung 6.35 dargestellt. Beim Einsatz einer Optik mit kleinen Öffnungswinkeln, wie sie beim Referenzsystem existiert, zeigt sich, dass vom kleinstmöglichen Wert $M = 2$ mit einer physikalischen Datenrate von 10 Mbit/s bis zu mindestens $M = 1024$ mit einer physikalischen Datenrate von 100 Mbit/s entlang der Mittelachse des Raumes eine vollständige Abdeckung möglich ist, falls zehn Infrastruktur-OAPs installiert sind (Szenario „10 OAP“). Beim Einsatz einer Optik mit annähernd doppelt so großen Öffnungswinkel und demzufolge auf fünf reduzierter Anzahl von Infrastruktur-OAPs, ist ebenfalls noch eine vollständige Abdeckung möglich. Erst bei einem Modulationsgrad von $M = 1024$ kommt es in diesem Szenario zu Einbrüchen des auf Anwendungsebene erzielten Durchsatzes in den Bereichen zwischen den Infrastruktur-OAPs. Dies stellt eine Auswirkung der bei identischer BER infolge des höheren Modulationsgrad M erhöhten Anforderung an die SNR dar. Im Vergleich zum Szenario mit schmal abstrahlender, respektive erfassender, Optik, wird hier die Sendeleistung auf größere Raumwinkel verteilt, so dass sich insbesondere in den Abstrahlungsrandbereichen der durch das Lambertsche Gesetz in (2.23) eine deutlich reduzierte effektive Abstrahlleistung einstellt. Infolgedessen sinkt der Signalrauschabstand auf ca. 48 dB, wodurch die Bitfehlerrate auf 2,35 % und die Paketfehlerrate trotz Vorwärtsfehlerkorrektur mittels RS(64,32) auf 92,04 % ansteigen. Der Abdeckungsbereich pro Infrastruktur-OAP reduziert sich dabei von 3,14 m² um ca. 5,86 % auf 2,96 m². Um die Abdeckungslücken auszugleichen, kann das System einerseits mit einem Modulationsgrad von maximal $M = 512$ betrieben werden, wodurch sich der maximal erzielbare Durchsatz um 10 % reduziert, oder die Anzahl der verfügbaren Infrastruktur-OAPs wird erhöht. Für das Szenario „6 OAP“ sind keine Einbrüche des Durchsatzes zu verzeichnen (siehe Abbildung 6.35c).

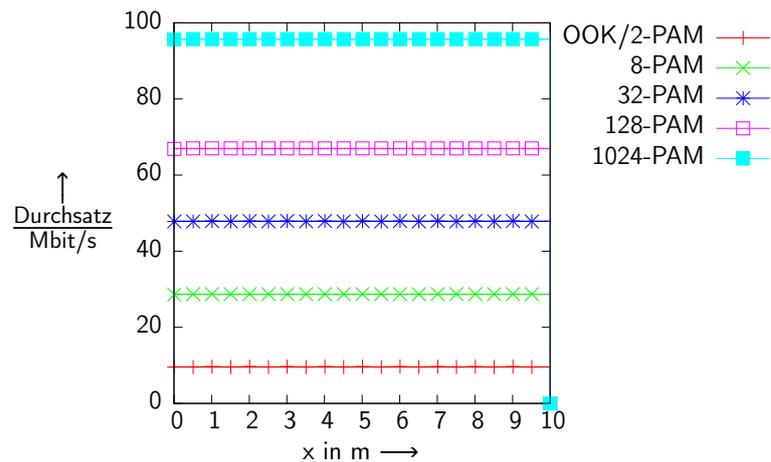
Wird statt M-PAM M-PPM als Modulationsverfahren eingesetzt, zeigen sich andere Resultate. In Abbildung 6.36 ist erneut der auf Anwendungsebene erzielte Durchsatz in Abhängigkeit von Zeit und Position in x-Richtung dargestellt. In keinem Szenario sind Abdeckungslücken auffällig, d.h. bei identischer Hintergrundrauschleistung und identischer optischer Sendeleistung wird eine geringere Bitfehlerrate erreicht. Dieses Verhalten liegt im eingesetzten Modulationsverfahren begründet. Bei M-PPM handelt es sich, im Gegensatz zu M-PAM, um ein energie-, aber nicht bandbreiteneffizientes Verfahren. Die Information wird mit Hilfe der Position eines innerhalb eines zeitlichen Bezugsrahmens verschiebbaren Pulses mit variabler Länge, aber konstanter Energie, übertragen. Mit steigendem Modulationsgrad M reduziert sich die Pulsdauer, da der Bezugsrahmen zeitlich konstant ist. Durch die Verkürzung der Pulsdauer muss die Sendeleistung erhöht werden, um die Signalenergie konstant zu halten. Trotz der auf Grund der größeren notwendigen Bandbreite am Empfänger ebenfalls größeren Rauschenergie, sinkt die SNR-Anforderung zur Erzielung identischer Bitfehlerraten bei höheren Modulationsgraden M . Dies ist der Fall, da die rauschrelevante Bandbreite am Empfänger mit $M/\log_2 M$, die Spitzenleistung im Signal jedoch mit M wächst (siehe [108], [109], [110]). Das lässt einen hohen Modulationsgrad zunächst wünschenswert erscheinen. Um die kurzen Pulse mit



(a) Szenario „10 OAP“ mit $\theta_{TX} = 16,92^\circ$ und $\theta_{RX} = 14,93^\circ$



(b) Szenario „5 OAP“ mit $\theta_{TX} = 20^\circ$ und $\theta_{RX} = 30^\circ$



(c) Szenario „6 OAP“ mit $\theta_{TX} = 20^\circ$ und $\theta_{RX} = 30^\circ$

Abbildung 6.35: Durchsatz auf Anwendungsebene für verschiedene Modulationsgrade einer PAM sowie unterschiedliche Öffnungswinkel θ_{TX} bzw. θ_{RX} bei Vorwärtsfehlerkorrektur mittels RS(64,32) in den Szenarien „10 OAP“, „5 OAP“ und „6 OAP“

hoher Leistung zur erzeugen und zu detektieren, werden allerdings hohe Anforderungen an die Treiber- bzw. Empfängerschaltungen gestellt, was die Implementierungskosten deutlich steigen lässt [111].

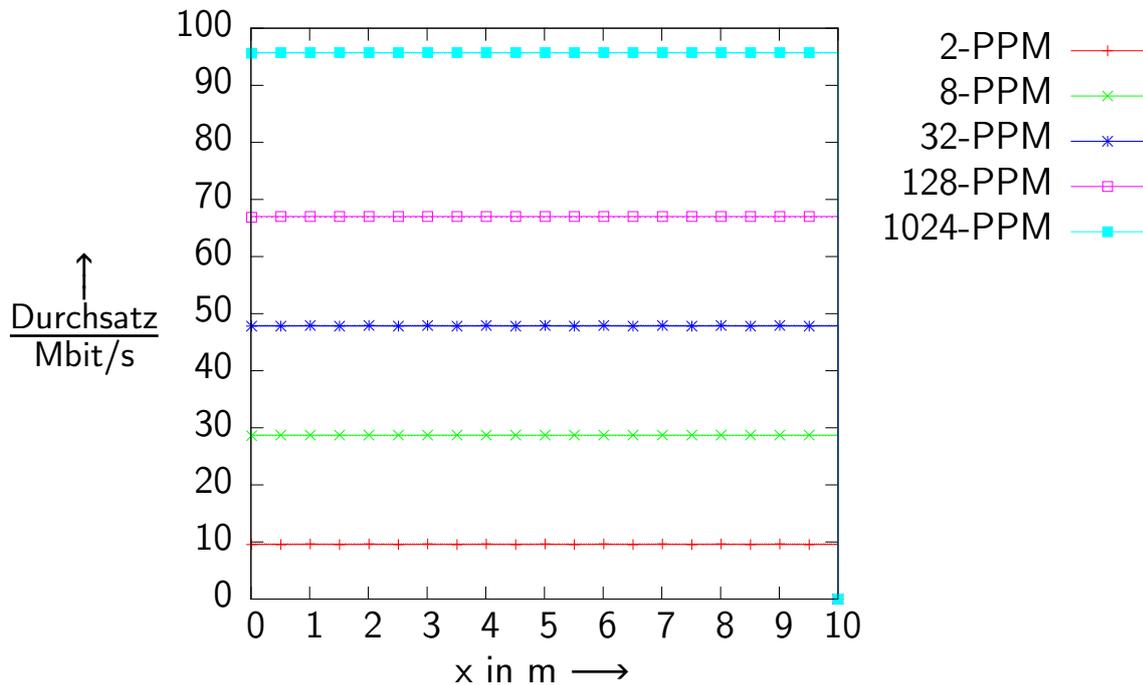


Abbildung 6.36: Durchsatz auf Anwendungsebene in Abhängigkeit der Anzahl von möglichen Pulspositionen M für das Modulationsverfahren M-PPM mit $\theta_{TX} = 20^\circ$ und $\theta_{RX} = 30^\circ$ im Szenario „5 OAP“

Vorwärtsfehlerkorrektur

Wie bereits in Kapitel 2.2.1 beschrieben, lässt sich der Datendurchsatz bei einer Übertragung über rauschbehaftete Kanäle mittels Vorwärtsfehlerkorrektur („Forward Error Correction“/FEC) verbessern. Vor allen Dingen bei widrigen Übertragungsbedingungen erlaubt es der Einsatz von FEC überhaupt erst, dass Daten übertragen werden können. Im Kontext von OWC bedeutet dies, dass sich die effektive Abdeckung vergrößern lässt, da die Anforderungen an den zur Übertragung notwendigen Signalrauschabstand verringert werden können. Zur quantitativen Untersuchung der Auswirkungen des Einsatzes von Algorithmen zur Vorwärtsfehlerkorrektur, wurden die Parameter für die im Simulationsmodell verfügbaren Reed-Solomon-Codes ($RS(n, k)$) in einer Parameterstudie verglichen. Dabei wird in Abhängigkeit der Parameter n und k , des Signalrauschverhältnisses, des Modulationsverfahrens sowie der Länge des zu empfangenden Pakets die Wahrscheinlichkeit für das Vorhandensein eines nicht korrigierbaren Fehlers berechnet und mit einem Zufallswert aus dem geschlossenen Intervall $(0, 1)$ verglichen. Ist das Paket fehlerhaft, wird es nach dem Empfang im OAP verworfen und fließt somit nicht in die Berechnung des Anwendungsdurchsatzes ein. Der simulativ ermittelte Anwendungsdurchsatz für verschiedene RS-Codes sowie 2- bzw. 1024-PAM mit zwei verschiedenen Öffnungswinkelkombinationen im Simulationsszenario „1 OAP“ ist in Abbildung 6.37 dargestellt.

In der Basiskonfiguration (siehe Parameterwerte in Tabelle 6.5) mit einer Kodierungsrate von 0,5, d.h. 50% Redundanz, wird ein Anwendungsdurchsatz von 9,571 Mbit/s erreicht. Dabei ergibt sich 2 m unterhalb des Infrastruktur-OAPs ein Abdeckungskreis mit einem Durchmesser von ca. 0,98 m. Ohne Redundanz verdoppelt sich der Anwendungsdurchsatz, aufgrund der relativ kleinen Öffnungswinkel

und des daraus resultierenden günstigen Signalrauschverhältnisses, reduziert sich der Durchmesser des Abdeckungskreises selbst ohne FEC fast nicht. Der Einfluss des gewählten RS-Codes offenbart sich erst bei Nutzung von 1024-PAM (siehe 6.37b) oder Optiken mit größeren Öffnungswinkeln, wie in den Abbildungen 6.37c und 6.37d dargestellt.

Bei Öffnungswinkeln von $\theta_{TX} = 20^\circ$ und $\theta_{RX} = 30^\circ$ und Nutzung einer 1024-PAM zeigen sich deutliche Auswirkungen des gewählten RS-Codes auf den erzielbaren Durchsatz, respektive den Abdeckungsbereich. Da sich durch die größere Aufspreizung der optisch emittierten Leistung bei konstanter Hintergrundrauschleistung ein geringeres Signalrauschverhältnis ergibt, steigt die Bitfehlerwahrscheinlichkeit in den Abdeckungsrandbereichen. Mit einer Kodierungsrate von 0.5, d.h. 50% Redundanz, erzielt man unter Nutzung von RS(64,32) einen Abdeckungsbereich mit einem Durchmesser von ca. 1,84 m bzw. 2,66 m² Abdeckungsfläche bei einem Anwendungsdurchsatz von 95,71 Mbit/s. Mit RS(255,127) kann der Abdeckungskreis auf einen Durchmesser von 1,98 m, d.h. um ca. 7,6% vergrößert werden. Die Abdeckungsfläche steigt auf 3,08 m² bzw. vergrößert sich um 15,8%. Durch die von $M = 6$ auf $M = 8$ erhöhte Symbollänge ergeben sich insgesamt mehr Korrekturmöglichkeiten, weshalb große Werte für n , bzw. k erstrebenswert scheinen. Der für eine schnelle En- sowie Dekodierung notwendige Hardwareaufwand wächst jedoch ebenfalls vor allem mit der Redundanz [112],[113], so dass heute üblicherweise Codes wie RS(255, 239) und RS(255,251) zum Einsatz kommen (z.B. xDSL, oder Audio-CD). Verwendet man im simulierten Szenario RS(255,251), erhöht sich der maximal erzielbare Anwendungsdurchsatz auf 188,4 Mbit/s, womit im Vergleich zu den Ergebnissen für RS(255,127) allerdings eine Verringerung der abgedeckten Fläche um 62,0% einhergeht. Diese beträgt nun 1,17 m² bzw. der Durchmesser des Abdeckungskreises misst 1,22 m.

Auch in diesem Aspekt des Anwendungsbeispiels zeigen sich die Wechselwirkungen zwischen erzielbarem Durchsatz und abgedeckter Fläche sowie den Systemparametern auf der Ebene der Bitübertragungsschicht. Sind die Kanaleigenschaften eines Übertragungskanals bekannt, beispielsweise ob es sich um Gaußsches oder Ricesches Fading handelt, sowie der sich einstellende Signalrauschabstand, lassen sich, wie in [114] gezeigt, optimale Kodierungsraten wählen, um ein auszulegendes optisches Kommunikationssystem gemäß den Nutzer- oder Gebäudeanforderungen zu entwerfen.

6.2.6 Optimierungen in der Datensicherungsschicht

Da das Simulationsmodell an das Experimentalsystem angepasst, bzw. diesem nachempfunden wurde, werden über die optische Verbindungsstrecke sog. Ethernet-Frames übertragen. Jeder dieser Datenrahmen enthält dabei eine im IEEE-Standard 802.3 definierte Prüfsumme („Frame Check Sequence“ bzw. FCS) mit einer Länge von vier Oktetten, die mittels 32-Bit CRC ermittelt wird.

Auf der optischen Übertragungsstrecke kommen darüber hinaus weder zusätzliche Datensicherungsmaßnahmen noch ein Medienzugriffsverfahren zum Einsatz, da das zur Verfügung stehende Experimentalsystem diese ebenfalls nicht unterstützt. Die eingesetzte Vorwärtsfehlerkorrektur mittels RS-Codes ist der Bitübertragungsschicht zuzuordnen.

6.2.7 Optimierungen in der Vermittlungsschicht

Die Vermittlungsschicht wird, wie in Kapitel 6.2.1 beschrieben, mittels IPv4 realisiert. Ihre Aufgabe ist die Ermittlung einer Route zum Zielsystem, wozu das „Address Resolution Protocol“, kurz ARP, genutzt wird. Hauptsächlich besteht somit eine Abhängigkeit zur Topologie des Netzwerks. Das zur Simulation verwendete Netzwerk (siehe Abbildung 6.25) enthält einen Switch zur Verbindung der Infrastruktur-OAPs mit dem Host A. Nach dem Start der Simulation muss zunächst eine Route zum Zielsystem,

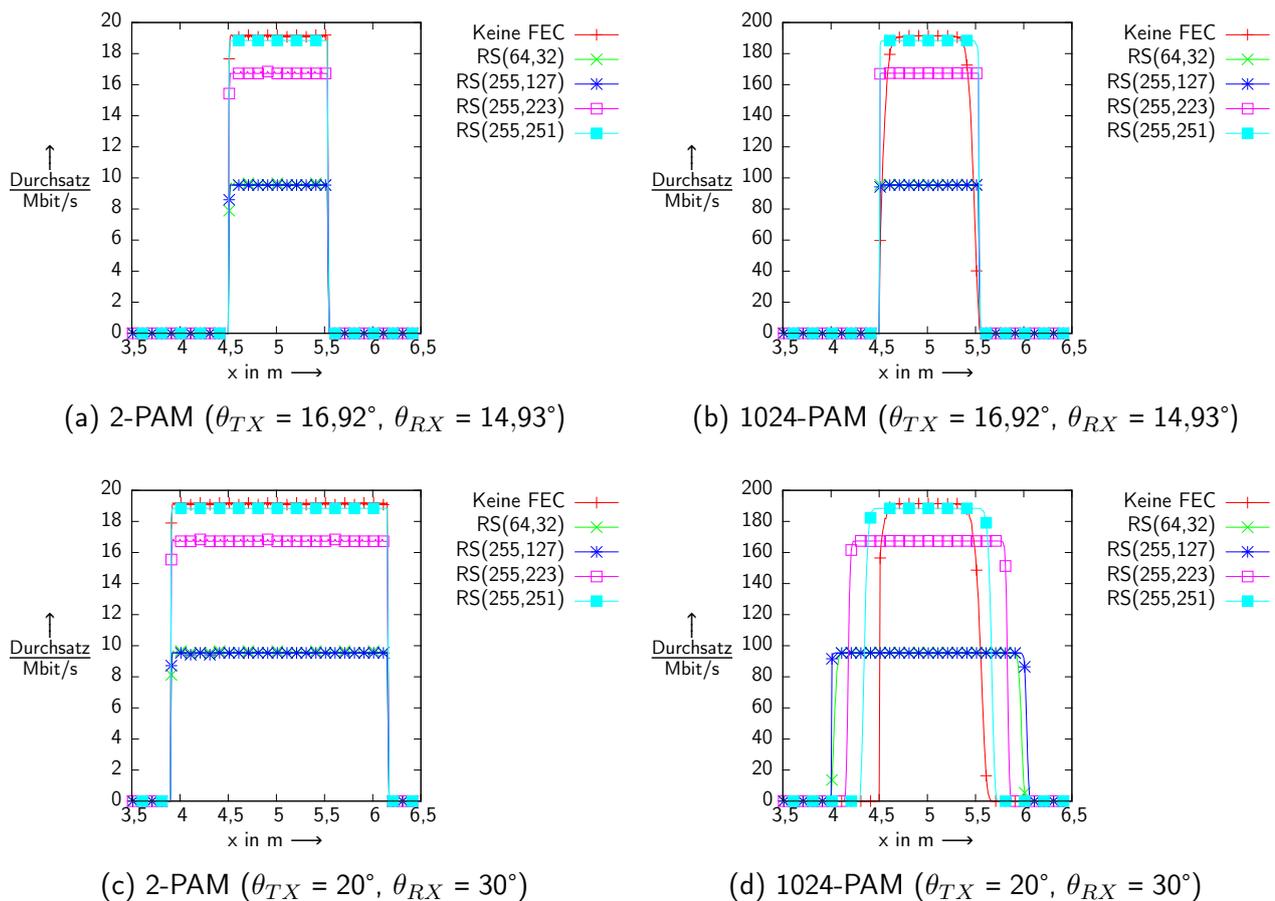


Abbildung 6.37: Durchsatz auf Anwendungsebene für 2-PAM und 1024-PAM für verschiedene RS-Codes zur Vorwärtsfehlerkorrektur sowie unterschiedliche Öffnungswinkel θ_{TX} bzw. θ_{RX}

dem Host B, ermittelt werden. Dazu werden zunächst sog. *ARP-Requests* als Broadcast-Nachricht vom Host A versendet, welche der Switch über alle seine Ausgangs-Ports an alle Infrastruktur-OAPs weiterleitet. Ist eine optische Kommunikation zwischen Infrastruktur- und Mobil-OAP möglich, wird der ARP-Request an den Ziel-Host weitergeleitet und dort mittels *ARP-Response* beantwortet. Nach anschließender optischer Übertragung, gelangt diese Antwort über den Port, an welchem der zur Übertragung benutzte Infrastruktur-OAP angeschlossen ist, zum Switch. Dieser extrahiert die im Datenrahmen enthaltene MAC-Adresse und trägt die Kombination aus MAC-Adresse und Port für zukünftige Weiterleitungen in seine „Source Address Table“ ein.

