
Selbstständigkeitserklärung

Hiermit erkläre/n ich/wir, dass die Arbeit selbstständig verfasst, in gleicher oder ähnlicher Fassung noch nicht in einem anderen Studiengang als Prüfungsleistung vorgelegt wurde und keine anderen als die angegebenen Hilfsmittel und Quellen, einschließlich der angegebenen oder beschriebenen Software, verwendet wurden.

Köthen, 09.01.2014

Ort, Datum

Unterschrift/en der/des Studierenden

Inhaltsverzeichnis

Selbstständigkeitserklärung	i
1 Einleitung	1
1.1 Hintergrund zur Forschung zu der Transportschicht	1
1.2 Zielsetzung der Arbeit	3
2 Grundlagen	4
2.1 Der Standard IEEE 802.11.	4
2.2 802.11 Systemarchitektur	5
2.2.1 <i>Ad-hoc</i> Modus	5
2.2.2 <i>Infrastruktur</i> Modus	6
2.3 Die MAC-Schicht nach IEEE 802.11	7
2.3.1 <i>Distribution Coordination Function(DCF)</i>	7
2.3.2 <i>Point Coordination Function(PCF)</i>	9
2.4 Transportschicht Protokolle im WLAN	9
2.4.1 <i>Transmission Control Protocol</i>	10
2.4.2 <i>User Datagram Protocol</i>	12
3 Messaufbau und Messumgebung	14
3.1 Hardware und Software Komponente	14
3.2 Messplatz und Durchführung	14
4 Messergebnisse und Auswertung	20
4.1 das Zugriffsverfahren <i>CSMA/CA</i> zur schlechten <i>TCP</i> -Performance	22
4.2 Die kumulative Verteilung der <i>OWDs</i>	28
4.3 die kumulative Verteilung der Abstände zwischen den Spitzenwerten der <i>OWD</i>	37
5 Zusammenfassung und Ausblick	44
Abbildungsverzeichnis	46
Tabellenverzeichnis	47
Literaturverzeichnis	48

Abkürzungsverzeichnis

TCP	Transmission Control Protocol
CSMA/CD	Carrier Sense Multiple Access/Collision Detection
CSMA/CA	Carrier Sense Multiple Access/Collision Avoidance
OWD	Ein-Weg-Verzögerung
NAV	Network Allocation Vector
CP	Contention Period
CF-ACK	Contention Free-Acknowledgment
IFS	Interframe Space
RTS	Ready To Send
CTS	Clear To Send
RTS/CTS	Ready to Send/Clear to Send
CW	Contention Window
cwnd	congestion window
IEEE	Institute for Electrical and Electronic Engineers
FHSS	Frequency Hopping Spread Spectrum
DSSS	Direct Sequence Spread Spectrum
DCF	Distribution Coordination Function
PCF	Point Coordination Function

1 Einleitung

1.1 Hintergrund zur Forschung zu der Transportschicht

Das Internet hat sich in den letzten zehn Jahren sehr schnell entwickelt und hat eine größere Anzahl von Benutzern erreicht, bis 2012 liegt die Anzahl ungefähr bei 2,4 Billion. Mobile Geräte werden immer beliebter, während die Übertragungsmedien heterogener werden. Neben den üblichen Telefon Sprachdaten, werden moderne mobile Zugangsnetze zunehmend eingesetzt, um die Daten sowohl für private auch geschäftliche Nutzer ins Internet zu transportieren. Neben den reinen Mobilitätsaspekten werden drahtlose Technologien in vielen Regionen der Welt zunehmend als Letzte-Meile-Infrastruktur für den normalen Netzwerkzugang verwendet. Mit dieser Tendenz transportieren drahtlose Zugangsnetze zunehmend die persönlichen und geschäftlichen Daten. Folglich ist die effiziente Datenlieferung über mobile Zugangsnetze sehr bedeutungsvoll für einen kommerziellen Erfolg vieler geschäftskritischer Anwendungen. Drahtlose Netze weisen andere Eigenschaften als diejenigen drahtgebundener Netze. Das traditionelle *Transmission Control Protocol (TCP)* wurde ursprünglich für drahtgebundene Netze implementiert. *TCP* ist ein zuverlässiges, verbindungsorientiertes, full-duplex, Bytestrom, Transportschichtprotokoll [1]. Es ist ein Ende-zu-Ende Protokoll, das Fluss- und Staukontrolle unterstützt und wird verwendet, durch viele Endbenutzer-Anwendungen, einschließlich Email-Clients und Web-Browser. In der Tat nutzt die überwiegende Mehrheit des heutigen Internet-Verkehrs *TCP* [2]. Das zuverlässige Transportprotokoll *TCP* hat in den konventionellen Netzwerken, wo Paketverluste meist wegen Überlastung treten, bis zu gewissen Datenraten sehr gut gedient. Allerdings drahtlose Netzwerke leiden von erheblichen nicht von Überlastung verursachten Verlusten aus Gründen wie Bitfehler und Handoffs. Außerdem ist die Anpassung an die erheblichen Verzögerungen und Rateschwankungen über den drahtlosen Kanal eine der größten Herausforderungen. Diese Schwankungen verursachen *TCP*-Leistungseinbußen, weil die *TCP*-Quelle nicht in der Lage ist, den entsprechenden Durchsatz von Ende-zu-Ende-Pfad korrekt zu schätzen. Nachdem *TCP* in die drahtlose Umgebung eingeführt wurden, hat sie nur eine degradierte Leistung gezeigt. Die Optimierung der *TCP*-Performance über diese Netzwerke müsste einen starken und deutlichen Einfluss auf die Anwendungsleistung haben.

In der letzten Zeit wurde *TCP* über das drahtlose Netzwerk viel recherchiert, um eine bessere Performance erzielt zu werden. Die früheren Forschungsarbeiten wie z.B. [3], [4], [5], [6] zeigten, dass Funkverbindungsverluste dramatisch negative Auswirkungen auf die *TCP*-Performance haben, weil *TCP* nicht in der Lage ist, die Funkverbindungsverluste von *Congestion-Verlusten* zu unterscheiden. Die Untersuchung in [7] hat zum Ziel die *SACK* Option in *TCP* hinzugefügt, somit werden genügend Informationen angeboten, um eine schnelle Wiederholung zu haben, wenn mehrere Pakete in einem *Retransmission Window* verloren gehen. In den weiteren Untersuchungen [8] [9] [10] wurde es gezeigt, dass *lokale Retrans-*

missions und *Error Correction* Paketverluste Ratio reduziert haben, aber lange Übertragungszeit und häufige Veränderung der Übertragungszeit werden als Nebenprodukte dadurch mitgebracht. Die meisten Methoden zum Verbessern der Performance des Transportprotokolls für mobile Netze beinhalten folgende drei Modelle:

1. Ende-zu-Ende Protokoll, wo die drahtlose Verbindung von dem Absender bekannt ist,
2. Link-Schicht Protokoll, das die lokale Zuverlässigkeit bietet
3. aufgeteilte Verbindung Protokoll, das die Verbindung in zwei Teile an der Basisstation aufbricht [11].

In Untersuchungen [12] [13] [14] und [15] werden die Auswirkungen von Datenrate- und Latenzschwankungen auf die *TCP*-Performance auf drahtlosen Verbindungen ausgewertet. In [12] wurde eine Erweiterung der *TCP*-Fehlerkorrektur vorgeschlagen. Zusätzliche Informationen im *TCP*-Header werden verwendet, um die verdächtigen Timeouts zu beseitigen, dann wird die Last wieder hergestellt. In [13] wurden eine Reihe von Empfehlungen, wie die Verwendung von *Selektive Acknowledgments*, große Fenstergröße und *Explicit Congestion Notification* und Aktivierung der Option Zeitstempel vorgestellt, um die *TCP*-Performance auf drahtlose Verbindungen zu verbessern. In [14] ist der *ACK Regulator* als Lösung präsentiert worden, um Paketverluste bei überlasteten Routern zu vermeiden für langlebige Ströme. In [15], ist ein *Window Regulator Algorithmus* vorgeschlagen, um langlebige *TCP*-Performance in Anwesenheit von Kanaländerungen zu jeder gegebenen Puffergröße auf der überlasteten Router maximieren. Erst seit kurzem werden die Auswirkungen der erheblichen Verzögerungen und Rateschwankungen über den drahtlosen Kanal von den Forschungsarbeiten [16] [17] ausgewertet. Es gibt auch einige Studien [18] [19] [20], in denen die Data-Link-Schicht zum Verbessern der *TCP*-Performance analysiert wurde. In der Studie [18] wurde es gezeigt, dass die richtige Wahl der Parameter des Backoff-Mechanismus einen großen Einfluss auf die Netzwerk-Performance hat. Der wichtigste Punkt ist die Auswahl von *minimal Contention Window* (CW_{min}) und *maximal Contention Window* (CW_{max}) je nach Anzahl der vom Netz abgedeckten aktiven Stationen. Studie [19] hat einen einfachen *Self Adaptive Contention Window Adjustment Algorithm-MIMLD for Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA)* vorgestellt. Es stellt den ersten *Contention Window (CW)* dynamisch zu einem nahezu optimalen Punkt nach der Verkehrsaktivität. In Studie [20] wurde ein neues Schema vorgeschlagen, um den Overhead, der die Ineffizienz des *CSMA/CA* verursacht, zu verringern und eine höhere Datenverbindungsrate zu erzielen. In den neusten Forschungsarbeiten [21] [22] wurde die Lösung zur Erkennung von *Congestion Retransmission Timeouts* und *Non-Congestion Retransmission Timeouts* vorgestellt. In Forschungsarbeit [22] wurde ein neuer *TCP*-Algorithmus zur Unterscheidung der *Non-Congestion Retransmission Timeouts*, die durch Übertragungsfehler und plötzliche Verzögerung auf dem Netzwerk entstehen, entwickelt.

Diese Arbeit wird aufgrund der Ergebnisse in den neueren Untersuchungen [23] [24] [25] begonnen. In [23] [24] wird es gezeigt, dass 70% der verworfenen Pakete nach Ablauf des *Retransmission Timeouts* wiederhergestellt werden. Darüber hinaus hat die Studie [25] gezeigt, dass mehr als 20% der *Retransmission Timeouts* fälschlicherweise von Paketübertragungsverzögerung verursacht werden, die führt zu unnötigen Wiederholungen und zur unnötigen Reduzierung des *Congestion Window (cwnd)*.

1.2 Zielsetzung der Arbeit

Wie vorhin erwähnt, mit dem Boom der Handys und Laptops wird die moderne mobile letzte Meile immer populärer, aber das Transportprotokoll *TCP* zeigt eine degradierte Performance auf dem Funknetz, deshalb ist die Forschung über Algorithmen für die effizienteste Verwendung der Bandbreite auf der drahtlosen letzten Meile dringend erforderlich. Die Arbeit versucht aus einer anderen Perspektive, das Problem zu betrachten. Während Schicht 1 und 2 der drahtlosen Kommunikationsnetzwerke in den letzten zehn Jahren signifikante Veränderungen wahrgenommen haben, das „unveränderliche“ Transportprotokoll *TCP* passt sich nicht so gut auf sie an. Schicht 1 ist die physikalische Schicht, die stellt das Medium für die Übertragung zur Verfügung. In dieser Arbeit soll versucht werden, die Einflüsse von Schicht 2 auf die *TCP*-Performance auf dem Funkmedium zu untersuchen.

Im Rahmen dieser Arbeit wird hauptsächlich die Auswirkung der MAC-Schicht auf die Transportschicht untersucht. Im ersten Schritt soll geklärt werden, ob der in dem MAC-Layer von WLAN-Netzen verwendete *CSMA/CA*-Algorithmus zu Paketverlusten im *TCP*-Stack führen kann. Falls es wirklich so wäre, wird es festgelegt, dass die langen *Ein-Weg-Verzögerungen (OWDs)* zum Ablauf des *Retransmission Timeouts* führen könnten, somit wird die *TCP*-Performance beeinträchtigt. Dann wird im zweiten Schritt die Verteilung der *OWDs* untersucht, damit die Beziehung zwischen *OWD* und Paketverlustwahrscheinlichkeit bekommen wird. Durch die Experimente in verschiedenen Szenarien mittels eines speziellen Netzwerktools „LTest [26]“ werden die Rohdaten gespeichert. Danach werden die entsprechend nachbearbeitet und analysiert. Es wird endgültig erwartet, dass unter einer bestimmten Bedingung ein bestimmtes Verhältnis zwischen der *OWD* und der Wahrscheinlichkeit der Paketverluste existiert. Außerdem ist die Verteilung der Abstände zwischen den Spitzenwerten der *OWD* ebenfalls zu bestimmen. Sollte es gelingen die gesuchten Abhängigkeiten zu bestimmen, könnte die *TCP*-Performance in WLAN-Netzen in der Zukunft durch die Veränderung des Algorithmus von *Retransmission Timeout* verbessert werden.

2 Grundlagen

Der Ethernet-Standard beschreibt das Zugriffsverfahren mit Kollisionserkennung und die Vorgehensweise nach dem Auftreten einer Kollision auf der Ebene des MAC-Layer. *Carrier Sense Multiple Access/Collision Detection (CSMA/CD)* ist für den Zugriff auf das gemeinsame Übertragungsmedium eingeführt. Da das Funkmedium ein Shared Media darstellt, können zeitgleich mehrere Stationen auf den Funkkanal zugreifen. Deshalb können Kollisionen auf Funkmedium oft vorkommen, die zur unvollständigen Übertragung führt. Um Kollisionen zu vermeiden und verlorene Daten erneuert auszusenden, übernimmt 802.11-MAC-Layer die primäre Aufgabe, es stellt ein neues Zugriffsverfahren *CSMA/CA* zur Verfügung[27]. Im Folgenden werden die unterschiedlichen Zugriffsverfahren auf Ethernet und drahtlose Netzwerke erklärt. Außerdem werden auch die wichtigen Charakteristika von TCP eingegangen.

2.1 Der Standard IEEE 802.11.

Nach der siebenjährigen Arbeit hat IEEE (Institute for Electrical and Electronic Engineers) einen Standard für drahtlose Netze, den sogenannten Standard 802.11 in 1997 verabschiedet. Es beschreibt den MAC Layer und den Physical Layer im OSI-Schichtenmodell und legt eine Datenübertragung mit 1 oder 2 Mbps im 2.4-GHz-Band fest.

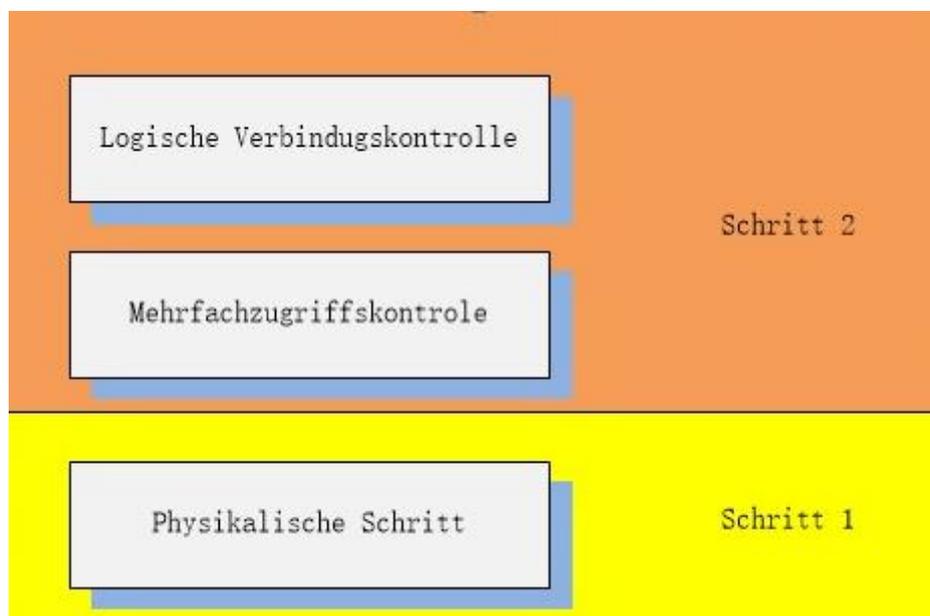


Abbildung 2.1 : Netzwerkschicht des Standards 802.11

Wie in Abbildung 2.1 dargestellt, beschäftigt sich der Standard 802.11 mit den unteren zwei Schichten des ISO/OSI-Modells (International Standards Organisation/Open System Interconnection). Zwei Jahre später wurde der Standard IEEE 802.11b dargestellt. Er definiert im 2.4-GHz-Frequenzband eine Datenrate maximal bis 11 Mbps. Im lizenzfreien 2,400 bis

2,485 GHz Frequenzband stehen für 802.11b 14 Kanäle mit jeweils 22 MHz Bandbreite zur Verfügung. In Europa sind nur 13 Kanäle davon freigegeben. Bei 802.11b WLANs sind alle Funktionen eines verkabelten Netzwerks möglich. Die einzige Einschränkung, liegt in der geringen Übertragungskapazität von Brutto 11Mb, netto sind nur 40 % davon erreichbar. Im selben Jahr wurde der Standard IEEE 802.11a verabschiedet, der eine maximale Datenrate von 54 Mbps im 5-GHz-Frequenzband anbietet, in dem 19 nicht überlappende Kanäle in Europa nutzbar sind. Auch dieser Frequenzbereich ist in Deutschland lizenzfrei nutzbar. Aufgrund der unterschiedlichen Frequenz sind die beiden Standards 802.11a und 802.11b nicht kompatibel zueinander und können nicht über denselben Sender im Access Point genutzt werden. In 2003 wurde der Standard 802.11g verabschiedet, der ebenfalls im 2.4-GHz-Frequenzband arbeitet und eine maximale Datenrate von 54 Mbps erreicht. Der Standard 802.11g ist zum Standard 802.11b für die Übertragung kompatibel, jedoch fällt das 802.11g-Gerät dann in einen Kompatibilitätsmodus zurück, der die Nettogeschwindigkeit reduziert. In 2009 wurde der endgültige Standard 802.11n verabschiedet, er kann sowohl im 2.4-GHz-Frequenzbereich auch im 5-GHz-Frequenzbereich arbeiten und wird zum Standard 802.11a, 802.11b, 802.11g kompatibel. Die maximale Bruttodatenrate wird dabei auf 600 Mbps erhöht.

2.2 802.11 Systemarchitektur

Die IEEE 802.11-Architektur besteht aus mehreren Komponenten, die ein drahtloses Netzwerk bilden, damit die Mobilität der Station in einem bestimmten Bereich garantiert wird. Die *Basic Service Set (BSS)* und der *Independent Basic Service Set (IBSS)* sind die Grundbausteine von IEEE 802.11-basierten WLANs. Die aus einer Reihe von Stationen bestehende *BSS* wird durch eine einzelne *Coordination Function* gesteuert. Sie bestimmt, wann eine Station in der Lage ist, Daten zu senden oder zu empfangen über das drahtlose Medium. Es lassen sich grundsätzlich zwei Formen von drahtlosen Netzwerken bilden. Das sind *Ad-hoc*- und *Infrastruktur*-Netzwerke.

2.2.1 Ad-hoc Modus

Im *Ad-hoc* Modus werden Endgeräte über die kabellose Netzwerkkarte direkt miteinander verbunden. Das *Ad-hoc* Netzwerk ist ein Peer-to-Peer Netzwerk ohne Access Point. Es hat die wesentliche Eigenschaft von Einfachheit und lässt sich sehr schnell aufbauen. Wie in Abbildung 2.2 dargestellt, besteht das *Ad-hoc* Netzwerk aus zwei Stationen, die jeweils eine MAC- und PHY-Schnittstelle zu dem drahtlosen Medium enthalten. Ein Teilnehmer erstellt eine direkte Funkverbindung mit dem anderen, es wird keine Anbindung an ein festes Netz vorgenommen. Die Topologie dieses *Ad-hoc*-Netzwerks wird als *IBSS* genannt.

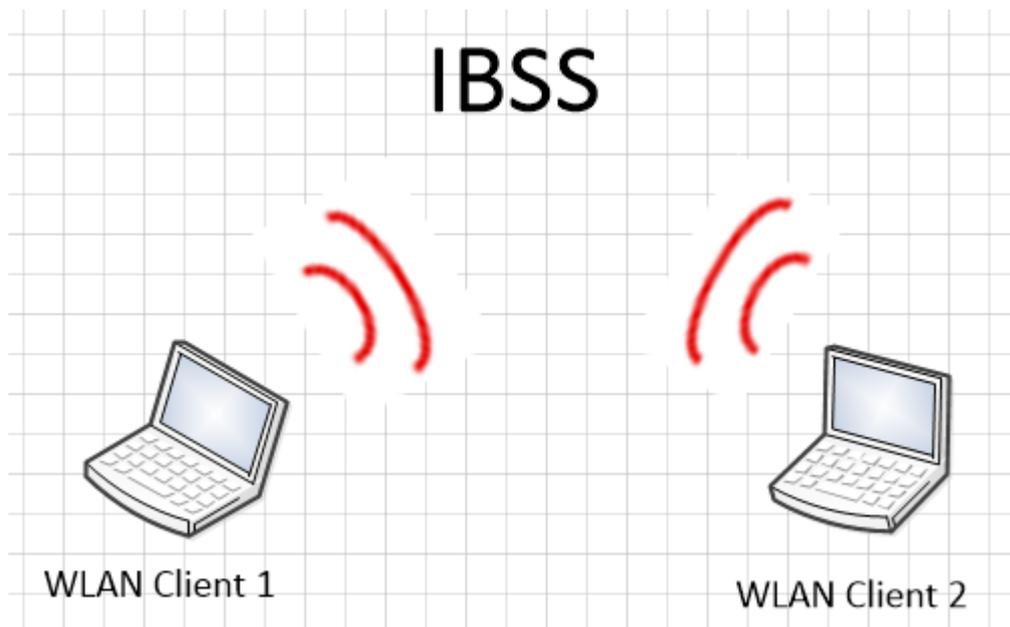


Abbildung 2.2 : Ad-hoc-Netzwerk Topologie

Solange die WLAN-Geräte sich gegenseitig in Reichweite befinden, ist eine Kommunikation zwischen ihnen realisierbar. Aber *IBSS*-Modus wurde nur sehr grob spezifiziert, deshalb gibt es heute noch Probleme, wenn WLAN-Geräte verschiedener Hersteller *Ad-hoc* miteinander verbunden werden sollen. Außerdem ist eine sichere Verschlüsselung im *IBSS*-Modus unmöglich.

