

# **ABSCHLUSSARBEIT**

Alexander Kinder

**Analyse der Kommunikation über die V2V Schnittstelle für  
große Datenpakete**

2019

## **ABSCHLUSSARBEIT**

# **Analyse der Kommunikation über die V2V Schnittstelle für große Datenpakete**

Autorin: **Alexander Kinder**

Studiengang: Elektrotechnik/Kommunikationstechnik  
Seminargruppe: KI12w1-F

Erstprüfer: Prof. Dr. Thomas Haenselmann  
Zweitprüfer: M. Eng. Benjamin Heidenreich

Wolfsburg, 20.05.2019

# Inhalt

<b>Abbildungsverzeichnis .....</b>	<b>III</b>
<b>Tabellenverzeichnis.....</b>	<b>IV</b>
<b>Abkürzungsverzeichnis .....</b>	<b>VI</b>
<b>1 Einleitung .....</b>	<b>1</b>
<b>2 Grundlagen .....</b>	<b>3</b>
2.1 <i>OSI-Referenzmodell.....</i>	<i>3</i>
2.1.1 Protokolle.....	4
2.1.2 Schnittstellen .....	5
2.2 <i>TCP/IP-Referenzmodell .....</i>	<i>5</i>
2.3 <i>Client-Server-Struktur .....</i>	<i>6</i>
2.4 <i>Verbindungsaufbau mit TCP.....</i>	<i>6</i>
2.5 <i>Peer-to-Peer Architektur .....</i>	<i>8</i>
2.6 <i>Torrent-Protokoll.....</i>	<i>9</i>
2.6.1 Funktionsweise des Torrent-Protokolls.....	9
2.7 <i>Kodierungstheorie .....</i>	<i>11</i>
2.7.1 Fehlererkennende Codes.....	11
2.7.1.1 Quersummenverfahren .....	11
2.7.1.2 Paritätsbit.....	12
2.7.1.3 Zyklische Redundanzprüfung (CRC) .....	13
2.7.2 Fehlerkorrigierende Codes.....	14
2.7.2.1 Hamming-Code .....	15
2.7.2.2 Fountain-Codes.....	17
<b>3 Stand der Technik .....</b>	<b>21</b>
3.1 <i>Ziele .....</i>	<i>21</i>
3.2 <i>Standards .....</i>	<i>21</i>
3.3 <i>WAVE und VANET.....</i>	<i>23</i>
3.4 <i>C-V2X .....</i>	<i>24</i>
3.5 <i>Zusammenfassung.....</i>	<i>25</i>
<b>4 Ausarbeitung .....</b>	<b>26</b>

---

4.1	<i>Update OTA</i> .....	26
4.2	<i>Torrent-Protokoll im Fahrzeugumfeld</i> .....	27
4.3	<i>Begegnungsarten</i> .....	29
4.4	<i>Festlegung der Dateigrößen</i> .....	32
4.5	<i>Gewichtung der Begegnungsarten</i> .....	34
4.6	<i>Bestimmung der Dateigröße</i> .....	38
4.7	<i>Fahrzeugbestand und Neuzulassungen</i> .....	39
4.8	<i>Prognose</i> .....	42
<b>5</b>	<b>Fazit</b> .....	<b>47</b>
<b>6</b>	<b>Literatur</b> .....	<b>49</b>
	<b>Selbstständigkeitserklärung</b> .....	<b>50</b>

# Abbildungsverzeichnis

Abbildung 1: OSI-Referenzmodell.....	4
Abbildung 2: TCP/IP-Referenzmodell .....	5
Abbildung 3: Client-Server Architektur .....	6
Abbildung 4: Verbindungsaufbau bei TCP .....	7
Abbildung 5: Peer-to-Peer Netzwerk (1) .....	8
Abbildung 6: Funktionsweise eines Torrent .....	10
Abbildung 7: Bitkombination .....	14
Abbildung 8: Erstellung eines Codierten Pakets .....	18
Abbildung 9: Beispiel einer Codierten Nachricht.....	19
Abbildung 10: WAVE Protokoll (2) .....	22
Abbildung 11: LTE-V Kommunikation (3).....	23
Abbildung 12: Vergleich Verbreitung Update .....	28
Abbildung 13: Szenario 1 - Fahrt in gleiche Richtung.....	29
Abbildung 14: Szenario 2 - Fahrt in entgegengesetzte Richtung .....	30
Abbildung 15: Szenario 3 - Kommunikation mit Gateway.....	32
Abbildung 16: Kraftwagen Neuzulassungen pro Monat.....	41

# Tabellenverzeichnis

Tabelle 1: Vor- und Nachteile einer Client-Server-Architektur .....	6
Tabelle 2: Vor- und Nachteile Peer-to-Peer-Netzwerke .....	9
Tabelle 3: Gewichtete Quersumme .....	12
Tabelle 4: Werte des Paritätsbit .....	12
Tabelle 5: Fehlerkorrektur mit Paritätsbit .....	13
Tabelle 6: Parameterkombination bei Hamming-Codes .....	16
Tabelle 7: Positionen der Paritätsbits .....	16
Tabelle 8: Spezifikationen IEEE802.11p (4) .....	24
Tabelle 9: Spezifikationen C-V2X (4) .....	25
Tabelle 10: Kommunikationszeiten für verschiedene Szenarien (Reichweite 1 km) .....	31
Tabelle 11: Kommunikationszeiten für verschiedene Szenarien (Reichweite 300 m) .....	31
Tabelle 12: Datenmengen Vergleich für verschiedene Szenarien (Reichweite 300 m) .....	34
Tabelle 13: Gewichtung - Paarweiser Vergleich .....	36
Tabelle 14: Prozentuale Gewichtung .....	37
Tabelle 15: Gewichtung der Begegnungsarten .....	37
Tabelle 16: Fahrzeugbestand in Deutschland .....	40
Tabelle 17: Prozentueller Vergleich neu Zugelassener Fahrzeuge .....	42
Tabelle 18: Fahrzeugbestand nach Hersteller .....	43
Tabelle 19: Neuzulassungen nach Hersteller – 2018 .....	44
Tabelle 20: Neuzulassungen nach Hersteller - 2019 1. Quartal .....	45

---

Tabelle 21: Vergleich neu Zugelassener Fahrzeuge - 2015 & 2016 .....	45
Tabelle 22: Vergleich neu Zugelassener Fahrzeuge - 2017 & 2018 .....	46

## Abkürzungsverzeichnis

<b>P2P</b>	Peer-to-Peer
<b>C2C</b>	Car-to-Car
<b>C2CC</b>	Car-to-Car-Consortium
<b>RSU</b>	Roadside Unit
<b>IRS</b>	Intelligent Roadside Station
<b>ETSI</b>	Europäisches Institut für Telekommunikationsnormen
<b>FCC</b>	Federal Communication Commission
<b>TDM</b>	Time Division Multiplex
<b>FDM</b>	Frequency Division Multiplex
<b>RiLSA</b>	Richtlinien für Lichtsignalanlagen
<b>KBA</b>	Kraftfahrt-Bundesamt



# 1 Einleitung

Die Automobilhersteller stehen aktuell vor einem Umbruch. Was früher nur durch unterschiedliche Motoren und Leistungsstufen der Autos ausgetragen wurde ist heute in Richtung Komfort gewandert. Begriffe wie „Autonomes Fahren“, „In Car Office“ oder „Elektrifizierung“ prägen die Automobilbranche. Zusammengefasst ist heute die Rede von der Digitalisierung eines Fahrzeugs.

Hersteller wie Tesla sind auf diesem Gebiet Pioniere und treiben die Entwicklung stark voran. Aber auch die anderen großen Hersteller sind zum Umdenken angehalten.

Ein großes Themengebiet in diesem Bereich ist die Kommunikation zwischen Fahrzeugen (Vehicle-to-Vehicle (V2V)) und der Infrastruktur (Vehicle-to-Infrastructure(V2I)).

Im ersten Augenblick zielt diese Technologie dahin ab den Verkehr sicherer zu machen und die Straßen in überfüllten Gebieten zu entlasten. Eine erkannte Gefahr wird an alle anderen Verkehrsteilnehmer kommuniziert. Diese können dann den Fahrer darauf aufmerksam machen, aber auch vorbereitende Maßnahmen ergreifen.