Da die optische Übertragungsstrecke oberhalb der Bitübertragungsschicht vollständig transparent ist, kommt es im Verlauf der Simulation zu einem Weiterleitungsfehler und somit Datenverlust, falls UDP eingesetzt wird. Der mobile OAP verlässt nach einiger Zeit das FOV des zur ursprünglichen Routenfindung benutzten Infrastruktur-OAPs. Da der Switch die für das Zielsystem bestimmten Datenrahmen aber weiterhin an den initial bestimmten Port weiterleitet, können diese das Zielsystem nicht mehr erreichen. Um dieses Problem zu beheben, muss der Prozess der Routenfindung erneut gestartet werden. Dazu ist es möglich, den Parameterwert für die Gültigkeitsdauer der in den Hosts gespeicherten ARP-Auflösungsergebnisse auf einen kleinen Wert zu setzen (Standardwert nicht definiert, aber oft zwischen 60 s und 300 s).

Die Auswirkungen dieses Verhaltens sind in Abbildung 6.38 dargestellt. Durch die fehlerhafte Paketweiterleitung und die ausbleibende Zustellung werden die TCP-internen Antwortzeitschranken

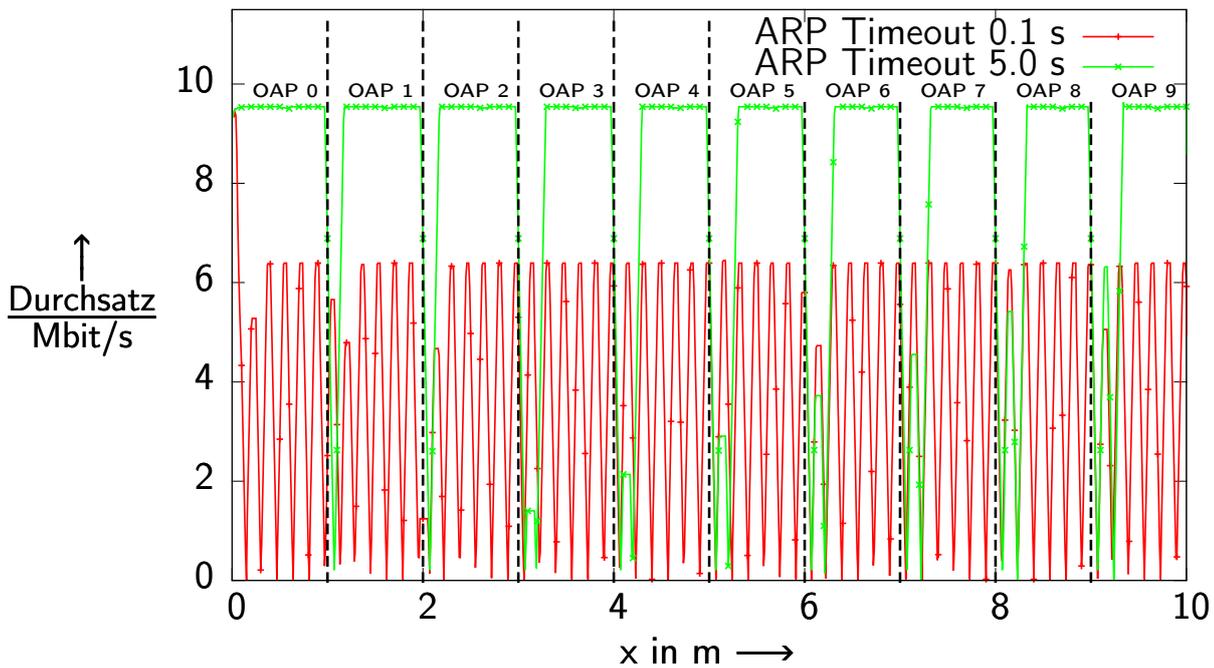
überschritten und das „Slow Start“-Verfahren läuft wiederholt an. Infolgedessen sinkt der erzielte Durchsatz signifikant. Kommt hingegen UDP zum Einsatz, fällt der Durchsatz zwar während der Zeit der fehlerhaften Weiterleitung, steigt danach aber unmittelbar wieder auf das Maximalniveau an, da es sich bei UDP um ein verbindungsloses Protokoll handelt. Die deutlich kürzere Gültigkeitsdauer der ARP-Auflösungsergebnisse bewirkt eine häufigere Adressauflösung, wodurch sich der auf Anwendungsebene erzielte Maximaldurchsatz um ca. 1,26 % reduziert (UDP). Im Gegenzug kann durch diese Maßnahme der im Simulationsszenario erzielte durchschnittliche Durchsatz auf Anwendungsebene für TCP-Übertragungen von 6,94 Mbit/s auf 7,64 Mbit/s, d.h. um 9,61 %, gesteigert werden. Kommt UDP als Transportprotokoll zum Einsatz, erhöht sich der durchschnittliche Anwendungsdurchsatz durch die häufigere Aktualisierung von 5,54 Mbit/s auf 9,47 Mbit/s um 70,94 %. Diese deutliche Verbesserung ist darauf zurück zu führen, dass keine gesicherte Datenübertragung stattfindet. Verlässt der mobile OAP das FOV des aktuell via ARP ermittelten Infrastruktur-OAPs, werden bei längerer Gültigkeit der ARP-Auflösungsergebnisse deutlich mehr Pakete nicht zugestellt. Erst nach neu ausgelöster Adressauflösung kann wieder erfolgreich übertragen werden. In Abbildung 6.38b ist dieser Effekt deutlich an den ca. 0,5 m langen Abschnitten ohne erfolgreiche Übertragungen erkennbar.

6.2.8 Einfluss des Transportprotokolls

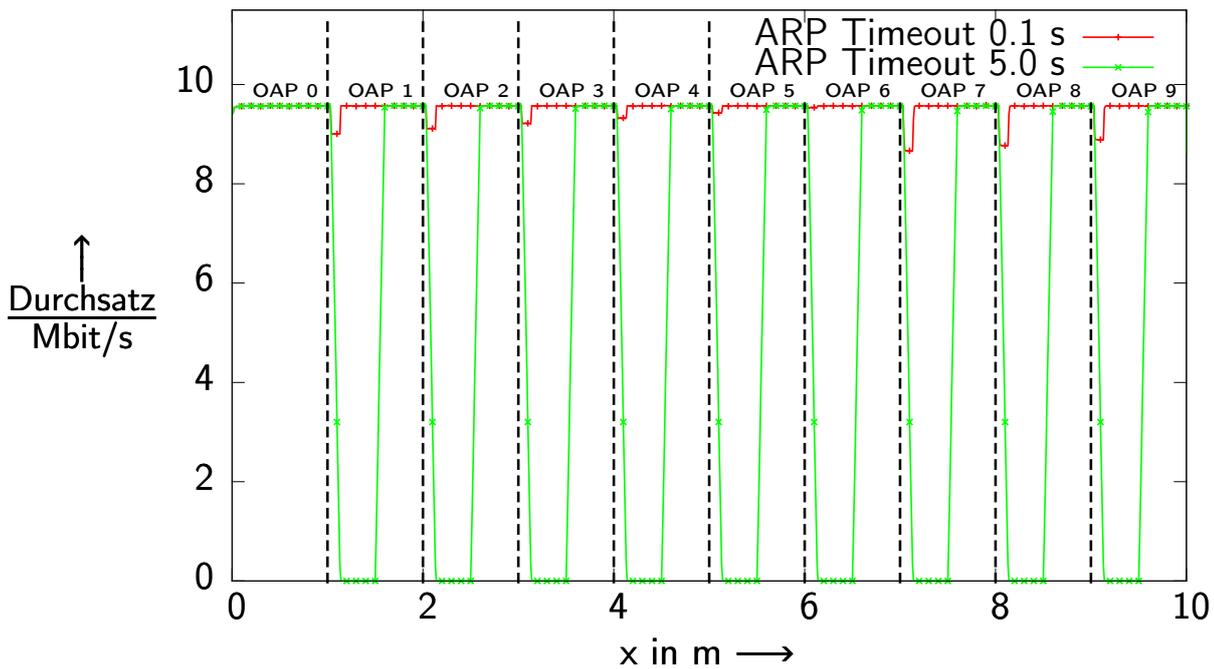
Jedes der bei der Datenübertragung eingesetzten Protokolle trägt durch das Hinzufügen von Verwaltungsinformationen („Header“ und „Footer“) zur Expansion der Anwendungsdaten bei. Weiterhin haben auch die protokollspezifischen Kontrollmechanismen einen signifikanten Einfluss auf die Übertragung. Ein Beispiel hierfür ist das eingesetzte Transportprotokoll. Je nach Art der zu übertragenden Daten kommen hier in der Regel TCP [115] oder UDP [116] zum Einsatz, in seltenen Fällen auch IPX [117], AppleTalk [118] oder NetBIOS [119].

Um die Auswirkungen der Wahl des Transportprotokolls sowie dessen Konfiguration auf den Datendurchsatz auf Anwendungsebene darstellen zu können, wurden vergleichende Simulationen durchgeführt. Dabei wurden für das UDP Paketgröße und Sendeintervall vorgegeben, die in Kombination dem theoretischen Durchsatzmaximum entsprechen. Bei Übertragungen mittels TCP ist dies nicht möglich, da Pakete in Abhängigkeit vom Eintreffen der Empfangsbestätigungen (ACKs), d.h. selbsttaktend, gesendet werden. Für TCP-Simulationen wurde deshalb eine zu übertragende Datenmenge vorgegeben, die ausreichend groß dimensioniert ist, um auch bei maximalem Durchsatz nicht innerhalb der simulierten Zeit übertragen werden zu können (siehe Tabelle 6.5). Weiterhin kann das Verhalten von TCP durch die Wahl der TCP-Variante, die Größe des maximalen Übertragungsfensters („Maximum Window Size“/MSS), ob verzögerte Empfangsbestätigungen genutzt werden sollen, ob selektive Empfangsbestätigungen gesendet werden sollen sowie ob kleine Pakete unmittelbar oder erst nach Wartezeit versendet werden sollen (Nagle-Algorithmus) bestimmt werden.

In Abbildung 6.39 ist der auf Anwendungsebene gemessene Durchsatz in Abhängigkeit vom zeitlichen Bewegungsverlauf bzw. Position in x-Richtung dargestellt. Bereits zu Beginn zeigt sich deutlich der Unterschied im Verhalten. Während die UDP-basierte Übertragung unmittelbar nach der erfolgreichen Ermittlung der Route zum Ziel-Host mittels ARP und IP beginnt, muss zunächst die für die TCP-Übertragung notwendige TCP-Verbindung hergestellt werden. Danach beginnt der sog. „Slow Start“ der in RFC 5681 „TCP Congestion Control“ [120] definiert ist. Zu Beginn wird das Übertragungsfenster auf $2 \cdot MSS$, $3 \cdot MSS$ oder $4 \cdot MSS$ initialisiert. Anschließend wird das Übertragungsfenster pro empfangener Empfangsbestätigung (ACK) um maximal ein MSS vergrößert, was zu einem exponentiellen Wachstum führt. Sobald nun der sog. „Soft start threshold“ erreicht ist, wird das Übertragungsfenster um maximal ein MSS pro Paketumlaufzeit (Round-Trip-Time RTT), d.h. linear, erhöht, solange der Empfang aller Pakete vom Empfänger rechtzeitig quittiert worden ist („Congestion avoidance“). Das Wachstum



(a) Übertragung mittels TCP



(b) Übertragung mittels UDP

Abbildung 6.38: Datendurchsatz auf Anwendungsebene für verschiedene Gültigkeitsdauern von ARP-Auflösungsergebnissen und die Transportprotokolle TCP und UDP im Szenario „10 OAP“, d.h. mit 10 optischen Access Points. Die kontinuierliche Bewegung in x-Richtung bewirkt einen „Zellenwechsel“, der einen Einbruch der effektiven Durchsatzrate zur Folge hat.

Tabelle 6.6: Maximaler und durchschnittlicher Durchsatz auf Anwendungsebene in Abhängigkeit der Gültigkeitsdauer für ARP-Auflösungsergebnisse im Szenario „10 OAP“

Topologie	Konfiguration		Durchsatz		
	Gültigkeitsdauer	Transportprotokoll	Mittelwert	Maximum	Standardabweichung
Switch	0,1 s	TCP	3,73 Mbit/s	6,39 Mbit/s	4,54 Mbit/s
		UDP	9,52 Mbit/s	9,54 Mbit/s	0,55 Mbit/s
	5,0 s	TCP	8,00 Mbit/s	9,54 Mbit/s	3,44 Mbit/s
		UDP	5,54 Mbit/s	9,57 Mbit/s	4,70 Mbit/s
Hub	0,1 s	TCP	3,73 Mbit/s	9,46 Mbit/s	4,51 Mbit/s
		UDP	9,56 Mbit/s	9,57 Mbit/s	0,30 Mbit/s
	5,0 s	TCP	8,31 Mbit/s	9,54 Mbit/s	3,10 Mbit/s
		UDP	9,56 Mbit/s	9,57 Mbit/s	0,31 Mbit/s

endet, sobald die vom Empfänger angezeigte maximale Übertragungsfenstergröße erreicht ist oder der Empfang eines Pakets nicht rechtzeitig quittiert wird („Timeout“). Ist letzteres der Fall, wird das Übertragungsfenster wieder auf den Initialwert zurück gesetzt und der „Slow-Start“-Algorithmus beginnt erneut.

Dieses Verhalten lässt sich in Abbildung 6.39 gut erkennen. Ab einer Position von ca. 0,37 m auf der x-Achse, sind die Übertragungsbedingungen stabil genug, so dass alle Pakete erfolgreich zugestellt werden können. Im Abdeckungsübergangsbereich des bei 0,5 m platzierten ersten von insgesamt zehn Infrastruktur-OAPs, treten erste Paketverluste auf (siehe $x = 1,1$ m), woraufhin der Anwendungsdurchsatz abrupt einbricht, da hier der „Slow-Start“-Algorithmus erneut gestartet wird. Wie auch im vorangegangenen Abschnitt 6.2.7 beschrieben, fällt infolgedessen der durchschnittliche Anwendungsdurchsatz bei Einsatz von TCP mit 7,64 Mbit/s um ca. 19,3 % geringer aus als beim Einsatz von UDP mit 9,47 Mbit/s (siehe Tabelle 6.6).

Es besteht die Möglichkeit, das Verhalten von TCP durch Parameter zu beeinflussen. Diese entstammen zum einem dem ursprünglichen RFC 793 [115], aber auch den darauf folgenden Erweiterung (z.B. RFC 2018 „TCP Selective Acknowledgment Options“ [121], RFC 6582 „The NewReno Modification to TCP’s Fast Recovery Algorithm“ [122] oder RFC 6691 „TCP Options and Maximum Segment Size (MSS)“ [123]).

Ein wichtiges Maß bzw. Eigenschaft einer Datenverbindung stellt das Verzögerungs-Bandbreiten-Produkt (englisch „bandwidth-delay product“, BDP) dar. Es symbolisiert die Menge an Daten, die notwendig ist, um bei einem Übertragungskanal mit Verzögerung den maximalen Durchsatz zu erzielen. Sendet ein Kommunikationsteilnehmer eine geringere Datenmenge, wird der Datendurchsatz durch die verzögerte, da paketorientierte Empfangsbestätigung limitiert (z.B. 1 ACK pro Paket) und die Verbindung erfährt Leerlaufzeiten. Bei Überschreiten der optimalen Datenmenge, muss ein Teil der Pakete im ausgangsseitigen Sendepuffer zwischengespeichert werden, da die Verbindung konstant ausgelastet ist. Die Menge an Daten, die bei einer TCP-basierten Übertragung maximal unbestätigt ausstehend sein darf ist durch das vom Empfänger angegebene Empfangsfenster bestimmt. Die TCP-Optionen erlauben es, die Größe dieses beworbenen Übertragungsfensters zu wählen, wodurch indirekt Einfluss auf die maximal erzielbaren Durchsatz genommen wird.