2.2.2 Infrastruktur Modus

Im *Infrastruktur*-Modus können Stationen mit Hilfe von Access Point an ein drahtgebundenes Netzwerk angebunden werden. Access Point und Stationen können über Kabel nach 802.3 oder über Funk nach 802.11 verbunden werden. Die Kommunikation zwischen zwei Stationen wird dann immer durch den Access Point gesteuert. Der Infrastruktur-Modus, welcher aus mindestens zwei WLAN Endgeräten und einem Access Point bestehen muss, wird auch als *BSS* bezeichnet. Die *BSS*-Topologie wird in Abbildung 2.3 gezeigt. Wichtigster Bestandteil ist der Access Point, der den Übergang zum drahtgebundenen Netzwerk bildet. Er stellt innerhalb einer Funkzelle den Zugriff auf das drahtgebundene Netzwerk und umgekehrt her. Der Access Point versorgt eine Funkzelle, welche räumlich begrenzt ist. In dieser Funkzelle müssen sich die WLAN-Endgeräte anmelden und authentifizieren. Die Authentifizierung erfolgt nur dann, wenn Das WLAN-Endgerät die Zugangsdaten für die Funkzelle besitzt. Außerdem garantiert der Access Point eine festgelegte Übertragungsrate. Alle Funkteilnehmer müssen sich jedoch diese Übertragungsrate teilen.

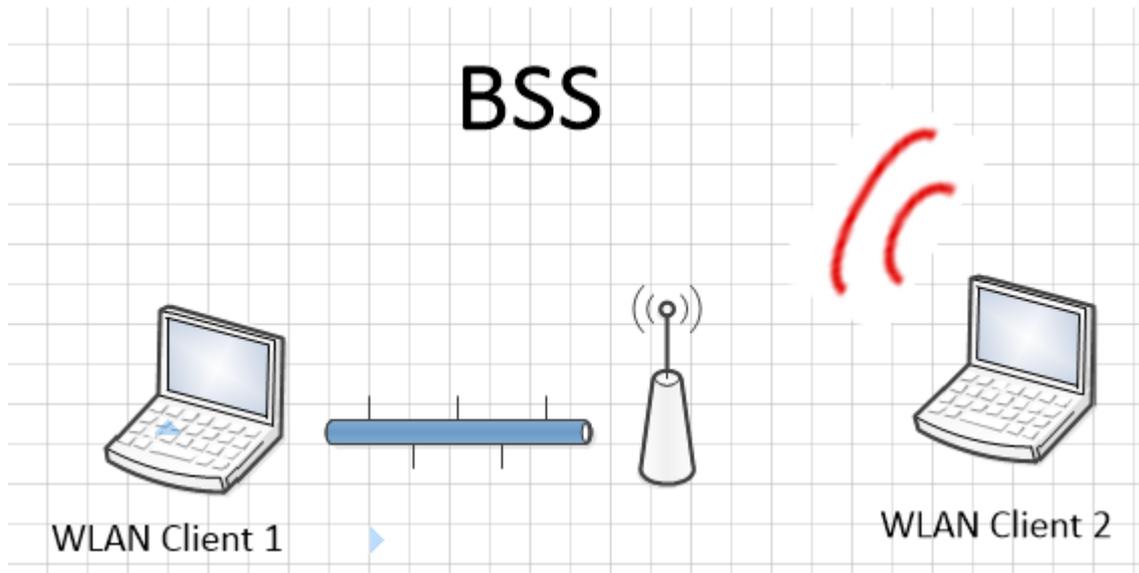


Abbildung 2.3 : Infrastruktur-Netzwerk Topologie

Im Vergleich zum *Ad-hoc* Modus ist *Infrastruktur* Modus besser planbar, geringer stör anfällig und größer reichweit

2.3 Die MAC-Schicht nach IEEE 802.11

Innerhalb des 802.11-MAC unterscheidet man zwischen zwei Methoden, mit diesen der Zugriff von mehreren Teilnehmern auf das Funkmedium geregelt wird. Zum einen existiert eine optionale zentralistische Methode *Point Coordination Function (PCF)*, mit dieser Methode wird der Funkverkehr via Access Point geregelt. Zum anderen eine dezentralistische Methode, die *Distribution Coordination Function (DCF)*.

2.3.1 *Distribution Coordination Function (DCF)*

In Anlehnung an Ethernet benutzt das 802.11-MAC-Protokoll als Basiszugriffsmethode in der *DCF CSMA/CA*. Das in 802.3 standardisierte Zugriffsverfahren *CSMA/CD* wurde in den drahtlosen Netzen verzichtet, weil aus technischen Gründen die Kollisionserkennung während der Aussendung der Daten nicht möglich ist. Außerdem lassen sich Kollisionen auf dem drahtlosen Medium nicht von anderen Störungen unterscheiden. Kollisionen beim WLAN könnten erst nach vollständiger Übertragung festgelegt werden, da ein Sender nicht gleichzeitig empfangen kann, deswegen würde die Verzögerung bei jeder Kollision sehr groß werden. Indem das Verfahren Kollisionsvermeidung verwendet wird, soll die Wahrscheinlichkeit für das Vorkommen einer Kollision klein gehalten, damit eine vollständige Datenübertragung sichergestellt wird.

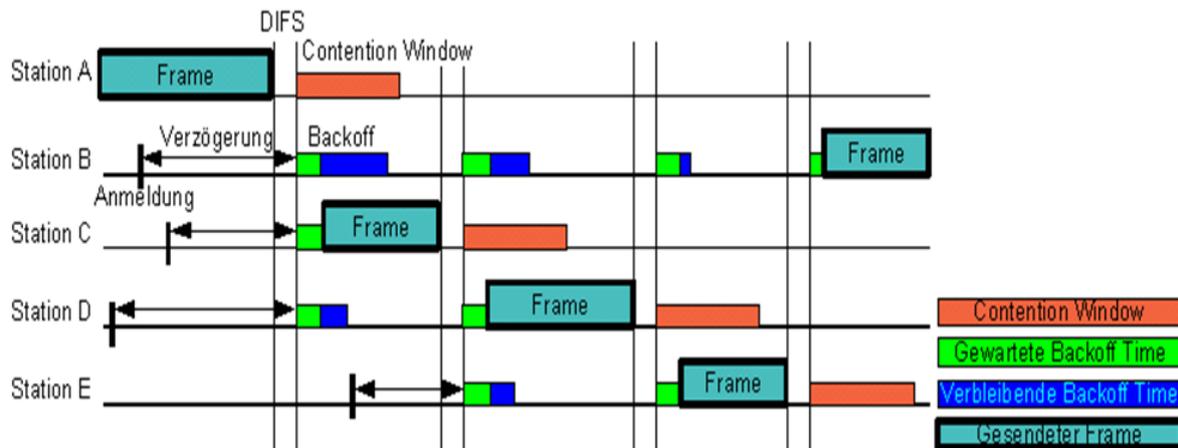


Abbildung 2.4 : Distributed Coordination Function[28]

In Abbildung 2.4 wird das Arbeitsprinzip von DCF dargestellt. Beim CSMA/CA-Verfahren überprüft eine sendwillige Station das drahtlose Medium, ob es bereits belegt ist, indem es dieses abhört. Ist das Medium belegt, wartet die Station bis zum Ende der laufenden Datenübertragung. Während der Datenübertragung sammeln sich mit größter Wahrscheinlichkeit mehrere Stationen, die eventuell auch senden wollen. Wenn nun das Medium frei ist, werden mehrere Stationen mehr oder weniger zeitgleich versuchen, auf das Medium zuzugreifen, daher besteht die Kollisionsgefahr. In dieser Phase werden mehrere Stationen untereinander in Konkurrenz um den Mediengriff stehen. Und diese Phase bzw. Zeitspanne wird als *Contention Period (CP)* genannt. Damit nach Ablauf des *Distributed Coordination Function Interframe Space (DIFS)* nicht alle Stationen gleichzeitig senden, wird die zufällige Wartezeit gewartet. Die Station mit der kleinsten Wartezeit wird als erste auf das Medium zugreifen und ihre Daten aussenden. Jeder ausgesendeter Datenframe enthält den Wert *Duration*, der angibt, wie lange das Medium besetzt ist. Alle Stationen kopieren diese Zeit in ihren eigenen *Network Allocation Vector (NAV)* und verwalten diesen wie einen Timer. Solange NAV ungleich 0 ist, ist das Medium belegt. Da jedoch Kollisionen nicht ausgeschlossen werden können, wird der erfolgreiche Empfang der Daten dem Sender vom Empfänger durch eine Empfangsbestätigung mitgeteilt. Das Fehlen der Empfangsbestätigung wirkt wie eine Kollision, so werden die Daten nach einer bestimmten Wartezeit erneut ausgesendet. Die Wartezeit bei Kollisionen steigt nach jedem erneuten Sendungsversuch exponentiell bis zu Grenze an, bei Empfangsbestätigung wird wieder mit dem ersten Versuch gestartet. Die Wartezeit wird durch die Formel $\text{Backoff Time} = \text{Random (CW)} * \text{Slot Time}$ ermittelt. CW stellt eine Integerzahl aus dem Bereich zwischen CW_{min} und CW_{max} dar. CW-Größe und Slot Time sind von der verwendeten physikalischen Schicht abhängig.

Frequenzspreizungsverfahren	CW_{min}	CW_{max}
FHSS	15	1023
DHSS	31	1023

Tabelle 2.1 : CW-Größe mit verschiedenen Frequenzspreizungsverfahren

In Tabelle 2.1 wird die CW-Größe mit verschiedenen Frequenzspreizungsverfahren gelistet. Bei FHSS liegen CW_{min} und CW_{max} jeweils bei 15 und 1023. Slot Time dauert bei Frequency

Hopping Spread Spectrum (*FHSS*) 50us. Deshalb die Wartezeit bei *FHSS* liegt zwischen 750 us und 51,2 ms. Bei *DSSS* liegen CW_{min} und CW_{max} jeweils bei 31 und 1023. Slot Time dauert bei *DSSS* 20 us. Deshalb die Wartezeit bei *DSSS* liegt zwischen 620 us und 20,5 ms.

2.3.2 Point Coordination Function(PCF)

Die PCF ist im 802.11-Standard eine optionale Methode des MAC-Layers, die ein Zugriffsverfahren definiert. Sie arbeitet nur im Infrastruktur-Modus und ist für den Access Point optional. Beim *PCF*-Verfahren bildet ein Access Point den Koordinator. Die Arbeitsweise des *PCF* wird wie in Abbildung 2.5 dargestellt. Der *Point Coordinator* wartet bis das Medium frei ist, und sendet nach einer *PIFS*-Zeit(*Point Coordination Interframe Space*), die zwischen *SIFS* und *DIFS* liegt, den Start des *PCF*-Modus an alle Stationen. Mit dieser Ankündigung wird auch eine *NAV*-Zeit angegeben, in der sämtliche Backoff-Verfahren ausgesetzt werden. Dann pollt der *Point Coordinator* nacheinander jede einzelne Stationen, die sich im selben *BSS* befindet, das dann dieser Station das Senderecht für ein Frame gibt. Die Wartezeit ist immer *SIFS*, dh. der kürzeste Zeitraum, dadurch ist ein hoher Durchsatz möglich. Mit dieser zentralen Zuteilung werden somit auch Kollisionen ausgeschlossen. Stationen senden im Wesentlichen Daten und *Contention Free-Acknowledgment* (*CF-ACK*) an den *Point Coordinator*. Nachdem alle Stationen abgefragt wurden, beendet der Access Point den *PCF*-Modus nach einem weiteren *SIFS* durch das Aussenden eines *CF-End-Frames*.

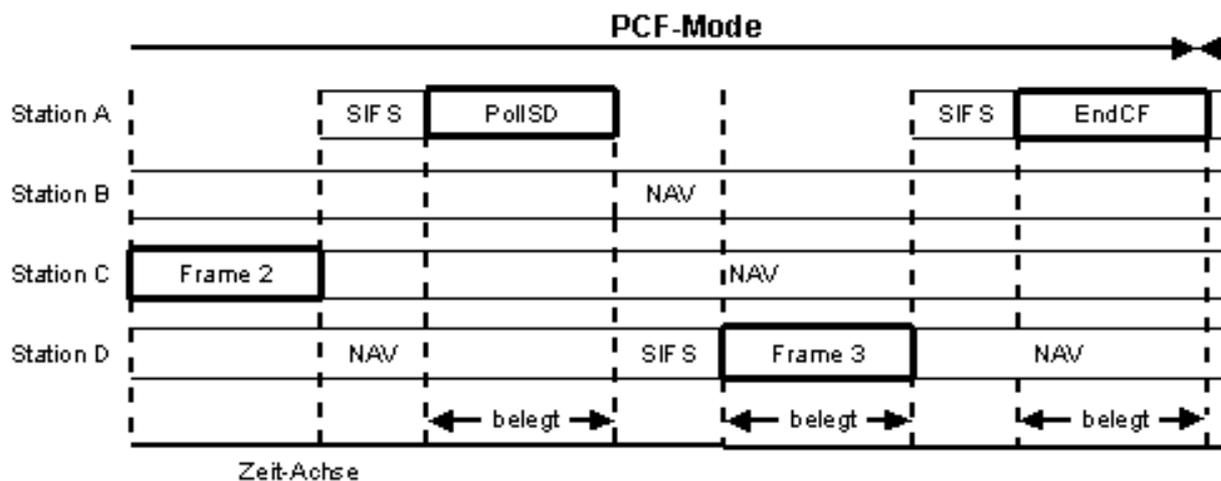


Abbildung 2.5 : Point Coordination Funktion[29]

Allerdings ist *PCF* in der Praxis nicht verwendet, weil es in der Regel keine Möglichkeit gibt, zu verhindern, dass Stationen in einem anderen nahe gelegenen Netzwerk den konkurrierenden Verkehr übertragen

2.4 Transportschicht Protokolle im WLAN

Die zwischen der Anwendungs- und Vermittlungsschicht angesiedelte Transportschicht ist die Schicht, in der eine direkte logische Ende-Zu-Ende-Kommunikation zwischen zwei Teilnehmern realisiert wird. Sie bildet das Kernstück der geschichteten Netzwerkarchitektur.

Die berühmtesten Internet-Protokollen, bzw *TCP* und *UDP* sind auf dieser Ebene angesiedelt.

2.4.1 *Transmission Control Protocol*

TCP ist ein verbindungsorientiertes und verlässliches Protokoll mit Fehlererkennung, -korrektur sowie Flusskontrolle. Die aufgesetzten Anwendungen sind bei Verwendung von *TCP* auch immer sicher, dass die Daten vollständig und ohne Fehler übertragen werden könnten. Der Preis dafür ist eine langsamere Übertragung auf einer schlechten oder fehlerbehafteten Verbindungsstrecke.

- ***Slow Start und Congestion Avoidance***

Durch den Bezug auf Bestätigungspakete ist die Versandrate von Datenpaketen, und damit der Datendurchsatz, prinzipiell von der Umlaufzeit der einzelnen Pakete abhängig, wäre es unnötige Verschwendung von Bandbreite. Um von dieser Beschränkung gelöst zu sein, hat ausgefeilte Fenstertechnik eingesetzt. *TCP* nutzt ein Senderfenster, das es erlaubt, dass schon nachfolgende Datenpakete zum Empfänger gesendet werden, obwohl die originalen Pakete von dem Empfänger noch nicht bestätigt wurden. Die Fenstertechnik hat eine Möglichkeit angeboten, die vorhandene Verbindungsbandbreite auszunutzen. Für jedes abgeschickte Paket startet *TCP* einen separaten Timer, der das mögliche Ausbleiben der Bestätigung anzeigt und dann ein Retransmission für verlorene Pakete einleitet.

Slow Start bestimmt die Größe des ausgefeilten Fensters, also wie viele Pakete auf einmal gesendet werden. Das *cwnd* wird benutzt, um die Anzahl der gleichzeitig zu schickenden Pakete anzugeben. Das *cwnd* wird mit 1 initialisiert und bei Ankunft der Bestätigung für das erste Paket auf 2 erhöht. Das *cwnd* erfährt durch *Slow Start* eine exponentielle Steigerung. Mit *Slow Start* kann es sehr schnell zu hohen Senderaten kommen, mit denen viele Server im Internet einfach nicht fertig werden, deshalb greift *Congestion Avoidance* ein. *Congestion Avoidance* sorgt für ein vorsichtiges Erhöhen des erzeugten Verkehrs, damit das Netzwerk nicht überlastet wird. Wenn eine Verstopfung auftritt, so wird die Hälfte der Paketanzahl als *threshold size (ssthresh)* gespeichert, und *cwnd* wird als 1 gesetzt. Dann übernimmt die Steigerung der Paketanzahl. Wenn nun das *cwnd* die gleiche Menge an Paketen wie *ssthresh* angibt, übernimmt *Congestion Avoidance*. Nun wird die Anzahl der Pakete bei jeder Round Trip Time nur um $1/cwnd$ gesteigert.

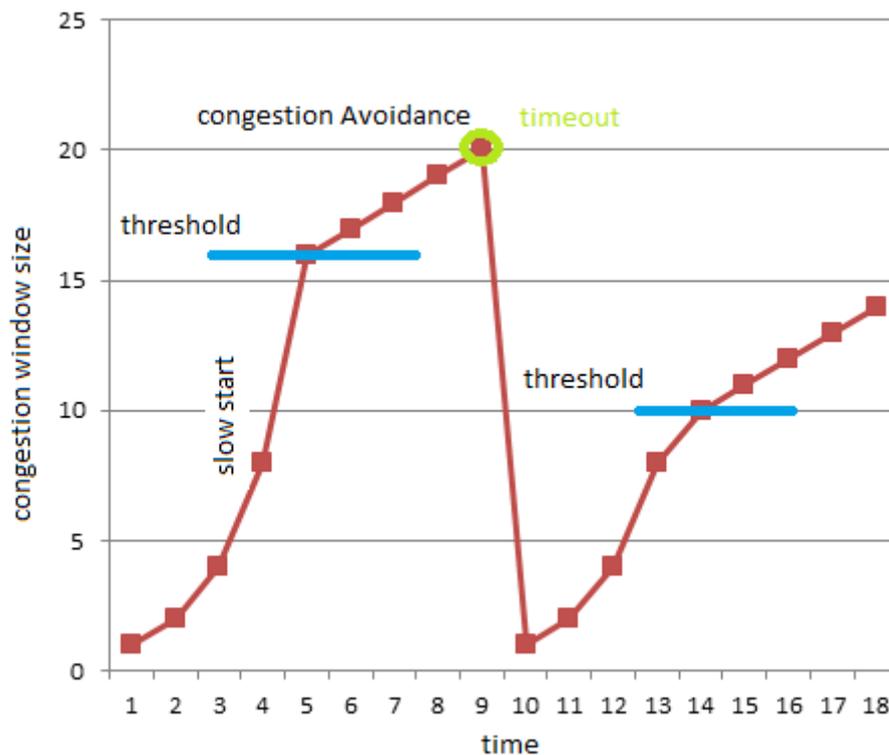


Abbildung 2.6 : *Slow Start* und *Congestion Avoidance*

In Abbildung 2.6 ist es erkannt, *cwnd* steigt von 1 s bis 5 s exponentiell und dann erreicht den Schwellwert. Von 5 s greift *Congestion Avoidance* ein und *cwnd* nimmt linear auf. Um 9 s tritt ein Paketverlust auf, dann wird Schwellwert als die Hälfte von *cwnd* gesetzt und fällt *cwnd* auf 1.

- **TCP Retransmission Timeout**

Der wichtigste Timer von *TCP* ist *Retransmission Timeout (RTO)*, wenn ein Paket gesendet wird, wird ein Timer gleichzeitig getriggert. Wenn das Paket bestätigt wird, bevor die Zeit abgelaufen ist, wird der Timer gestoppt. Wenn auf der anderen Seite der Timer abgelaufen ist, bevor die Bestätigung eintrifft, wird das Paket erneut gesendet. *TCP* ist konfrontiert mit einer radikal anderen Umwelt. Es ist sehr schwer festzulegen, wie lange der Timeout sein sollte. Wenn Timeout zu kurz eingestellt ist, wird unnötige erneute Übertragung auftreten, somit wird Internet mit nutzlosen Paketen verstopfen. Aber wenn es zu lang eingestellt ist, wird die Leistung degradiert aufgrund der langen erneuten Übertragungsverzögerung, solange ein Paket verloren geht. Die *Round Trip Time (RTT)* ist eine für *TCP* fundamentale Funktion denn sie hilft dabei sinnvolle Entscheidung zu treffen, ob ein Paket nun wirklich verloren gegangen ist, oder es noch unterwegs ist, damit unnötige erneute Übertragungen verhindert werden,

Die *RTT* wird beim ersten Paket gemessen und dann durch einen Formel in eine geschätzte Zeit umgewandelt, die eventuelle Routenänderungen einkalkuliert. Die *RTT* wird mit jeder ankommenden Bestätigung neu berechnet, wobei die neu berechnete Zeit zu 12,5% mit in die Berechnung für die gesamt Zeit eingeht. Die *RTO* wird nach folgender Formel berechnet:

$$SRTT = a \cdot SRTT + (1-a) \cdot R$$

$$RTO = 2 * SRTT$$

SRTT gibt die geschätzte gesamte *RTT* an, *R* ist die gemessene *RTT* und *a* ist ein Glättungsfaktor, der empfehlenerweise immer bei 1/8 liegen sollte (RFC793). Erste Implementierung von *TCP* verwendet $2 * SRTT$ als *RTO*.

Aber ein konstanter Wert zu unflexibel war, weil es nicht richtig reagiert, wenn die Varianz aufgestiegen ist. Um dieses Problem zu beheben, wird der Timeout empfindlich auf die Varianz in *RTT* gemacht. Diese Änderung erfordert die Verfolgung der anderen Variable *Round Trip Time Variation (RTTVAR)*.

$$RTTVAR = b * RTTVAR + (1 - b) * |SRTT - R|$$

$$RTO = SRTT + 4 * RTTVAR$$

Normalerweise liegt *b* bei 3/4. Diese Gleichung ist verwendet, wenn die Standardabweichung von *RTT* sehr groß ist.