Diese Technologie kann aber auch genutzt werden, um andere Informationen zwischen den Fahrzeugen auszutauschen. Das können Daten wie die Informationen über die Verkehrslage, aber auch Updatedaten für ein im Fahrzeug verbautes Steuergerät sein. Die Verteilung von Multimediadaten, wie zum Beispiel Videos, könnte ein weiteres Anwendungsgebiet darstellen. Wenn die Entwicklung in Richtung autonomes Fahren weiter so voranschreiten wird hat der Fahrer mehr Zeit sich auf die oben genannten Punkte zu konzentrieren.

Solche Daten sind selbstverständlich größer, als nur kurze Informationen über eine Gefahrenstelle. Somit liegt es nahe das der Austausch der Daten koordiniert und geregelt werden muss.

Auch müssen Kommunikationspartner vorhanden sein, mit welchen die Daten ausgetauscht werden können. Bis Fahrzeuge, ausgestattet mit der Technologie, in ausreichender Menge vorhanden sein werden wird Zeit benötigt, basierend auf der Fahrzeugmenge auf den Straßen. Der zweite Teil der Arbeit beschäftigt sich mit der Analyse der Zulassung neuer Fahrzeuge. Mit diesen Resultaten soll eine Prognose erstellt werden, zu welchem Zeitpunkt ausreichend Kommunikationsteilnehmer auf der Straße vorhanden sind und somit ein erkennbarer Vorteil dieser Technologie besteht.

Um eine Absicherung der Kommunikationsstrecke zwischen den Fahrzeugen zu gewährleisten sollte zudem die Codierungstheorie betrachtet und aktuelle fehlerkorrigierende

Codes analysiert werden. Diese Analyse soll als eine Empfehlung für den Standard dienen. Während der Recherche für die Arbeit kam heraus, dass bereits ein aktueller Code, aus dem Bereich der Fountain-Codes, verwendet wird. Folglich wird keine Empfehlung an dieser Stelle abgegeben, da bereits ein aktueller und optimaler Code Verwendung findet.

## 2 Grundlagen

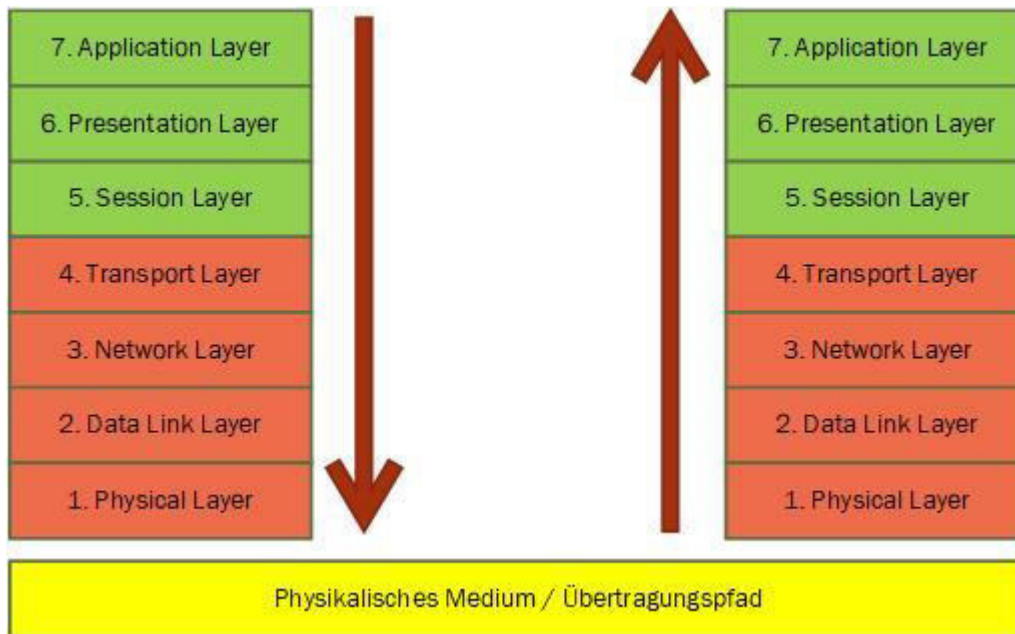
In diesem Kapitel werden einige grundlegende Beschreibungen dargestellt. Diese sind nicht explizit für eine Car2Car-Kommunikation relevant, sondern viel mehr die Grundlagen für die Kommunikation zwischen zwei oder mehreren Systemen.

### 2.1 OSI-Referenzmodell

Wie in einer menschlichen Unterhaltung sind auch in der Informationstechnologie Regeln für eine Kommunikation notwendig. Für diesen Zweck wurde das OSI-Referenzmodell erstellt. Dieses beschreibt insgesamt sieben verschiedene Schichten (Layer), die jeweils aufeinander aufbauen. In jeder einzelnen Schicht werden genau definierte Aufgaben ausgeführt. Auch die Schnittstellen zwischen den Schichten sind genau beschrieben. Somit können einzelne Schichten ausgetauscht werden, ohne dass andere davon betroffen sind. Als Beispiel kann das Übertragungsmedium von Kupferleitung gegen W-Lan ausgetauscht werden.

Bei einer Kommunikation zweier Systeme durchläuft die Nachricht jede Schicht und wird mit den Informationen der jeweiligen Schicht verpackt. Beim Empfänger muss diese Nachricht in umgekehrter Reihenfolge die Schichten durchlaufen. Die Informationen der jeweiligen Schicht werden dann entpackt und ausgewertet. Sind in der Kommunikation Zwischenstationen notwendig, kommen weitere Durchläufe durch das OSI-Modell hinzu. In diesem basieren die Zwischenstationen je nach Umfang nur auf einer begrenzten Anzahl an Schichten: Ein Router beispielsweise arbeitet auf den Schichten 1 bis 3.

Das folgende Bild soll einen Nachrichtendurchlauf darstellen.



**Abbildung 1: OSI-Referenzmodell**

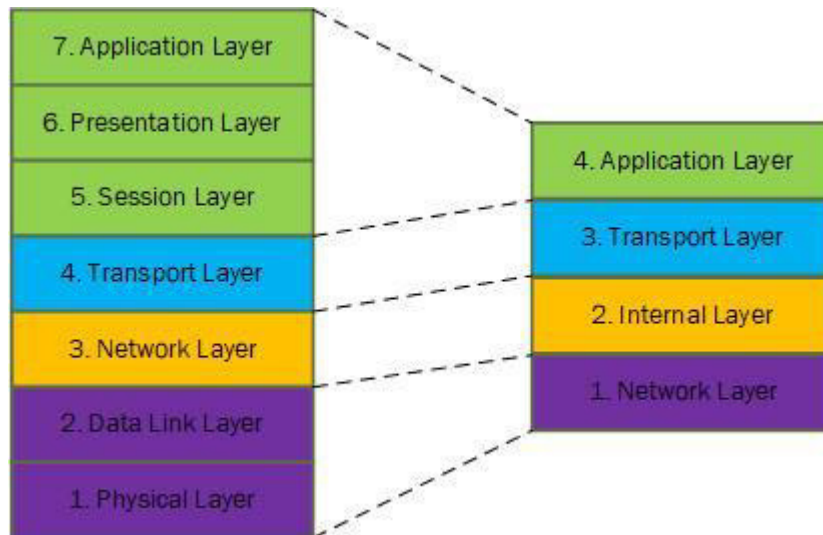
Folgende Aufzählung stellt die sieben einzelnen Schichten des OSI-Modells und eine Kurzbeschreibung ihrer Aufgaben dar:

- **Schicht 1:** Der Physical Layer oder die Bitübertragungsschicht: physikalische Anpassung der Bits auf das Übertragungsmedium
- **Schicht 2:** Der Data Link Layer oder die Scherungsschicht: Segmentierung der Daten und Sicherung mit Prüfsumme
- **Schicht 3:** Der Network Layer oder die Vermittlungsschicht: Vermittlung der Pakete zum nächsten Netzknoten
- **Schicht 4:** Der Transport Layer oder die Transportschicht: Zuteilung von Daten zur Anwendung
- **Schicht 5:** Der Session Layer oder die Sitzungsschicht: Steuerung und Kontrolle der Verbindung
- **Schicht 6:** Der Presentation Layer oder die Darstellungsschicht: Wandlung der Daten in unabhängige Formate
- **Schicht 7:** Der Application Layer oder die Anwendungsschicht: Bereitstellung von Funktionen der Anwendung wie die Datenein- und -ausgabe

### 2.1.1 Protokolle

Jede Schicht im OSI-Referenzmodell bietet seine Dienste den Schichten darüber und darunter an. Für diesen Zweck müssen für jede Schicht Protokolle beschrieben werden, damit Empfänger und Sender die Daten interpretieren können, ähnlich der menschlichen Sprache.