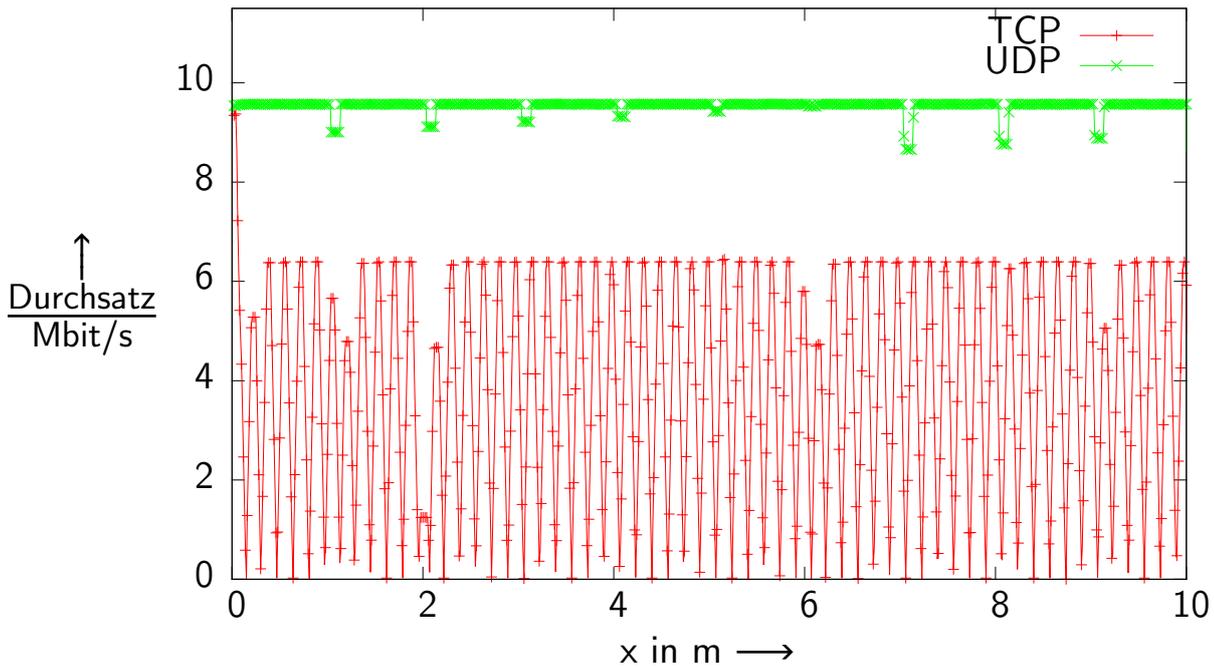


Abbildung 6.39: Vergleich des auf Anwendungsebene erzielten Durchsatzes in Abhängigkeit des verwendeten Transportprotokolls im Szenario „10 OAP“ mit einer Gültigkeitsdauer für ARP-Auflösungsergebnisse von 0,1 s

Der Zusammenhang zwischen dem maximal zu erwartenden Durchsatz auf Transportebene (OSI Layer 4) und der durch die Teilstrecken akkumulierten Paketumlaufzeit (RTT) sowie der Paketverlustrate p kann mit Hilfe der sog. Mathis-Gleichung [124] approximativ beschrieben werden:

$$\text{Durchsatz} \leq \frac{\text{MSS}}{\text{RTT}\sqrt{p}} \tag{6.1}$$

Padhye et al. stellten 1998 ebenfalls ein empirisch validiertes Modell zur Errechnung des Durchsatzes unter Einbeziehung der maximalen TCP-Fenstergröße und des initialen „Retransmit Timers“ vor [125]. Das Modell kann durch (6.2) approximiert werden, wobei die Abweichung bei der in [125] durchgeführten Validierung höchstens 0,5% betrug. Es gilt zu beachten, dass sich die Angabe des Durchsatzes auf die Transportebene bezieht (OSI Layer 4). Aus dieser kann mit Kenntnis der MTU sowie der Größe der durch IPv4 und TCP hinzugefügten Zusatzinformationen die pro Paket enthaltene Nutzdatenmenge errechnet werden. Die Standard-MTU bei Ethernet-Frames beträgt 1500 B, die Header von IPv4 und TCP sind zusammen mindestens 40 B groß, wodurch pro Ethernet-Frame maximal 1460 B an Nutzdaten versendet werden dürfen, um Fragmentierung durch das IP zu vermeiden. Die „Maximum Segment Size“ (MSS) müsste in diesem Fall auf 1460 B eingestellt werden.

$$\text{Durchsatz}(p) \approx \min \left(\frac{W_{max}}{\text{RTT}}, \frac{1}{\text{RTT}\sqrt{\frac{2bp}{3}} + T_O \cdot \min \left(1, 3\sqrt{\frac{3bp}{8}} \right) p(1 + 32p^2)} \right) \tag{6.2}$$

Hier bezeichnet p erneut die Paketverlustrate, b die Anzahl der von einem ACK bestätigten Pakete ($b = 1$, falls keine selektiven ACKs verwendet werden), W_{max} die maximale Größe des TCP-

Übertragungsfensters und T_O den initialen Wert des „Retransmit Timers“ (üblicherweise $T_O = 1$ s, kann adaptiert werden).

Um den erzielbaren Durchsatz gemäß (6.2) abschätzen zu können, müssen für RTT und p entweder Werte angenommen oder beispielsweise mit Hilfe einer Simulation bestimmt werden. Um die RTT so unverfälscht wie möglich zu bestimmen, sollte eine Messung im unbelasteten Zustand erfolgen. Hierfür bieten sich die zum Aufbau der TCP-Verbindung im Rahmen des sog. „Three-Way-Handshakes“ versendeten Pakete an, welcher in Abbildung 6.40 dargestellt ist. Zunächst sendet der TCP-Client ein sog. SYN-Paket zum TCP-Server. Dieser bestätigt den Erhalt mit Hilfe eines SYN+ACK-Pakets. Nachdem dieses beim Empfänger eingetroffen ist, bestätigt er seinerseits den Empfang mit Hilfe eines ACK-Pakets. Danach wird die Verbindung von beiden Seiten als hergestellt betrachtet. Die RTT im unbelasteten Zustand entspricht der Verzögerung vom initialen Senden des SYN-Pakets bis zum Eintreffen des SYN+ACK-Pakets.

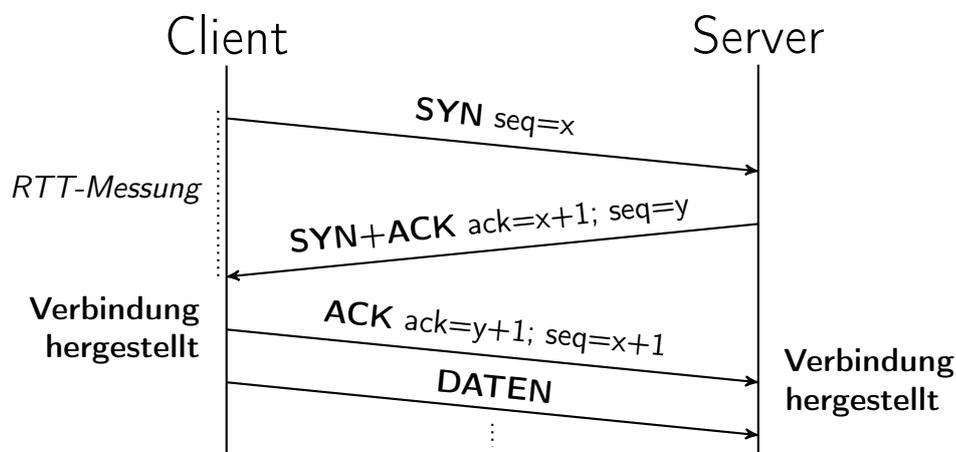


Abbildung 6.40: Im Rahmen des TCP-Verbindungsaufbaus übertragene Pakete und Messzeitraum zur Bestimmung der unverfälschten RTT

Der simulierte Verlauf von RTT für die in Tabelle 6.8 aufgelistete TCP-Konfiguration A ist in Abbildung 6.41 dargestellt. Die RTT im unbelasteten Zustand beträgt ca. 1,65 ms, wodurch sich bei einer Datenrate von 10 Mbit/s ein Verzögerungs-Bandbreiten-Produkt von aufgerundet 2062 B ergibt. Wird die Größe des TCP-Übertragungsfensters größer gewählt, stellt sich nach einem Einschwingvorgang ein konstant erhöhter RTT-Wert ein, der sich aus dem Füllstand der innerhalb der Datenverbindungsstrecke existenten Puffer ergibt (siehe Abbildung 6.42). Die Warteschlangen ermöglichen das nahezu vollständige Ausnutzen der zur Verfügung stehenden Bandbreite. Der sich aus den Puffern ergebende Nachteil ist jedoch die Vergrößerung von Latenz bzw. RTT. Beim Auftreten von Überlastspitzen oder dauerhaften Überlastsituationen (die Bandbreite der Verbindung wird erschöpft) wächst der Füllstand in den Warteschlangen. Die sich vergrößernde RTT bewirkt ein verzögertes Eintreffen von ACKs und somit einen verzögerten Versand von neuen TCP-Segmenten. Insgesamt ergibt sich dadurch im eingeschwungenen Zustand ein konstanter Füllstand, dessen Wert sich aus der Differenz zwischen Verzögerungs-Bandbreiten-Produkt und dem TCP-Übertragungsfenster ergibt. Wählt man die Größe des TCP-Übertragungsfensters hingegen zu klein, d.h. im schlechtesten Fall so groß wie die MSS, ergibt sich ein signifikant anderer Verlauf. Da die Verbindungskapazität nun nicht mehr ausgeschöpft wird, bleibt die RTT konstant bei einem sehr niedrigen Wert, welcher dem der minimalen RTT beim Verbindungsaufbau entspricht. Zusätzliche Pufferzeiten treten in diesem Fall nicht auf. Das beschriebene Verhalten stellt eine Fehlkonfiguration dar, die sich allerdings ebenfalls mit Hilfe des Simulationsmodells darstellen bzw. untersuchen lässt.

Tabelle 6.7: Gegenüberstellung RTT-basierter Abschätzungen mit simulationsbasierten Ergebnissen für die mittels TCP durchschnittlich erzielte Durchsatzrate im Szenario „10 OAP“ ($T_O = 1$, $b = 1$)

RTT (eingeschwungen)	Paketfehler- wahrscheinlichkeit p	TCP- Fenstergröße	Mathis- Gleichung (6.1)	erweitertes Modell (6.2)	Simulation
9,36 ms	$4,10 \times 10^{-4}$	65535	108,29 Mbit/s	0,50 Mbit/s	84,25 Mbit/s

Tabelle 6.8: Standardparameter für die TCP-assoziierten Untersuchungen

Parameter	Wert
TCP-Variante	TCPreno
TCP-Fenstergröße	65 535 B
Maximum Segment Size (MSS)	1460 B
Nagle-Algorithmus (RFC 896 [126])	nein
verzögerte ACKs (RFC 1122 [127])	nein
begrenzte Übertragung (RFC 3042) [128]	nein
vergrößertes Startfenster (RFC 3390) [129]	nein
selektive ACKs (RFCs 2018 [121], 2883 [130] & 3517 [131])	nein
Fensterskalierung (RFC 1323) [132]	nein
Zeitstempel (RFC 1323)	nein

Die Paketverlustrate kann experimentell bestimmt oder angenommen werden. Wird der erzielbare Durchsatz in Abhängigkeit von der Paketfehlerrate bestimmt, zeigt sich, dass das von Mathis et al. beschriebene Modell (6.1) für sehr kleine Paketfehlerraten nur eingeschränkt nutzbar ist, da es eine viel zu optimistische obere Grenze angibt. Das von Padhye et al. beschriebene erweiterte Modell (6.2) trifft eine Unterscheidung zwischen paketfehlerdominierter und bandbreiten- bzw. laufzeitdominierter Datenverbindung und ist zur Vorhersage des maximal zu erwartenden Durchsatzes theoretisch deutlich besser geeignet. Ein numerischer Vergleich für verschiedene Konstellationen von RTT, Paketfehlerwahrscheinlichkeit p und vorgegebenem TCP-Übertragungsfenster ist in Tabelle 6.7 gegeben. Dabei wurde die Paketverlustrate innerhalb der Datenverbindung auf Basis des Quotienten der vom Infrastruktur-Host ausgehenden und beim mobilen Host eingehenden Pakete ermittelt. Der Vergleich zeigt, dass die mittels 6.1 ermittelten Werte den tatsächlichen Durchsatz um 28,54 % überschätzen, während das in 6.2 ausgedrückte Modell in diesem Anwendungsszenario um den Faktor 168,5 unterschätzt. Beide Modelle sind also nur eingeschränkt bzw. gänzlich ungeeignet, um in den zu Grunde liegenden – hochgradig dynamischen – Szenarien den beim Einsatz von TCP erzielbaren Durchsatz sinnvoll abschätzen zu können. Weiterhin unterstreicht dieser Sachverhalt, dass ein geeigneter Mechanismus für einen koordinierten Zellenwechsel („Handover“) dringend notwendig ist, um trotz Fortbewegung innerhalb des Gebäudes eine gleichbleibend gute „Quality of Service“ sicherzustellen.

Tabelle 6.8 verdeutlicht die Anzahl der verfügbaren Konfigurationsparameter mit deren Hilfe sich das Verhalten von TCP beeinflussen lässt. Die einzelnen Implementierungsvarianten unterscheiden sich im Wesentlichen durch die Reaktion auf das Auftreten von sog. „duplicate ACKs“. Diese werden immer dann versendet, wenn innerhalb einer Sequenz von TCP-Paketen einzelne Pakete den Empfänger nicht erreichen. Der Empfänger quittiert zwar den Empfang der weiteren Pakete, behält allerdings die Sequenznummer innerhalb des ACKs bei, so dass der Empfang des letzten Pakets vor Auftreten des

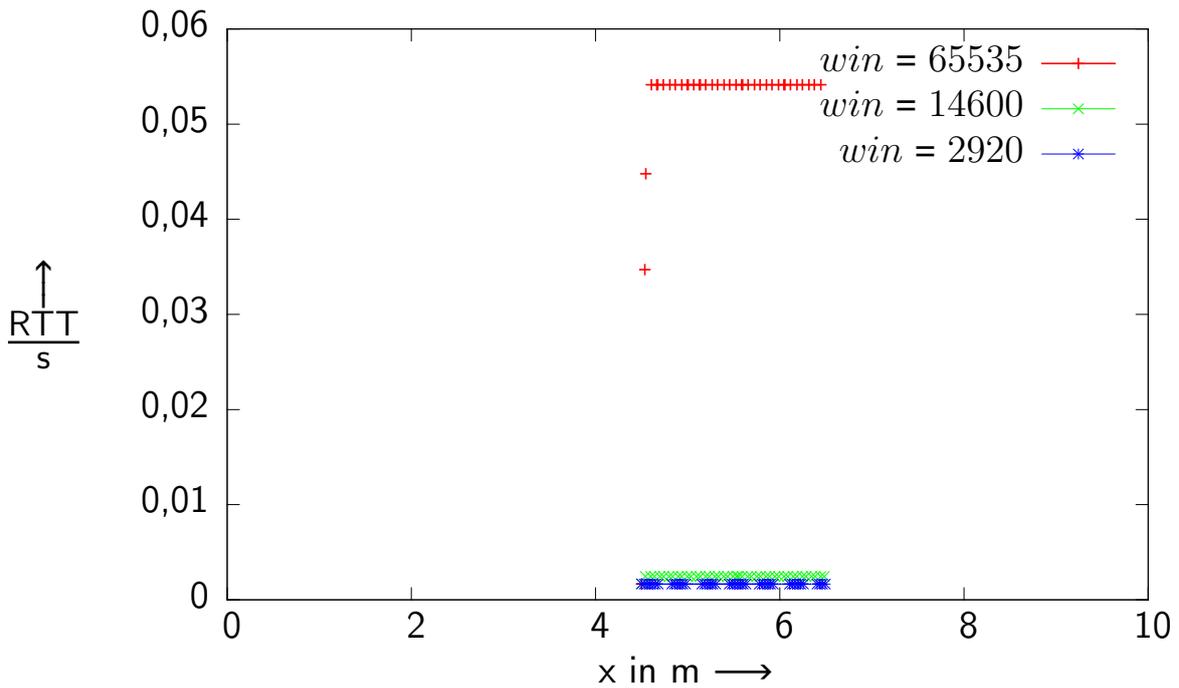


Abbildung 6.41: Resultierende Paketumlaufzeit RTT für das Simulationsszenario „1 OAP“ und TCP-Konfiguration A mit TCP-Fenstergrößen von $win = 65535$ bis $win = 1460$

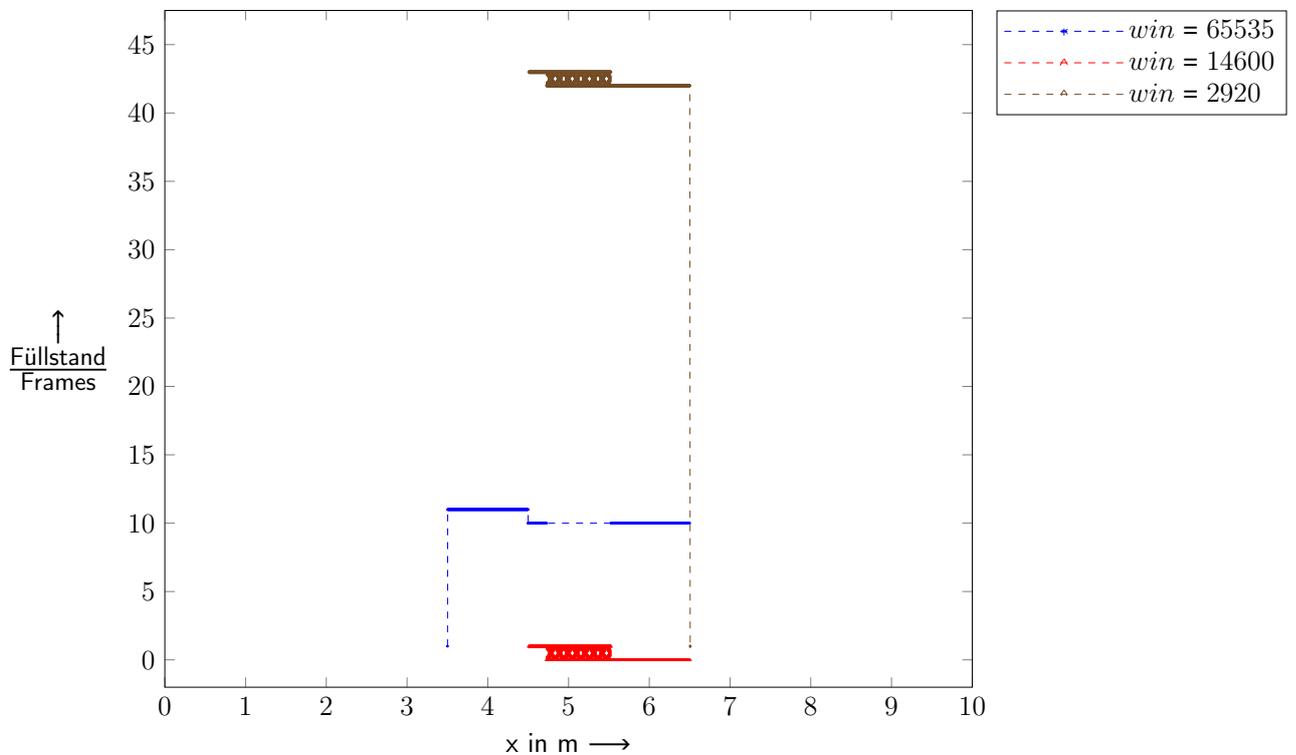


Abbildung 6.42: Resultierender Füllstand des Puffers innerhalb des optischen Übertragungsmoduls für TCP-Übertragungsfenstergrößen von $win = 1460$ bis $win = 65535$

Verlusts mehrfach quittiert wird. Der Sender zählt die Anzahl der mit identischer Sequenznummer eingehenden ACKs und führt bei Erreichen eines Schwellwerts von üblicherweise drei ein „Fast Retransmit“ des bzw. der fehlenden Pakete durch. Dieser Mechanismus hilft das Überschreiten der Zeitschranke für die Bestätigung mittels ACK zu vermeiden, was sonst unmittelbar zu Reduzierung des erzielbaren Durchsatzes führen würde.