- **Faktoren zur schlechten *TCP*-Performance auf Funkverbindung**

1. Handoff
2. Hohe Bitfehler-Rate
3. Möglicher Backoff von *CSMA/CA*

Bei Handoff sind Paketverluste immer nicht vermeidbar. Außerdem nimmt die Umschaltung in dieser Phase zu viel Zeit, somit sind die Verzögerungen extrem groß. Die großen Verzögerungen könnten höchst wahrscheinlich *TCP*-*RTO* triggern. Aber dieser Faktor wird hier nicht berücksichtigt, weil die Experimente in *WLAN*-Netzen durchgeführt wurden. Die Hohe Bitfehler-Rate des Funkmediums verursacht auf einer Seite die Paketverluste auf der *TCP*-Schicht, auf der anderen Seite die *MAC*-Schicht-Retransmissions. Wenn Paketverluste auf der *TCP*-Retransmissions auftreten, werden sofort Datendurchsatz verringert und die verloren gegangenen Pakete erneut übertragen. *MAC*-Schicht-Retransmissions führen zur hohen Verzögerung. Die zwei Faktoren zur schlechten *TCP*-Performance wurden schon in der letzten Zeit viel diskutiert.

Der dritte Faktor ist die Annahme von dieser Arbeit, wird gezielt untersucht. wenn *TCP* auf *Ethernet* im *LAN* läuft, wird die erwartete Verzögerung in Mikrosekunden gemessen und höchst wahrscheinlich vorhersehbar. Bei der drahtlosen Verbindung ist es ganz anders. Die hohen Verzögerungen der Funkverbindung wurden schon lange anerkannt. Das zugriffsverfahren *CSMA/CA* hat festgelegt, vor jeder Übertragung muss der Sender eine Zufällige Zeit warten. Außerdem ist die Funkverbindung eine lose Kopplung. Das zugriffsverfahren *CSMA/CA* muss durch Retransmissions eine fehlerfreie Übertragung bewahren. Nach jedem Retransmission wird die zufällige Wartezeit von *CSMA/CA* aus einem vergrößerten Bereich ausgewählt. Dadurch ist es offensichtlich, dass das Zugriffsverfahren die Verzögerung erhöht. Es reagiert dann entsprechend mit ablaufenden Timern und Retransmissions. Die Timer werden neu und länger eingestellt und die Datenrate fällt unter.

2.4.2 User Datagram Protocol

UDP ist auch ein verbindungsloses Transportschichtprotokoll, aber es hat keine Mechanismen zur Fehlererkennung und -korrektur sowie Flusssteuerung. Ein *UDP*-Sender sendet ein-

fach die Daten zum Empfänger und interessiert sich überhaupt nicht, ob die Daten den Empfänger erreichen könnten oder nicht. *UDP*-Empfänger wartet auf die herkommenden Daten. Wenn die vom *UDP*-Sender gesendeten Pakete auf dem Weg zum Empfänger verloren gegangen sind, werden die Pakete nicht erneut übertragen. Deshalb hat *UDP* den Ruf eines ungesicherten Protokolls bekommen. Wegen dieser Besonderheit ist *UDP* sehr geeignet für die Untersuchung der Eigenschaft von *CSMA/CA*. Ein-Weg-Verzögerungen werden die wahren Eigenschaften von *CSMA/CA* reflektieren. Stattdessen ist *TCP* nicht förderlich die Eigenschaften von *CSMA/CA* zu beobachten, weil ein großer OWD durch eine erneute Übertragung verursacht werden könnte, dann die Übertragungsgeschwindigkeit von *TCP* wird reduziert. Diese Faktoren haben eine negative Wirkung auf die Beobachtung der *CSMA/CA* Eigenschaft.

3 Messaufbau und Messumgebung

Die Messungen wurden in zwei Räumen durchgeführt, um eine gute und schlechte Verbindungsqualität zu realisieren. Insgesamt wurden die gesamten Messungen in 3 Szenarios eingeteilt, um die gängigsten Arten von drahtlosen Umgebungen zu simulieren. Der ausgewählte Drahtlos-Modus ist der häufigste *Infrastruktur*-Modus, der vergleichend mit dem *Ad-hoc*-Modus den Vorteil der Stabilität, Skalierbarkeit, verbesserten Reichweite und zentralisierten Sicherheit. Im ersten Teil wird überzeugt, ob *CSMA/CA* zu Paketverlusten führen könnten. Im zweiten Teil wird die Beziehung zwischen *OWD* und Paketverlustwahrscheinlichkeit untersucht.

3.1 Hardware und Software Komponente

Die in den Messungen benutzten Geräte beinhalten:

- 3 Gleiche Rechner, auf denen das System Linux openSUSE(kernel 3.7.10-1.16) läuft
- 1 Linksystem drahtloser Broadband-Router, der 802.11 b/g unterstützt
- 2 USB drahtlose Adapters, die 802.11 b/g/n unterstützen
- 2 Stück Kabel CAT.6 SFTP 4X2XAWG27/7



Abbildung 3.1 : USB drahtloser Adapter und drahtloser Router

Oben wurden die verwendeten Hardwarekomponenten aufgelistet. In Abbildung 3.1 werden der im Experiment verwendete drahtlose Router und Adapter dargestellt. Da es gewünscht ist, die Verzögerung von jedem ausgehenden Paket in der Funkstrecke verfolgt werden zu können, neben den genannten Hardwarekomponenten ein kommerzielles Werkzeug LTest noch gebraucht auf den Rechnern zu installieren. Wireshark (1.8.9) wird zur Bestimmung, wann *TCP* Retransmission vorkommt, verwendet.

3.2 Messplatz und Durchführung

Um die oben beschriebenen Szenarios zu simulieren, die Position des Prüfstands muss vorher angeordnet werden. Die räumliche Lage des Prüfstands wurde eingerichtet wie in Abbildung 3.2 dargestellt.

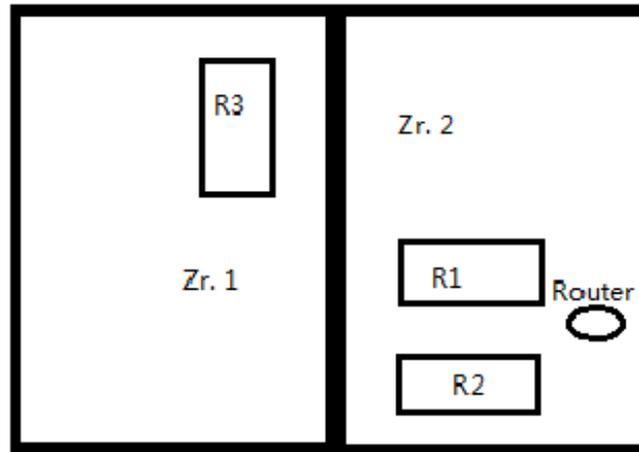


Abbildung 3.2 : die räumliche Lage des Prüfstands

Alle Geräte verteilen sich in zwei Räumen, Rechner 3 steht in Zimmer 1, Router, Rechner 1 und 2 befinden sich in Zimmer 2. Die Messungen werden in drei Szenarios eingeteilt, Verbindung mit guter Link-Qualität, schlechter Link-Qualität und Cross-Traffic. Das erste Szenario und zweite Szenario werden jeweils in Zimmer 2 und Zimmer 1 durchgeführt. Für Szenario 3 werden beide Zimmer gleichzeitig benutzt.

Die Test-Topologie von Szenario 1 und 2 wird in Abbildung 3.3 dargestellt. Diese Topologie gibt eine bessere Möglichkeit zu beobachten wie CSMA/CA jedes Paket beeinflusst, denn hier CSMA/CA wird nur im Downlink verwendet. Ansonsten CSMA/CA wird auf die Ergebnisse sowohl in Uplink und Downlink wirken, es ist nicht förderlich zur Beobachtung der Auswirkung von CSMA/CA auf Pakete.

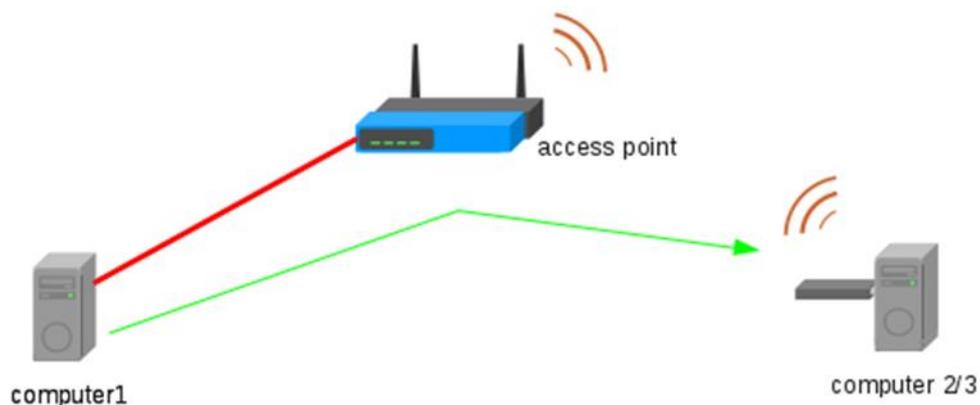


Abbildung 3.3 : Experiment-Topologie auf Szenario 1 und 2

Der Rechner 1 wird mit Access Point über ein 3-Meter-Kabel angeschlossen und Access Point wird drahtlos mit Rechner 2 oder 3 verbunden. Für Szenario 1 liegt Abstand zwischen Access Point und Rechner 2 ungefähr bei ein Meter. Access Point und Rechner 3 in Szenario 3 werden durch eine 20-Zentimeter-Betonwand getrennt, der linear Abstand zwischen den beiden ist ca. 8 Meter. LTest-Client auf Rechner 1 sendet Pakete zum Rechner 2 oder 3, wo LTest-Server läuft. In Szenario 1 sind Rechner 2 und Access Point nahe beieinander,

dadurch könnte eine Verbindung mit hoher Verbindungsqualität aufgebaut werden. Für Szenario 2 wird der Rechner 3 fern von Access Point gehalten, in der Messung befindet sich Rechner 3 in einem anderen Zimmer, damit wird eine Umgebung mit schlechter Verbindungsqualität erstellt. Die Verbindungsparameter werden in Tabelle 3.1 dargestellt.

Szenario	Verbindungsqualität	Signalpegel
1	70/70	-39 dBm +/- 5dBm
2	45/70	-65 dBm +/- 5dBm

Tabelle 3.1 : Verbindungsparameter auf Szenario 1 und 2

In Szenario 3 wird Cross-Traffic emuliert, eine zusätzliche Verbindung wird erforderlich, um die Link-Ressource zu kämpfen während der Kommunikation der anderen. In diesem Szenario wird Cross Traffic zwischen den Rechnern 1 und 3 und Probe Traffic zwischen Rechner 1 und 2 erzeugt. Cross Traffic wird während der Durchführung von Test auf Szenario 3 laufen. In Abbildung 3.4 werden die beschriebenen Verbindungen dargestellt.

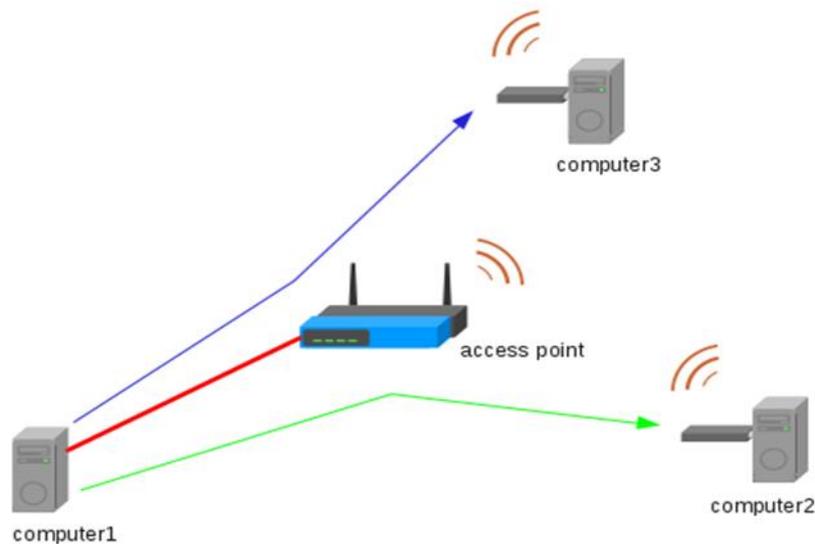


Abbildung 3.4 : Experiment-Topologie auf Szenario 3

In der Tat wird es gewünscht, die Beobachtung von Data Link Layer zu erreichen. Weil LTest auf Anwendungsschicht läuft und es teilweise Kabelverbindung gibt, wie in Abbildung 3.5 dargestellt, vor der Messung ist es notwendig, zu

versichern, ob diese Faktoren die *CSMA/CA* Charakter beeinflussen oder nicht.

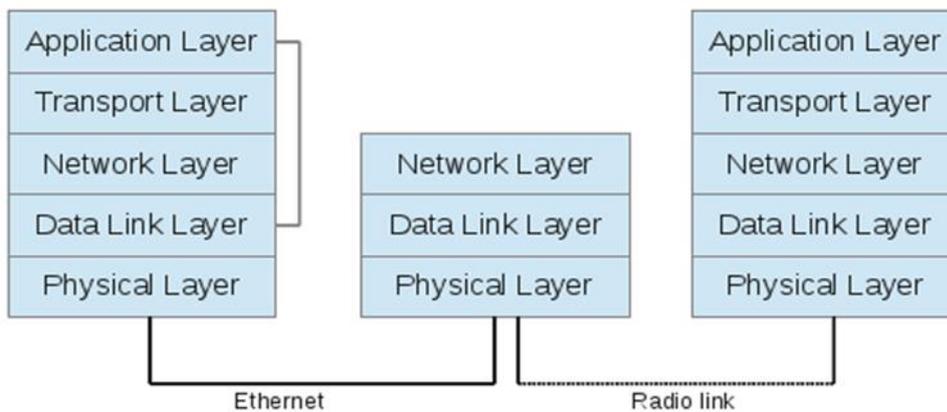


Abbildung 3.5 : Netzwerk-Schicht Topologie der Experimente

Wegen des Backoffs vor jeder Übertragung ist die Verzögerung von *CSMA/CA* normalerweise relativ hoch und verändert sehr oft mit einer großen Standardabweichung. Für drahtgebundene Verbindung *CSMA/CD* ist benutzt, das Backoff wird nur dann eingesetzt, wenn die Kollision aufgetreten ist, aber in dieser idealen Testumgebung ist die Kollision kaum aufzutreten. Die Verzögerung von *CSMA/CD* gegenüber *CSMA/CA* ist in einer niedrigeren Höhe und stabiler. Das wurde auch durch Experimente bewiesen.

Die Experiment-Topologie für *CSMA/CD* wird in Abbildung 3.6 dargestellt, Rechner 1 und Rechner 2 werden über zwei 3-Meter-Kabel mit Access Point verbunden. Rechner 1, wo LTest-Client läuft, sendet Datenpakete, deren Größe 1024 Byte ist, mit einer konstanten Bitrate von 5 Mbps mit *UDP* über den Access Point zum Rechner2.

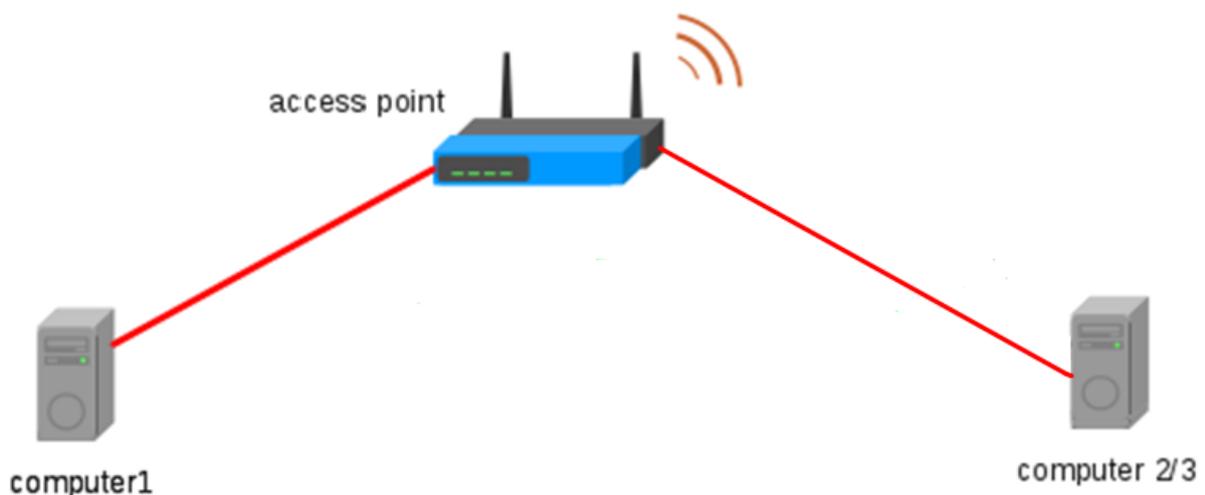


Abbildung 3.6 : Experiment-Topologie für *CSMA/CD* Verbindung

Der Mittelwert der *OWD* von etwa 200 μ s zwischen zwei Rechnern ist von LTest gemessen worden. Ein anderer Test wurde auf Szenario 1 gemacht. Rechner 1 sendet Datenpakete, deren Größe 1024 Byte ist, ebenfalls mit einer konstanten Bitrate von 5 Mbps mit *UDP* über den Access Point zum Rechner2. Eine

Zeitverzögerung von ungefähr 10 ms wurde hier bekommen. Um die Ergebnisse visuell darzustellen, wurden die von den oben durchgeführten Testen gespeicherten Daten geplottet.

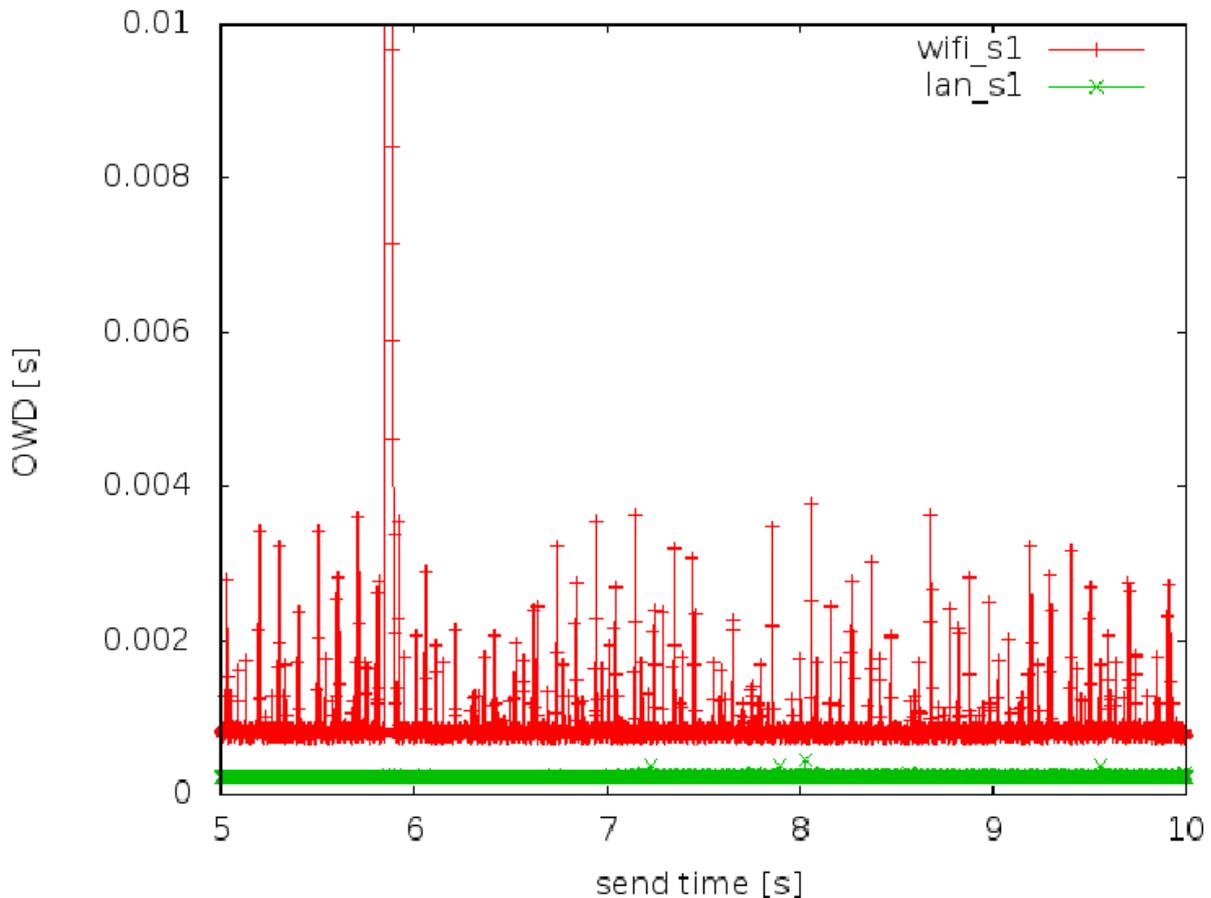


Abbildung 3.7 : OWDs auf CSMA/CA- und CSMA/CD-Verbindung

Abbildung 3.7 zeigt die OWDs abhängig von der Sendezeit. Die Grüne Linie stellt die OWDs in CDMA/CD Netzwerken dar. Im Vergleich zur Roten Linie in Szenario 1 ist es deutlich erkannt, dass die OWDs von Datenpaketen in CDMA/CD Netzwerken relativ klein sind und sich fast nicht verändern. Dadurch ist es offensichtlich, dass diese kurze Strecke CDMA/CD wird die Eigenschaft von CSMA/CA nicht so viel verändern. Gleichzeitig könnte die 10 us Verzögerung von Netzwerk- und Anwendungsschicht auch nicht merkbare Beeinflussungen bringen. Diese Faktoren könnten weggelassen werden, wenn die Eigenschaften von CSMA/CA untersucht werden.