Zum Beispiel werden die Informationen aus der Schicht 5 beim Empfänger erst in der Schicht 5 gelesen und ausgewertet.



**Abbildung 2: TCP/IP-Referenzmodell**

### 2.1.2 Schnittstellen

Schnittstellen, oder auch englisch Interfaces, werden die Dienste genannt, die einer Schicht der jeweiligen darüber oder darunter liegenden zur Verfügung stellt. Dabei muss die Schicht das Vorgehen dieser nicht kennen.

So braucht die Schicht 2 nicht zu Wissen mit welcher Geschwindigkeit das einzelne Bit auf dem Übertragungsmedium gesendet wird, sondern gibt nur die Daten an die 1. Schicht über die Schnittstelle weiter.

## 2.2 TCP/IP-Referenzmodell

Auf dem TCP/IP-Referenzmodell basiert das heutige Internet. Es ist einfacher gehalten als das OSI-Referenzmodell. Das OSI-Referenzmodell entstand aus dem TCP/IP-Modell.

Die einzelnen Schichten im TCP/IP-Referenzmodell erfüllen mehr Aufgaben und können nicht ohne weiteres ausgetauscht werden.

Nach diesem Schema funktioniert heute jegliche Kommunikation zwischen Informationssystemen, so wie beispielsweise das Internet.

## 2.3 Client-Server-Struktur

Im Internet hat sich die Client-Server-Struktur durchgesetzt. Ein Client muss immer zuerst eine Anfrage an den Server stellen, bevor er die Informationen zurückbekommt. Der Server entscheidet, ob und wann er die Anfrage beantwortet und ob er dem Client sogar die Informationen vorenthält, wenn dieser sich nicht ausreichend verifiziert hat.

Hinzu kommt, dass die Server je nach Aufgabe unterschiedliche Software ausführen müssen. Diese verbrauchen Ressourcen vom Rechner, unabhängig davon, ob diese gerade genutzt werden oder nicht.

Wie auf dem Bild dargestellt können verschiedenen Clients einen Server anfragen und die Informationen erhalten.

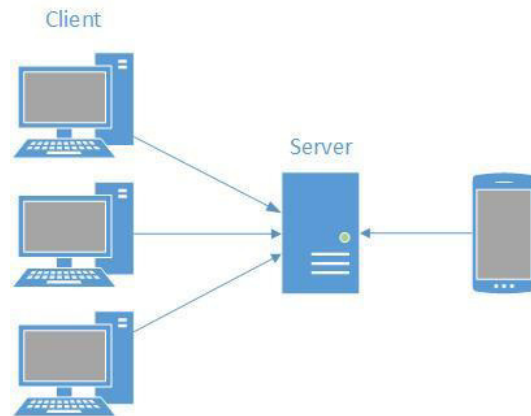


Abbildung 3: Client-Server Architektur

Vor- und Nachteile bei einem solchen System sind:

Vorteile	Nachteile
Leichtere Zugriffskontrolle durch Authentifizierung am Server	Fällt der Server aus, ist das ganze System lahmgelegt (Single Point of failure)
Clients müssen nur die Server kennen und mit diesen kommunizieren.	Viele Anfragen an einen Server können zu zeitversetzter Antwort führen bis hin zum Ausfall des Servers (Denial of Service)

Tabelle 1: Vor- und Nachteile einer Client-Server-Architektur

## 2.4 Verbindungsaufbau mit TCP

Als nächstes wollen wir uns einmal ansehen wie eine Verbindung zwischen zwei Kommunikationspartnern aufgebaut wird.

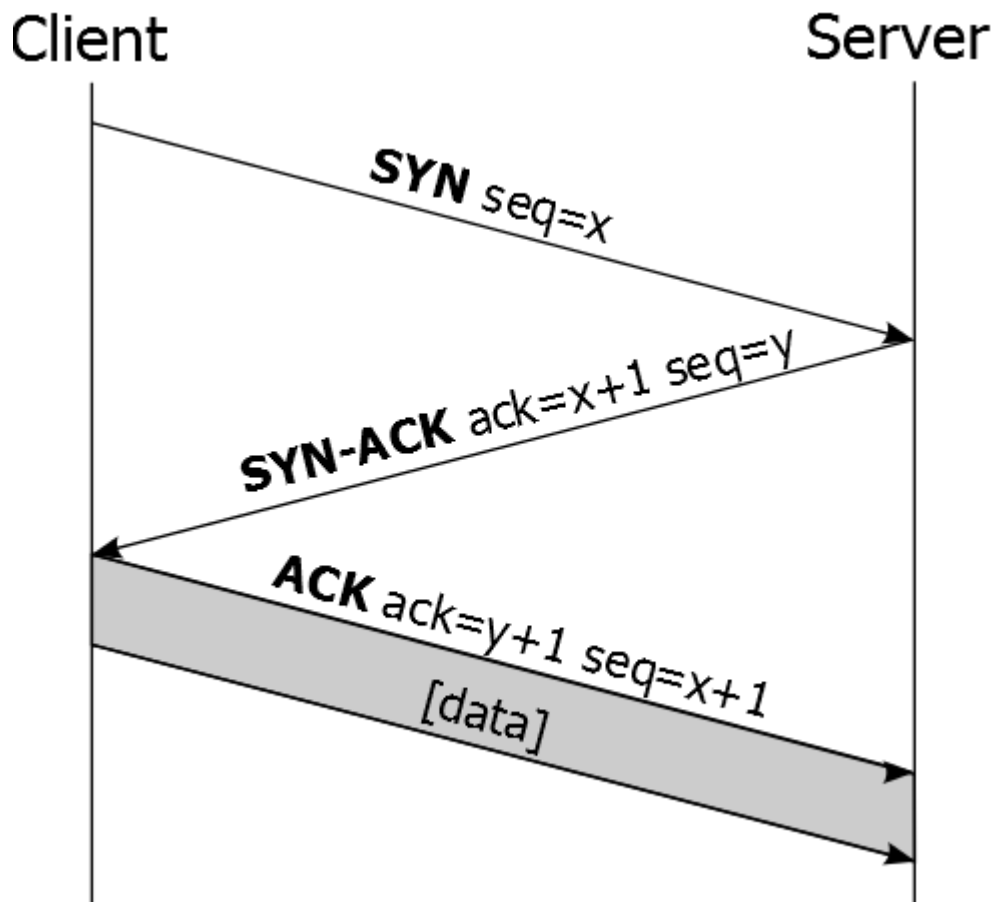


Abbildung 4: Verbindungsaufbau bei TCP

Der Verbindungsaufbau zwischen zwei Teilnehmern ist in Abbildung 4 abgebildet.

Der Verbindungsaufbau geschieht in drei Schritten:

- 1) Ein Client sendet ein TCP-Segment ohne Daten an den Server. In diesem Segment ist das SYN-Flag gesetzt und eine initiale Sequence Number (eine beliebige Zahl) wird übertragen.
- 2) Der Server empfängt das Segment. Sollte der angesprochene Port auf dem Server geschlossen sein, kann keine Verbindung aufgebaut werden. Ist er jedoch geöffnet, bestätigt er den Erhalt des TCP-Segments und stimmt dem Verbindungsaufbau zu. Dazu antwortet der Server mit einem TCP-Segment, welches seine eigene initiale Sequence Number im entsprechenden Feld des TCP-Headers enthält. Bei dem Segment sind außerdem das SYN-Flag und das ACK-Flag gesetzt und die initiale Sequence Number vom Client wird um eins erhöht und im Acknowledgment Number Feld eingetragen.
- 3) Im letzten Schritt bestätigt der Client den Erhalt des Segments des Servers. Hierzu erhöht er seine initiale Sequence Number um eins und trägt diese in das Feld Sequence Number ein. Zusätzlich wird die initiale Sequence Number des Servers um eins erhöht und im Feld Acknowledgment Number eingetragen. Das entsprechende

TCP-Segment wird dann mit dem gesetzten ACK-Flag an den Server verschickt und die Verbindung ist aufgebaut.