Je nach Wahl der Werte für die Konfigurationsparameter, stellt sich ein unterschiedliches Protokoll- respektive Systemverhalten ein, was zu unterschiedlichem Durchsatz führen kann. Mit Hilfe des vorgestellten Simulationsmodells ist es möglich, für ein Simulationsszenario alle gültigen Kombinationen von TCP-Parameterwerten innerhalb von Minuten zu vergleichen (s. bTabelle 6.11), das heißt den Entwurfsraum vollständig abzudecken. Ein Vergleich des durchschnittlichen Durchsatzes der sich ergebenden 256 Simulationen durchläufe ist in Abbildung 6.43 dargestellt. Die Werte schwanken zwischen 0,56 Mbit/s und 9,24 Mbit/s was den signifikanten Einfluss – und somit die Wichtigkeit – der Wahl geeigneter Parameterwerte offenbart.

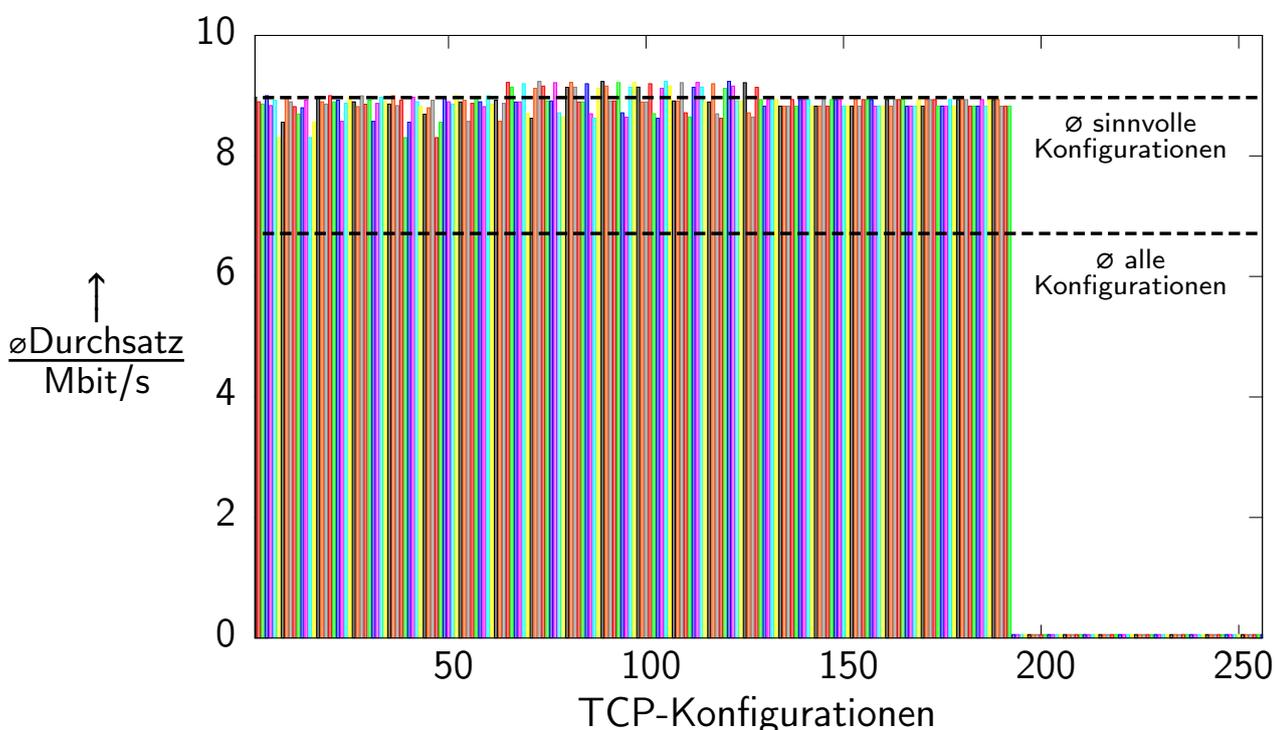


Abbildung 6.43: Übersicht über den auf Anwendungsebene durchschnittlich erzielten Durchsatz für alle 256 gültigen Parameterkonfigurationen und das Simulationsszenario „10 OAP“, die mittels paralleler Simulationsläufe innerhalb von wenigen Minuten erzeugt werden kann.

Die Parameterkombination, die im Rahmen des Simulationsszenarios zum größtmöglichen durchschnittlichen Durchsatz führt, lässt sich durch eine einfache Suche des Maximums bestimmen. Es bleibt allerdings die Frage, welche Parameter den größten Einfluss auf das Ergebnis haben. Um dieser Frage nachzugehen, wurden alle 256 Simulationsläufe statistisch ausgewertet. Dabei sollen die, sich aus den in Tabelle 6.8, insgesamt neun unabhängigen Variablen, davon sieben dichotome, eine mehrstufige und eine quantitative, die abhängige quantitative Variable „durchschnittlicher Durchsatz“ erklären. Der Wert für die MSS wurde unverändert beibehalten.

Zunächst kann mit Hilfe der in Tabelle 6.9 dargestellten Korrelationsmatrix untersucht werden, welche Parameter den durchschnittlichen Durchsatz positiv, negativ oder nur sehr geringfügig beeinflussen.

Tabelle 6.9: Korrelationsmatrix für die einzelnen TCP-Konfigurationsparameter im Bezug auf den durchschnittlichen Durchsatz auf Anwendungsebene

	Fenstergröße	verz. ACKs	vergröß. Startfenster	Nagle	begrenzte Übertragung	sel. ACKs	TCP-Reno	TCP-NewReno	TCP-Tahoe	Fenster-skalierung	Zeitstempel	∅ Durchsatz
Fenstergröße	1	0	0	0	0	0	0	0	0	–	0	0,579
verz. ACKs	0	1	0	0	0	0	0	0	0	–	0	-0,565
vergröß. Startfenster	0	0	1	0	0	0	0	0	0	–	0	0,000
Nagle-Algorithmus	0	0	0	1	0	0	0	0	0	–	0	0
begr. Übertragung	0	0	0	0	1	0	0	0	0	–	0	0,003
sel. ACKs	0	0	0	0	0	1	0,578	-0,333	-0,333	–	0	0,012
TCPReno	0	0	0	0	0	0,578	1	0,578	0,578	–	0	0,007
TCPNewReno	0	0	0	0	0	-0,333	-0,578	1	-0,333	–	0	-0,006
TCP Tahoe	0	0	0	0	0	-0,333	-0,578	-0,333	1	–	0	-0,013
Fenster-skalierung	–	–	–	–	–	–	–	–	–	1	–	–
Zeitstempel	0	0	0	0	0	0	0	0	0	0	1	-0,013

Es fällt auf, dass die Parameter Fenstergröße, begrenzte Übertragung und selektive ACKs den durchschnittlichen Durchsatz positiv beeinflussen. Negativ wirken sich hingegen die Verwendung verzögerter ACKs sowie der Einsatz der TCP-Varianten „TCPNewReno“ bzw. „TCPTahoe“ aus. Besonders auffällig ist der extrem geringe Durchsatz für die Konfigurationen 193 bis 256 (siehe Abbildung 6.43) als Ergebnis der Verwendung eines Übertragungsfensters mit der minimalen Größe einer MSS, also 1460 B, in Kombination mit verzögerten ACKs (RFC 1122 [127]). Da hier nur ein einziges Segment ausstehend sein darf und gleichzeitig die Bestätigungen um maximal bis zu 0,5s verzögert werden können, um sowohl die Verbindung als auch beide Verbindungspartner zu entlasten, kommt es häufig zu deutlichen RTT-Vergrößerungen sowie Zeitüberschreitungen und somit Übertragungswiederholungen. Es handelt sich in diesem Fall um eine zu vermeidende Fehlkonfiguration, die jedoch mit Hilfe des Simulationsmodells aufgezeigt werden kann.

Um die Vermutung zu bestätigen, dass bereits eine geeignete Wahl der zwei Parameter TCP-Fenstergröße sowie verzögerte ACKs in Verbindung mit der Implementierungsvariante „TCP Reno“ zu einem signifikant besseren durchschnittlichen Durchsatz führt, wurde die entsprechende Teilmenge $S_{Optimum}$ aus der Grundgesamtheit der 256 Ergebnisse selektiert und analysiert. Es zeigt sich, dass der durchschnittliche Durchsatz im Mittel 9,06 Mbit/s beträgt und somit 1,91 % über dem gemitteltem Durchsatz von 8,89 Mbit/s für die Teilmenge der 192 sinnvollen Konfigurationen $S_{Sinnvoll}$ liegt (siehe Tabelle 6.10). Weiterhin steigt der minimal erzielte Durchsatz um 7,01 % von 8,31 Mbit/s auf 8,90 Mbit/s.

Die durch die Wahl geeigneter TCP-Parameter erzielten Verbesserungen im Bezug auf den Anwendungsdurchsatz fallen im untersuchten Szenario relativ gering aus. Die Untersuchung zeigt allerdings, dass eine Verbesserung auf Basis der Entwurfsraumexploration mit geringem Aufwand möglich ist. Das Verbesserungspotential in anderen Szenarien kann mit Hilfe des Simulationsmodells ebenfalls leicht bestimmt werden.

6.2.9 Simulationsperformance

Wie bereits in Kapitel 6.1.6 erläutert, bestimmt die Simulationsperformance den Aufwand für die Entwurfsraumexploration. Das erstellte Simulationsmodell soll bei Wahrung der Exaktheit in möglichst kurzer Zeit Ergebnisse generieren, um so Parameterstudien oder vollständige Suchen innerhalb des Entwurfsraums zu ermöglichen. Die zur vollständigen Durchführung der Simulation nötige Zeit richtet sich dabei nach der Anzahl von Ereignissen, die simuliert werden müssen. Diese steigt mit der Menge der übertragenen Daten bzw. der Datenrate und der Anzahl der Teilnehmer. Weiterhin sind Netzwerkeffekte, die beispielsweise zur Vervielfältigung von Nachrichten führen, zu berücksichtigen.

Tabelle 6.10: Vergleich der im Szenario „10 OAP“ auf Anwendungsebene erzielten durchschnittlichen Durchsatzraten für eine Teilmenge $S_{Optimum}$ mit einer Übertragungsfenstergröße von 65535, selektiven ACKs und der TCP-Variante „TCPReno“ mit der Grundgesamtheit aller Simulationsergebnisse S_{Gesamt} sowie der Grundgesamtheit abzüglich der Fehlkonfigurationen 192 - 256 $S_{Sinnvoll}$

Konfiguration	Durchsatz				
	Minimum	Mittelwert	Maximum	Median	Standardabweichung
S_{Gesamt}	0,06 Mbit/s	6,69 Mbit/s	9,24 Mbit/s	8,83 Mbit/s	3,84 Mbit/s
$S_{Sinnvoll}$	8,31 Mbit/s	8,89 Mbit/s	9,24 Mbit/s	8,90 Mbit/s	0,18 Mbit/s
$S_{Optimum}$	8,90 Mbit/s	9,06 Mbit/s	9,24 Mbit/s	9,06 Mbit/s	0,13 Mbit/s

Zu beachten ist hierbei, dass die einzelnen Simulationsdurchläufe als eigener Prozess gestartet werden und mit nur einem Thread arbeiten.

Das Szenario mit der geringsten Komplexität und somit auch Simulationszeit ist das Szenario „1 OAP“ in Verbindung mit einer 2-PAM ohne Vorwärtsfehlerkorrektur und UDP als Transportprotokoll. Auf einem unbelasteten Server mit einer CPU vom Typ Intel Xeon E5-2683 v4 und 32 physisch vorhandenen Kernen sowie 256 GB Hauptspeicher ist der Simulationsdurchlauf nach durchschnittlich 38s CPU-Zeit abgeschlossen. Die durchschnittlichen Ausführungszeiten weiterer Durchläufe in Abhängigkeit der gewählten Simulationsparameter sind in Tabelle 6.11 aufgeführt.

In Szenarien mit einer größeren Anzahl von Infrastruktur-OAPs steigt das Nachrichtenaufkommen durch den insgesamt größeren Abdeckungs- bzw. Kommunikationsbereich. Im Szenario „10 OAP“ mit FEC via RS(64,32) und ansonsten unveränderten Simulationsparametern benötigt der Simulationsdurchlauf deshalb 317s. Weiterhin bewirkt eine Erhöhung des Modulationsgrads von $M = 2$ auf $M = 1024$ einen Anstieg der Anzahl versendeter Pakete um den Faktor 5,55. Da jede dieser Nachrichten mindestens ein zu simulierendes Ereignis darstellt, steigen deren Anzahl sowie die Ausführungszeit um einen Faktor von 5,93 bzw. 5,97 (siehe Tabelle 6.11).

Wird statt eines Switchs ein Hub zur Verbindung der Infrastruktur-OAPs eingesetzt, verlängern sich die Ausführungszeiten von ca. 32 Minuten auf fast sieben Stunden (Faktor ≈ 13), wobei die Anzahl zu simulierender Ereignisse auf das 6,6-Fache ansteigt.

Dieses Verhalten offenbart den Nachteil der diskreten und ereignisbasierten Simulationsmethode im Vergleich zu stochastischen oder analytischen Methoden, wo wie im Abschnitt 3.2.1 beschrieben, Parameteränderungen nur numerische Veränderungen bedeuten, aber keine oder nur sehr geringe Auswirkung auf die Ausführungszeit haben.

6.2.10 Diskussion der Ergebnisse

Das hier dargestellte Anwendungsbeispiel eignet sich, um zum einen die Auswirkungen der für die optische Übertragung spezifischen Übertragungsparameter auf bspw. den erzielbaren Durchsatz zu untersuchen, aber auch, um den Einfluss der eingesetzten Protokolle sowie deren Parametrierung aufzuzeigen. Der Vergleich zwischen experimentell und simulativ ermittelten Durchsatzraten (6.2.3) offenbart eine gute Übereinstimmung im Bezug auf die räumliche Abbildung des Übertragungsverhaltens für die geometrisch ermittelten Übertragungseigenschaften. Die Abweichung von 14,53% des maximal

Tabelle 6.11: Vergleich der über drei Durchgänge gemittelten durchschnittlichen CPU-Zeiten für Simulationsläufe in Abhängigkeit der konfigurierten Simulationsparameter bei Ausführung auf einem Server mit Intel Xeon E5-2683v4 Prozessor und 256 GB Arbeitsspeicher

Topologie	Anzahl OAPs	Modulation	Gültigkeitsdauer ARP-Ergebnisse	Transportprotokoll	Datenaufzeichnung	Anzahl Nachrichten	Anzahl Ereignisse	Ø CPU-Zeit
Switch	10	2-PAM	0,1 s	TCP	nein	9017665	10005638	317 s
Switch	10	2-PAM	0,1 s	TCP	ja	9017665	10005638	339 s
Switch	10	2-PAM	0,1 s	UDP	nein	4258335	5143310	144 s
Switch	10	2-PAM	0,1 s	UDP	ja	4258335	5143310	151 s
Switch	10	2-PAM	5,0 s	TCP	nein	5413160	6375130	183 s
Switch	10	2-PAM	5,0 s	TCP	ja	5413160	6375130	199 s
Switch	10	2-PAM	5,0 s	UDP	nein	3297200	4022045	106 s
Switch	10	2-PAM	5,0 s	UDP	ja	3297200	4022045	123 s
Switch	10	1024-PAM	0,1 s	TCP	nein	50084473	59305703	1716 s
Switch	10	1024-PAM	0,1 s	TCP	ja	50084473	59305703	1894 s
Switch	10	1024-PAM	0,1 s	UDP	nein	35543988	44435684	1170 s
Switch	10	1024-PAM	0,1 s	UDP	ja	35543988	44435684	1297 s
Switch	10	1024-PAM	5,0 s	TCP	nein	63872989	75768629	2258 s
Switch	10	1024-PAM	5,0 s	TCP	ja	63872989	75768629	2380 s
Switch	10	1024-PAM	5,0 s	UDP	nein	33130396	40552356	1076 s
Switch	10	1024-PAM	5,0 s	UDP	ja	33130396	40552356	1165 s
Hub	10	1024-PAM	0,1 s	TCP	ja	496272054	392652218	24 531 s
Hub	10	1024-PAM	0,1 s	UDP	ja	268861428	214336645	10 605 s

erzielten Durchsatzes lässt sich indes leider nicht abschließend klären. Da aber Übertragungsfehler durch die Analyse von Mitschnitten der gesamten Verbindungen ausgeschlossen werden konnten, ist die Ursache im Bereich der Interaktion zwischen Netzwerkschnittstellentreiber und Betriebssystem zu vermuten. Diese lässt sich jedoch nicht ohne erheblichen Aufwand analysieren. Leider besteht aus rechtlichen Gründen auch nicht die Möglichkeit die interne Funktionsweise des zur Verfügung stehenden Experimentalsystems näher zu untersuchen bzw. im Rahmen dieser Arbeit darzustellen, weshalb dieses als sog. „Black Box“ behandelt werden musste. Weiterhin beschränken sich die Vergleichsmöglichkeiten im Rahmen der experimentellen Validierung, bedingt durch die geringe Verfügbarkeit beschaffbarer OWC-Systeme, mit dem „Li-Fi Hotspot“ des Fraunhofer IPMS auf lediglich eine Referenz.