Die sehr wichtige und dringende Sache ist, zu beweisen, dass CSMA/CA zum TCP Retransmission führen könnte. Der Test wurde auf Szenario 2 übernommen. Wie vorher erwähnt, TCP ist nicht geeignet für die Beobachtung von CSMA/CA Mechanism. Deshalb werden eine TCP- und eine UDP-Verbindung jeweils nacheinander auf derselben Link-Schnittstelle mit verschiedenen Porten erstellt. Gleichzeitig wird Wireshark auf Rechner1 gestartet. Die Verbindungen werden wie in Abbildung 3.8 aufgebaut.

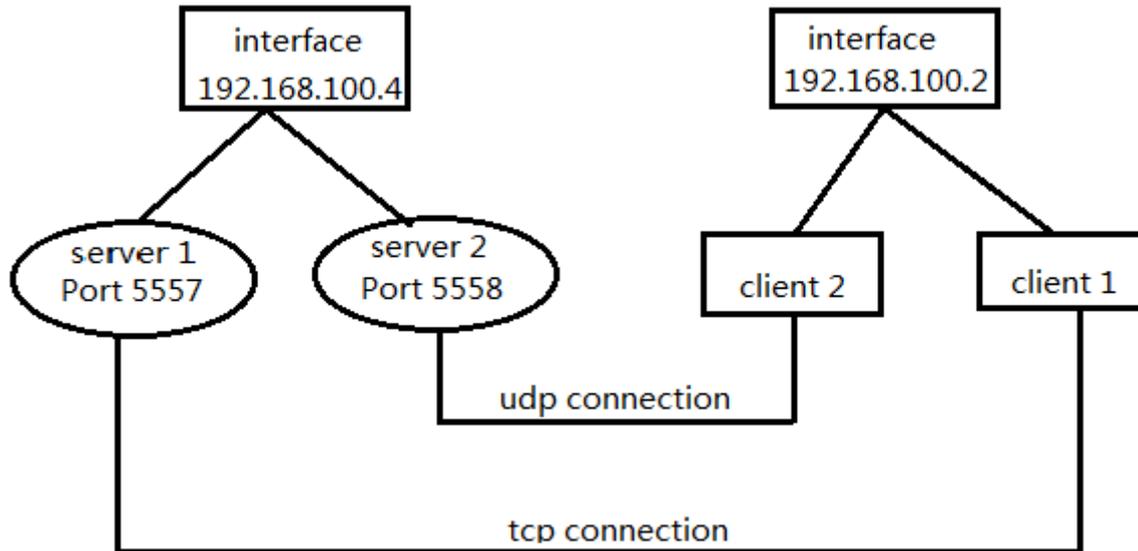


Abbildung 3.8 : Verbindungsaufbau der Experimente

Beide LTest-Client senden 1024-Byte-Pakete innerhalb von 3 Minuten mit einer konstanten Bitrate von 5 Mbps, damit ergibt sich ein Inter-Packet-Time von 1,638 ms. In Wireshark werden die erneut gesendeten *TCP*-Pakete ausgefiltert, dann die genauen Zeitpunkte von verlorenen *TCP*-Paketen werden festgelegt. Durch die Beobachtung der *OWDs* von *UDP*-Pakete ist es zu bestimmen, ob die zufällige große Verzögerung wegen *CSMA/CA* zum *TCP* Retransmission führt.

Im nächsten Schritt wird auf den oben genannten Szenarios die Beziehung von *OWD* und Paketverlustwahrscheinlichkeit untersucht. Das Experiment wurde auf jedem Szenario bestehend aus 20 aufeinander folgenden Testen durchgeführt. Für jeden Test wird zuerst LTest-Server auf Rechner 2 oder 3 gestartet. Dann sendet Rechner 1, wo LTest-Client läuft, Datenpakete, deren Größe 1024 Byte ist, mit einer konstanten Bitrate von 5 Mbps mit *UDP* über den Access Point zum Empfänger. Die Sitzungszeit für jeden Test liegt bei 5 Minuten. Als Testergebnisse werden Sendezeit, Empfangszeit, *OWD* für jedes Packet gespeichert. Dann ist die kumulative Verteilungsfunktion zu bekommen.

4 Messergebnisse und Auswertung

Oben wurde schon besprochen dass *UDP*-Verbindung förderlich ist, die Eigenschaft von *CSMA/CA* zu betrachten. Hier wird es gezeigt, wie *CSMA/CA* sich auf *UDP*-Verbindung verhält. Um die Eigenschaft von *CSMA/CA* darzustellen, wird das Ergebnis von einer zufälligen Messung auf Szenario 1 genommen. Mittels Sendezeit und *OWD* ist die visuelle Darstellung zu bekommen.

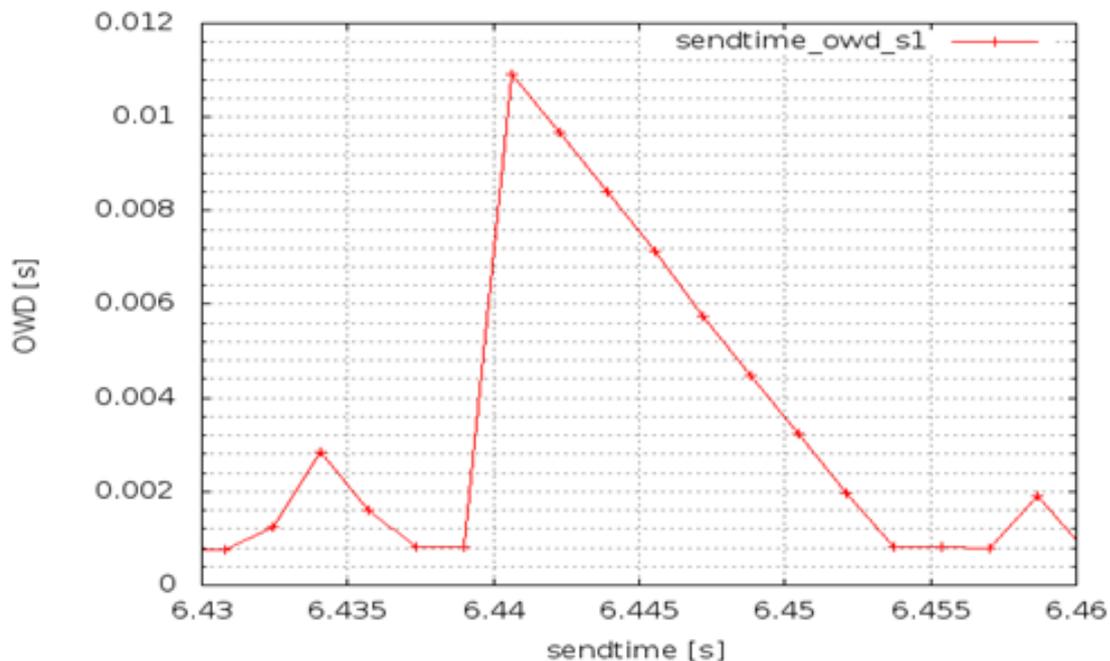


Abbildung 4.1 : Spitzenwerte der *OWD*

In Abbildung 4.1 ergibt sich ein Spitzenwert von *OWD*. In *CSMA/CA* Netzwerken wartet der Sender vor jeder Übertragung eine zufällige Zeitperiode, um Kollisionen zu vermeiden. In der ganzen Übertragungsphase kommen viele Spitzenwerte wie in Abbildung 4.1 dargestellt vor. Der Sender sendet Pakete mit einem konstanten Bitrate von 5Mbps. Deshalb ist die Interpaketzeit zu bekommen mit Gleichung 1.

$$\text{Interpaketzeit} = \frac{P_{gr}}{DR} = \frac{1000 \text{ ms}}{5000000 \text{ bit}} * \frac{1024 * 8 \text{ bit}}{1 \text{ paket}} = 1,638 \frac{\text{ms}}{\text{paket}}$$

Gleichung 1

P_{gr} : Paketgröße

DR : Datenrate

Nach ungefähr 6.44 s ein Paket ist um ungefähr 10.9 ms verzögert wegen Backoffs in *CSMA/CA*, werden die anderen 7 Pakete im Puffer gespeichert. Die Menge der gepufferten Pakete kann mit Gleichung 2 bekommen werden.

$$n = \frac{OWD_{sp}}{T_{int}} = \frac{10.9 \text{ ms}}{1,638 \text{ ms/paket}} = 7 \text{ Paket}$$

Gleichung 2

OWD_{sp} : Spitzenwert der Ein-Weg-Verzögerung

T_{int} : Interpaketzeit

Dann sind die im Puffer stehenden Pakete mit einer gleichen Rate abgebaut, deshalb fallen die *OWDs* von den 7 folgenden Paketen ab. Da die Empfangszeit von jedem Paket und Paketgröße bekannt sind, kann die Link-Rate mit Gleichung 3 auch berechnet werden.

$$LR = \frac{n * P_{gr}}{TR_n - TR_o} = \frac{7 * 1024 \text{ byte} * 8 \text{ bit/byte}}{(1384261661,868814043 - 1384261661,866313180)_s} = 22,93 \text{ Mbps}$$

Gleichung 3

LR: Link-Rate

TR_n : Empfangszeit vom letzten im puffer stehenden Paket

TR_o : Empfangszeit vom ersten im puffer stehenden Paket

n: Anzahl der im Puffer stehenden Pakete minus 1

P_{gr} : Paketgröße

Das Ergebnis von 22,93 Mbps nimmt ungefähr 45% von der maximalen Link-Rate. Das hat auch bewiesen was vorher erzählt wurde, 802.11g hat theoretisch die Link-Rate von 54 Mbps, praktisch kann nur bis oder ein bisschen mehr 40% erreichen.

Bisher wurde es gezeigt, wie *CSMA/CA* auf die *UDP*-Übertragung wirkt. Es ist schon bekannt, dass *TCP* ungeeignet für die Betrachtung von *CSMA/CA*-Eigenschaft ist. Wenn die Pakete wirklich verloren gegangen sind, wird das *TCP*-Retransmission getriggert. Dann könnte die Eigenschaft von *CSMA/CA* nicht richtig reflektiert werden. Deshalb wird hier die Eigenschaft von *CSMA/CA* auf *UDP*-Verbindung untersucht. Im folgend wird die *CSMA/CA* auf *UDP*- und *TCP*-Verbindung in Abbildung 4.2 gezeigt. Die grüne und rote Linie sind jeweils die *TCP*- und *UDP*-Verbindung. Der Spitzenwert der *OWD* um 92,96 Sekunden auf *TCP*-Verbindung ist auffällig. Dieser Spitzenwert ist viel größer als der auf *UDP*-Verbindung, aber die anderen *OWDs* auf den beiden Verbindungen sind fast gleich. Diese Erscheinung hat eine Vermutung aufgerufen, dass *CSMA/CA* zum *TCP*-Retransmission um 92,96 Sekunden geführt hätte, deshalb die Ein-Weg-Verzögerung um diesen Zeitpunkt ist viel größer als die auf *UDP*-Verbindung. Wenn diese Annahme gültig ist, es ist sinnvoll, die *CSMA/CA*-Eigenschaft auf *UDP*-Verbindung zu untersuchen. Durch die Analysierung von *OWDs* könnten auch *TCP*-Retransmission vermieden werden.

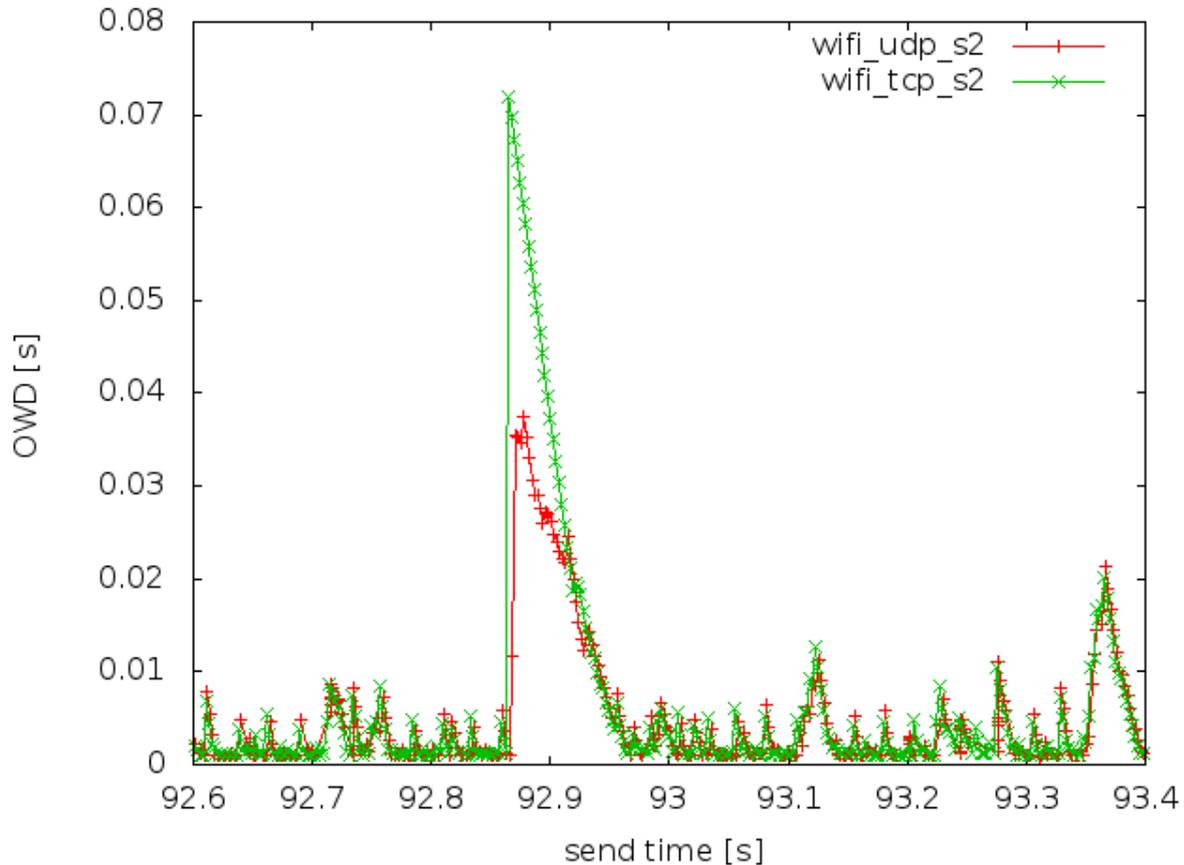


Abbildung 4.2 : OWDs auf TCP- und UDP-Verbindung

4.1 das Zugriffsverfahren CSMA/CA zur schlechten TCP-Performance

Anschließend wird versucht, nachzuweisen, dass CSMA/CA zum TCP-Retransmission geführt hat. Bei der oben schon vorgestellten Topologie hat UDP zwei Pakete verloren, aber TCP hat sechsmal Retransmissions durchgeführt. UDP hat kein Timeout, das heißt, diese zwei Pakete sind wirklich verloren gegangen. Wireshark hat folgende Informationen in Abbildung 4.3 notiert.

```
"No.", "Time", "Source", "Destination", "Protocol", "Info"
"14196", "1385387012.917094000", "192.168.100.2", "192.168.100.3", "TCP", "[TCP Retransmission]"
"31245", "1385387028.990114000", "192.168.100.2", "192.168.100.3", "TCP", "[TCP Retransmission]"
"33557", "1385387031.104234000", "192.168.100.2", "192.168.100.3", "TCP", "[TCP Retransmission]"
"78067", "1385387072.727116000", "192.168.100.2", "192.168.100.3", "TCP", "[TCP Retransmission]"
"101623", "1385387095.029089000", "192.168.100.2", "192.168.100.3", "TCP", "[TCP Retransmission]"
"101625", "1385387123.192973016", "192.168.100.2", "192.168.100.3", "TCP", "[TCP Retransmission]"
```

Abbildung 4.3 : TCP-Retransmissions in Wireshark

Die um 1385387028,927832025 s und 1385387123,137368025 s gesendeten Paketen auf UDP-Verbindung sind verloren gegangen. Das erste Paket auf UDP-

Verbindung wurde um 1385387002,105553112 s gesendet. Unten wird die Sendezeit in Sekunden relativ zu der des ersten Pakets gezeigt.

Sendezeit von <i>TCP</i> -Retransmission, [s]	Sendezeit von <i>UDP</i> -Paketverlust [s]
10,8115627766	
26,8845827579	26,822278913
28,9987027645	
70,6215848923	
92,9235577583	
121,087419904	121,031814913

Tabelle 4.1 : Sendezeit von *TCP*-Retransmission und *UDP*-Paketverlust

Der Tabelle 4.1 ist es zu entnehmen, dass um 26,8 s das *TCP*-Retransmission und der *UDP*-Paketverlust fast gleichzeitig vorkommt. Dann um 121 s trifft diese Erscheinung wieder auf. Bei dem gleichen Medium und unter der gleichen Bedingung übertragen *TCP* und *UDP* gleichzeitig Pakete. Es ist nicht schwer zu beschließen, dass der Paketverlust zum *TCP*-Retransmission geführt hat. Daher ist es bekannt, dass *TCP* viermal Retransmissions von den anderen Ursachen außerhalb Paketverlust verursacht worden sind. Um die Ursache zu finden, werden die vier Retransmissions näher betrachtet. Im Folgenden werden die *OWDs* auf *TCP*- und *UDP*-Verbindung in der Nähe von 10,8115627766 dargestellt.

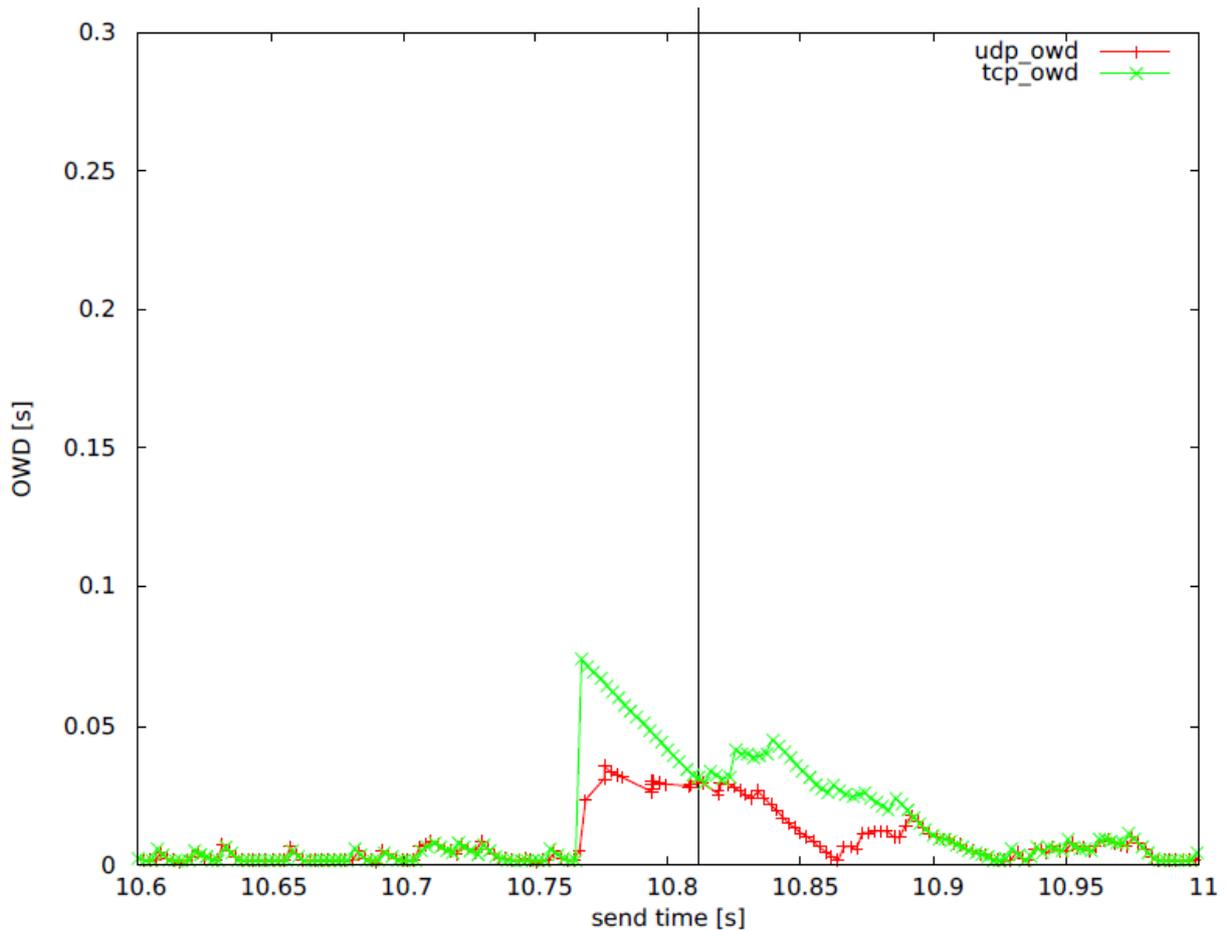


Abbildung 4.4 : OWDs auf TCP- und UDP-Verbindung

In Abbildung 4.4 werden die OWDs in der Nähe von TCP-Retransmission auf TCP- und UDP-Verbindung dargestellt. Von 10.6 S bis 10.75 S sind die Ein-Weg-Verzögerungen auf TCP- und UDP-Verbindung fast identisch. Um 10,77 S erscheinen sich die Spitzenwerte der OWD auf TCP- und UDP-Verbindung. Es ist auch auffällig, dass der Spitzenwert der OWD auf TCP-Verbindung deutlich größer als der auf UDP-Verbindung. Neben den Spitzenwerten der OWD kommt das TCP-Retransmission vor. Nach ungefähr 10.9 S sind die OWDs auf den beiden Verbindungen wieder gleich. Anschließend werden die OWDs auf TCP- und UDP-Verbindung in der Nähe von 28,9987027645 S dargestellt. Wenn die Ergebnisse den obigen entsprechen, wird die Annahme mehr überzeugt.