Nach dem Verbindungsaufbau können die Daten zwischen den Teilnehmern übertragen werden. Diese Art der Verbindungsaufbaus wird auch als „Drei-Wege-Handschlag“ bezeichnet.

## 2.5 Peer-to-Peer Architektur

Aus der Problematik einer Client-Server-Architektur hat sich mit der Zeit die Peer-to-Peer-Architektur (P2P) etabliert. Alle Teilnehmer in einem Netzwerk besitzen dieselben Rechte. Alle Partner können die Funktionen und Dienstleistungen des jeweils anderen nutzen und umgekehrt. Ein P2P-Netzwerk arbeitet dezentral, das bedeutet das die Daten der Rechner, die an das Netzwerk angebunden sind, nicht auf einem Server gespeichert werden, sondern auf den Rechnern verteilt sind.



**Abbildung 5: Peer-to-Peer Netzwerk (1)**

Im Grunde kann man sich eine P2P-Architektur so vorstellen, dass alle teilnehmenden Rechner über ein Kabel miteinander verbunden werden. Da dies in der heutigen Internet-Welt nicht einfach so möglich ist, wurde auch das P2P weiterentwickelt. Heute besteht das P2P-Netzwerk hauptsächlich nur noch als Overlay Netzwerk, welches über das Internet aufgebaut wird.

Eine Weiterentwicklung des P2P-Netzwerks mit zentralen Serverkomponenten sind die Super-Peer-Netzwerke. In einer solchen Konfiguration werden besonders leistungsfähige Peers zu Super-Peers zusammengeschlossen, die die Serverdienste erbringen und das



Netzwerk organisieren. Sie sorgen für das Routing der Daten zu den dezentralen Clients und bilden ein Backbone-Netzwerk.

Vorteile	Nachteile
Je mehr Teilnehmer am Netzwerk beteiligt sind, desto leistungsfähiger wird das gesamte Netzwerk	Die Suche nach Informationen innerhalb des Netzwerks ist sehr aufwendig.
Hohe Robustheit und Verfügbarkeit, da die Informationen auf alle Teilnehmer verteilt werden.	

**Tabelle 2: Vor- und Nachteile Peer-to-Peer-Netzwerke**

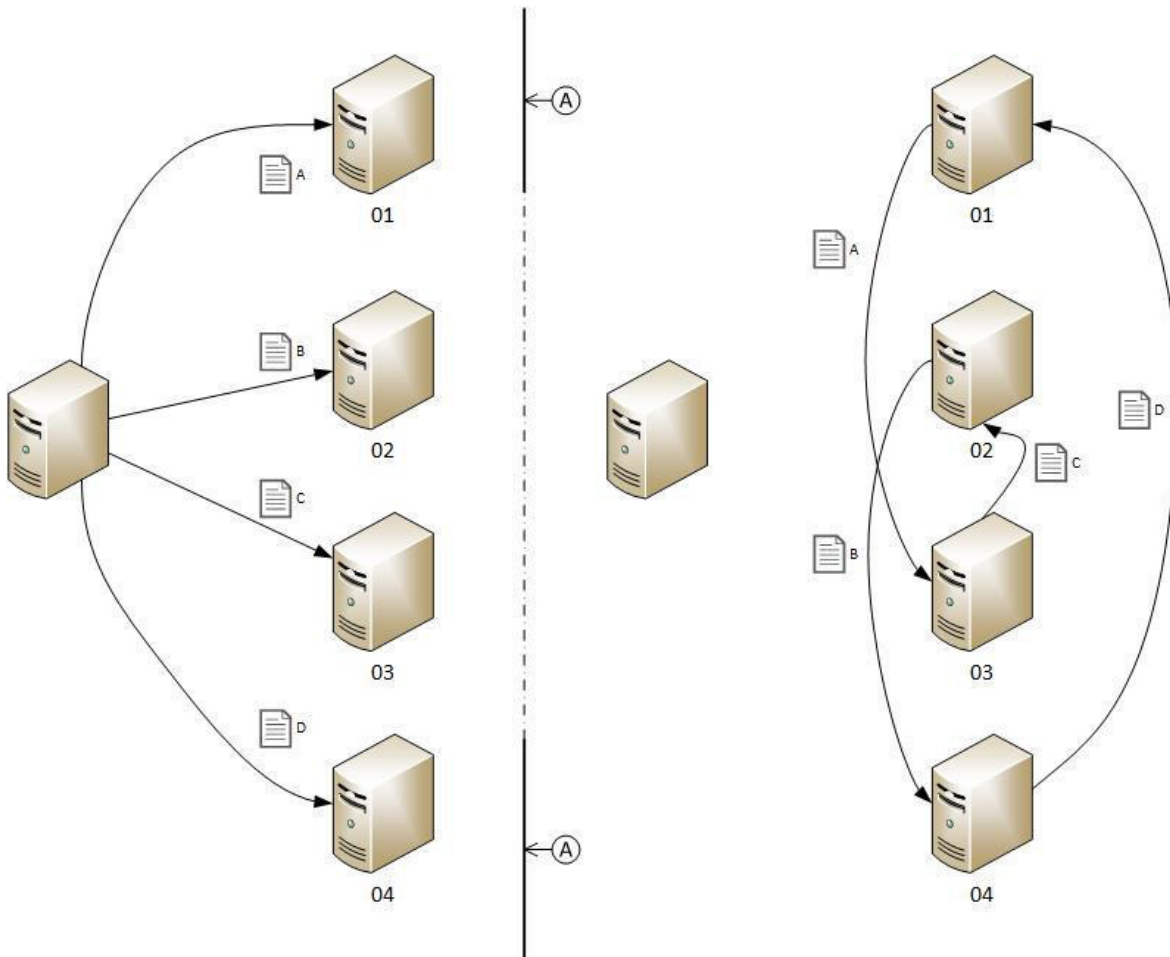
## 2.6 Torrent-Protokoll

Der Kernpunkt des Torrent-Protokolls ist die Informationen, meist in Form von Dateien, in kleine Blöcke zu unterteilen und diese dann zu versenden. Erst wenn alle Blöcke einer Information vollständig sind kann der Empfänger diese zusammenführen. Der Vorteil daran ist das die Blöcke gespeichert werden und eine Übertragung später fortgesetzt wird, ohne wieder von vorne zu beginnen.

Nach den negativen Schlagzeilen, im Zusammenhang mit dem Filesharing, ist das Torrent-Protokoll wieder in den Schatten der Allgemeinheit getreten. Trotzdem wurde das Protokoll weiterentwickelt und bietet heute mehr Funktionen denn je.

### 2.6.1 Funktionsweise des Torrent-Protokolls

Eine Datei soll an mehrere Empfänger gesendet werden. Bevor jeder Empfänger die Datei herunterladen kann wird diese in kleinere Parts aufgeteilt. Danach kann jeder Peer einen Part runterladen. Nachdem die Peers die Parts empfangen haben stellen Sie diesen für andere zur Verfügung. Im Beispiel wird eine Datei in vier Parts geteilt. Vier Peers laden sich die Datei herunter. Jeder Peer erhält zuerst jeweils einen Part der Datei. Im nächsten Schritt können die Peers untereinander die noch fehlenden Parts beziehen, auch wenn der Initiator in der Zwischenzeit vom Netzwerk abgemeldet und offline ist. Durch dieses Verfahren wird die Sicherheit erzeugt, dass eine Datei noch verfügbar bleibt. Diese muss nur auf eine ausreichende Anzahl von Peers verteilt sein. Damit wird zusätzlich erreicht das die Netzwerklast zwischen den Peers aufgeteilt wird.