Aufgrund der konsequenten Modularisierung des Simulationsmodells sind vergleichende Simulationen sehr leicht umsetzbar. So ließ sich beispielsweise die Auswirkung des Einsatzes verschiedener Modulationsverfahren auf das Verhalten des Gesamtsystems untersuchen. Hier zeigte sich, dass die Wahl des Modulationsgrads nicht nur einen Einfluss auf den maximal erzielbaren Durchsatz hat, sondern auch erheblich den Abdeckungsbereich bestimmt. Gleichzeitig konnte in Abschnitt 6.2.5 gezeigt werden, dass in bestimmten Konstellationen eine Steigerung des Modulationsgrads und somit des Durchsatzes nicht unbedingt mit einer Reduktion der Abdeckung einhergeht, da die SNR noch ausreichend hoch ist. Weiterhin konnte in Abschnitt 6.2.5 der Einfluss des gewählten FEC-Verfahrens auf den Abdeckungsbereich gezeigt werden. Bei der Verwendung von RS-Codes, in Verbindung mit einer 1024-PAM modulierten Datenübertragung, ergaben sich Unterschiede der abgedeckten Flächen von bis zu 62,3 %. Das Simulationsmodell, respektive der simulative Ansatz, sind an dieser Stelle ideal geeignet, um bei der Auslegung eines optischen Übertragungssystems dessen Leistungsfähigkeit mit der aktuellen Parametrierung im Betrieb abzuschätzen.

Ein signifikanter Anteil des dynamischen Systemverhaltens wird durch die zum Einsatz kommenden Protokolle in den OSI-Schichten 2 bis 7 bestimmt. Hier wurden vor allen Dingen die Auswirkungen von Parameteränderungen in der Transportschicht (OSI-Schicht 4) untersucht. Die diskrete eventbasierte Simulation erlaubt es, dynamische Vorgänge darzustellen und nachzuvollziehen. Dabei zeigte sich, dass

im Fall des Einsatzes mehrerer optischer Femtozellen eine Verbindungsübergabe („Handover“) analog zum Mobilfunknetz zum Einsatz kommen sollte oder alternativ die Routenaktualisierung durch sehr kleine Gültigkeitsdauern für ARP-Antworten provoziert werden muss. Dieses Vorgehen sorgt jedoch, wie in 6.38a dargestellt, für Durchsatzeinbrüche bei Datenübertragungen mittels TCP, da hier der Paketverlust detektiert wird und das „Slow Start“-Verfahren wiederholt anläuft.

Ebenso ist erkennbar, dass sich die in den TCP-RFCs festgelegten Zeitkonstanten an den Anforderungen des Internets orientieren und zur Reaktion auf transiente Veränderungen auf optischen oder drahtlosen Übertragungstrecken in kleinen Netzwerken zu groß bemessen sind. Beispielsweise ist das in RFC 6298 [133] definierte „Retransmission Timeout“ mindestens 1 s lang, d.h. der Verlust von Segmenten bzw. Empfangsbestätigungen sorgt erst nach dieser Zeit für eine Neuübertragung, sollte der Mechanismus der „Fast Retransmissions“ nicht genutzt werden. Dadurch können auf Anwendungsebene deutlich spürbare Verzögerungen entstehen, die durch große Datenpuffer ausgeglichen werden müssten.

6.3 Auswertung der Optimierungsergebnisse der Anwendungsbeispiele

Wie in den verschiedenen Aspekten der Anwendungsbeispiele gezeigt wurde, ist die Entwurfsraumexploration mit Hilfe der erstellten diskreten Netzwerksimulationsmodelle in kurzer Zeit möglich. Dadurch können geeignete Werte für die Auslegungs- oder Betriebsparameter von funkbasierten oder optisch arbeitenden Kommunikationssystemen auf Basis von Parameterstudien bzw. schichtübergreifenden Analysen bestimmt werden. Weiterhin ist es aber auch möglich, das Systemverhalten insgesamt statistisch zu analysieren, um die Bedeutsamkeit der einzelnen Parameter zu bestimmen. Ein Ansatz hierfür ist die sukzessive Untersuchung des Bestimmtheitsmaßes R^2 von unabhängigen Variablen (Systemparametern), die das Systemverhalten (abhängige Variable) im Rahmen eines linearen Regressionsmodells erklären sollen. Dabei wird jeweils für jede unabhängige Variable, aber auch für jede Variablenkombination, überprüft inwieweit die Variation bzw. Varianz der abhängigen Variablen durch sie erklärt werden kann. Diese Analyse wurde exemplarisch für die gegen Ende des Abschnitts 6.2.8 dargestellte Untersuchung von, zur Steigerung des durchschnittlichen Anwendungsdurchsatzes, geeigneten Parameterwerten für TCP durchgeführt. Hierbei kam das für die Programmiersprache R verfügbare Paket „hier.part“ [134] zum Einsatz, welches automatisiert die Anpassungsgüte für die aus N unabhängigen Variablen entstehenden Modellhierarchien bestimmt. Aus der Differenz der Ergebnisse für verschiedene Partitionierungen lassen sich die individuellen Beiträge zum Bestimmtheitsmaß der einzelnen unabhängigen Variablen errechnen, welche in Abbildung 6.44 dargestellt sind.

Es zeigt sich, dass die Variablen „Fenstergröße“ und „verzögerte ACKs“ einen wesentlichen Erklärungsanteil im Rahmen der linearen Regression aufweisen. Dabei muss allerdings angemerkt werden, dass insbesondere im Bereich der Netzwerksimulation nichtlineare Zusammenhänge existieren können, die durch ein lineares Modell nur schlecht nachgebildet werden können. In diesem Fall sollten nichtlineare Regressionsmodelle zum Einsatz kommen, deren Bestimmtheitsmaß R^2 deutlich von Null verschiedene Werte aufweisen wird.

Im Vergleich mit alternativen Simulationsansätzen erlaubt es, die diskrete Simulation das dynamische Verhalten und somit auch dynamische Effekte aufzuzeigen. Als Beispiele lassen sich hier die Routingvorgänge aus dem Anwendungsbeispiel der elektronischen Schließenanlage nennen. Beispielsweise profitiert das System bei Nutzung vom simplen Fluten von einer endständigen Position des Access Points, da die Ausbreitung dann in Form einer Wellenfront erfolgt. Bei einer flurmittigen Positionierung kollidieren hier Anfragen aus mehreren Raumrichtungen, so dass Übertragungsfehler resultieren. Weiterhin lassen

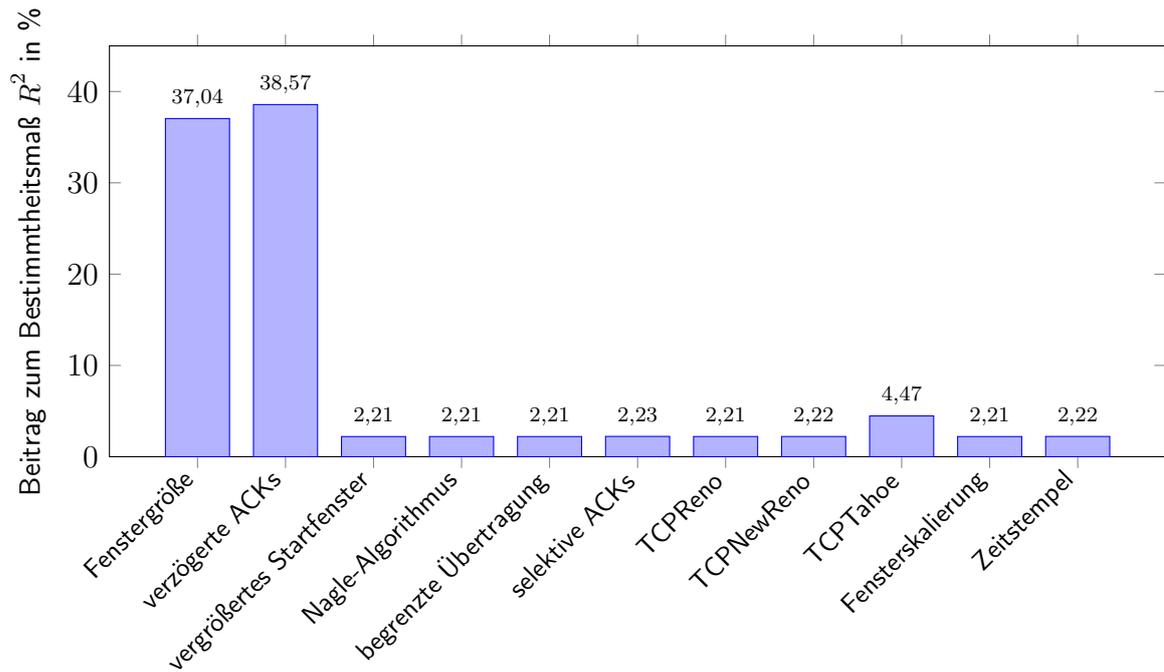


Abbildung 6.44: Individuelle Beiträge der unabhängigen Variablen (TCP-Parameter) zum Bestimmtheitsmaß R^2 für die Modellierung der zu erklärenden Variablen „durchschnittlicher Durchsatz“ mittels linearer Regression, d.h. die Stärke des Einflusses einzelner Parameter auf den Durchsatz

sich die Überlastbedingungen sehr präzise evaluieren, da die konkret auftretenden Paketwiederholungen zusätzlich zur, durch das gemeinsame Aufwachintervall zeitlich verdichteten, Grundlast auftreten und ebenso abgearbeitet werden müssen. Mittels Bedientheorie, Petrinetzen oder „Network Calculus“ lassen sich nur Verteilungen analysieren, so dass das dynamische auftretende Übertragungsvolumen zusätzlich modelliert werden müsste.

Ähnlich stellt sich die Situation bei der Durchsatzanalyse im Anwendungsbeispiel des optischen Kommunikationssystems dar. Hier führen die Übertragungsbedingung beim Zellenwechsel sowie die anschließenden Änderungen der zur Übertragung notwendigen Route zu Reaktionen innerhalb der verwendeten Transportschicht, in diesem Beispiel TCP bzw. UDP. Infolgedessen zeigen sich irreguläre Muster im ortsbezogenen Verlauf des anwendungsbezogenen Datendurchsatzes.

Zusammenfassung

Der Trend einer fortschreitenden Digitalisierung und Vernetzung spiegelt sich auch im Bereich der Gebäude-Infrastruktur wider. Hier versprechen vernetzte Systeme deutliche Energieeinsparungen, eine Erhöhung der Gebäudesicherheit und auch einen maßgeblichen Komfortgewinn durch Automatisierung. Die hieraus erwachsenden Anforderungen an Durchsatzrate bzw. verfügbare Bandbreite sind gering. Im Fokus steht stattdessen die Möglichkeit, viele Geräte oder Komponenten im gleichen limitierten Spektrum kommunizieren zu lassen, die effektive Verlustleistungsaufnahme der Endgeräte sowie bei latenzkritischen Anwendungen die Einhaltung maximaler Schranken für die Verzögerung. Aus Kostengründen und aus Gründen der Nachrüstbarkeit bei Altbeständen sind hier drahtlose Lösungen klar zu präferieren. Ebenso gehört auch die Bereitstellung von drahtlosen Kommunikationsnetzwerken für mobile Nutzer zur Standardaufgabe im Gebäudekontext (z.B. WLAN nach IEEE 802.11). Hieraus resultieren sehr hohe Anforderungen an die zur Verfügung stehende Bandbreite, während die energetischen Anforderungen für verbundene mobile Geräte wichtig, für die Infrastruktur jedoch zumeist nachrangig sind. Vor dem Hintergrund dieser gegensätzlichen Auslegungsziele sowie des vieldimensionalen Entwurfsraums, stellt der anwendungsbezogene Entwurf von drahtlosen Kommunikationssystemen eine besondere Herausforderung dar.

Im Kapitel 3 dieser Arbeit, wurden deshalb zunächst Möglichkeiten der modellbasierten Entwurfsraumexploration untersucht und hinsichtlich ihrer Präzision, des entstehenden Modellierungsaufwands und der Qualität der Ergebnisse bewertet. Dabei zeigte sich, dass analytische Modelle die komplexen Vorgänge innerhalb eines Netzwerks nur approximieren oder über Zufallsprozesse abbilden können. Das Verfahren der diskreten, ereignisbasierten Simulation ermöglicht die Abbildung mit der größten Genauigkeit, was jedoch mit einem vergleichsweise hohen Aufwand für die Erstellung des Simulationsmodells sowie dessen Verifikation einhergeht. Weiterhin ist die benötigte Simulationszeit abhängig vom Ereignisaufkommen, das heißt der Anzahl der Teilnehmer sowie der Modellierungsgranularität. Da jedoch eine schichtübergreifende Betrachtung aller Vorgänge innerhalb des Netzwerks sowie seiner Teilnehmer erfolgt und diese aufgrund der zugrunde liegenden Zeitbasis direkt mit Abläufen in real existierenden Netzwerken vergleichbar sind, kann der Aufwand als vertretbar angesehen werden.

Ein Vergleich möglicher Werkzeuge zur integrierten Durchführung diskreter Netzwerksimulationen ist in Kapitel 4 gegeben. Unter Berücksichtigung der Kriterien Quelloffenheit, kostenlose Verfügbarkeit sowie akzeptable Simulationsleistung reduzierte sich die Menge der in Frage kommenden Kandidaten

auf ns-2, ns-3 sowie OMNeT++. Während ns-2, der innerhalb des akademischen Umfelds immer noch den De-facto-Standard darstellt, aufgrund veralteter Simulator- und Modellbasis allerdings als überholt zu bezeichnen ist, war ns-3 zum Zeitpunkt der Erstellung dieser Arbeit noch nicht ausreichend ausgereift. Beiden gemein war und ist der Mangel einer integrierten, die verschiedenen Arbeitsabläufe unterstützenden, Entwicklungsumgebung. Diese ist jedoch fester Bestandteil bei OMNeT++, weshalb dieses Werkzeug zum Einsatz kam. Weiterhin ist der modulare Entwurf hier ein Kerngedanke, wodurch das Ziel einer weitestgehenden Modularisierung bei der Erstellung umfangreicher diskreter Simulationsmodellen erfüllt werden konnte.

Wie in Kapitel 5 beschrieben, muss die Umsetzung dahingehend erfolgen, dass beispielsweise in Standards wie IEEE 802.15.4 aufgeführte Methoden, Abläufe oder Datenstrukturen berücksichtigt werden. Existiert eine solche Vorgabe nicht, erfolgt eine inhaltliche Gliederung nach Funktion bzw. vor dem Hintergrund der Austauschbarkeit dieser. Durch diese Maßnahmen kann die Verifikation sehr gut mittels Modul- und Integrationstests erfolgen, wie sie auch im Rahmen des angewendeten V-Modells vorgegeben sind. Die letzte Stufe stellte hier die funktionale Verifikation eines Simulationsszenarios, d.h. der Vergleich des simulierten mit dem zu erwartenden Verhalten, dar. Parallel dazu musste das entstandene Simulationsprogramm auf statische und Laufzeitfehler überprüft werden, da diese eine Ausführung unmöglich machen, zum vorzeitigen Simulationsabbruch oder zu funktionalem Fehlverhalten führen können. Hier stellte sich heraus, dass insbesondere die Integrationstests sowie das abschnittsweise Testen im Rahmen des Simulationsszenarios sehr zeitaufwändig waren. Das Debuggen innerhalb der Benutzeroberfläche von OMNeT++ erlaubt es zwar die Ausführung des Programms mit Hilfe so genannter *Breakpoints* an beliebigen Stellen im Quellcode zu unterbrechen, jedoch muss dazu das verursachende Ereignis im Simulationsverlauf zunächst bestimmt werden. Dabei ist es jedoch nicht möglich, bereits zu vergangenen Zeitpunkten/Ereignissen zurück zu kehren, so dass der Simulationslauf hierfür erneut gestartet werden muss. Insbesondere im Fall von Abweichungen zum erwarteten Verhalten innerhalb eines Simulationsszenarios, wird die Rückverfolgung der Ursache so erschwert.

Im Anschluss an die Erstellung der Simulationsmodelle, wurden diese innerhalb zweier Anwendungsfälle mit gegensätzlichen Anforderungsprofilen zur simulationsbasierten und anwendungsorientierten Systemauslegung eingesetzt. Die erste, in Kapitel 6 dargestellte, Fallstudie ist ein Netzwerk von elektronischen Schließzylindern. Hier ergibt sich als Anforderung eine möglichst niedrige durchschnittliche Verlustleistungsaufnahme. In der Variante als Online-System ist die sich im Betrieb für den Benutzer bei der Türschließung wahrnehmbare Verzögerung zu minimieren. Bereits im ersten Beispiel offenbart der alle OSI-Schichten überspannende Simulationsansatz sein Potential zur Analyse der leistungsrelevanten Systemparameter. Angefangen von der Darstellung der Übertragungseigenschaften des Kanals mittels austauschbarer Pfadverlustmodelle bis hin zur Abbildung alternativer Umsetzungen auf Anwendungsebene, können die Einflüsse auf das vom Benutzer erfahrene Systemverhalten veranschaulicht werden. Im Rahmen der in Summe 1392 durchgeführten Simulationsläufe wurden unter anderem die Auswirkungen des Einsatzes zweier möglicher SoCs mit unterschiedlichen Eigenschaften verglichen. Dabei zeigte sich, dass die Verlustleistungsaufnahme im Empfangsbetrieb im dargestellten Anwendungsfall von größerer Bedeutung ist als schnelle Wechsel des Transceiver-Betriebszustands. Die zur Weiterleitung von Paketen eingesetzten Routingalgorithmen haben ebenfalls einen deutlichen Einfluss auf die wesentlichen Kennwerte Lebensdauer bzw. durchschnittliche Verlustleistungsaufnahme sowie orts- und szenarioabhängige Verzögerung. Weiterhin kann, wie im Fall von „Minimal Energy Relay Routing“ (MERR), ein Algorithmus auf Basis der Zustellerfolgsquote als sehr geeignet erscheinen, wohingegen sich dessen Einsatz unter Berücksichtigung der Verteilung der energetischen Belastung verbietet. Darüber hinaus konnten verschiedene Varianten der Benutzerauthentifizierung bezüglich ihres Einflusses auf die Systemleistung hin untersucht werden. Es lässt sich belegen, dass der Wechsel von einer Authentifikation mittels leicht zu fälschender Hardware-ID des RFID-Transponders („Unique

Identifizier“/UID) hin zu einer energieaufwändigeren Authentifikation über einen auf dem Transponder gespeicherten Schlüssel bei einem Online-System die Batterielebensdauer nur geringfügig verkürzt. Der geringe zusätzliche Energieaufwand steht mit ca. 11 % Anteil am Gesamtenergiebedarf deutlich hinter dem durch die dauerhafte Empfangsbereitschaft verursachten Anteil zurück (ca. 76 %).