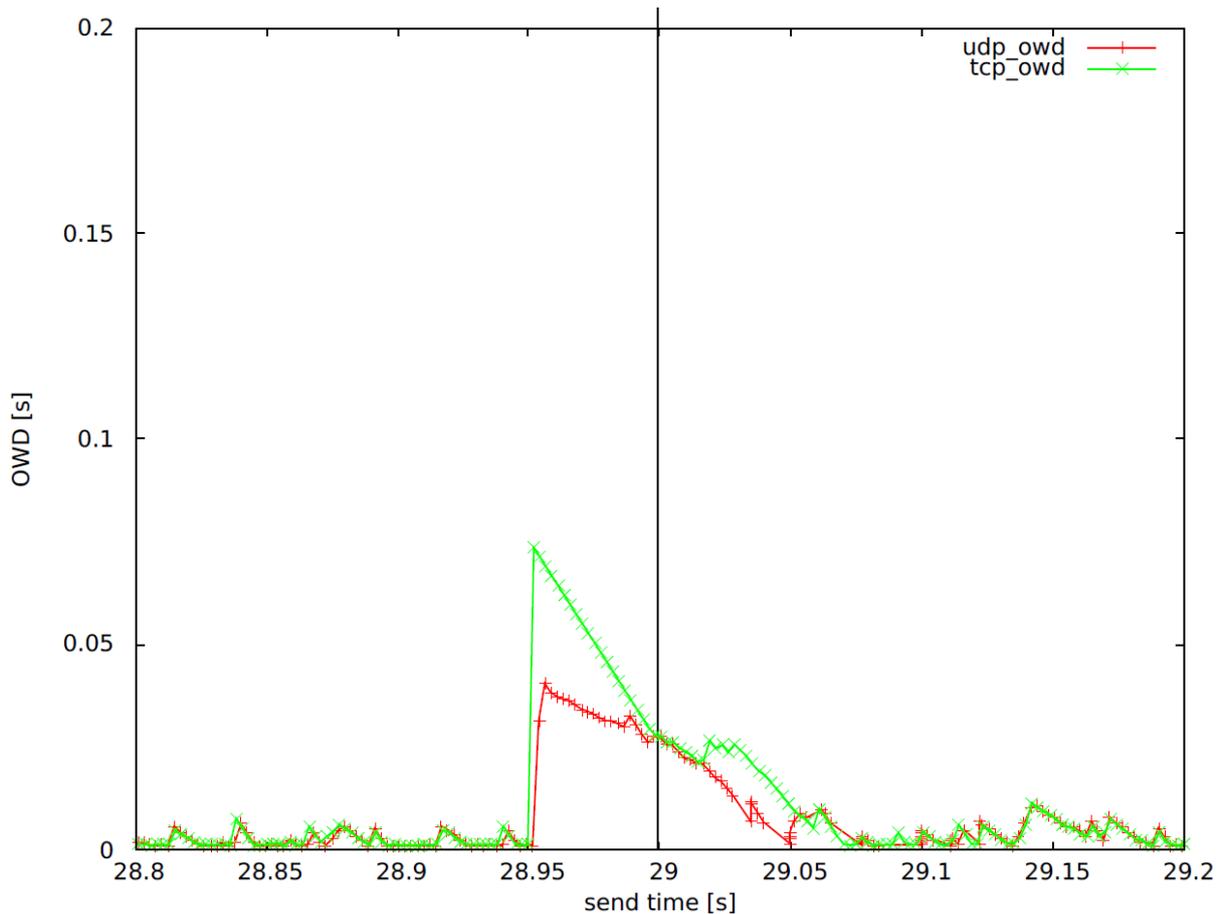


Abbildung 4.5 : OWDs auf TCP- und UDP-Verbindung

In Abbildung 4.5 werden die *OWD* auf der *TCP*- und *UDP*-Verbindungen in der Nähe von 28,9987027645 S gezeigt. Innerhalb der Zeitperiode von 28.8 S bis 28.95 S sind die *OWD*s auf der *TCP*- und *UDP*-Verbindung fast gleich. Kurz nach 28.95 S treten die Spitzenwerte der *OWD* auf der *TCP*- und *UDP*-Verbindung auf. Ebenfalls ist der Spitzenwert der *OWD* auf *TCP*-Verbindung viel größer als der auf *UDP*-Verbindung. Neben den Spitzenwerten wurde das *TCP*-Retransmission getriggert. Nach 29.05 S sind die *OWD*s auf der *TCP*- und *UDP*-Verbindung wieder identisch.

Von oben gezeigten Bildern, ist es nicht schwer zu erkennen, wenn kein Spitzenwert vorkommt, sind die *OWD*s auf *TCP*- und *UDP*-Verbindung fast gleich. Dann werden die *OWD*s auf *TCP*- und *UDP*-Verbindung in der Nähe von 70,6215848923 S dargestellt.

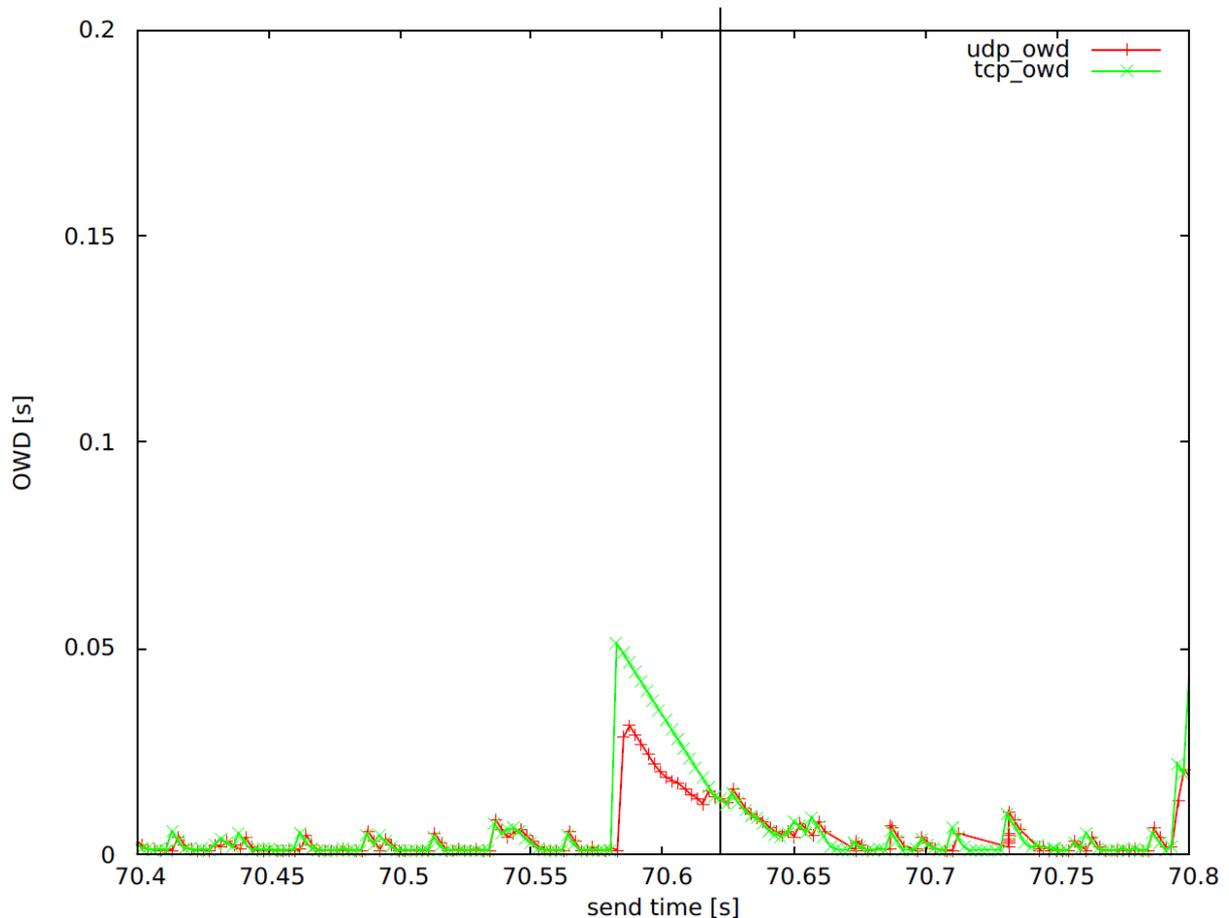


Abbildung 4.6 : OWDs auf TCP- und UDP-Verbindung

Die OWDs auf der TCP- und UDP-Verbindung werden in der Nähe von 70,6215848923 S in Abbildung 4.6 dargestellt. In dem Zeitraum von 70,4 S bis 70,58 S sind die OWDs auf der TCP- und UDP-Verbindung verändern sich gleich. Nach 70,58 S treten die Spitzenwerte der OWDs auf den beiden Verbindungen auf. Kurz nach dem Zeitpunkt, wann die Spitzenwerte der OWDs sich erschienen, wurde das TCP-Retransmission getriggert. Der Spitzenwert der OWD auf TCP-Verbindung ist nicht überraschend viel größer als der auf UDP-Verbindung. Nach 70,63 S verändern sich dann die OWDs auf beiden Verbindungen wieder im gleichen Schritt. Bisher haben drei von den vier Retransmissions die Annahme überzeugt, dass der Spitzenwert der OWD zu TCP Retransmission führen könnte. Unten werden OWDs in der Nähe von dem letzten TCP Retransmission dargestellt.

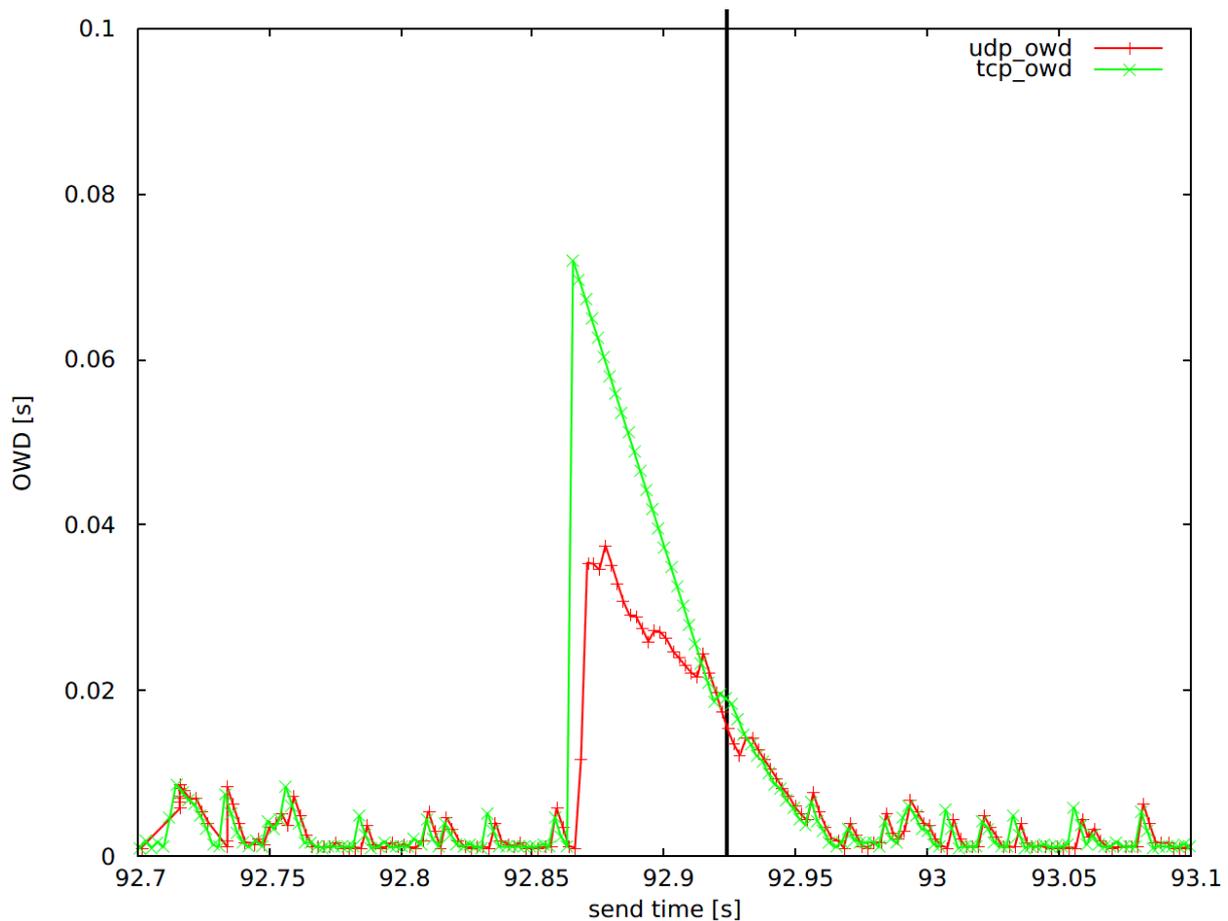


Abbildung 4.7 : OWDs auf TCP- und UDP-Verbindung

Am Ende werden die OWDs auf der TCP- und UDP-Verbindung in der Nähe von 92,9235577583 S in Abbildung 4.7 vorgestellt. Innerhalb des Zeitintervalls von 92,7 S bis 92,85 S sind die Schwankung der OWDs auf TCP-Verbindung und die auf UDP-Verbindung fast gleich. Nach 92,95 S erschienen sich die Spitzenwerte der OWDs auf TCP- und UDP-Verbindung. Kurz nach dem Auftritt der Spitzenwerte der OWD wurden TCP-Pakete erneut gesendet. Wie erwartet, der Spitzenwert der OWD auf TCP-Verbindung viel größer als der auf UDP-Verbindung. Nach 92,4 S sind die Veränderung der OWDs auf TCP-Verbindung und die auf UDP-Verbindung wieder gleich.

In den oben gezeigten Bildern, ist es nicht schwer zu erkennen, wenn kein Spitzenwert der OWD vorkommt, sind die Schwankung der OWDs auf TCP-Verbindung und die auf UDP-Verbindung fast identisch. Jedes Mal wurde das TCP-Retransmission kurz nach dem Vorkommen des Spitzenwertes der OWD getriggert. Dann ist es zu bekommen, dass die Überschreitung von TCP-RTO zum TCP-Retransmission geführt hat. Auf TCP-Verbindung wird dann die Datenrate verringert und das Paket wieder erneut gesendet. Es ist auch angemessen, dass der Spitzenwert der OWD auf TCP-Verbindung viel größer als der auf UDP-Verbindung ist. Nach einem Intervall wird dann die Datenrate wieder erholt, deshalb sind die Veränderung der OWD auf TCP-Verbindung und die auf UDP-Verbindung nach einer Zeitperiode wieder gleich. Bisher ist es schon erkannt,

dass der Spitzenwert der *OWD* zu *TCP*-Retransmission führen könnte. Oben schon erwähnt, dass der Spitzenwert der *OWD* von dem Backoff-Mechanism, welches eine Kernkomponente von *CSMA/CA* zur Kollisionsvermeidung ist, verursacht wird. Bisher wird es bewiesen, dass *CSMA/CA* zu *TCP*-Retransmission führen kann. Das Result des Experiments hat sechs *TCP*-Retransmissions gezeigt. Nach jedem *TCP*-Retransmission wird die Datenrate verkleinert. Die könnten schon zur heftigen Schädigung der *TCP*-Performance auf Funknetz führen. Vier Retransmissions von den sechs wurde von der Überschreitung des *TCP*-Retransmission-Timeouts verursacht. Daher ist es sinnvoll, die Eigenschaft von *CSMA/CA* zu untersuchen.

4.2 Die kumulative Verteilung der *OWDs*

Im Folgend wird die Verteilung der *OWDs* auf 3 Szenarios betrachtet. Anschließend wird die Beziehung von *OWD* und Paketverlustwahrscheinlichkeit untersucht. Das könnte zum Verbessern der *TCP*-Performance beitragen. Die Topologie für die Tests auf 3 Szenarios wurde oben schon beschrieben. Hier werden die Ergebnisse jeweils gezeigt. Zuerst werden die *OWDs* auf Szenario 1 dargestellt.

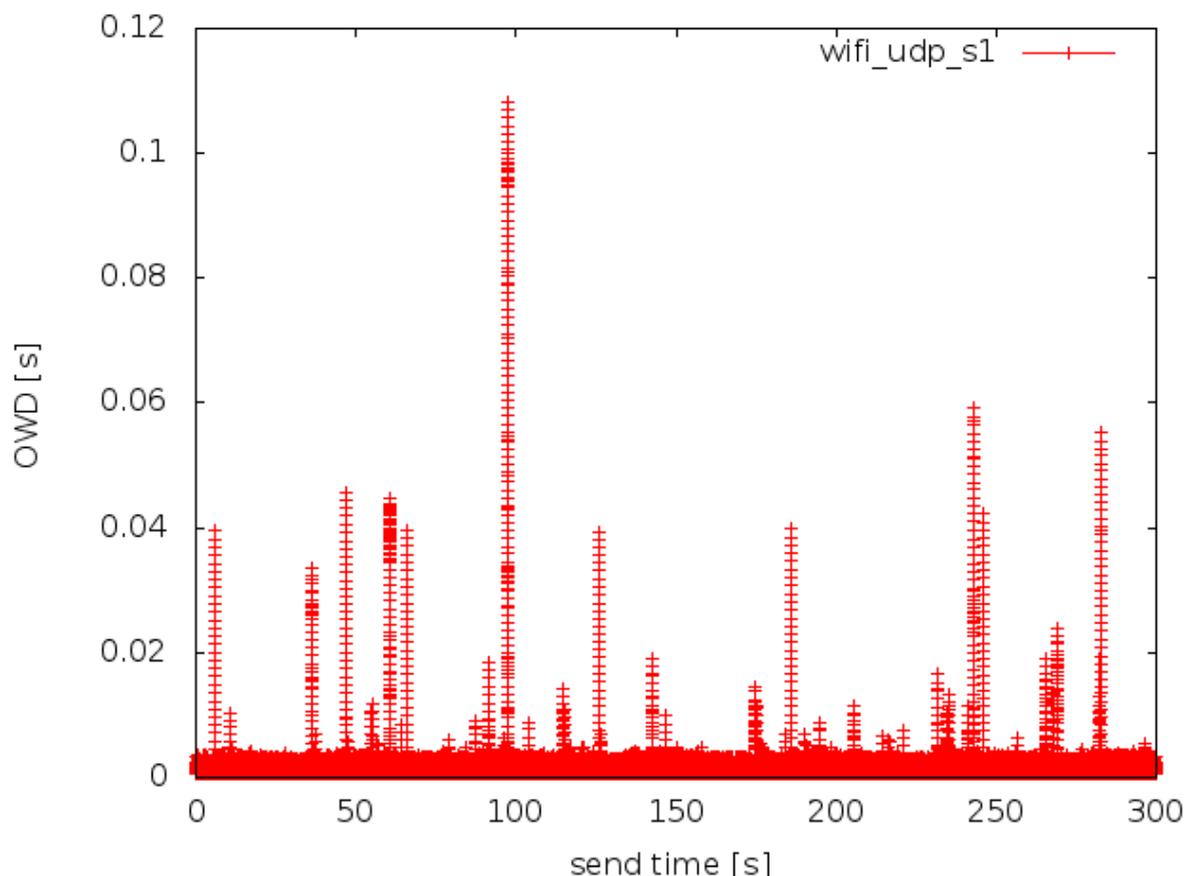


Abbildung 4.8 : *OWDs* der *UDP*-Verbindung auf Szenario 1

In Abbildung 4.8 werden die *OWDs* abhängig von der Sendezeit auf Szenario 1 dargestellt. Der maximale Spitzenwert liegt ungefähr bei 0,11 s. Die Spitzenwerte der *OWD* sind zwar spärlich, aber sehr auffällig. Sie könnten auf *TCP*-Verbindung zum Retransmission führen. Unten werden Paket Anzahl, Paketverlust Prozent und Mittelwert der *OWD* von 20 Messungen auf Szenario 1 gelistet.

Messung Sequenz	Paketverlust Anzahl	Paketverlust Prozent	Mittelwert der <i>OWD</i>
1	0	0%	0,000964 s
2	0	0%	0,000958 s
3	0	0%	0,000894 s
4	0	0%	0,000884 s
5	0	0%	0,000896 s
6	0	0%	0,000890 s
7	0	0%	0,000881 s
8	0	0%	0,000875 s
9	0	0%	0,000886 s
10	0	0%	0,000879 s
11	0	0%	0,000870 s
12	0	0%	0,000873 s
13	0	0%	0,000875 s
14	0	0%	0,000878 s
15	0	0%	0,000882 s
16	0	0%	0,000972 s
17	0	0%	0,000894 s
18	0	0%	0,000902 s
19	0	0%	0,000911 s
20	0	0%	0,000914 s

Tabelle 4.2 : Paketverluste und Mittelwert der *OWDs* auf Szenario 1

In Tabelle 4.2 werden die Ergebnisse von 20 Messungen auf Szenario 1 gezeigt. In den 20 Messungen hat kein Paketverlust aufgetreten. Der Mittelwert der *OWD* liegt zwischen 0,000873 s und 0,000964 s. Die Ergebnisse haben die Übertragungsverhältnisse auf einer guten Linkqualität-Verbindung, dessen Signal-Level sich zwischen -34 dBm und -44 dBm variiert, beschrieben.

Anschließend werden die *OWDs* jeder Messung in einer Datei gesammelt. Mit diesen *OWDs* wird die Eigenschaft von *CSMA/CA* auf Szenario 1 untersucht. Unten wird die komplementäre Verteilungsfunktion dargestellt.