**Abbildung 6: Funktionsweise eines Torrent**

In Abbildung 6 ist eine beispielhafte Funktionsweise des Torrent-Protokolls dargestellt. Ein Rechner besitzt eine vollständige Datei, die er im Netzwerk zur Verfügung stellt. Die Datei wird in vier kleinere Parts geteilt (A, B, C, D). Diese werden an vier Interessenten verteilt, die jeweils einen vollständigen Part der Datei besitzen. Sobald jeder einen Part heruntergeladen hat, kann der jeweilige Part bei diesem heruntergeladen werden. So kann der Rechner 03 den Part A vom Rechner 01 zusätzlich herunterladen, sodass alle Teilnehmer schon zwei Parts besitzen und den Server entlasten.

Wie bereits erwähnt wird das Protokoll immer noch weiterentwickelt und um weitere Funktionen erweitert. So wird das Protokoll heute neben dem bekannten Filetransfer auch noch für Anwendungen wie beispielsweise

- Videostream und
- Private Torrents

verwendet.

Das Protokoll wurde außerdem so weiterentwickelt, dass es im Stande ist sogenannte Leecher (Clients die nur Runterladen, aber nicht die Dateien wieder zur Verfügung stellen) zu identifizieren und aus dem Netz auszuschließen.

## 2.7 Kodierungstheorie

Die Kodierungstheorie beschäftigt sich mit fehlererkennenden und -korrigierenden Codes.

Dies kommt sowohl in der Speicherung wie auch beim Übertragen von Informationen zur Geltung. Eine so codierte Datei kann beim Auslesen geprüft werden. Sollte dann ein Fehler festgestellt werden kann die Datei bei Verwendung eines fehlerkorrigierenden Codes eventuell wiederhergestellt werden. Dabei werden der Information weitere codespezifischen Informationen hinzugefügt. Dies hat zur Folge, dass die gespeicherte oder übertragene Information werden größer ist. Auch der Rechenaufwand für den Empfänger wird größer, da er zusätzliche Informationen herausrechnen muss, um an die eigentliche Nachricht zu gelangen.

### 2.7.1 Fehlererkennende Codes

Fehlererkennende Codes bestehen in den meisten Fällen aus einer Prüfsumme, die am Ende der Information nach einem festen Rechenalgorithmus berechnet wird. Der Empfänger berechnet nach demselben Algorithmus die Prüfsumme und vergleicht diese. Ist das Ergebnis identisch wurde die Information korrekt übertragen. Diese Methoden haben alle eine Gemeinsamkeit: Sie können einen Fehler identifizieren, die empfangene Information muss jedoch verworfen und erneut gesendet werden, da diese korrumpiert ist.

Im Folgenden werden einige Varianten von Prüfsummen vorgestellt und die Funktionsweise kurz erläutert. Letztendlich sollen noch einige Vor- und Nachteile der jeweiligen Methoden beschrieben werden.

#### 2.7.1.1 Quersummenverfahren

Die einfachste Methode einer Prüfsumme ist die Bits der Information zu addieren. Das Ergebnis ist die Prüfsumme. Diese Methode ist jedoch nicht sicher und auch nur bedingt dazu in der Lage einen Fehler in der Information zu erkennen.

Nehmen wir als Beispiel eine Zeichenkette aus folgenden Informationen:

1; 3; 6; 10; 84; 52

Nach der Addition dieser Zahlen beträgt die Prüfsumme 156.

Ist die Reihenfolge der Informationen relevant hat diese Methode keine Möglichkeit mehr einen Fehler zu identifizieren. Das Ergebnis wäre, unabhängig der Reihenfolge das gleiche.

Ein weiteres Problem dieser Methode: Treten zwei Fehler auf können diese sich aufheben und die gleiche Prüfsumme zum Ergebnis haben, zum Beispiel wird aus der 6 eine 7 und aus der 84 eine 83 interpretiert. Addiert ergibt das die gleiche Prüfsumme, obwohl die Informationen nicht mehr stimmen. Das macht das Verfahren unsicher gegen Fälschungen und Manipulationen.

Eine bessere Methode bietet hier das gewichtete Quersummenverfahren. Hierbei wird den einzelnen Werten eine Gewichtung festgelegt. Durch das Multiplizieren der Werte mit der Gewichtung und dem anschließenden Addieren der Ergebnisse wird ein besseres Ergebnis erzielt.

	$X_0$	$X_1$	$X_2$	$X_3$
Gewichtung	1	2	4	8
Information	5	12	31	50

**Tabelle 3: Gewichtete Quersumme**

Im obigen Beispiel würde die gewichtete Quersumme 136 800 betragen. Wenn durch einen Übertragungsfehler die Stellen  $X_1$  und  $X_2$  vertauscht empfangen werden dann ergibt die Quersumme 133 000. Der Empfänger erkennt somit, dass die empfangenen Informationen nicht den gesendeten entsprechen, kann diese verwerfen und neu beim Sender anfordern.

### 2.7.1.2 Paritätsbit

Beim Paritätsbit-Verfahren wird einem Bitstrom ein weiteres Bit hinzugefügt. Dieses Bit zeigt an, ob die Summe der Einsen im Bitstrom gerade ist oder nicht.

Ist die Summe gerade so kann das Paritätsbit mit einer eins ungerade bedeuten, dann handelt es sich um eine Even-Parity. Bedeutet eine eins handelt es sich dabei um eine Odd-Parity.

Summe der Einsen eines Informationsworts	Wert des Paritätsbit bei Even-Parity	Wert des Paritätsbit bei Odd-Parity
Gerade	0	1
Ungerade	1	0

**Tabelle 4: Werte des Paritätsbit**

Der Empfänger muss nur die empfangenen Informationen summieren und das Ergebnis mit dem gesendeten Paritätsbit vergleichen. Ist das Ergebnis unterschiedlich ist die empfangene Information fehlerhaft. Mit diesem Verfahren lässt sich eine ungerade Bitfehleranzahl identifizieren. Sind zwei Bits fehlerhaft wird das Paritätsbit korrekt berechnet und die Information fälschlicherweise akzeptiert.

Eine Erweiterung des Paritätsbit-Verfahren, die auch eine größere Sicherheit bietet und teilweise sogar Fehlerkorrigierend eingesetzt werden kann, ist die mehrdimensionale Paritätskontrolle, auch Blockparität genannt. Hierbei werden die Informationsbits in einer Matrix angeordnet. Die Paritätsbits werden sowohl bei den Spalten als auch bei den Zeilen berechnet und hinzugefügt.

0	1	0	1	0
1	0	1	0	1
0	1	1	1	1
0	0	0	1	1
1	0	1	1	1

**Tabelle 5: Fehlerkorrektur mit Paritätsbit**

Im Beispiel wird eine gerade Parität verwendet und den Informationsbits hinzugefügt (grün). Hier ist das 3. Bit in der zweiten Nachricht falsch. Über die Koordinaten der Paritätsbits lässt sich eine genaue Lokalisierung des Fehlers durchführen. Das betroffene Bit kann nun durch invertieren korrigiert und die Information wiederhergestellt werden. Das blaue Feld ist die sogenannte Kreuzparität. Diese ergibt sich aus den Spalten- und Zeilen-Paritätsbits. Mit diesem Kreuzparitätsbit lassen sich selbst Fehler in den Paritätsbits identifizieren und korrigieren.

Aber auch diese Methode hat ihre Grenzen: Ab zwei Bitfehler lässt sich beispielsweise die Korrektur nicht mehr durchführen. Bei Vierbitfehlern kann die Konstellation sogar so auftreten, dass die Paritätsbits keinen Fehler erkennen lassen.

### **2.7.1.3 Zyklische Redundanzprüfung (CRC)**

Mit einer zyklischen Redundanzprüfung werden der zu sendenden Nachricht Prüfbits hinzugefügt. Diese Prüfbits berechnen sich aus der Nachricht und einem vorher definierten Generatorpolynom verrechnet. Der Rest aus der Polynomdivision dieser beiden Werte wird

der Nachricht hinzugefügt. Der Empfänger führt die Berechnung ebenfalls durch und vergleicht die Werte miteinander. Sind die Werte nach dem Vergleich nicht identisch ist die Empfangene Nachricht nicht korrekt und muss erneut gesendet werden.