Das zweite Anwendungsbeispiel thematisiert ein optisches beleuchtungsintegriertes Kommunikationssystem, welches Konnektivität für mobile Benutzer bietet. Im Gegensatz zu batteriebetriebenen, vernetzten elektronischen Schließzylindern, wird in diesem Kontext von Kommunikationseinheiten ausgegangen, die wie das Leuchtmittel dauerhaft elektrisch versorgt werden. Insbesondere dadurch standen weder die durchschnittliche Verlustleistungsaufnahme noch die Latenz im Vordergrund, sondern der anwendungsbezogene Durchsatz und der pro Einheit effektiv erzielte Abdeckungsbereich. Weiterhin folgte die Umsetzung keinem der gegenwärtig in Entwicklung bzw. Weiterentwicklung befindlichen Standards, wodurch sich vor allem im Bereich der zu verwendenden Übertragungsparameter deutlich mehr Entwurfsfreiheiten ergaben als noch im ersten Beispiel. Nach experimenteller Verifikation mit Hilfe eines prototypischen „Optical Wireless Communication“-Systems (OWC), wurden zunächst Simulationen vorgenommen, welche die quantitativen Zusammenhänge zwischen Hintergrundbeleuchtungsintensität, Modulationsverfahren und -grad sowie Grad der Vorwärtsfehlerkorrektur auf den Durchsatz sowie den effektiven Abdeckungsbereich verdeutlichen sollten. Bei Verwendung kleiner Öffnungswinkel oder dem Einsatz eines niedrig modulierten Signals, zeigen sich zwischen den mittels unterschiedlicher Varianten von RS-Codes abgesicherten Übertragungen nur äußerst geringe Unterschiede. In solchen Anwendungsszenarien kann also auf eine Vorwärtsfehlerkorrektur verzichtet und so der erzielbare Durchsatz deutlich gesteigert werden. Zur Beurteilung des sich für eine Anwendung ergebenden Durchsatzes wurden die Vermittlungsprotokolle TCP und UDP parametrisiert und in die Simulationen eingebunden. Dadurch offenbarten sich deutlich Wechselwirkungen zwischen der Netzwerktopologie, der Parametrisierung des zur Routenfindung eingesetzten Protokolls und dem resultierenden Durchsatz. Weiterhin konnte gezeigt werden, dass sich die simulierten Durchsatzraten nicht mit gebräuchlichen analytischen Ansätzen vorhersagen ließen, da diese die im Simulationsszenario vorhandene Dynamik nicht berücksichtigen können.

In beiden Fallstudien überzeugte der gewählte simulative Ansatz durch eine hohe Abbildungspräzision. Es ist jedoch anzumerken, dass die Entwicklung der Modelle mittels OMNeT++ inklusive der durchgeführten Verifikation sehr zeitintensiv war. Zusätzlich benötigen die Simulationsläufe Rechenzeit, die mit der Anzahl der vom Simulationskernel zu verarbeitenden Nachrichten, nahezu linear skaliert. Im Kontext der auf Parameterstudien basierenden Entwurfsraumexploration, können und sollten Simulationsläufe parallelisiert gestartet werden, solange es die zur Verfügung stehenden Ressourcen erlauben. Weiterhin lassen sich innerhalb eines Modells vorhandene Teilschritte zusammenzufassen (z.B. Beitritt zu einem Netzwerk als ein Ereignis anstatt Suche, Beitrittsanfrage und Beitrittsbestätigung), um das Abstraktionsniveau zu erhöhen bzw. die Ausführungszeiten der Simulationen zu verkürzen.

Insgesamt erlaubt es die Mischung aus Modularität und Parametrisierbarkeit des Ansatzes sowie der erstellten Simulationsmodelle nahezu beliebige Anwendungsszenarien zu definieren, um so das zu erwartende Systemverhalten vorab zu schätzen und dahingehend anzupassen, dass die aus der Anwendung resultierenden Anforderungen erfüllt werden. Die vorliegende Arbeit leistet damit einen wertvollen Beitrag zur Verfügbarkeit systematischer simulationsbasierter Ansätze zur Entwurfsraumexploration sowie zur anwendungsspezifischen Optimierung von drahtlos kommunizierenden Systemen.

A.1 Parameter für das empirische Pfadverlustmodell der ITU-R

Tabelle A.1: Verlustleistungskoeffizienten, N , zur Berechnung von Übertragungsverlusten im Innenbereich (aus [135])

Frequency	Residential	Office	Commercial	Factory	Corridor
900 MHz	–	33	20	–	–
1,25 GHz	–	32	22	–	–
1,9 GHz	28	30	22	–	–
2,1 GHz	–	25,5 ⁴	20	21,1	17 ⁹
2,4 GHz	28	30	–	–	–
2,625 GHz	–	44 ⁵	–	33 ⁶	–
3,5 GHz	–	27	–	–	–
4 GHz	–	28	22	–	–
5,2 GHz	30 ² /28 ³	31	–	–	–
5,8 GHz	–	24	–	–	–
28 GHz	–	–	27,6 ⁸	–	–
60 GHz ¹	–	22	17	–	16 ⁷⁹
70 GHz ¹	–	22	–	–	–

¹ Die Werte für 60 GHz and 70 GHz setzen eine Übertragung innerhalb eines Raumes voraus und gelten nicht für Übertragungen durch Wände.

² Apartment/Wohnung: Ein- oder zweigeschossige Wohnstätte für mehrere Haushalte. Die meisten Wände sind aus Beton.

³ Haus: Ein- oder zweigeschossige Wohnstätte für einen Haushalt. Die meisten Wände sind aus Holz.

⁴ Computerraum mit vielen im Raum verteilten Computern.

⁵ Sender und Empfänger befinden sich auf derselben Etage und beide Antennen sind in der Deckenhöhe von 2,7 m angebracht.

⁶ Der Pfad zwischen Sender und Empfänger ist teilweise durch metallische Materialien abgeschirmt, wobei sich beide Antennen auf einer Höhe von 1,5 m befinden.

⁷ Sende- und Empfangsantenne haben einen Öffnungswinkel von 15,4°.

⁸ Bahnhof (170 m × 45 m × 21 m (Höhe)) und Flughafenterminal (650 m × 82 m × 20 m (Höhe)): Übertragung ohne Sichtverbindung, Sendeantenne mit Öffnungswinkel von 60° befindet sich in 8 m Höhe und die Empfangsantenne mit einem Öffnungswinkel von 10° ist auf dem Boden in 1,5 m Höhe platziert. Der Wert repräsentiert den maximalen Pfadgewinn bei mehreren Sende- und Empfangsantennenorientierungen.

⁹ Sender und Empfänger befinden sich in Sichtlinie auf einem Flur.

Tabelle A.2: Verlustfaktor bei Bode-/Deckendurchdringung, L_f (dB), wobei n die Anzahl der durchdrungenen Böden/Decken darstellt ($n \geq 1$), zur Berechnung von Übertragungsverlusten im Innenbereich(aus [136])

Frequency	Residential	Office	Commercial
900 MHz	–	9 ($n = 1$), 19 ($n = 2$), 24 ($n = 3$)	–
1,8 GHz bis 2 GHz	4 n	15 + 4($n - 1$)	6 + 3 ($n - 1$)
2,4 GHz	10 ¹ , 5 (house)	14	–
3,5 GHz	–	18 ($n = 1$), 26 ($n = 2$)	–
5,2 GHz	13 ¹ , 7 ²	16 ($n = 1$)	–
5,8 GHz	–	22 ($n = 1$), 28 ($n = 2$)	–

¹ Pro Betonwand

² Verputzte Holzwand

Aufbau von Nachrichten verwendeter Protokolle der Vermittlungsschicht

B.1 Aufbau der Nachrichten des Protokolls „Ad-Hoc On-Demand Distance Vector Routing“ (AODV)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Typ								J	R	G	D	U	Reserviert										Hop Anzahl								
RREQ ID																															
Netzwerkadresse Ziel																															
Zielsequenznummer																															
Netzwerkadresse Ursprung																															
Ursprungssequenznummer																															

(a) Aufbau der „RREQ“-Nachricht:

Typ = „1“, J = Join-Flag (Multicast), R = Repair flag (Multicast), G = Gratuitous RREP (unangeforderte RREP an Ziel senden), D = Destination only (Antwort nur durch Zielknoten), U = Unknown Sequence Number (Zielsequenznummer unbekannt), Hop Count = Anzahl Hops zwischen Ursprungs- und bearbeitendem Knoten

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Typ								R	A	Reserviert										Präfixgröße					Hop Anzahl						
Netzwerkadresse Ziel																															
Zielsequenznummer																															
Netzwerkadresse Ursprung																															
Lebensdauer																															

(b) Aufbau der „RREP“-Nachricht: Typ = „2“, R = Repair flag (Multicast), A = Acknowledgment required (Bestätigung erforderlich), Präfixgröße = angegebener nächster Hop darf für alle Knoten mit identischem Routen-Präfix wie das Ziel verwendet werden, Hop Count = Anzahl Hops von Ursprung zu Ziel des dazugehörigen RREQs, Lebensdauer = Zeit in ms, die die Route als gültig betrachtet werden darf

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Typ								N	Reserviert										Zielanzahl												
Netzwerkadresse unerreichbares Ziel																															
Sequenznummer unerreichbares Ziel																															
weitere Netzwerkadresse unerreichbares Ziel																															
weitere Sequenznummer unerreichbares Ziel																															

⋮

(c) Aufbau der „RERR“-Nachricht: Typ = „3“, Zielanzahl = Anzahl der enthaltenen unerreichbaren Ziele

Abbildung B.1: Für AODV in RFC 3561 definierte Nachrichtentypen

B.2 Aufbau der Nachrichten des Protokolls „Energy-efficient Multi-Path Routing“ (EEMP)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Message Sequence											Message Type						Sender ID (1)														
Sender ID (2)						Node Type						Hop Count						Forward Node ID (1)													
Forward Node ID (2)						Forward Node Energy Level (1)																									
Forward Node Energy Level (2)																															

Abbildung B.2: Aufbau der „HELLO“-Nachricht:

Typ = „1“, J = Join-Flag (Multicast), R = Repair flag (Multicast), G = Gratuitous RREP (unangeforderte RREP an Ziel senden), D = Destination only (Antwort nur durch Zielknoten), U = Unknown Sequence Number (Zielsequenznummer unbekannt), Hop Count = Anzahl Hops zwischen Ursprungs- und bearbeitendem Knoten

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Message Sequence											Message Type						Sender ID (1)														
Sender ID (2)						Sink Numbers						Sink 1 ID																			
Sink 1 Hops						Sink 2 ID												Sink 2 Hops													
⋮																															

Abbildung B.3: Aufbau der „CONNECTIVITY“-Nachricht:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Message Sequence											Message Type						Source ID (1)														
Source ID (2)						Sink ID												Route ID													
Path Cost																															
Forward Node ID (2)						Forward Node Energy Level (1)																									
Forward Node Energy Level (2)																															

Abbildung B.4: Aufbau der „ASSIGN“-Nachricht: Typ = „3“, Zielanzahl = Anzahl der enthaltenen unerreichbaren Ziele

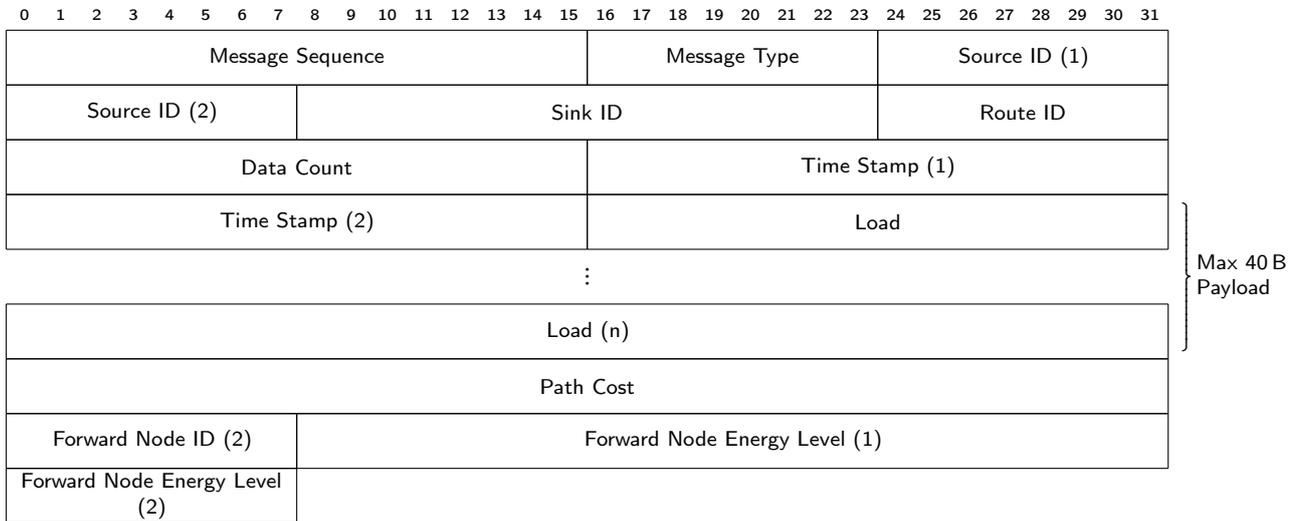


Abbildung B.5: Aufbau der „DATA“-Nachricht: Typ = „3“, Zielanzahl = Anzahl der enthaltenen unerreichbaren Ziele

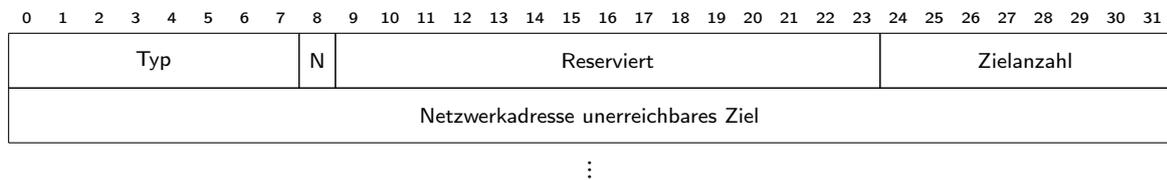


Abbildung B.6: Aufbau der „RESET“-Nachricht: Typ = „3“, Zielanzahl = Anzahl der enthaltenen unerreichbaren Ziele

- [1] G. E. Moore, „Cramming more components onto integrated circuits“, *Electronics*, Bd. 38, Nr. 8, S. 114–117, Apr. 1965.
- [2] G. F. Gilder, *Telecosm: How Infinite Bandwidth Will Revolutionize Our World*. New York: Free Press, 2000, 351 S., ISBN: 978-0-684-80930-4.
- [3] C. Shapiro und H. R. Varian, *Information Rules: A Strategic Guide to the Network Economy*. Boston, Mass: Harvard Business School Press, 1999, 352 S., ISBN: 978-0-87584-863-1.
- [4] (). Global IoT connections by access technology 2016-2021 | Statistic, Statista, Adresse: <https://www.statista.com/statistics/774002/worldwide-connected-devices-by-access-technology/> (besucht am 24. 10. 2018).
- [5] *ISO/IEC Standard 7498-1:1994 - Information Technology – Open Systems Interconnection – Basic Reference Model: THE BASIC MODEL*, 1. Aufl. Geneva, Switzerland: International Organization for Standardization, Nov. 1994, 59 S.
- [6] „ISO 11898-1:2015“, International Organization for Standardization, Geneva, Switzerland, International Standard ISO 11898-1, Dez. 2015, S. 65. Adresse: <http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/36/63648.html> (besucht am 05. 11. 2018).
- [7] „KNX Specifications v2.1“, KNX Association cvba, Specification 2.1, 29. Okt. 2013. Adresse: <https://my.knx.org/en/shop/knx-specifications>.
- [8] „ISO/IEC 14908-1:2012“, International Organization for Standardization, Geneva, Switzerland, International Standard ISO 14908-1, Nov. 2012, S. 420. Adresse: <http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/06/02/60203.html> (besucht am 05. 11. 2018).
- [9] „Iec 62386-101:2014“, International Electrotechnical Commission, Geneva, Switzerland, International Standard IEC 62386-101, 25. Mai 2018, S. 306. Adresse: <https://webstore.iec.ch/publication/63236> (besucht am 05. 11. 2018).
- [10] „ISO 16484-1:2010“, International Organization for Standardization, Geneva, Switzerland, International Standard ISO 16484-1, Nov. 2010, S. 24. Adresse: <http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/03/73/37300.html> (besucht am 05. 11. 2018).

- [11] (19. Feb. 2017). Knx kosten für ein eff. detaillierte übersicht aller kosten!, KNX Anleitungen, Adresse: <https://www.knx-anleitungen.de/knx-grundlagen/was-kostet-ein-smarthome-knx-kosten-813> (besucht am 29.10.2018).
- [12] „IEEE 802.11ac-2013 - 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–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz.“, IEEE, IEEE Standard 802.11ac-2013, 11. Dez. 2013, S. 425. Adresse: https://standards.ieee.org/standard/802_11ac-2013.html (besucht am 13.11.2018).
- [13] „IEEE 802.11ah-2016 - 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 Amendment 2: Sub 1 GHz License Exempt Operation“, IEEE, IEEE Standard IEEE 802.11ah, 5. Mai 2017. Adresse: https://standards.ieee.org/standard/802_11ah-2016.html (besucht am 16.11.2018).
- [14] „IEEE 802.15.4-2015/Cor 1-2018 - IEEE Standard for Low-Rate Wireless Networks Corrigendum 1“, IEEE, IEEE Standard IEEE 802.15.4-2015, 9. Juli 2018. Adresse: https://standards.ieee.org/standard/802_15_4-2015-Cor1-2018.html (besucht am 16.11.2018).
- [15] „ITU-T Rec. G.9959 (02/2012) Short range narrow-band digital radiocommunication transceivers - PHY and MAC layer specifications“, ITU, Geneva, Switzerland, Recommendation G.9959, S. 126. Adresse: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-G.9959-201202-S!!PDF-E&type=items (besucht am 16.11.2018).
- [16] „ISO/IEC 14543-3-10:2012“, International Organization for Standardization, Geneva, Switzerland, International Standard ISO 14543-3-10, März 2012, S. 29. Adresse: <http://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/05/98/59865.html> (besucht am 05.11.2018).
- [17] „Specification of the Bluetooth System Version 4.0“, Bluetooth SIG, Specification, 30. Juni 2010, S. 2303.
- [18] „LoRaWAN [®] L2 1.0.4 Specification (TS001-1.0.4)“, LoRa Alliance Technical Committee, 5177 Brandin Court Fremont, CA 94538 United States, Specification, Okt. 2020, S. 90. Adresse: <https://loro-alliance.org/resource-hub/lorawan-104-specification-package> (besucht am 27.12.2020).
- [19] „IEEE 802.15.7-2011 - IEEE Standard for Local and Metropolitan Area Networks–Part 15.7: Short-Range Wireless Optical Communication Using Visible Light“, IEEE, IEEE Standard IEEE 802.15.7-2011, 6. Sep. 2011. Adresse: https://standards.ieee.org/standard/802_15_7-2011.html (besucht am 16.11.2018).
- [20] obs/iHaus AG. (25. Aug. 2017). *Ihaus all-in-one smart home app*, Adresse: <https://www.presseportal.de/pm/127690/3717732> (besucht am 11.07.2018).
- [21] „IEEE 802.11g-2003 - IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Further Higher Data Rate Extension in the 2.4 GHz Band“, IEEE, IEEE Standard 802.11g-2003, 12. Juni 2003, S. 78. Adresse: https://standards.ieee.org/standard/802_11g-2003.html (besucht am 13.11.2018).