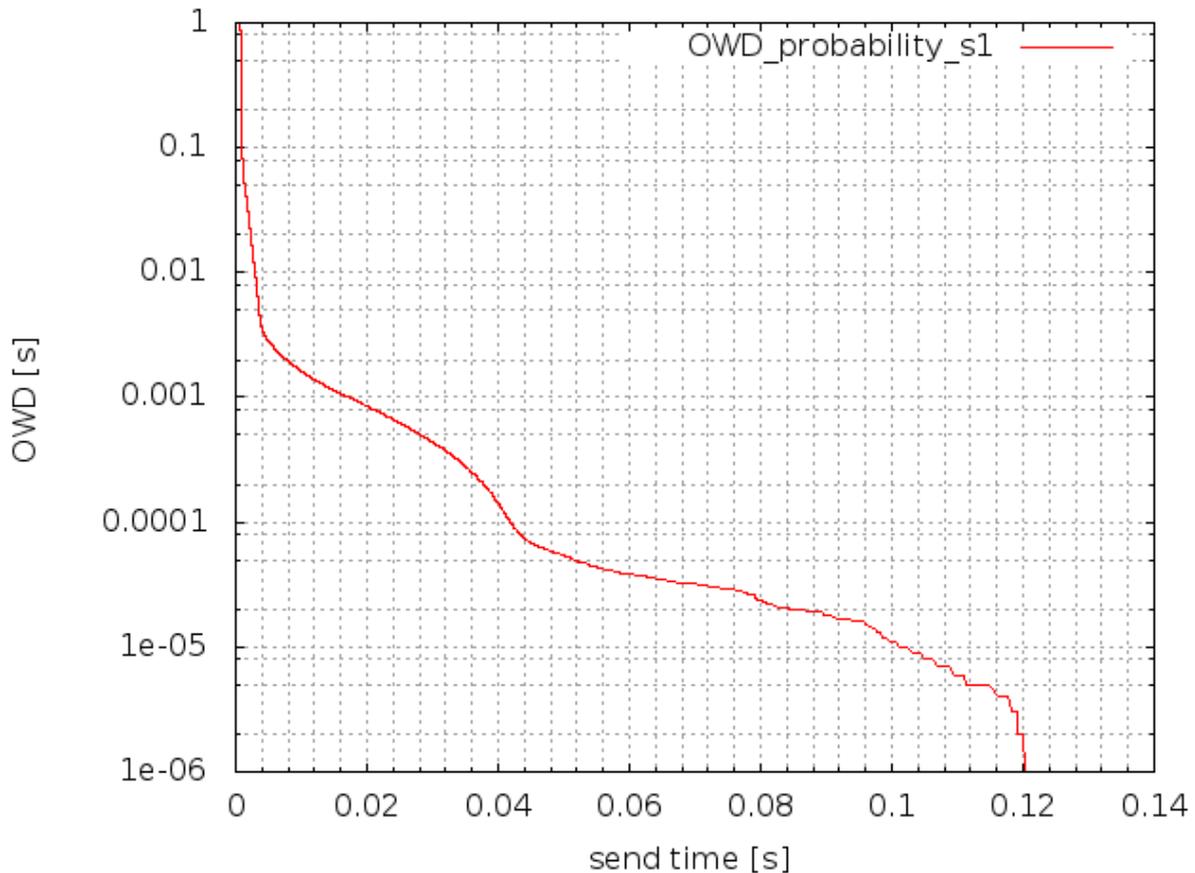


Abbildung 4.9 : die kumulative Verteilung der *OWDs* auf Szenario 1

In Abbildung 4.9 wird die kumulative Verteilung der *OWDs* dargestellt. Mit steigender *OWD* nimmt die Paketverlustwahrscheinlichkeit ab. Von 0 s bis 0,004 s fällt die Kurve sehr schnell ab. Die Paketverlustwahrscheinlichkeit sinkt von 1 auf 0,004375 ab. Die Kurve zwischen 0,004 s und 0,044 s liegt viel flacher. Mit steigender *OWD* nimmt die Paketverlustwahrscheinlichkeit langsam ab. In diesem Intervall fällt die Paketverlustwahrscheinlichkeit von 0,004375 auf 0,000075. Von 0,044 s bis 0,116 s fällt die Kurve noch langsamer ab. Erreicht Paketverlustwahrscheinlichkeit eine Verlustrate von $1e-05$, muss die *OWD* steigt bis 0,1 s. Wenn die *OWD* größer als 0,12 s ist, fällt die Paketverlustwahrscheinlichkeit unter einer extremen kleinen Verlustrate von $1e-06$. Wenn das Verhältnis von *OWD* und Paketverlustwahrscheinlichkeit bekannt ist, könnte das *TCP-RTO* größer als zwei mal *OWD* gesetzt, dann nimmt die Wahrscheinlichkeit, dass *CSMA/CA* zum *TCP*-Retransmission führt, ab. Wird das *TCP-RTO* als 0,24 s gesetzt, fällt die Wahrscheinlichkeit der Paketverluste auf $1e-06$.

Anschließend werden die *OWDs* auf Szenario 2 dargestellt.

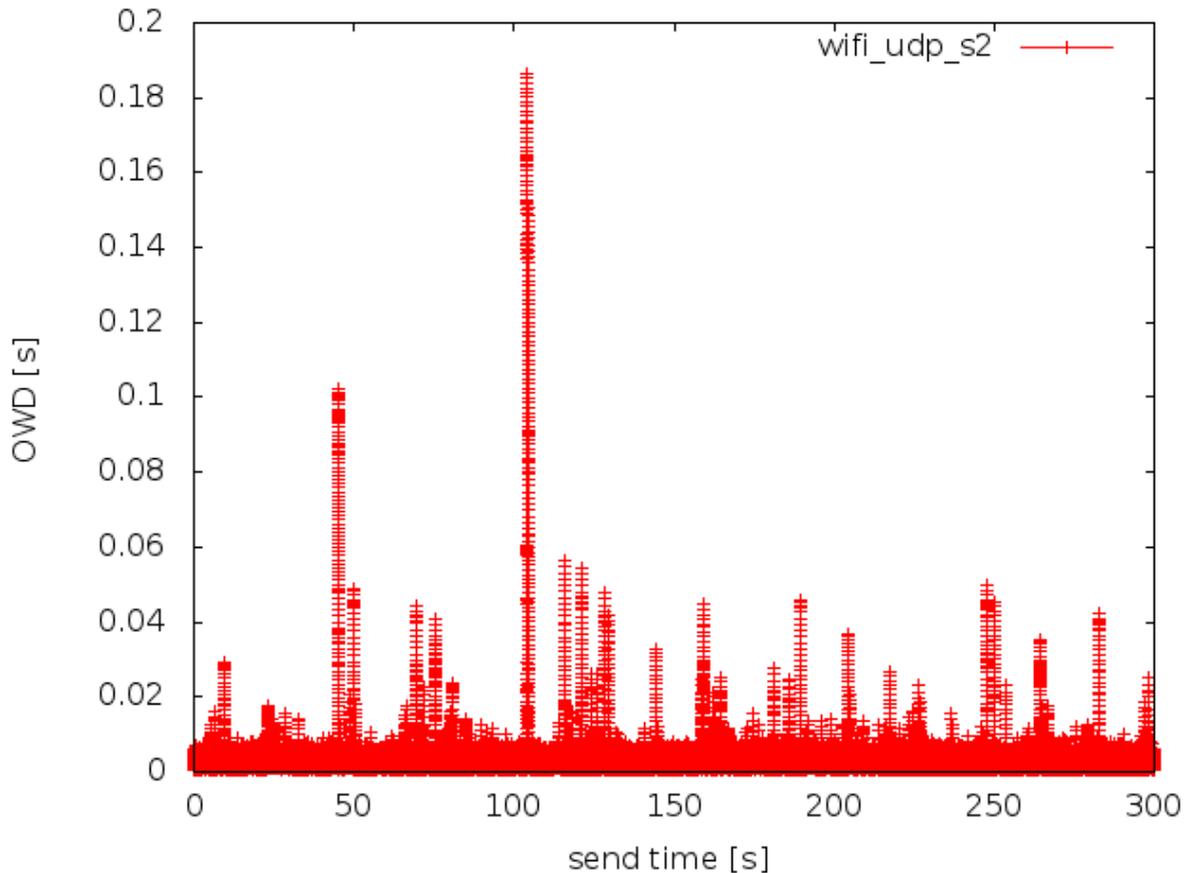


Abbildung 4.10 : OWDs der UDP-Verbindung auf Szenario 2

In Abbildung 4.10 werden die *OWDs* abhängig von Sendezeit auf Szenario 2 dargestellt. Im Vergleich zu Szenario 1 sind die Spitzenwerte deutlich mehr. Außerdem variieren sich die Spitzenwerte in einem größeren Bereich. Der maximale Spitzenwert der *OWD* liegt ungefähr bei 0,19 s. In diesem Fall könnten die Spitzenwerte mehr *TCP*-Retransmissions verursachen. Unten werden Paket Anzahl, Paketverlust Prozent und Mittelwert der *OWD* von 20 Messungen auf Szenario 2 gelistet.

Messung Sequenz	Paketverlust Anzahl	Paketverlust Prozent	Mittelwert der Ein-Weg-Verzögerung
1	6	0,0033%	0,001829 s
2	2	0,0011%	0,001730 s
3	0	0%	0,001575 s
4	0	0%	0,001579 s
5	0	0%	0,001703 s
6	0	0%	0,001611 s

7	0	0%	0,001681 s
8	0	0%	0,001637 s
9	0	0%	0,001550 s
10	0	0%	0,001710 s
11	1	0,0005%	0,001785 s
12	0	0%	0,001684 s
13	1	0,0005%	0,001893 s
14	1	0,0005%	0,001897 s
15	0	0%	0,001698 s
16	1	0,0005%	0,001683 s
17	0	0%	0,001835 s
18	0	0%	0,001768 s
19	1	0,0005%	0,001675 s
20	0	0%	0,000914 s

Tabelle 4.3 : Paketverluste und Mittelwert der *OWDs* auf Szenario 2

In Tabelle 4.3 werden die Paketverluste und der Mittelwert der *OWD* von 20 Messungen auf Szenario 2 dargestellt. Die Paketverluste nehmen maximal einen Anteil von 0,0033%. Die Mittelwerte der *OWD* variieren sich von 0,001550 s bis 0,001829 s. Von diesen Ergebnissen ist auch zu erkennen, dass die Linkqualität auf Szenario 2 deutlich schlechter als auf Szenario 1 ist. Die Mittelwerte der *OWD* auf Szenario 2 sind auch fast einmal größer als auf Szenario 1. Deshalb ist es sinnvoller, die *OWDs* auf Szenario 2 zu betrachten.

Die *OWDs* der 20 Messungen werden in eine Datei hinzugefügt. Dann ist die komplementäre Verteilungsfunktion zu bekommen. Die Komplementäre Verteilungsfunktion wird unten dargestellt.

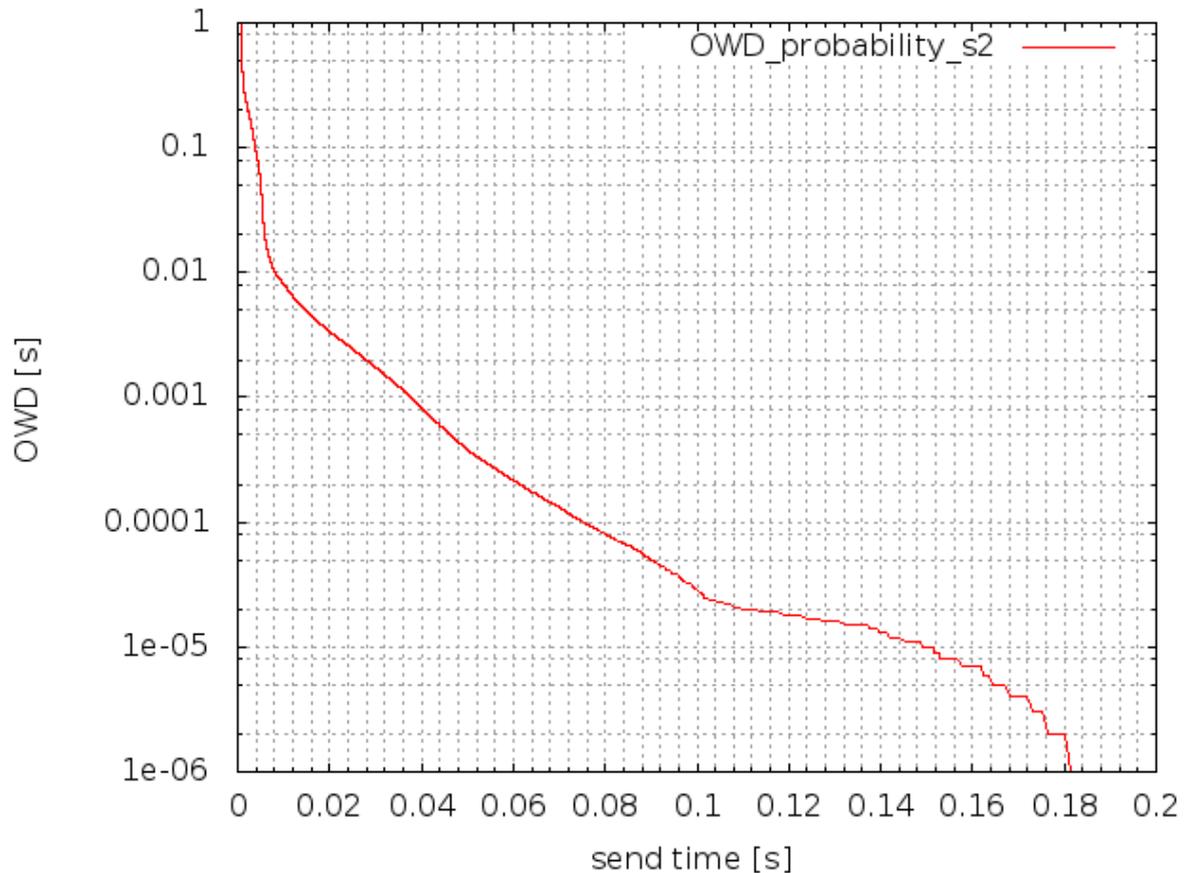


Abbildung 4.11 : die kumulative Verteilung der OWDs auf Szenario 2

In Abbildung 4.11 wird die Paketverlustwahrscheinlichkeit abhängig von der OWD auf Szenario 2 dargestellt. Die Kurve zeigt, dass mit steigender OWD die Paketverlustwahrscheinlichkeit abnimmt. Von 0 s bis 0,008 s fällt die Paketverlustwahrscheinlichkeit extrem schnell ab. Bis 0,008 s fällt die Paketverlustwahrscheinlichkeit auf 1%. Im Intervall von 0,008 s bis 0,104 s fällt die Paketverlustwahrscheinlichkeit langsamer von 1% auf 0,0025%. Von 0,104 s bis 0,164 s liegt die Kurve noch flacher. Paketverlustwahrscheinlichkeit fällt von 0,0025% auf 0,0005%. Erreicht die Paketverlustwahrscheinlichkeit eine Fehlerrate von 10^{-5} , muss die OWD bis 0,152 s steigen. Wenn die OWD größer als 0,181 s ist, dann fällt die Paketverlustwahrscheinlichkeit unter eine Fehlerrate von 10^{-6} . Wenn das TCP-RTO als 0,362 s gesetzt würde, dann liegt die Wahrscheinlichkeit, dass CSMA/CA zum TCP-Retransmission führt, unter 0,0001%.

Am Ende werden die OWDs auf Szenario 3 dargestellt.

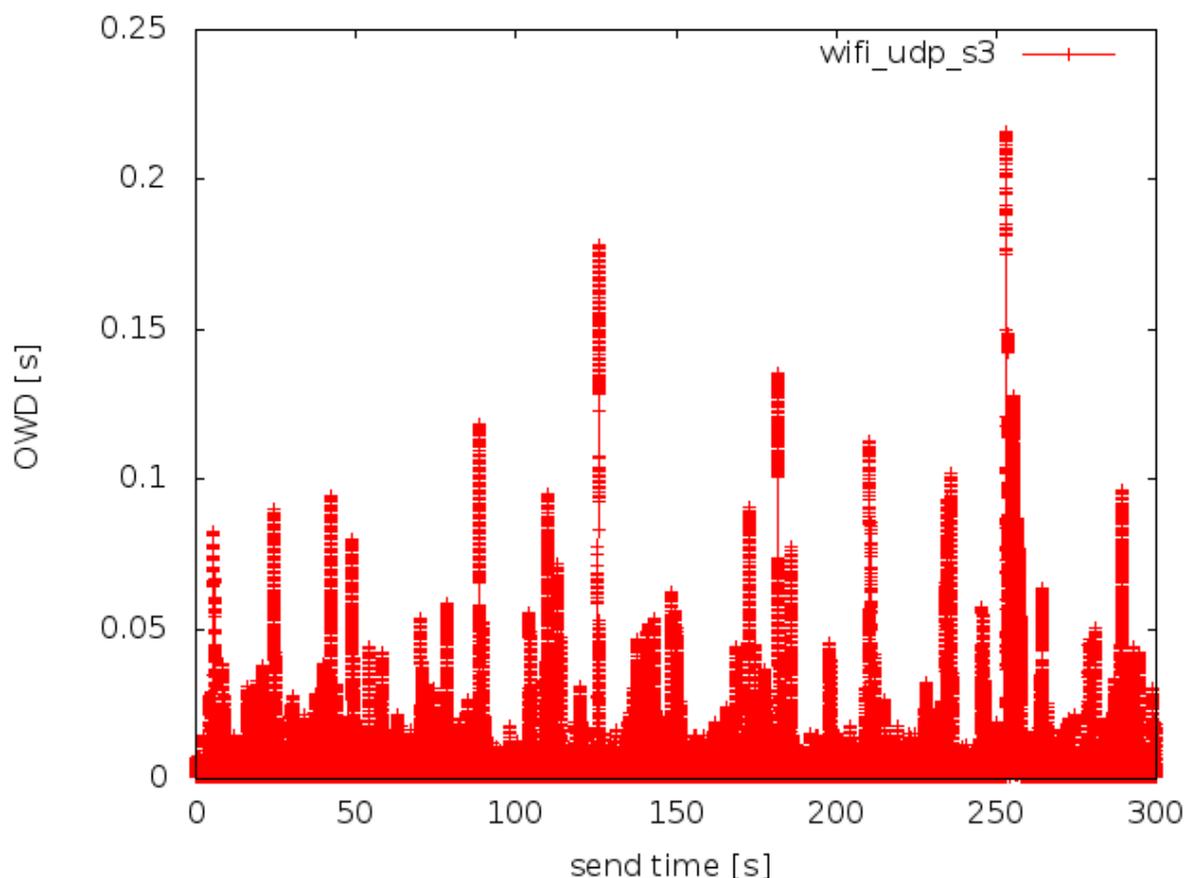


Abbildung 4.12 : OWDs der UDP-Verbindung auf Szenario 3

Die Abbildung 4.12 hat den Verlauf der *OWDs* mit der Sendezeit auf Szenario 3 gezeigt. Der Spitzenwert der *OWD* erreicht 0,22 s. Im Vergleich zum Szenario 1 und Szenario 2 sind die Spitzenwerte der *OWD* deutlich mehr. Die Spitzenwerte variieren sich in einem noch größeren Bereich. In diesem Fall kommt die höchste Wahrscheinlichkeit vor, dass nach dem Ablauf des *TCP-RTO* die erneute Übertragung getriggert wird. Unten werden Paket Anzahl, Paketverlust Prozent und Mittelwert der *OWD* von 20 Messungen auf Szenario 3 gelistet.

Messung Sequenz	Paketverlust Anzahl	Paketverlust Prozent	Mittelwert der Ein-Weg-Verzögerung
1	152	0,0830%	0,004375 s
2	384	0,2097%	0,004785 s
3	264	0,1442%	0,005209 s
4	239	0,1305%	0,002143 s
5	21	0,0115%	0,002275 s
6	156	0,0852%	0,002847 s

7	1507	0,8230%	0,008570 s
8	1503	0,8209%	0,007290 s
9	30	0,0164%	0,002335 s
10	508	0,2774%	0,002617 s
11	1002	0,5472%	0,004078 s
12	330	0,1802%	0,002687 s
13	6079	3,3194%	0,003248 s
14	66	0,0360%	0,002255 s
15	3	0,0002%	0,002100 s
16	105	0,0573%	0,002422 s
17	2	0,0001%	0,002455 s
18	53	0,0289%	0,002420 s
19	14	0,0008%	0,003376 s
20	1	0,0001%	0,002005 s

Tabelle 4.4 : Paketverluste und Mittelwert der *OWDs* auf Szenario 3

In Tabelle 4.4 werden die Paketverluste und der Mittelwert der *OWD* von 20 Messungen auf Szenario 3 dargestellt. Die Paketverluste nehmen maximal einen Anteil von 3,3194% und minimal von 0,0001%. Im Vergleich zu Szenario 1 und 2 treten mehr Paketverluste auf Szenario 3 auf. Die Mittelwerte der *OWD* variieren sich von 0,002005 s bis 0,008570 s. Daher wurde es erkannt, dass neben der Link-Qualität der Cross-Traffic auch eine heftige Beeinflussung auf Messergebnisse hat. Die Mittelwerte der *OWD* auf Szenario 3 sind fast 1 bis 4 mal größer als auf Szenario 2. Das drahtlose Netzwerk besteht mindestens aus mehreren aktiven Stationen, die auch gleichzeitig Daten senden oder empfangen könnten. Deshalb ist es sinnvoller, die *OWDs* auf Szenario 3 zu betrachten.

Die *OWDs* der 20 Messungen auf Szenario 3 werden in eine Datei hinzugefügt. Dann ist die komplementäre Verteilungsfunktion zu bekommen. Die Komplementäre Verteilungsfunktion wird unten grafisch gezeigt.