Im Folgenden soll der Berechnungsalgorithmus beispielhaft vorgestellt werden.

Gegeben ist das Generatorpolynom  $g = x^5 + x^2 + x + 1$

Dies entspricht dem Wort 1 0 0 1 1 1

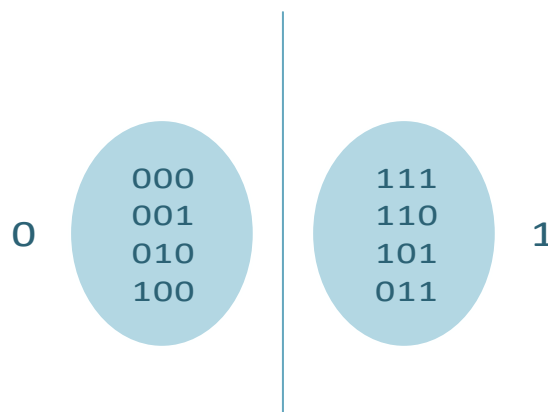
Und eine Nachricht: 1 0 0 1 0 1 1 1 0 0 1 1 1 0 1

- 1) Die Nachricht wird mit  $x^5$  (höchster Grad des Generatorpolynoms) multipliziert (5 Nullen an die Nachricht anhängen)
- 2) Das Ergebnis wird durch das Generatorpolynom dividiert und der Rest gebildet.
- 3) Der Rest zu dem Ergebnis aus Schritt 1 wird hinzuaddieren.

Das Ergebnis ist dann die Nachricht und der Rest aus der Polynomdivision. Diese Bits müssen nun an den Empfänger gesendet werden.

## 2.7.2 Fehlerkorrigierende Codes

Das Ziel der Fehlerkorrigierenden Codes ist die empfangene Nachricht nicht nur darauf zu überprüfen, ob sie korrekt ist, sondern auch einzelne Informationen der Nachricht korrigieren zu können. Einen solchen Code haben wir schon im Kapitel Paritätsbit kennengelernt. Eine Wichtige Messeinheit in der Codierungstheorie, besonders bei fehlererkennenden Codes, ist die sogenannte Hamming Distanz.



**Abbildung 7: Bitkombination**

Nehmen wir an wir senden drei Bits an den Empfänger. Als gültige Information sind aber nur die beiden Kombinationen „000“ und „111“ möglich. Der Empfänger kann, wenn er eine Nachricht erhält, die nicht einer gültigen Kombination entspricht, einen Fehler in der Übertragung identifizieren. Wenn der Empfänger nun die Signalfolge „001“ empfängt kann er

diese Signalfolge in eine gültige korrigieren. Die Wahrscheinlichkeit, dass nur ein Bit fehlerhaft empfangen worden ist, ist höher, als dass zwei Signale fehlerhaft sind. Die Nachricht kann somit zu „000“ korrigiert werden. Dieser Code wird Hamming-Code (3,1) genannt und ist der kürzeste seiner Art.

Die Hamming-Distanz berechnet sich daran wie viele Bits geändert werden müssen, um von einem Gültigen Wert zum anderen zu gelangen. Im Beispiel werden alle drei Bits geändert. Somit ist die Hamming-Distanz drei. An diesem Wert lassen sich die fehleridentifizierenden oder fehlerkorrigierenden Codes sowie die Anzahl der zuverlässig korrigierten Bits errechnen. Diesbezüglich ist die Hamming-Distanz relevant. Die Formel dafür ist:

$$\text{Korrekturbits} = \frac{HD_{\min} - 1}{2}$$

$HD_{\min}$  entspricht dem Mindestabstand zwischen zwei gültigen Signalen. Im Beispiel wäre, das der Wert 3. Die Werte werden nun in die Formel eingesetzt, das Ergebnis ist dann 1. Dies bedeutet, dass ein solcher Code 1-Bit Fehler korrigieren kann. Im weiteren Verlauf sollen effektivere Codes vorgestellt werden.

Weiter wird bei den fehlerkorrigierenden Codes die sogenannte Code Ratio wichtig. Dieser gibt das Verhältnis, wie viele Nutzbits (Nachricht) und wie viele Sicherungsbits der Code enthält, an. Als Beispiel verwenden und senden wir ein Bit als Nutzbit, und zwei Bits zusätzlich als Sicherungsbits. Daraus ergibt sich folgende Formel:

$$\text{Fehler Ratio} = \frac{\text{Nutzbits}}{\text{Nachrichtlänge}} = \frac{1}{3}$$

Je kleiner die Fehler Ratio desto besser können Fehler korrigiert werden. Dies bedeutet jedoch, dass die Gesamtmenge der zu sendenden Bits höher wird.

### **2.7.2.1 Hamming-Code**

Der Hamming-Code basiert auf den Paritätsbits, welche bereits in Kapitel 2.7.1.2 erläutert wurden. Bei diesem Code kommen mehr als nur ein Paritätsbit zum Einsatz. Dies ermöglicht nicht nur eine Fehlererkennung, sondern auch die Korrektur der Fehler einzelner Bits.

Abhängig von der Anzahl genutzter Datenbits werden entsprechende Paritätsbits dem Datenwort hinzugefügt. In der folgenden Tabelle sind Kombinationen bis zu Datenbitgröße von 247 Bit dargestellt.

Parameterkombination bei Hamming-Codes		
n	k	$N = n + k$
Datenbits (Datenwort)	Paritätsbits	Nachrichtenbits (Gesamtlänge)
1	2	3
4	3	7
11	4	15
26	5	31
57	6	63
120	7	127
247	8	255

**Tabelle 6: Parameterkombination bei Hamming-Codes**

Die Position der Paritätsbits ist in dem Code immer fest. Diese werden grundsätzlich an die Positionen geschrieben deren Stelle eine Zweierpotenz ist, also 1, 2, 4, 8, 16, ...

Daraus ergibt sich am Beispiel einer 8-Bit Datenübertragung, die mit 4 zusätzlichen Paritätsbits abgesichert werden muss, folgende Aufteilung.

Stelle	1	2	3	4	5	6	7	8	9	10	11	12
P/D	P <sub>1</sub>	P <sub>2</sub>	D <sub>1</sub>	P <sub>3</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	P <sub>4</sub>	D <sub>5</sub>	D <sub>6</sub>	D <sub>7</sub>	D <sub>8</sub>

**Tabelle 7: Positionen der Paritätsbits**

Die Paritätsbits werden wie folgt berechnet: Um P<sub>1</sub> zu erhalten fängt man an erster Stelle an. Dabei handelt es sich um das Paritätsbit selbst. Paritätsbits werden für die Berechnung nicht mitangezogen, der Wert bleibt erstmal offen. Dann wird ein Bit übersprungen und das nächste Bit für die Berechnung hinzugezogen. Dieser Vorgang wiederholt sich bis zum Ende des Datenwortes. Daraus ergeben sich folgende, für P<sub>1</sub> zu verwendenden Bits: 1, 3, 5, 7, 9, 11. Ergibt die Summe der Bits eine ungerade Anzahl Einsen wird das Paritätsbit auf „1“ gesetzt, sonst auf „0“.



Um  $P_2$  zu berechnen beginnt man an der zweiten Stelle und verwendet zwei Bits für die Berechnung. Zwei Bits werden für die Berechnung übersprungen und die nächsten zwei verwendet. Daraus ergeben sich folgende, für  $P_2$  zu verwendende, Bits: 2, 3, 6, 7, 10, 11. Da hier das zweite Bit das Paritätsbit ist kann dieses nicht zur Berechnung hinzugezogen werden.

Für  $P_3$  müssen die nächsten 4 Bits zur Berechnung verwendet werden, 4 Bits übersprungen und wieder 4 Bits verwendet werden. Bei  $P_4$  werden im Wechsel alle 8 Bits verwenden und übersprungen.