- [22] L. D. Braun und M. Pormann, „The Comprehensive MAC Taxonomy Database: Comatose“, Bielefeld University, 2018. DOI: 10.4119/unibi/2918509. Adresse: <https://pub.uni-bielefeld.de/publication/2918509> (besucht am 08.11.2018).
- [23] A. Sarkar und T. Senthil Murugan, „Routing protocols for wireless sensor networks: What the literature says?“, *Alexandria Engineering Journal*, Bd. 55, Nr. 4, S. 3173–3183, Dez. 2016, ISSN: 11100168. DOI: 10.1016/j.aej.2016.08.003. Adresse: <https://linkinghub.elsevier.com/retrieve/pii/S1110016816302125> (besucht am 08.11.2018).
- [24] P. Park, P. D. Marco und K. H. Johansson, „Cross-Layer Optimization for Industrial Control Applications Using Wireless Sensor and Actuator Mesh Networks“, *IEEE Transactions on Industrial Electronics*, Bd. 64, Nr. 4, S. 3250–3259, Apr. 2017, ISSN: 0278-0046, 1557-9948. DOI: 10.1109/TIE.2016.2631530. Adresse: <http://ieeexplore.ieee.org/document/7755835/> (besucht am 09.11.2018).
- [25] F. Foukalas, V. Gazis und N. Alonistioti, „Cross-layer design proposals for wireless mobile networks: A survey and taxonomy“, *IEEE Communications Surveys & Tutorials*, Bd. 10, Nr. 1, S. 70–85, 21–2008, ISSN: 1553-877X. DOI: 10.1109/COMST.2008.4483671. Adresse: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4483671> (besucht am 09.11.2018).
- [26] C.-F. Chiasserini und M. Garetto, „An Analytical Model for Wireless Sensor Networks with Sleeping Nodes“, *IEEE Transactions on Mobile Computing*, Bd. 5, Nr. 12, S. 1706–1718, Dez. 2006, ISSN: 1536-1233. DOI: 10.1109/TMC.2006.175. Adresse: <http://ieeexplore.ieee.org/document/1717439/> (besucht am 22.03.2018).
- [27] S. Chouhan, R. Bose und M. Balakrishnan, „A Framework for Energy-Consumption-Based Design Space Exploration for Wireless Sensor Nodes“, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Bd. 28, Nr. 7, S. 1017–1024, Juli 2009, ISSN: 0278-0070, 1937-4151. DOI: 10.1109/TCAD.2009.2018865. Adresse: <http://ieeexplore.ieee.org/document/5075821/> (besucht am 18.11.2018).
- [28] H. Q. Nguyen, J. H. Choi, M. Kang, Z. Ghassemlooy, D. H. Kim, S. K. Lim, T. G. Kang und C. G. Lee, „A MATLAB-based simulation program for indoor visible light communication system“, in *2010 7th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP 2010)*, 21.–23. Juli 2010, S. 537–541. Adresse: <https://ieeexplore.ieee.org/document/5580355>.
- [29] D. Hillenbrand, „A Flexible Design Space Exploration Platform for Wireless Sensor Networks“, Karlsruhe Institut für Technologie, Karlsruhe, 2010, 211 S. Adresse: <https://publikationen.bibliothek.kit.edu/1000015671>.
- [30] E. Fischer und G. P. Fettweis, „An accurate and scalable analytic model for round-robin arbitration in network-on-chip“, IEEE, Apr. 2013, S. 1–8, ISBN: 978-1-4673-6492-8. DOI: 10.1109/NoCS.2013.6558403. Adresse: <http://ieeexplore.ieee.org/document/6558403/> (besucht am 23.06.2018).
- [31] I. Minakov und R. Passerone, „PASES: An energy-aware design space exploration framework for wireless sensor networks“, *Journal of Systems Architecture*, Bd. 59, Nr. 8, S. 626–642, Sep. 2013, ISSN: 13837621. DOI: 10.1016/j.sysarc.2013.05.020. Adresse: <https://linkinghub.elsevier.com/retrieve/pii/S138376211300101X> (besucht am 18.11.2018).

- [32] S. Abadal, M. Iannazzo, M. Nemirovsky, A. Cabellos-Aparicio, H. Lee und E. Alarcon, „On the Area and Energy Scalability of Wireless Network-on-Chip: A Model-Based Benchmarked Design Space Exploration“, *IEEE/ACM Transactions on Networking*, Bd. 23, Nr. 5, S. 1501–1513, Okt. 2015, ISSN: 1063-6692, 1558-2566. DOI: 10.1109/TNET.2014.2332271. Adresse: <http://ieeexplore.ieee.org/document/6847751/> (besucht am 18.11.2018).
- [33] B. Cattelan und S. Bondorf, „Iterative design space exploration for networks requiring performance guarantees“, *IEEE*, Sep. 2017, S. 1–10, ISBN: 978-1-5386-0365-9. DOI: 10.1109/DASC.2017.8102106. Adresse: <http://ieeexplore.ieee.org/document/8102106/> (besucht am 12.04.2018).
- [34] „Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 300 MHz to 100 GHz“, ITU, Geneva, Recommendation ITU-R P.1238-9, Juni 2017, S. 28. Adresse: https://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.1238-9-201706-I!!PDF-E.pdf (besucht am 17.06.2017).
- [35] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed, Ser. Prentice Hall Communications Engineering and Emerging Technologies Series. Upper Saddle River, N.J: Prentice Hall PTR, 2002, 707 S., ISBN: 978-0-13-042232-3.
- [36] Chang-Fa Yang, Boau-Cheng Wu und Chuen-Jyi Ko, „A ray-tracing method for modeling indoor wave propagation and penetration“, *IEEE Transactions on Antennas and Propagation*, Bd. 46, Nr. 6, S. 907–919, Juni 1998, ISSN: 0018926X. DOI: 10.1109/8.686780. Adresse: <http://ieeexplore.ieee.org/document/686780/> (besucht am 27.04.2018).
- [37] M. Salem, M. Ismail und N. Misran, „Validation of Three-Dimensional Ray-Tracing Algorithm for Indoor Wireless Propagations“, *ISRN Communications and Networking*, Bd. 2011, S. 1–5, 2011, ISSN: 2090-4355. DOI: 10.5402/2011/324758. Adresse: <http://www.hindawi.com/journals/isrn.communications.networking/2011/324758/>.
- [38] A. Kaya, L. Greenstein und W. Trappe, „Characterizing Indoor Wireless Channels via Ray Tracing Combined with Stochastic Modeling“, *IEEE Transactions on Wireless Communications*, Bd. 8, Nr. 8, S. 4165–4175, Aug. 2009, ISSN: 1536-1276. DOI: 10.1109/TWC.2009.080785. Adresse: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5200977>.
- [39] Z.-Y. Liu, L.-X. Guo und C.-G. Jia, „GPU-based ray tracing algorithm for fast coverage zone prediction under urban microcellular environment“, B. Huang, A. J. Plaza und Z. Wu, Hrsg., 23. Okt. 2013, S. 88950L. DOI: 10.1117/12.2029530. Adresse: <http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.2029530> (besucht am 27.04.2018).
- [40] A. Vilhar, A. Hrovat, I. Ozimek und T. Javornik, „Efficient open-source ray-tracing methods for rural environment“, in *Proceedings of the 16th WSEAS International Conference on Computers*, 2012.
- [41] S. Hosseinzadeh. (11. Okt. 2017). *3D Ray Tracing For Indoor Radio Propagation*. Version 20.0.0.0, Adresse: <https://de.mathworks.com/matlabcentral/fileexchange/64695-3d-ray-tracing-for-indoor-radio-propagation>.
- [42] T. Instruments, „CC430F613x, CC430F612x, CC430F513x MSP430™ SoC With RF Core“, Datasheet, Mai 2009, S. 129. Adresse: <http://www.ti.com/lit/ds/symlink/cc430f5137.pdf> (besucht am 24.09.2018).
- [43] —, „CC430 Family User’s Guide“, User’s Guide SLAU259E, Mai 2009, S. 782. Adresse: <http://www.ti.com/lit/pdf/slau259> (besucht am 24.09.2018).

- [44] P. Huang, L. Xiao, S. Soltani, M. W. Mutka und N. Xi, „The Evolution of MAC Protocols in Wireless Sensor Networks: A Survey“, *IEEE Communications Surveys Tutorials*, Bd. 15, Nr. 1, S. 101–120, First 2013, ISSN: 1553-877X. DOI: 10.1109/SURV.2012.040412.00105.
- [45] „IEEE 802.15.4-2015 - IEEE Standard for Low-Rate Wireless Networks“, IEEE Standard 802.15.4, 2015. Adresse: https://standards.ieee.org/standard/802_15_4-2015.html (besucht am 03.10.2018).
- [46] J. Zheng und M. J. Lee, „A comprehensive performance study of IEEE 802.15. 4“, *Sensor network operations*, Bd. 4, S. 218–237, 2006.
- [47] S. Pollin, M. Ergen, S. Ergen, B. Bougard, L. Der Perre, I. Moerman, A. Bahai, P. Varaiya und F. Catthoor, „Performance Analysis of Slotted Carrier Sense IEEE 802.15.4 Medium Access Layer“, *IEEE Transactions on Wireless Communications*, Bd. 7, Nr. 9, S. 3359–3371, Sep. 2008, ISSN: 1536-1276. DOI: 10.1109/TWC.2008.060057. Adresse: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4626310>.
- [48] M. Petrova, J. Riihijarvi, P. Mahonen und S. LaBellá, „Performance study of IEEE 802.15.4 using measurements and simulations“, in *IEEE WIRELESS COMMUNICATIONS and Networking Conference, 2006. WCNC 2006.*, Las Vegas, NV, USA: IEEE, 2006, S. 487–492, ISBN: 978-1-4244-0269-4. DOI: 10.1109/WCNC.2006.1683512. Adresse: <http://ieeexplore.ieee.org/document/1683512/> (besucht am 03.10.2018).
- [49] Y. Sun, O. Gurewitz und D. B. Johnson, „RI-MAC: A receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks“, ACM Press, 2008, S. 1, ISBN: 978-1-59593-990-6. DOI: 10.1145/1460412.1460414. Adresse: <http://portal.acm.org/citation.cfm?doid=1460412.1460414> (besucht am 27.04.2018).
- [50] C. Perkins, E. Belding-Royer und S. Das, „Ad Hoc On-Demand Distance Vector (AODV) Routing“, IETF, RFC RFC3561. Adresse: <https://www.ietf.org/rfc/rfc3561.txt> (besucht am 03.08.2017).
- [51] C. Chen und J. Ma, „Simulation Study of AODV Performance over IEEE 802.15.4 MAC in WSN with Mobile Sinks“, IEEE, 2007, S. 159–164, ISBN: 978-0-7695-2847-2. DOI: 10.1109/AINAW.2007.335. Adresse: <http://ieeexplore.ieee.org/document/4224098/> (besucht am 04.08.2017).
- [52] A. Gupta, H. Sadawarti und A. K. Verma, „Performance Analysis of AODV, DSR & TORA Routing Protocols“, *International Journal of Engineering and Technology*, Bd. 2, S. 226, 2010.
- [53] Y. Sasson, D. Cavin und A. Schiper, „Probabilistic Broadcast for Flooding in Wireless Mobile Ad Hoc Networks“, Bd. 2, IEEE, 2003, S. 1124–1130, ISBN: 978-0-7803-7700-4. DOI: 10.1109/WCNC.2003.1200529. Adresse: <http://ieeexplore.ieee.org/document/1200529/> (besucht am 07.08.2017).
- [54] M. Zimmerling, W. Dargie und J. M. Reason, „Energy-Efficient Routing in Linear Wireless Sensor Networks“, IEEE, Okt. 2007, S. 1–3, ISBN: 978-1-4244-1454-3. DOI: 10.1109/MOBHOC.2007.4428618. Adresse: <http://ieeexplore.ieee.org/document/4428618/> (besucht am 04.08.2017).
- [55] M. Bhardwaj, T. Garnett und A. Chandrakasan, „Upper Bounds on the Lifetime of Sensor Networks“, Bd. 3, IEEE, 2001, S. 785–790. DOI: 10.1109/ICC.2001.937346. Adresse: <http://ieeexplore.ieee.org/document/937346/> (besucht am 04.08.2017).

- [56] Y. Ming Lu und V. W. S. Wong, „An Energy-Efficient Multipath Routing Protocol for Wireless Sensor Networks“, *International Journal of Communication Systems*, Bd. 20, Nr. 7, S. 747–766, Sep. 2006, ISSN: 10745351, 10991131. DOI: 10.1002/dac.843. Adresse: <http://doi.wiley.com/10.1002/dac.843> (besucht am 04.08.2017).
- [57] H. Frank, Phrood und Anony, *Spektrum elektromagnetischer wellen*, 21. Juni 2009. Adresse: https://de.wikipedia.org/wiki/Elektromagnetisches_Spektrum.
- [58] S. Heissmeyer, L. Overmeyer und A. Muller, „Indoor positioning of vehicles using an active optical infrastructure“, in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sydney, Australia: IEEE, Nov. 2012, S. 1–8, ISBN: 978-1-4673-1954-6. DOI: 10.1109/IPIN.2012.6418914. Adresse: <http://ieeexplore.ieee.org/document/6418914/> (besucht am 30.08.2018).
- [59] geodesy. (2018). Geodesy FSO Laser Datenübertragung Laserlink optischer Richtfunk Gigabit Wireless ab 2600 EUR, Adresse: <https://www.gepanet.com/geodesy.htm> (besucht am 30.08.2018).
- [60] Cree® XLamp® XHP35 LEDs, 24. Apr. 2018. Adresse: <http://www.cree.com/led-components/media/documents/ds-XHP35.pdf> (besucht am 14.08.2018).
- [61] J. Kahn und J. Barry, „Wireless Infrared Communications“, *Proceedings of the IEEE*, Bd. 85, Nr. 2, S. 265–298, Feb./1997, ISSN: 00189219. DOI: 10.1109/5.554222. Adresse: <http://ieeexplore.ieee.org/document/554222/> (besucht am 06.03.2017).
- [62] F. Gfeller und U. Bapst, „Wireless In-House Data Communication via Diffuse Infrared Radiation“, *Proceedings of the IEEE*, Bd. 67, Nr. 11, S. 1474–1486, 1979, ISSN: 0018-9219. DOI: 10.1109/PROC.1979.11508. Adresse: <http://ieeexplore.ieee.org/document/1455777/> (besucht am 06.03.2017).
- [63] S. Dimitrov und H. Haas, *Principles of LED Light Communications: TOWARDS NETWORKED LI-Fi*. Cambridge: Cambridge Univ. Press, 2015, 207 S., ISBN: 978-1-107-04942-0.
- [64] A. M. Khalid, G. Cossu, R. Corsini, P. Choudhury und E. Ciaramella, „1-Gb/s Transmission over a Phosphorescent White LED by Using Rate-Adaptive Discrete Multitone Modulation“, *IEEE Photonics Journal*, Bd. 4, Nr. 5, S. 1465–1473, 26. Juli 2012, ISSN: 19430655. DOI: 10.1109/JPHOT.2012.2210397.
- [65] D. Tse und P. Viswanath, *Fundamentals of Wireless Communication*, 4. printing with corr. Cambridge: Cambridge Univ. Press, 2008, 564 S., ISBN: 978-0-521-84527-4.
- [66] J. G. Proakis, *Digital Communications*, 3. ed, Ser. McGraw-Hill Series in Electrical and Computer Engineering. New York: McGraw-Hill, 1995, 928 S., ISBN: 978-0-07-113814-7.
- [67] E. Kang, E. Jackson und W. Schulte, „An Approach for Effective Design Space Exploration“, in *Foundations of Computer Software. Modeling, Development, and Verification of Adaptive Systems*, R. Calinescu und E. Jackson, Hrsg., Bd. 6662, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, S. 33–54, ISBN: 978-3-642-21291-8. DOI: 10.1007/978-3-642-21292-5_3. Adresse: http://link.springer.com/10.1007/978-3-642-21292-5_3 (besucht am 22.03.2018).
- [68] L. Cai, A. Gerstlauer und D. Gajski, „Retargetable profiling for rapid, early system-level design space exploration“, ACM Press, 2004, S. 281, ISBN: 978-1-58113-828-3. DOI: 10.1145/996566.996651. Adresse: <http://portal.acm.org/citation.cfm?doid=996566.996651> (besucht am 12.04.2018).