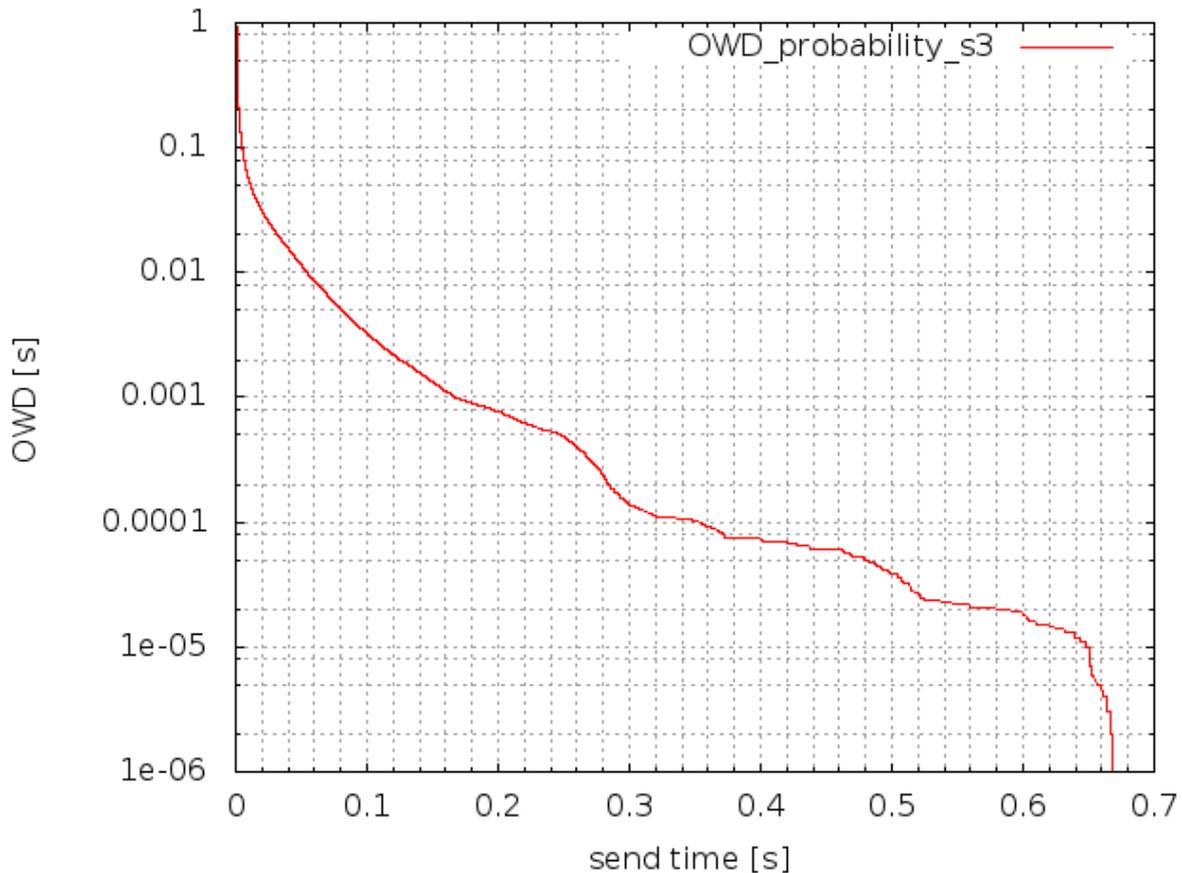


Abbildung 4.13 : die kumulative Verteilung der *OWDs* auf Szenario 3

In Abbildung 4.13 wird die kumulative Verteilung der *OWDs* auf Szenario 3 dargestellt. Die Kurve zeigt, dass mit steigender *OWD* die Paketverlustwahrscheinlichkeit abnimmt. Von 0 s bis 0,01 s fällt die Paketverlustwahrscheinlichkeit extrem schnell ab. Bis 0,01 s fällt die Paketverlustwahrscheinlichkeit auf 5%. Das heißt, die *OWDs* unter 0,01s nehmen 95% der Gesamt. Im Intervall von 0,01 s bis 0,24 s fällt die Paketverlustwahrscheinlichkeit langsamer von 5% auf 0,05%. Von 0,24 s bis 0,32 s liegt die Kurve ein bisschen steiler. Paketverlustwahrscheinlichkeit fällt von 0,05% auf 0,01%. Im Intervall von 0,32 s bis 0,65 s fällt die Paketverlustwahrscheinlichkeit von 0,01% auf 0,001%. Nach 0,65 s fällt die Kurve wieder schneller ab. Erreicht die Paketverlustwahrscheinlichkeit eine Fehlerrate von $1e-05$, muss die *OWD* bis 0,65 s steigen. Wenn die *OWD* größer als 0,67 s ist, dann fällt die Paketverlustwahrscheinlichkeit unter eine Fehlerrate von $1e-06$. Wenn das *TCP-RTO* als 1,34 s gesetzt würde, dann liegt die Wahrscheinlichkeit, dass *CSMA/CA* zum *TCP*-Retransmission führt, unter 0,0001%.

Bisher wurde das Verhältnis zwischen *OWD* und Paketverlustwahrscheinlichkeit auf 3 Szenarios grafisch dargestellt und erklärt. Das könnte für das Verbessern der *TCP*-Performance von großer Bedeutung haben. Oben wurde es erwähnt, dass die Spitzenwerte der *OWD* zur großen Schädigung auf *TCP*-Performance

führen könnten. Deshalb hat die Verteilung der Abstände zwischen den Spitzenwerten der *OWD* das Interesse geweckt. Am Ende wird die Verteilung der Abstände zwischen den Spitzenwerten der *OWD* mit den Ergebnissen auf 3 Szenarios analysiert. Es wird bekannt, wie oft ein Spitze auftritt.

4.3 die kumulative Verteilung der Abstände zwischen den Spitzenwerten der *OWD*

Um die Verteilung der Abstände zwischen den Spitzenwerten der *OWD* grafisch darstellen zu können, müssen die Ergebnisse der 3 Szenarios bearbeitet werden. Die Bearbeitung der Rohdaten umfasst 4 Schritte.

1. Die Spitzenwerte werden in jeder Messung gesucht und gleichzeitig wird die Sendezeit gespeichert.
2. Indem die nächste Sendezeit die vorige subtrahiert, werden die Abstände zwischen den Spitzenwerten bekommen.
3. Die Abstände zwischen den Spitzenwerten von 20 Messungen werden dann gesammelt, sortiert.
4. Die kumulative Wahrscheinlichkeit für den Abstand wird berechnet.

```
#<sequence>/t<distance>/t<probability>
0      0.000000000    0.000092920
1      0.001640081    0.000199777
2      0.001640320    0.000236945
3      0.001641035    0.000274113
4      0.001641750    0.000278759
5      0.001642942    0.000297343
...    ...    ...
```

Abbildung 4.14 : verarbeitende Daten für die kumulative Verteilung der Abstände

In Abbildung 4.14 wird ein Teil der bearbeitenden Ergebnisse gezeigt. Als wichtige Informationen werden die Sequenznummer, der Abstand zwischen den Spitzenwerten und die kumulative Wahrscheinlichkeit für den Abstand dargestellt. Nach der Datenverarbeitung ist die komplementäre kumulative Verteilungsfunktion der Abstände zwischen den Spitzenwerten auf jedem Szenario zu bekommen. Zu dem ersten Schritt wurde ein Programm geschrieben, um die Spitzenwerte zu definieren und filtern. *OWDs* müssen zwei Bedingungen erfüllen, dann könnten sie auch als Spitzenwerte betrachtet werden. Zuerst müssen die einzelnen *OWDs* größer als 60% der Standardabweichung der *OWDs* sein.

```
$stdviation = &stdev(\@value);
```

```
#my $level = 0.6*$stdviation;
```

```
my $temptime = $sndtime[0], $distance = 0;
```

```

for (my $x=0; $x<$len;$x++)
{
  $distance = 0;
  if($value[$x] > $level)
  {
    ...
  }
}

```

Wenn die erste Bedingung erfüllt worden ist, müssen die OWDs größer als ihre Nachbarn.

```

if($value[$x] > $level)
{
  if($x==0)
  {
    $distance = 0;
    printf PEAK ("%5.9f\t%10.9f\n",$value[$x],$distance);
  }
  else
  {
    if(($sndtime[$x] - $temptime) > 0.00164)
    {
      $distance = $sndtime[$x] - $temptime;
      printf PEAK ("%5.9f\t%10.9f\n",$value[$x],$distance);
    }
  }
  $temptime = $sndtime[$x];
}

```

Nach den oben genannten vier Schritten wird die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf 3 Szenarios bekommen. Unten wird zuerst die Kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 1 in Abbildung 4.15 dargestellt.

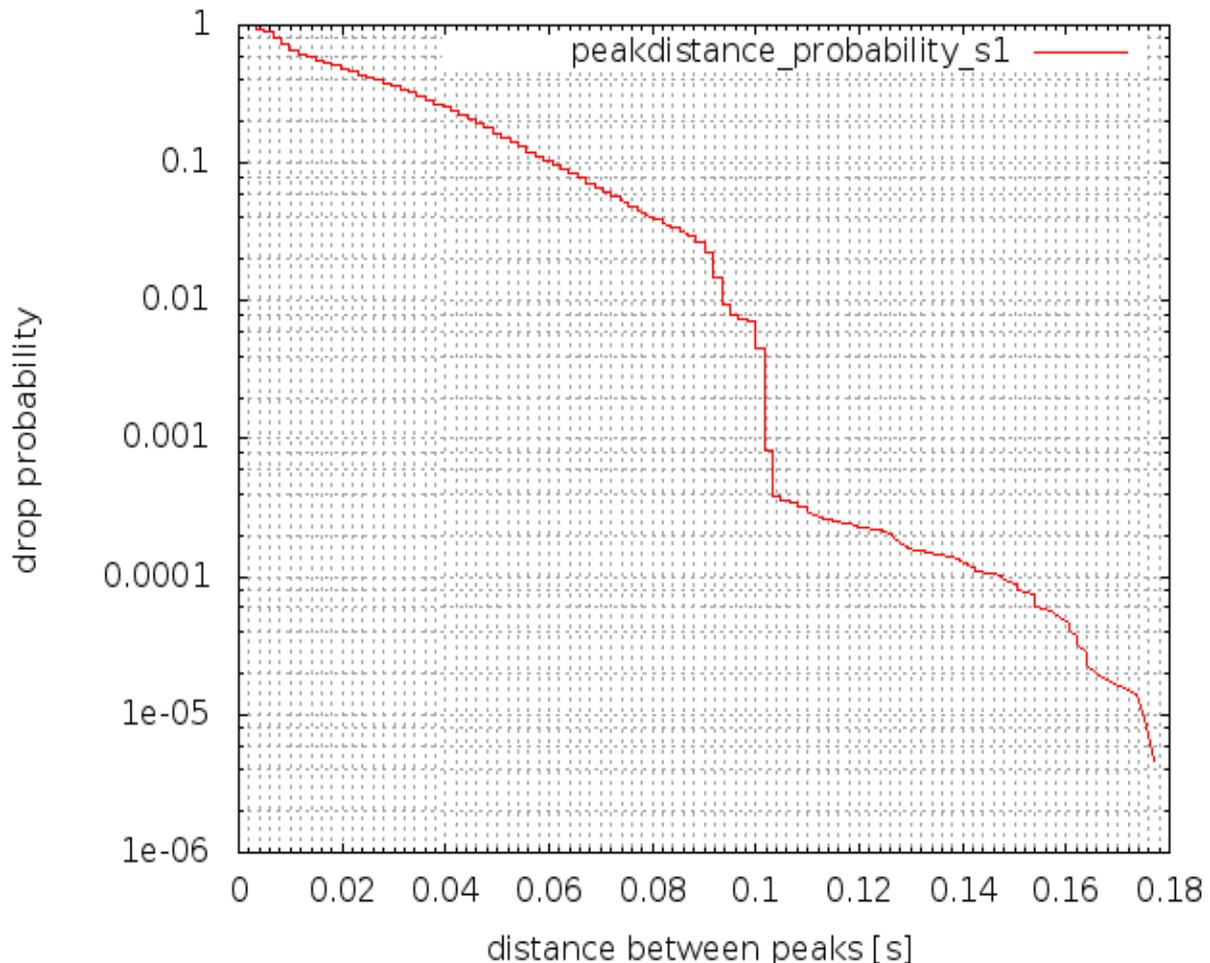


Abbildung 4.15 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 1

Eine Abfallende Kurve erscheint sich in Abbildung 4.15. Mit dem steigendem Abstand nimmt die Wahrscheinlichkeit der Spitzenwertverluste ab. Von 0 s bis 0,088 s fällt die Wahrscheinlichkeit der Spitzenwertverluste fast linear von 1 auf 0,037. Das heißt, die Abstände zwischen den Spitzenwerten unter 0,088 s nehmen 96,3% der Gesamt. Aber die Kurve in diesem Intervall ist nicht so steil, das heißt, die Abstände zwischen den Spitzenwerten verändern sich nicht so schnell. Von 0,088 s bis 0,104 s hat fällt die Kurve fast senkrecht ab. In diesem Intervall sinkt die Wahrscheinlichkeit der Spitzenwertverluste von 3,7% auf 0,046% ab. Die Abstände zwischen den Spitzenwerten nehmen 99,954% der Gesamt. Von der hohen Steilheit ist es auch bekannt, dass die Abstände zwischen den Spitzenwerten sich sehr schnell variieren. Von 0,104 s bis 0,177 s fällt die Wahrscheinlichkeit der Spitzenverlust von 0,046% auf 0,00047%. Von dem Ergebnis ist es bekannt, die Abstände zwischen den Spitzenwerten sich unter 0,104 s konzentrieren.

Anschließend wird die Kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 2 in Abbildung 4.16 dargestellt.

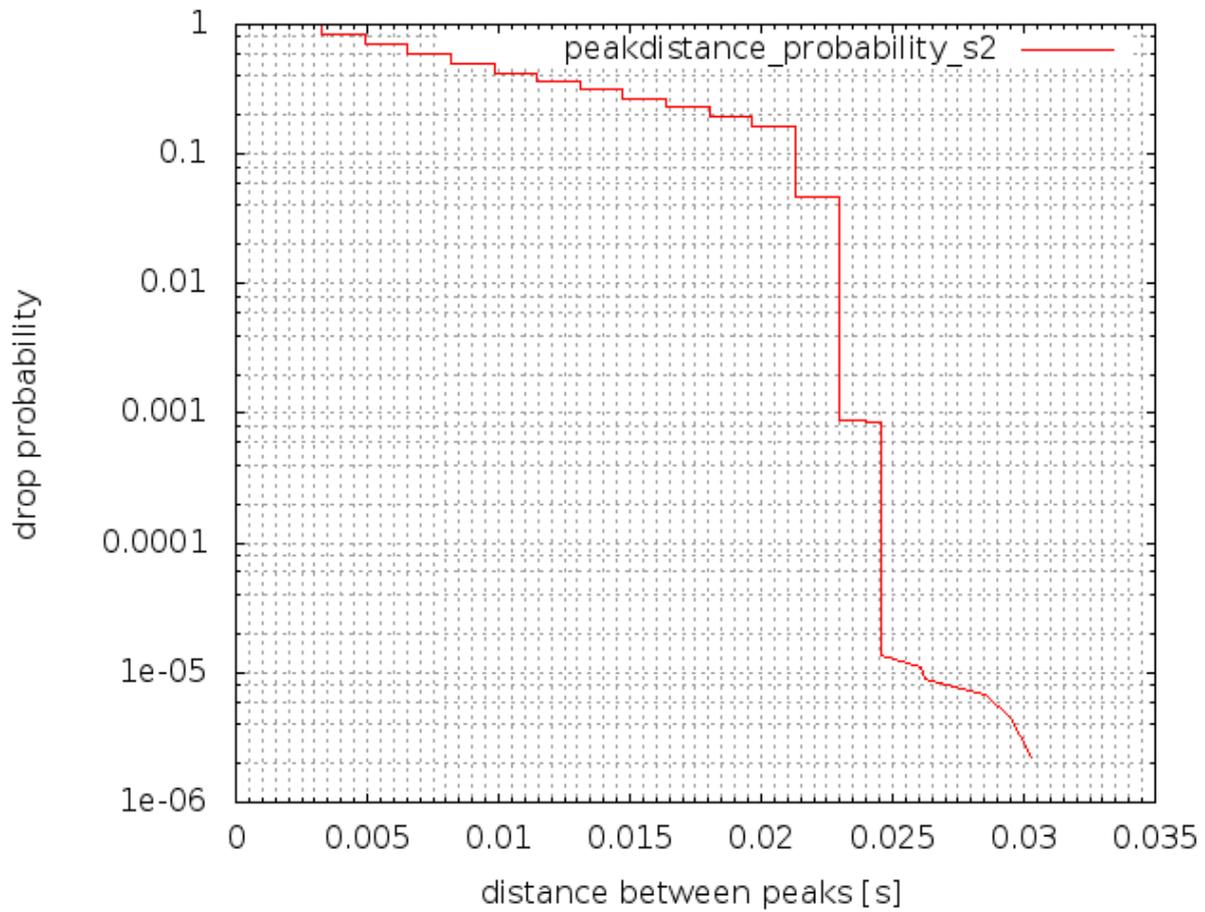


Abbildung 4.16 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 2

Eine Abfallende Kurve kommt vor. Mit dem steigenden Abstand nimmt die Wahrscheinlichkeit der Spitzenwertverluste ab. Von 0 s bis 0,021s fällt die Wahrscheinlichkeit der Spitzenwertverluste fast linear von 1 auf 0,22. Das heißt, die Abstände zwischen den Spitzenwerten unter 0,021 s nehmen 78% der Gesamt. Aber die Kurve in diesem Intervall ist nicht so steil, das heißt, die Abstände zwischen den Spitzenwerten verändern sich nicht so schnell. Von 0,021 s bis 0,0245 s fällt die Kurve fast senkrecht ab. In diesem Intervall sinkt die Wahrscheinlichkeit der Spitzenwertverluste von 22% auf 0,0008% ab. Die Abstände zwischen den Spitzenwerten unter 0,0245 s nehmen 99,9992% der Gesamt. Von der hohen Steilheit ist es auch bekannt, dass die Abstände zwischen den Spitzenwerten sich sehr schnell variieren. Auf Szenario 1 nehmen die Abstände zwischen den Spitzenwerten unter 0,104 s 99,954% und 0,174 s 99,9992% der Gesamt. Es ist erkannt, dass im Vergleich zu Szenario 1 die Spitzenwerte der *OWD* auf Szenario 2 frequenter und mehr auftreten. Von 0,0245 s bis 0,0303 s fällt die Wahrscheinlichkeit der Spitzenverlust von 0,0008% auf 0,00028%. Von dem Ergebnis ist es bekannt, die Abstände zwischen den Spitzenwerten sich unter 0,0245 s konzentrieren.

Am Ende wird die Kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 3 in Abbildung 4.17 dargestellt.

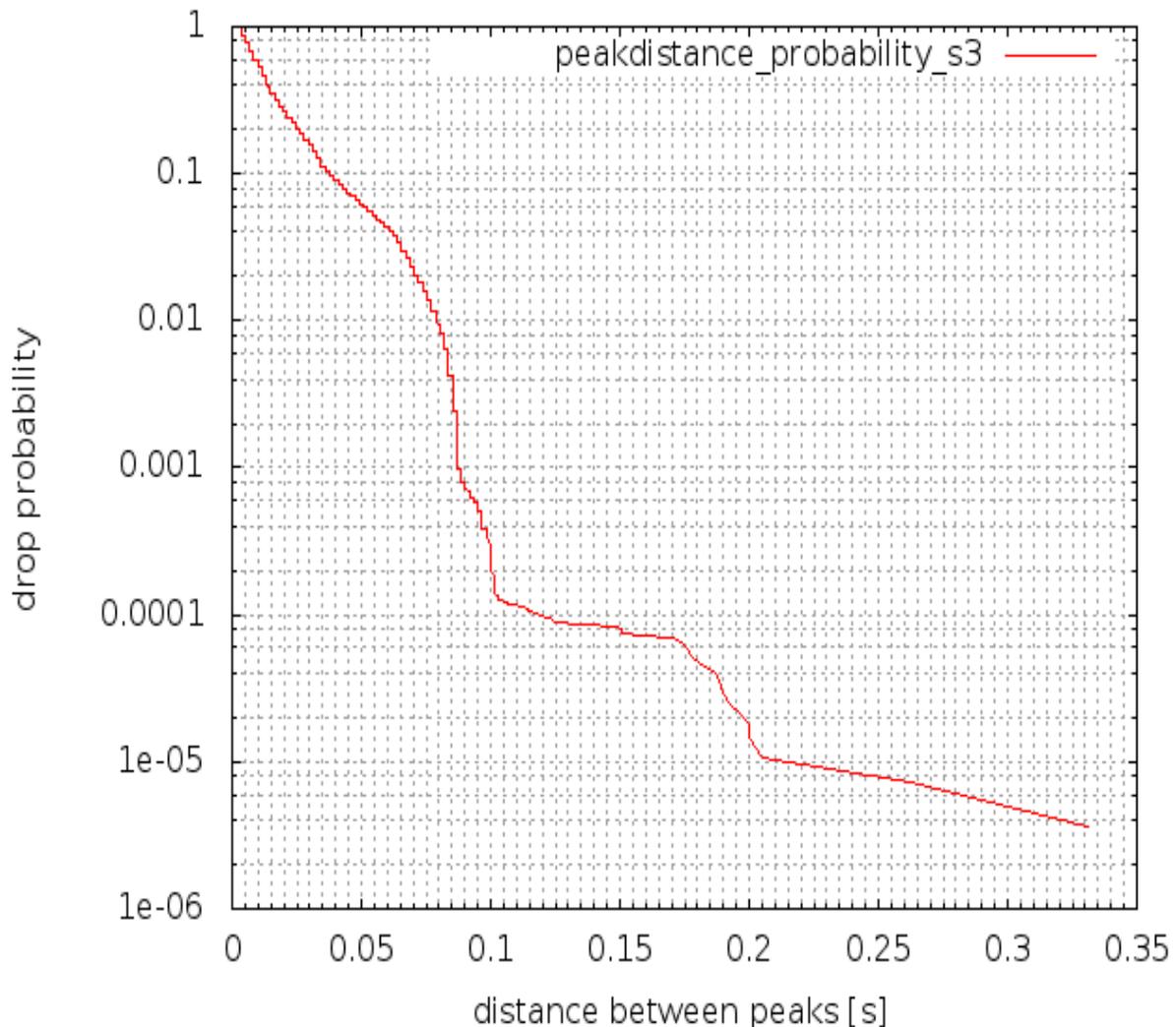


Abbildung 4.17 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 3

Eine Abfallende Kurve erscheint sich. Mit dem steigenden Abstand nimmt die Wahrscheinlichkeit der Spitzenwertverluste ab. Von 0 s bis 0,102s ist die Kurve sehr steil, die Wahrscheinlichkeit der Spitzenwertverluste fällt schnell von 1 auf 0,00017. Das heißt, die Abstände zwischen den Spitzenwerten unter 0,102 s nehmen 99,982% der Gesamt. Von 0,102 s bis 0,17 s fällt die Kurve sehr langsam ab. Die Wahrscheinlichkeit der Spitzenwertverluste sinkt von 0,00017 um 0,000097 ab. Danach liegt die Kurve von 0,17 s bis 0,205 s wieder steil. In diesem Intervall fällt die Wahrscheinlichkeit der Spitzenwertverluste ungefähr auf 0,001% ab. Von 0,205 bis 0,332 fällt von 0,00001 auf 0,0000045. Die Abstände zwischen den Spitzenwerten unter 0,102 s nehmen 99,982% der Gesamt. Von der hohen Steilheit ist es auch bekannt, dass die Abstände zwischen den Spitzenwerten sich sehr schnell verändern. Auf Szenario 1 nehmen die Abstände zwischen den Spitzenwerten unter 0,104 s 99,954% der Gesamt. Im Vergleich zu Szenario 1 treten die Abstände zwischen den Spitzenwerten unter ungefähr 0,1 s frequenter und mehr auf. Von den Ergebnissen ist es bekannt, die Abstände zwischen den Spitzenwerten sich unter 0,102 s konzentrieren.