Zusammengefasst werden folgende Bits für die Paritäten verwendet:

$$P_1 = 1, 3, 5, 7, 9, 11$$

$$P_2 = 2, 3, 6, 7, 10, 11$$

$$P_3 = 4, 5, 6, 7, 12$$

$$P_4 = 8, 9, 10, 11$$

Wenn die Übertragung der Nachricht einen Fehler hat wird mit Hilfe des Codes dieser Fehler identifiziert und durch einen Bitkipper korrigiert. Der Code ist jedoch nicht mehr anwendbar, wenn die übertragene Nachricht mehr als einen Bitfehler aufweist. Zwei Fehler können noch identifiziert werden, sodass die Nachricht beim Sender nochmal angefordert werden kann.

Schauen wir uns anhand eines Beispiels an, wie die Korrektur funktioniert.

Folgende Nachricht soll übertragen werden: 0010 0110

Nach hinzufügen der Paritätsbits zu dieser Nachricht sieht die codierte Nachricht folgendermaßen aus: 1101 0100 0110

Die unterstrichenen Bits stellen die Paritätsbits dar.

Der Empfänger erhält aber folgende Bits: 1101 1100 0110. Das unterstrichene Bit stellt den Fehler dar. Der Empfänger berechnet nun ebenfalls die Paritätsbits für die empfangenen Datenbits. Hier fällt dem Empfänger auf, dass  $P_1$  und  $P_3$  nicht korrekt sind. Er addiert nun die Stellung der Paritätsbits, also 1+4, und erhält das fehlerhafte Bit, in diesem Beispiel die Stelle 5. Nun kann das Bit an dieser Stelle korrigiert werden.

### **2.7.2.2 Fountain-Codes**

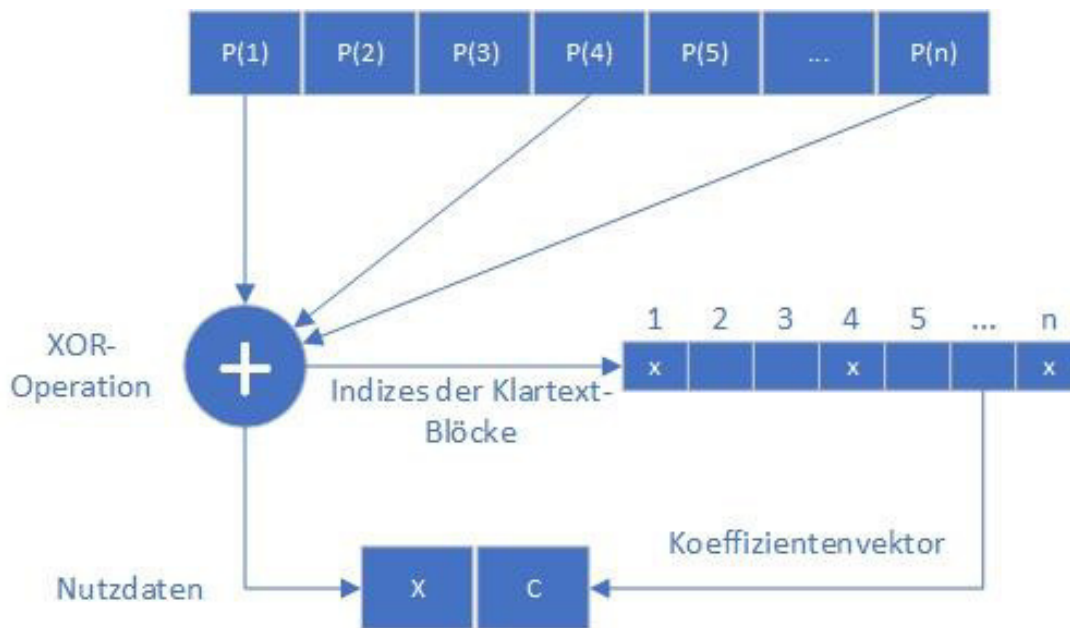
Unter den Fountain-Codes sind verschiedene Codes gemeint. Sie besitzen die Eigenschaft beim Erhalt einiger Nachrichteninformationen die komplette Nachricht rekonstruieren zu können. Der Unterschied zu den bisher kennengelernten Fehlerkorrigierenden Codes ist,

dass bei den bisher kennengelernten Codes die Informationen empfangen werden müssen. Danach fand eine Auswertung statt. Bei einem erkannten Fehler werden die Informationen korrigiert. Bei den Fountain-Codes ist es nicht zwingend erforderlich, dass alle Informationen empfangen werden, sondern nur ausreichend viele Informationen. Ein weiterer Vorteil der Fountain-Codes ist, dass diese keine feste Größe der zu sendenden Nachricht vorschreiben.

Bekannte Vertreter von Fountain-Codes sind:

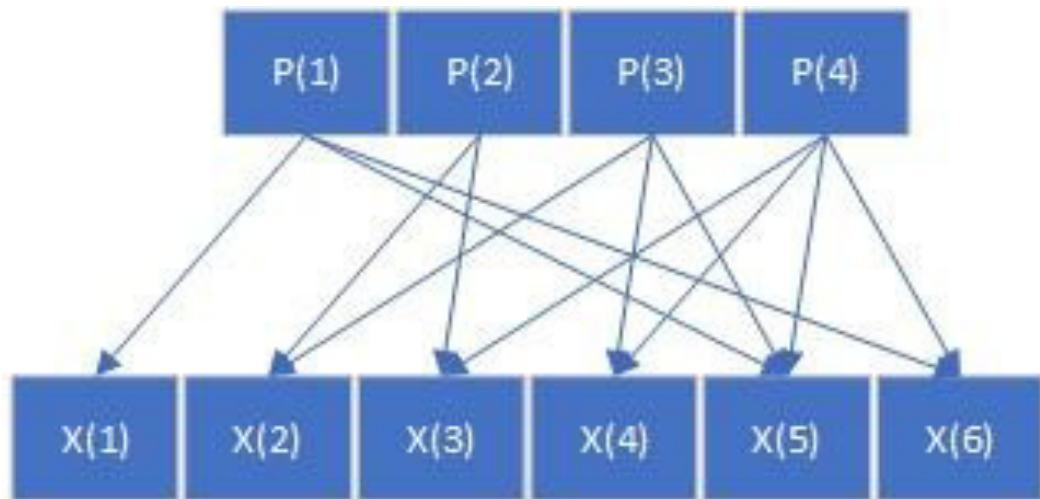
- Luby Transform-Codes (LT-Codes)
- Raptor Codes
- Online Codes

Im folgenden Abschnitt wird die generelle Funktionsweise der Fountain Codes beschrieben. Die oben beschriebenen Vertreter werden letztendlich miteinander verglichen und die jeweiligen Unterschiede gegenübergestellt.



**Abbildung 8: Erstellung eines Codierten Pakets**

Als erstes wird die zu sendende Nachricht in gleich große Blöcke unterteilt ( $P(n)$ ). Die Nachrichtenpakete werden dann nach einem festen Schema, der sogenannten Degree-Distribution, miteinander verknüpft. Hier kommt die XOR-Verknüpfung zum Einsatz. Das Ergebnis der Operation ist ein Codierter Nachrichtenblock, welcher der Koeffizientenvektor angehängt wird. Dieser Vektor gibt dem Decoder an, aus welchen ursprünglichen Nachrichten das codierte Wort besteht. Diese Information ist unerlässlich für den Decodiervorgang. Zusätzlich sei erwähnt, dass jedes Packet mehrfach für die Berechnung verwendet werden kann. Oft wird ein Nachrichtenpaket uncodiert weitergegeben, um den Startpunkt der Decodierung zu vereinfachen.



**Abbildung 9: Beispiel einer Codierten Nachricht**

Als Beispiel nehmen wir eine Nachricht, bestehend aus 4 Quellblöcken, und Codieren diese wie in Abbildung 9 dargestellt. Daraus ergeben sich folgende Zuordnungen:

$$p1 = x1$$

$$p2 \oplus p3 = x2$$

$$p2 \oplus p4 = x3$$

$$p3 \oplus p4 = x4$$

$$p1 \oplus p3 \oplus p4 = x5$$

$$p1 \oplus p4 = x6$$

Die empfangene Nachricht X1 kann direkt in P1 decodiert werden. Die anderen Blöcke werden mithilfe der decodierten Nachricht verarbeitet. Das Ergebnis besteht aus den XOR-Verknüpften Blöcken. So kann P4 berechnet werden, indem X6 mit P1 XOR verknüpft wird. Zudem werden nicht alle Codierten Blöcke benötigt, um die Nachricht vollständig wiederherzustellen.