- [69] B. Ristau, *Entwurfsraumexploration heterogener Multi-Prozessor-Systeme*, Ser. Beiträge aus der Informationstechnik Mobile Nachrichtenübertragung 53. Dresden: Vogt, 2010, 109 S., ISBN: 978-3-938860-42-7.
- [70] U. Hofmann, *Modellierung von Kommunikationssystemen*, Ser. Fortis. Wien: Manz-Verl. Schulbuch [u.a.], 2000, 182 S., ISBN: 978-3-7068-0595-7.
- [71] D. Baum, *Grundlagen der Warteschlangentheorie*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, ISBN: 978-3-642-39631-1. DOI: 10.1007/978-3-642-39632-8. Adresse: <http://link.springer.com/10.1007/978-3-642-39632-8> (besucht am 30.03.2018).
- [72] C. Lindemann, *Performance Modelling with Deterministic and Stochastic Petri Nets*, Ser. Wiley-Interscience Series in Systems and Optimization. Chichester ; New York: Wiley, 1998, 405 S., ISBN: 978-0-471-97646-2.
- [73] H. Blume, T. von Sydow, D. Becker und T. Noll, „Application of deterministic and stochastic Petri-Nets for performance modeling of NoC architectures“, *Journal of Systems Architecture*, Bd. 53, Nr. 8, S. 466–476, Aug. 2007, ISSN: 13837621. DOI: 10.1016/j.sysarc.2006.11.001. Adresse: <http://linkinghub.elsevier.com/retrieve/pii/S1383762106001408> (besucht am 30.03.2018).
- [74] Y. Jiang und Y. Liu, *Stochastic Network Calculus*. London: Springer, 2008, 229 S., ISBN: 978-1-84800-126-8.
- [75] J.-Y. Le Boudec und P. Thiran, Hrsg., *Network Calculus*, bearb. von G. Goos, J. Hartmanis und J. van Leeuwen, Ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, Bd. 2050, ISBN: 978-3-540-42184-9. DOI: 10.1007/3-540-45318-0. Adresse: <http://link.springer.com/10.1007/3-540-45318-0> (besucht am 30.03.2018).
- [76] G. A. Korn und J. V. Wait, *Digital Continuous System Simulation*. Eaglewood Cliffs, N.J.: Prentice-Hall, 1978.
- [77] F. E. Cellier, *Continuous System Modeling*. New York, NY: Springer New York, 1991, ISBN: 978-1-4757-3924-4. DOI: 10.1007/978-1-4757-3922-0. Adresse: <http://link.springer.com/10.1007/978-1-4757-3922-0> (besucht am 17.04.2018).
- [78] A. Beck, „Simulation: The practice of model development and use“, *Journal of Simulation*, Bd. 2, Nr. 1, S. 67–67, März 2008, ISSN: 1747-7778, 1747-7786. DOI: 10.1057/palgrave.jos.4250031. Adresse: <https://www.tandfonline.com/doi/full/10.1057/palgrave.jos.4250031> (besucht am 30.03.2018).
- [79] M. C. Jeruchim, P. Balaban und K. S. Shanmugan, *Simulation of Communication Systems*, Ser. Information Technology: Transmission, Processing, and Storage. Boston: Kluwer Academic Publishers, 2002, ISBN: 978-0-306-46267-2. DOI: 10.1007/b117713. Adresse: <http://link.springer.com/10.1007/b117713> (besucht am 30.03.2018).
- [80] (4. Nov. 2011). *Ns-2*. Version 2.35, Adresse: <https://www.isi.edu/nsnam/ns/> (besucht am 01.05.2018).
- [81] (5. Jan. 2018). *Ns-3*. Version 3-28, Adresse: <https://www.nsnam.org/> (besucht am 01.05.2018).
- [82] R. Technology. (5. Jan. 2018). OPNET is now part of Riverbed Steelcentral™, Riverbed, Adresse: <https://www.riverbed.com/de/products/steelcentral/opnet.html> (besucht am 01.05.2018).

- [83] J. Prokkola, „OPNET - Network Simulator“, Presentation slides, Tietotalo, University of Oulu, 22. Apr. 2008, Adresse: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.472.5542> (besucht am 27. 12. 2020).
- [84] (). *NetSim-Network Simulator & Emulator | NetSim Academic*, Adresse: <https://www.tetcos.com/netsim-acad.html> (besucht am 02. 05. 2018).
- [85] (). *OMNEST - High-Performance Simulation for All Kinds of Networks*, Adresse: <https://www.omnest.com/> (besucht am 02. 05. 2018).
- [86] A. Varga und R. Hornig. (). *INET Framework - INET Framework*, Adresse: <https://inet.omnetpp.org/> (besucht am 02. 05. 2018).
- [87] E. Weingartner, H. vom Lehn und K. Wehrle, „A Performance Comparison of Recent Network Simulators“, IEEE, Juni 2009, S. 1–5. DOI: 10.1109/ICC.2009.5198657. Adresse: <http://ieeexplore.ieee.org/document/5198657/> (besucht am 14. 05. 2018).
- [88] A. Khan, S. M. Bilal und M. Othman, „A performance comparison of open source network simulators for wireless networks“, IEEE, Nov. 2012, S. 34–38, ISBN: 978-1-4673-3143-2. DOI: 10.1109/ICCSCE.2012.6487111. Adresse: <http://ieeexplore.ieee.org/document/6487111/> (besucht am 03. 05. 2018).
- [89] M. Hawa, „A Graphical User Interface for the ns-3 Simulator“, in *Proceedings of the 12th International Conference on Computer Modeling and Simulation*, Ser. ICCMS '20, New York, NY, USA: Association for Computing Machinery, 22. Juni 2020, S. 159–163, ISBN: 978-1-4503-7703-4. DOI: 10.1145/3408066.3408088. Adresse: <https://doi.org/10.1145/3408066.3408088> (besucht am 14. 09. 2020).
- [90] R. Cruz, H. Fontes, J. Ruela, M. Ricardo und R. Campos, „On the Reproduction of Real Wireless Channel Occupancy in ns-3“, in *Proceedings of the 2020 Workshop on Ns-3*, Gaithersburg MD USA: ACM, 17. Juni 2020, S. 41–48, ISBN: 978-1-4503-7537-5. DOI: 10.1145/3389400.3393754. Adresse: <https://dl.acm.org/doi/10.1145/3389400.3393754> (besucht am 14. 09. 2020).
- [91] K. Kravlevska, M. Garau, M. Førlund und D. Gligoroski, „Towards 5G Intrusion Detection Scenarios with OMNeT++“, in *Proceedings of 6th International OMNeT++ Community Summit 2019*, Sep. 2019, S. 44–35. DOI: 10.29007/2jg6. Adresse: <https://easychair.org/publications/paper/sNcK> (besucht am 14. 09. 2020).
- [92] N. Elmangosh und I. Ighneiwa, „Using Li-Fi to Improve SCADA Communications: Comparative Simulated Study Using OMNet++“, in *Proceedings of the 6th International Conference on Engineering & MIS 2020*, Almaty Kazakhstan: ACM, 14. Sep. 2020, S. 1–8, ISBN: 978-1-4503-7736-2. DOI: 10.1145/3410352.3410830. Adresse: <https://dl.acm.org/doi/10.1145/3410352.3410830> (besucht am 14. 09. 2020).
- [93] P. A. B. Bautista, L. F. Urquiza-Aguilar, L. L. Cardenas und M. A. Igartua, „Large-Scale Simulations Manager Tool for OMNeT++: Expediting Simulations and Post-Processing Analysis“, *IEEE Access*, Bd. 8, S. 159 291–159 306, 2020, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.3020745. Adresse: <https://ieeexplore.ieee.org/document/9181563/> (besucht am 14. 09. 2020).

- [94] A. Varga und R. Hornig, „An Overview of the OMNeT++ Simulation Environment“, in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops*, Ser. Simutools '08, ICST, Brussels, Belgium, Belgium: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, 60:1–60:10, ISBN: 978-963-9799-20-2. Adresse: <http://dl.acm.org/citation.cfm?id=1416222.1416290> (besucht am 25.05.2018).
- [95] A. Varga, *Learn OMNeT++ with TicToc*, 2017. Adresse: <https://docs.omnetpp.org/tutorials/tictoc/> (besucht am 24.05.2018).
- [96] B. W. Boehm, „Guidelines for Verifying and Validating Software Requirements and Design Specifications“, in *Euro IFIP 79*, North Holland, 1979, S. 711–719.
- [97] K. Schwaber, „SCRUM Development Process“, in *Business Object Design and Implementation*, Springer, London, 1997, S. 117–134, ISBN: 978-3-540-76096-2. DOI: 10.1007/978-1-4471-0947-1_11. Adresse: https://link.springer.com/chapter/10.1007/978-1-4471-0947-1_11 (besucht am 21.12.2017).
- [98] K. Beck und C. Andres, *Extreme Programming Explained: EMBRACE CHANGE*, 2. ed., 6. printing, Ser. The XP Series. Boston: Addison-Wesley, 2007, 189 S., ISBN: 978-0-321-27865-4.
- [99] A. Viklund. (26. Juni 2018). *MiXiM*. Version 2.3, Adresse: <http://mixim.sourceforge.net/> (besucht am 20.12.2017).
- [100] A. Chou. (2017). *Coverity Scan Static Analysis*, Synopsys, Adresse: <https://scan.coverity.com/>.
- [101] J. Seward. (14. Juni 2017). *Valgrind*. Version 3.13.0, Adresse: <http://valgrind.org/>.
- [102] H. Lundgren, D. Lundberg, J. Nielsen, E. Nordstrom und C. Tschudin, „A large-scale testbed for reproducible ad hoc protocol evaluations“, Bd. 1, IEEE, 2002, S. 412–418, ISBN: 978-0-7803-7376-1. DOI: 10.1109/WCNC.2002.993531. Adresse: <http://ieeexplore.ieee.org/document/993531/> (besucht am 21.06.2018).
- [103] Jindong Hou und D. O'Brien, „Adaptive Inter-System Handover for Heterogeneous RF and IR Networks“, IEEE, 2005, 125a–125a, ISBN: 978-0-7695-2312-5. DOI: 10.1109/IPDPS.2005.78. Adresse: <http://ieeexplore.ieee.org/document/1419958/> (besucht am 23.08.2017).
- [104] R. Narasimhan, J. Kahn und M. Audeh, „Effect of Electronic-Ballast Fluorescent Lighting on Wireless Infrared Links“, *IEE Proceedings - Optoelectronics*, Bd. 143, Nr. 6, S. 347–354, 1. Dez. 1996, ISSN: 1350-2433, 1359-7078. DOI: 10.1049/ip-opt:19960877. Adresse: http://digital-library.theiet.org/content/journals/10.1049/ip-opt_19960877 (besucht am 22.03.2017).
- [105] F. Gesellschaft. (19. Apr. 2017). *Li-Fi HotSpot - Fraunhofer IPMS*, Li-Fi HotSpot, Adresse: <http://www.ipms.fraunhofer.de/de/research-development/wireless-microsystems/lifi-hotspot.html> (besucht am 19.04.2017).
- [106] iperf. (21. Apr. 2017). *iPerf - The TCP, UDP and SCTP Network Bandwidth Measurement Tool*, Adresse: <https://iperf.fr/> (besucht am 21.04.2017).
- [107] G. Combs. (18. Juli 2017). *Wireshark*. Version 2.2.8, Wireshark Foundation, Adresse: <https://www.wireshark.org/>.

- [108] J. M. G. Balsells, A. Jurado-Navas, J. F. Paris, M. Castillo-Vázquez und A. Puerta-Notario, „Closed-Form BER Analysis of Variable Weight MPPM Coding under Gamma-Gamma Scintillation for Atmospheric Optical Communications“, *Optics Letters*, Bd. 37, Nr. 4, S. 719, 15. Feb. 2012, ISSN: 0146-9592, 1539-4794. DOI: 10.1364/OL.37.000719. Adresse: <https://www.osapublishing.org/abstract.cfm?URI=ol-37-4-719> (besucht am 03.04.2017).
- [109] M. Audeh, J. Kahn und J. Barry, „Performance of Pulse-Position Modulation on Measured Non-Directed Indoor Infrared Channels“, *IEEE Transactions on Communications*, Bd. 44, Nr. 6, S. 654–659, Juni 1996, ISSN: 00906778. DOI: 10.1109/26.506380. Adresse: <http://ieeexplore.ieee.org/document/506380/> (besucht am 03.04.2017).
- [110] H. Bidgoli, Hrsg., *Handbook of Computer Networks: KEY CONCEPTS, Data Transmission, and Digital and Optical Networks*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 23. Nov. 2007, 1022 S., ISBN: 978-1-118-25605-3. DOI: 10.1002/9781118256053. Adresse: <http://doi.wiley.com/10.1002/9781118256053> (besucht am 03.04.2017).
- [111] M. Herceg, D. Zagar und D. Galic, „Multi Pulse Position Amplitude Modulation for Ultra-High Speed Time-Hopping UWB Communication Systems over AWGN Channel“, *IEEE*, März 2010, S. 1–4, ISBN: 978-1-4244-6285-8. DOI: 10.1109/ISCCSP.2010.5463491. Adresse: <http://ieeexplore.ieee.org/document/5463491/> (besucht am 24.08.2017).
- [112] Y.-M. Lin, C.-H. Hsu, H.-C. Chang und C.-Y. Lee, „A 2.56 Gb/s Soft RS (255, 239) Decoder Chip for Optical Communication Systems“, *IEEE Transactions on Circuits and Systems I: Regular Papers*, Bd. 61, Nr. 7, S. 2110–2118, Juli 2014, ISSN: 1549-8328, 1558-0806. DOI: 10.1109/TCSI.2014.2298282. Adresse: <http://ieeexplore.ieee.org/document/6731597/> (besucht am 24.08.2017).
- [113] S. Tirró, Hrsg., *Satellite Communication Systems Design*. Boston, MA: Springer US, 1993, ISBN: 978-1-4613-6309-5. DOI: 10.1007/978-1-4615-3006-0. Adresse: <http://link.springer.com/10.1007/978-1-4615-3006-0> (besucht am 24.08.2017).
- [114] B. Sklar, *Digital Communications: FUNDAMENTALS and Applications*, 2. ed., 17. print. Upper Saddle River, NJ: Prentice Hall PTR, 2009, 1079 S., ISBN: 978-0-13-084788-1.
- [115] J. Postel, „Transmission Control Protocol“, RFC Newtork Working Group, RFC 793, Sep. 1981, S. 85. Adresse: <https://tools.ietf.org/html/rfc793> (besucht am 06.03.2017).
- [116] —, „User Datagram Protocol“, RFC Newtork Working Group, RFC 768, 28. Aug. 1980, S. 3. Adresse: <https://www.ietf.org/rfc/rfc768.txt> (besucht am 06.03.2017).
- [117] L. McLaughlin III, „RFC 1132 - Standard for the Transmission of 802.2 Packets over IPX Networks“, IETF, RFC 1132, Nov. 1989, S. 4. Adresse: <https://tools.ietf.org/html/rfc1132> (besucht am 25.08.2017).
- [118] G. S. Sidhu, R. F. Andrews und A. B. Oppenheimer, *Inside AppleTalk*, 2. ed., 6. print. Reading, Mass.: Addison-Wesley Pub. Co, 1994, 500 S., ISBN: 978-0-201-55021-4.
- [119] J. S. Haugdahl, *Inside NetBIOS*, 3rd. Architecture Technology Corporation, März 1990, 75 S., ISBN: 978-0-939405-01-5.
- [120] E. Blanton und M. Allman, „TCP Congestion Control“, RFC Newtork Working Group, RFC 5681, Sep. 2009, S. 18. Adresse: <https://tools.ietf.org/html/rfc5681> (besucht am 06.03.2017).
- [121] S. Floyd, J. Mahdavi, M. Mathis und A. Romanow, „TCP Selective Acknowledgment Options“, RFC Newtork Working Group, RFC RFC 2018, Okt. 1996, S. 12. Adresse: <https://tools.ietf.org/html/rfc2018> (besucht am 27.02.2017).

- [122] S. Floyd, A. Gurtov, Y. Nishida und T. Henderson, „The NewReno Modification to TCP’s Fast Recovery Algorithm“, IETF, RFC, Apr. 2012, S. 16. Adresse: <https://tools.ietf.org/html/rfc6582> (besucht am 03.04.2017).
- [123] D. Borman, „TCP Options and Maximum Segment Size (MSS)“, IETF, RFC 6691, Juli 2012, S. 9. Adresse: <https://tools.ietf.org/html/rfc6691> (besucht am 03.04.2017).
- [124] M. Mathis, J. Semke, J. Mahdavi und T. Ott, „The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm“, *SIGCOMM Comput. Commun. Rev.*, Bd. 27, Nr. 3, S. 67–82, Juli 1997, ISSN: 0146-4833. DOI: 10.1145/263932.264023. Adresse: <http://doi.acm.org/10.1145/263932.264023> (besucht am 06.03.2017).
- [125] J. Padhye, V. Firoiu, D. F. Towsley und J. F. J. Kurose, „Modeling TCP Throughput: A Simple Model and Its Empirical Validation“, in *Proceedings of the ACM SIGCOMM ’98 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, Ser. SIGCOMM ’98, Vancouver, British Columbia, Canada: Vancouver, British Columbia, Canada, 1998, S. 303–314, ISBN: 1-58113-003-1. DOI: 10.1145/285243.285291. Adresse: <http://doi.acm.org/10.1145/285237.285291> (besucht am 03.06.2017).
- [126] J. Nagle, „Congestion Control in IP/TCP Internetworks“, RFC Newtork Working Group, RFC RFC 896, 6. Jan. 1984, S. 1–9. Adresse: <https://tools.ietf.org/html/rfc896> (besucht am 27.02.2017).
- [127] R. Braden, „Requirements for Internet Hosts – Communication Layers“, RFC Newtork Working Group, RFC RFC 1122, Okt. 1989, S. 1–21. Adresse: <https://tools.ietf.org/html/rfc1122> (besucht am 27.02.2017).
- [128] S. Floyd, H. Balakrishnan und M. Allman, „Enhancing TCP’s Loss Recovery Using Limited Transmit“, RFC Newtork Working Group, RFC RFC 3042, Jan. 2001, S. 9. Adresse: <https://tools.ietf.org/html/rfc3042> (besucht am 27.02.2017).
- [129] S. Floyd, C. Partridge und M. Allman, „Increasing TCP’s Initial Window“, RFC Newtork Working Group, RFC RFC3390, Okt. 2002, S. 1–14. Adresse: <https://tools.ietf.org/html/rfc3390> (besucht am 27.02.2017).
- [130] S. Floyd, J. Mahdavi, M. Mathis und M. Podolsky, „An Extension to the Selective Acknowledgement (SACK) Option for TCP“, RFC Newtork Working Group, RFC RFC 2883, Juli 2000, S. 17. Adresse: <https://tools.ietf.org/html/rfc2018> (besucht am 27.02.2017).
- [131] E. Blanton, K. Fall und M. Allman, „A Conservative Selective Acknowledgment (SACK)-Based Loss Recovery Algorithm for TCP“, RFC Newtork Working Group, RFC RFC3517, Apr. 2003. Adresse: <https://tools.ietf.org/html/rfc3517> (besucht am 27.02.2017).
- [132] V. Jacobson, R. Braden und D. Borman, „TCP Extensions for High Performance“, RFC Newtork Working Group, RFC RFC 1323, Mai 1992, S. 37. Adresse: <https://tools.ietf.org/html/rfc2018> (besucht am 27.02.2017).
- [133] M. Sargent, V. Paxson, M. Allman und J. Chu, „Computing TCP’s Retransmission Timer“, IETF, RFC RFC 6298, Juni 2011, S. 11. Adresse: <https://tools.ietf.org/html/rfc6298> (besucht am 07.06.2017).
- [134] C. Walsh und R. Mac Nally. (3. März 2020). *Package ‘hier.part’*. Version 1.0-6, Adresse: <https://cran.r-project.org/web/packages/hier.part/hier.part.pdf> (besucht am 21.08.2017).
- [135] ITU-R, „Recommendation ITU-R P.1238-7“, Bd. 7, 2012.

- [136] „RECOMMENDATION ITU-R P.1411-6 - Propagation data and prediction methods for the planning of short-range outdoor radiocommunication systems and radio local area networks in the frequency range 300 MHz to 100 GHz“, S. 35,