Durch die Analysierung der *OWDs* sind die Größe und die Verteilung der *OWD* auf 3 Szenarios erkannt.

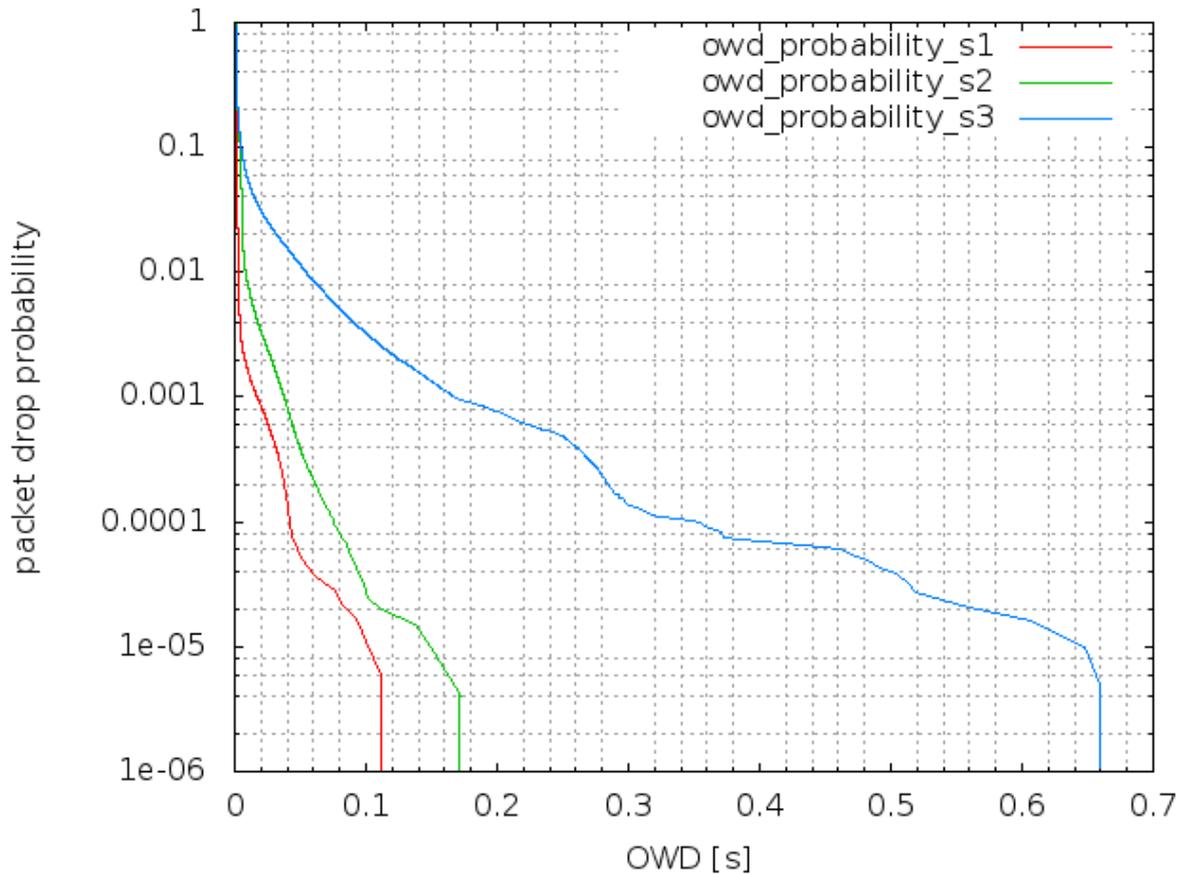


Abbildung 4.18 : die kumulative Verteilung der *OWDs* auf 3 Szenarios

In Abbildung 4.18 wird die kumulative Verteilung der *OWDs* auf 3 Szenarios dargestellt. Die *OWDs* auf Szenario 1, 2 und 3 verteilen sich in dem Intervall von 0 s bis 0,12 s., von 0 s bis 0,18 s und von 0 s bis 0,67 s. Die Kurven von Szenario 1 und 2 sind viel steiler als die von Szenario 3. Das heißt die Paketverlustwahrscheinlichkeit von Szenario 1 und 2 nimmt sehr schnell mit Zeit ab. Erreicht eine Paketverlustrate von 10^{-5} , Szenario 1 und 2 brauchen jeweils 0,101 s und 0,152 s, aber Szenario 3 braucht 0,65 s. Um die Fehlerrate unter 10^{-6} zu unterdrücken, Szenario 3 braucht 0,67 s, aber Szenario 1 und 2 brauchen nur jeweils 0,12 s und 0,18 s.

Indem der Abstand zwischen den Spitzenwerten der *OWD* auf 3 Szenarios analysiert wurde, ist die Häufigkeit der Erscheinung der Spitzenwerte auf jedem Szenario bekannt. Mit anderen Worten ist es bekannt, wie oft die Spitzenwerte auftreten.

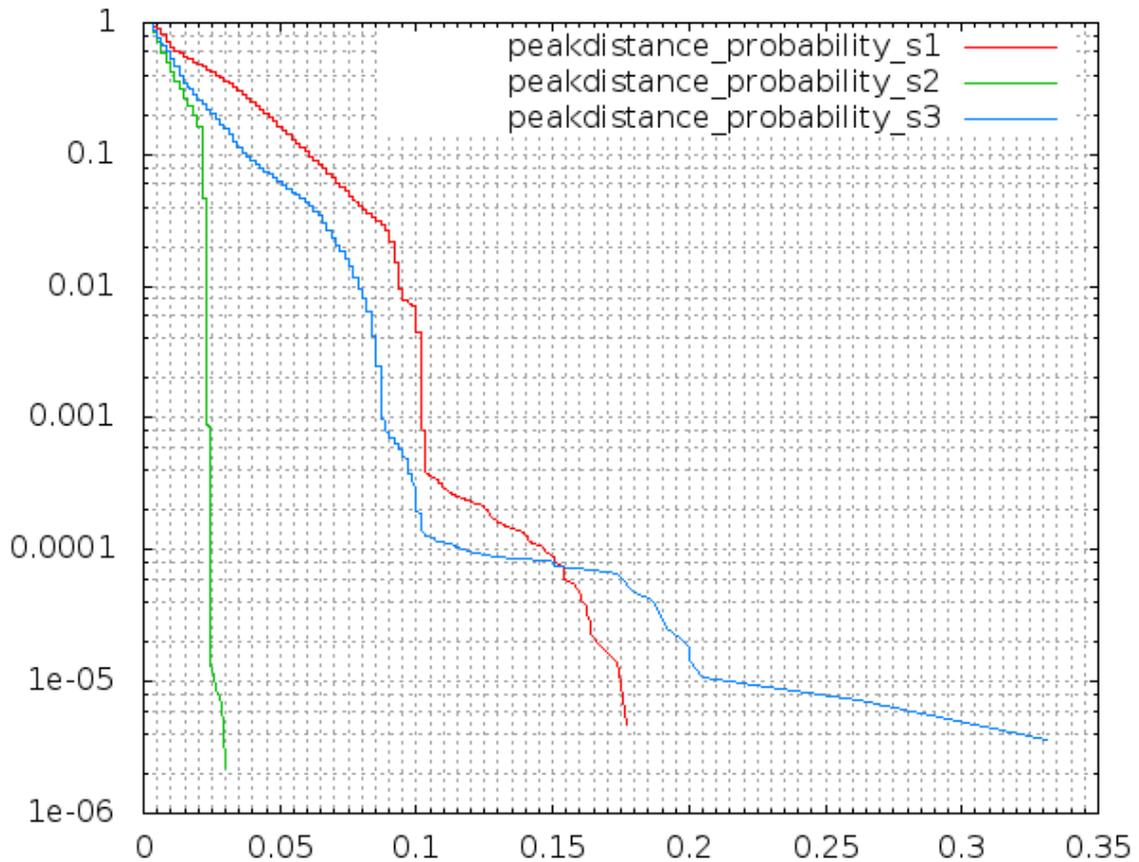


Abbildung 4.19 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf 3 Szenarios

In Abbildung 4.19 wird die Verteilung der Abstände zwischen den Spitzenwerten der *OWD* auf 3 Szenarios dargestellt. Alle Spitzenwerte auf Szenario 2 treten unter dem Abstand von 0,03 s auf. Auf Szenario 3 und 1 kommen jeweils 99,9855% und 99,963% der gesamten Spitzenwerte unter dem Abstand von 1,1 s vor. Spitzenwerte auf Szenario 3 und 1 nehmen unter 0,15 s fast gleich 99,9928% der Gesamt. Es ist auch erkannt, dass auf Szenario 2 die Spitzenwerte am häufigsten auftreten. Dann treten die Spitzenwerte auf Szenario 3 häufiger als auf Szenario 1 auf.

5 Zusammenfassung und Ausblick

Gegenstand der hier vorliegenden Arbeit war die Untersuchung der Auswirkung der MAC-Zugriffsverfahren *CSMA/CA* auf das Transportschicht-Protokoll in WLAN-Netzen. Zum Erreichen des Ziels dieser Arbeit wurde es zuerst nachgewiesen, dass *CSMA/CA* zum *TCP*-Retransmission führen könnte. Dann die kumulative Verteilung der *OWDs* auf jedem Szenario analysiert. Am Ende wurde die kumulative Verteilung der Abstände zwischen den Spitzenwerten der *OWD* nach der Datenverarbeitung auf jedem Szenario analysiert.

Wegen der Vorteile der Stabilität, Skalierbarkeit, verbesserten Reichweite und zentralisierter Sicherheit wurde der Infrastruktur-Modus ausgewählt. Die Messungen wurden in 3 Szenarios geteilt.

1. Messung mit guter Link-Qualität
2. Messung mit schlechter Link-Qualität
3. Messung mit Cross-Traffic

Dann wurden die entsprechenden Test-Topologien für die 3 Szenarios vorgestellt. Auf jedem Szenario wurden 20 Messungen durchgeführt. Weiterhin wurden die Messergebnisse ausgewertet. Da wurde es bekommen, dass die Spitzenwerte der *OWD* *TCP*-Retransmission verursachen könnten. Die Spitzenwerte der *OWD* wurden vom Backoff des Zugriffsverfahren *CSMA/CA* gebildet. Somit wurde es nachgewiesen, dass das Zugriffsverfahren *CSMA/CA* zum *TCP*-Retransmission führen könnten. Das heißt, das Zugriffsverfahren *CSMA/CA* übt tatsächlich eine negative Wirkung auf *TCP*-Performance aus. Bisher wurde die Kernaufgabe dieser Arbeit gelöst. Die Bemühungen zum Verbessern der *TCP*-Performance in der vorliegenden Arbeit sind gezielt in Forschung des Zugriffsverfahren *CSMA/CA*. Nach der Datenanalyse wurde die kumulative Verteilung der *OWDs* auf jedem Szenario zu bekommen. Damit könnte das Verhältnis von Paketverlustwahrscheinlichkeit und *OWD* festgelegt werden. Um die Paketverlustwahrscheinlichkeit unter $1e-06$ zu unterdrücken, brauchen Szenario 1, 2 und 3 jeweils 0,12 s, 0,18 s und 0,67 s. Am Ende wurde die kumulative Verteilung der Abstände zwischen den Spitzenwerten der *OWD* nach der Datenverarbeitung auf jedem Szenario bekommen. Damit könnte die Häufigkeit der Erscheinung der Spitzenwerte auch festgelegt werden. Von den Ergebnissen wurde es erkannt, dass auf Szenario 2 die Spitzenwerte am häufigsten auftreten. Die Spitzenwerte auf Szenario 3 kommen häufiger als auf Szenario 1 vor.

Zusammenfassend kann festgelegt werden, dass das Zugriffsverfahren *CSMA/CA* tatsächlich einen negativen Einfluss auf *TCP*-Performance hat. Die kumulative Verteilung der *OWDs* zeigt das Verhältnis von Paketverlustwahrscheinlichkeit und *OWD*. Außerdem stellt die kumulative Verteilung der Abstände zwischen den Spitzenwerten der *OWD* die Häufigkeit der

Erscheinung der Spitzenwerte der *OWD* dar. Die beiden wichtigen Informationen könnten genutzt werden, das *TCP-RTO* zu verändern. Wegen der Knappzeit könnten nicht alle Aufgaben auf einmal geschafft werden. In der Zukunft werden noch mehr Messungen auf verschiedenen Verbindungen mit unterschiedlicher Link-Qualität durchgeführt. Das endgültige Ziel ist ein neuer Algorithmus für Timeout zu bekommen.

Abbildungsverzeichnis

Abbildung 2.1 : Netzwerkschicht des Standards 802.11	4
Abbildung 2.2 : Ad-hoc-Netzwerk Topologie	6
Abbildung 2.3 : Infrastruktur-Netzwerk Topologie	7
Abbildung 2.4 : Distributed Coordination Function[30]	8
Abbildung 2.5 : Point Coordination Funktion[31]	9
Abbildung 2.6 : Slow Start und Congestion Avoidance	11
Abbildung 3.1 : Hardwarekomponente	14
Abbildung 3.2 : die räumliche Lage des Prüfstands	15
Abbildung 3.3 : Experiment-Topologie auf Szenario 1 und 2	15
Abbildung 3.4 : Experiment-Topologie auf Szenario 3	16
Abbildung 3.5 : Netzwerk-Schicht Topologie der Experimente	17
Abbildung 3.6 : Experiment-Topologie für CSMA/CD Verbindung	17
Abbildung 3.7 : Ein-Weg-Verzögerungen auf CSMA/CA- und CSMA/CD-Verbindung	18
Abbildung 3.8 : Verbindungsaufbau der Experimente	19
Abbildung 4.1 : Spitzenwerte der Ein-Weg-Verzögerung	20
Abbildung 4.2 : Ein-Weg-Verzögerungen auf TCP- und UDP-Verbindung	22
Abbildung 4.3 : TCP-Retransmissions in Wireshark	22
Abbildung 4.4 : Ein-Weg-Verzögerungen auf TCP- und UDP-Verbindung	24
Abbildung 4.5 : Ein-Weg-Verzögerungen auf TCP- und UDP-Verbindung	25
Abbildung 4.6 : Ein-Weg-Verzögerungen auf TCP- und UDP-Verbindung	26
Abbildung 4.7 : Ein-Weg-Verzögerungen auf TCP- und UDP-Verbindung	27
Abbildung 4.8 : Ein-Weg-Verzögerungen der UDP-Verbindung auf Szenario 1	28
Abbildung 4.9 : die kumulative Verteilung der Ein-Weg-Verzögerungen auf Szenario 1	30
Abbildung 4.10 : Ein-Weg-Verzögerungen der UDP-Verbindung auf Szenario 2	31
Abbildung 4.11 : die kumulative Verteilung der Ein-Weg-Verzögerungen auf Szenario 2	33
Abbildung 4.12 : Ein-Weg-Verzögerungen der UDP-Verbindung auf Szenario 3	34
Abbildung 4.13 : die kumulative Verteilung der Ein-Weg-Verzögerungen auf Szenario 3	36
Abbildung 4.14 : verarbeitende Daten für die kumulative Verteilung der Abstände	37
Abbildung 4.15 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 1	39
Abbildung 4.16 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 2	40
Abbildung 4.17 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf Szenario 3	41
Abbildung 4.18 : die kumulative Verteilung der Ein-Weg-Verzögerungen auf 3 Szenarios	42
Abbildung 4.19 : die kumulative Verteilung der Abstände zwischen den Spitzenwerten auf 3 Szenarios	43

Tabellenverzeichnis

Tabelle 2.1 : CW-Größe mit verschiedenen Frequenzspreizungsverfahren	8
Tabelle 3.1 : Verbindungsparameter auf Szenario 1 und 2	16
Tabelle 4.1 : Sendezeit von TCP-Retransmission und UDP-Paketverlust	23
Tabelle 4.2 : Paketverluste und Mittelwert der Ein-Weg-Verzögerungen auf Szenario 1	29
Tabelle 4.3 : Paketverluste und Mittelwert der Ein-Weg-Verzögerungen auf Szenario 2	32
Tabelle 4.4 : Paketverluste und Mittelwert der Ein-Weg-Verzögerungen auf Szenario 3	35

Literaturverzeichnis

- [1] J. B. Postel, *Transmission Control Protocol*, RFC 793, 09.1981
- [2] Cooperative Association for Internet Data Analysis:
<http://www.caida.org/research/traffic-analysis/tcpudpratio/> [Zugriff:17.07.2013]
- [3] Ajay Bakre and B.R.Badrinath, *Handoff and System Support for Indirect TCP/IP*, in proceeding of Second Usenix Symposium on Mobile and Location-Independent Computing, 04 1995
- [4] H.Balakrishnan et al., *Improving TCP/IP Performance over Wireless Networks*, in proceedings of ACM Mobicom, 11 1995
- [5] S. Paul et al., *An Asymmetric Link-Layer Protocol for Digital Cellular Communications*, in proceedings of INFOCOM, 1995
- [6] R. Ludwig et al., *Multi-layer Tracing of TCP over a Reliable Wireless Link*, in proceedings of ACM SIGMETRICS, 1999
- [7] M.Mathis, J. Mahdavi, S. Floyd and A. Romanow, *TCP Selective Acknowledgments Options*, 07 1996
- [8] E. Aynoglu, S. Paul, T. F. LaPorta, K. K. Sabnani and R. D. Gitlin, *AIRMAIL: A Link-Layer Protocol for Wireless Networks*, 02 1995
- [9] Third Generation Partnership Project, *RLC Protocol Specification(3G TS 25.322)*, 1999
- [10] TIA/EIA/IS-707-A-2.10, *Data Service Options for Spread Spectrum Systems: Radio Link Protocol Type 3*, 01 2000
- [11] Hari Balakrishnan, Venkata N. Padmanabhan, Srinivasan Seshan und Randy H. Katz, *A Comparison of Mechanisms for Improving TCP Performance over Wireless Links*, August 1996
- [12] R.Ludwig and R.H.Katz, *The Eifel Algorithm: Making TCP Robust Against Spurious Retransmissions*, in *ACM Computer Communications Review*, Vol. 30, No. 1, 01 2000
- [13] H. Inamura et.al, *TCP over 2.5G and 3G Wireless Networks*, draft-ietf-pilc-2.5g3g-07, 08 2002
- [14] M. C. Chan and R. Ramjee, *TCP/IP Performance over 3G wireless links with rate and variation*, in *Proceedings of ACM Mobicom'02*, 2002
- [15] Mun Choon Chan and Ramachandran Ramjee, *Improving TCP/IP Performance over Third Generation Wireless Networks*, *Mobile Computing, IEEE Transaction on Volume 7, Issue 4*, 04 2008
- [16] Mun Choon and Ramachandran Ramjee Bell Labs, *TCP/IP Performance over 3G Wireless Links with Rate and Delay Variation*, *Proceeding MobiCom '02 Proceedings of the 8th annual international conference on Mobile computing and networking*, 2002

- [17] Mun Choon Chan and Ramachandran Ramjee, Bell Labs, *Improving TCP/IP Performance over Third Generation Wireless Networks, Mobile Computing, IEEE Transactions Volume 7 Issue 4*, 04 2008
- [18] Marek Natkaniec and Andzej R. Pach, *An Analysis of the Backoff Mechanism used in IEEE 802.11 Networks, Computers and Communications*, 07 2000
- [19] Qixiang Pang, Soung Chang Liew Jack Z. B. Lee and S. -H. Gary Chan, *A TCP-like Adaptive Contention Window Scheme for WLAN , Communications, IEEE international Conference on Volume 6*, 06 2004
- [20] Tianji Li, *Improving Performance for CSMA/CA Based Wireless Networks, a dissertation submitted for the degree of Doctor of Philosophy*, 12 2007
- [21] P Mi-Young and C Sang-Hwa, *Distinguishing the cause of TCP retransmission timeouts in multi-hop wireless networks, 12th IEEE International Conference on High Performance Computing and Communications*, 09 2010
- [22] Prasanthi Sreekumari and Meejeong Lee, *TCP NRT: a new TCP algorithm for differentiating non-congestion retransmission timeouts over multihop wireless networks, EURASIP Journal on Wireless Communications and Networking*, 06 2013
- [23] NJ Kothari, BM Gambhava, KS Dasgupta, *RTT utilization by detecting avoidable timeouts, 14th IEEE International Conference on Networks, Singapore*, 09 2006
- [24] D Ciullo, M Mellia and M Meo, *Two schemes to reduce latency in short lives TCP flows, IEEE Communications Letters 13*, 10 2009
- [25] P Mi-Young and C Sang-Hwa, *Distinguishing the cause of TCP retransmission timeouts in multi-hop wireless networks, 12th IEEE International Conference on High Performance Computing and Communications*, 09 2010
- [26] Siemens E., Piger S., Grimm C. and Fromme M., *LTest - a tool for distributed network performance measurement, Consumer Communications and Networking Conference, 2004. CCNC 2004, first IEEE*, 01 2004
- [27] Jörg Rech, *Ethernet Technologien und Protokolle für die Computervernetzung*, Verlag: Heise 2002
- [28] Rechnernetze, Distributed Coordination Function:
<http://einstein.informatik.uni-oldenburg.de/rechnernetze/dcf.htm> [Zugriff:17.07.2013]
- [29] Wirtschaftsinformatik (Bachelor-Studiengang): Rechnernetze/Onlinedienste (2. Semester):
<http://www.it-infothek.de/fhtw/studium.html> [Zugriff:17.07.2013]