## 3 Stand der Technik

Die Entwicklung des Informationsaustauschs von Fahrzeugen wird schon seit einigen Jahren betrieben. Ein großes Problem ist die Umsetzung der Ideen durch die unterschiedliche Herangehensweise der Hersteller. Um eine möglichst einheitliche Umsetzung der Hersteller zu gewährleisten haben sich Arbeitsgruppen etabliert, die einen Standard definieren sollen. In diesen Arbeitsgruppen sind Hersteller, Zulieferer und Forschungsinstitute vertreten. Aktuell gibt es weder einen internationalen Standard noch eine einheitliche Arbeitsweise. Aktuell beschäftigen sich drei Arbeitsgruppen mit dem Thema: Das Car2Car-Communication-Consortium (C2CCC) in Europa, sowie das Research and Innovative Technology Administration (RITA) in Amerika und Japan, in welchem das Ministerium für Land, Infrastruktur, Transport und Tourismus die Forschung in diesem Gebiet vorantreibt.

### 3.1 Ziele

Mit Einführung der Technologie werden verschiedene Ziele verfolgt. Das primäre Ziel ist die Erhöhung der Straßenverkehrssicherheit. Durch Teilen der Sensordaten sollen andere Fahrzeuge bereits gewarnt werden, noch bevor die eigenen Fahrzeugsensoren die Gefahrensituation erkennen können.

Ein weiteres Ziel ist die Verkehrsführung in Großstädten zu entzerren und den Verkehrsfluss zu optimieren. Damit verbunden soll auch der Umweltschutz verbessert werden. Ein gerade entstehender Stau soll rechtzeitig identifiziert und die Fahrzeuge auf alternative Routen navigiert werden.

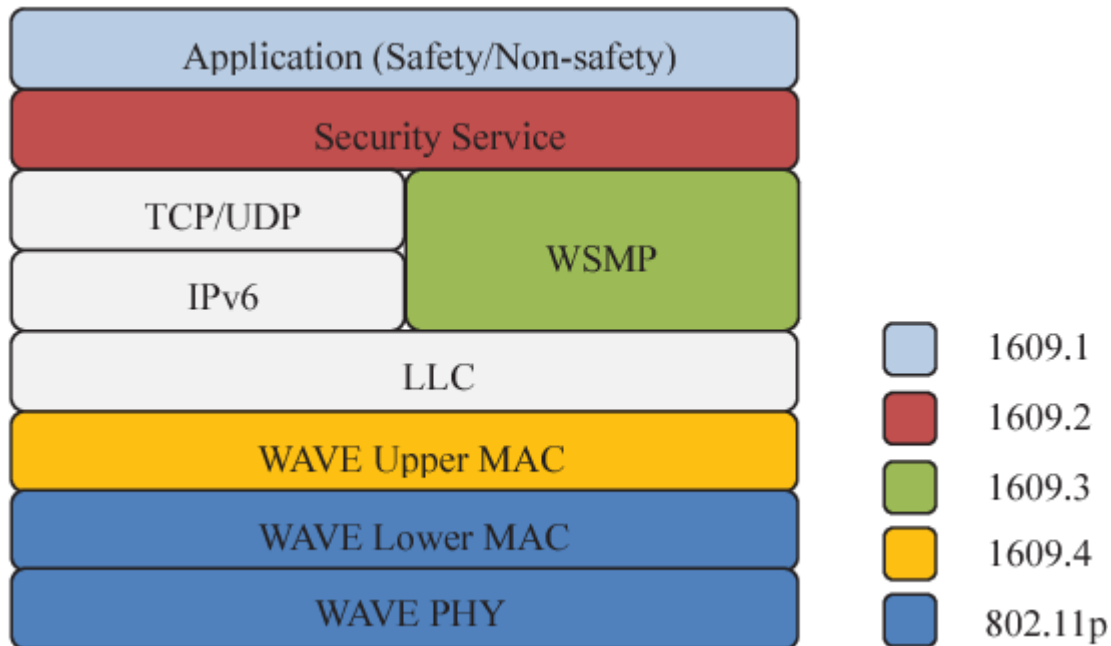
Der weitere Ausbau des Bereichs Entertainment ist ein untergeordnetes Ziel der Technologie. Ideen wären hier das Versenden von Nachrichten zwischen den Fahrzeugen oder aber das Spielen mit anderen am Verkehr teilnehmenden Fahrzeugen. Denkbar wäre auch dem Fahrer eine ortsbezogene Werbung anzuzeigen.

In Bezug auf die Standards selbst ist das Ziel eine latenzarme Kommunikation zu gewährleisten. Neben der Latenzzeit ist auch der Aspekt der Überlastung der Kommunikationskanäle ein wichtiger Faktor für die Technologie.

### 3.2 Standards

Wie bereits oben beschrieben gibt es aktuell keine internationalen Standards, sodass hier verschiedene Ansätze verfolgt werden. So ist beispielsweise das priorisierte Mittel der C2CCC die Kommunikation auf Basis des WLAN Standards zu entwickeln. Dem gegenüber

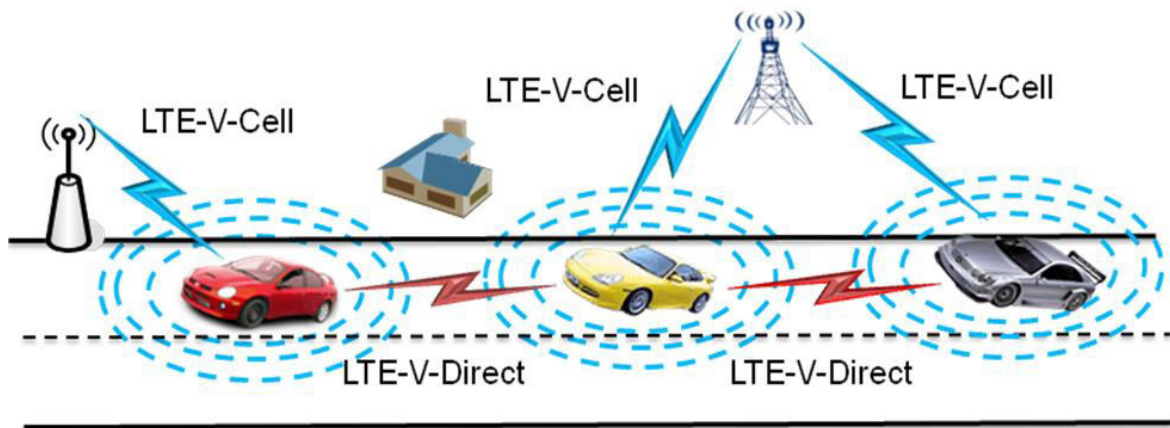
steht die Entwicklung der RITA, welches sich mehr auf die Nutzung des Mobilfunknetzes konzentriert. Aus beiden Herangehensweisen sind bereits Ideen entstanden und zur Demonstration umgesetzt worden. Da es sich hier nicht um einen Standard handelt ist nicht sichergestellt das Fahrzeuge unterschiedlicher Hersteller miteinander kommunizieren und Informationen austauschen können.



**Abbildung 10: WAVE Protokoll (2)**

Der WAVE (Wireless Access in Vehicular Environment) basiert auf Kommunikation der Fahrzeuge mit Einrichtungen am Straßenrand, den Roadside Units (RSU) oder den Intelligent Roadside Stations (IRS). Für die Kommunikation zwischen den Fahrzeugen direkt wird momentan das Vehicular Ad-hoc Network (VANET) entwickelt. Dieses ist zu einer Echtzeitkommunikation zwischen den Fahrzeugen im Stande.

Beide benutzen eine Weiterentwicklung des IEEE 802.11 Standards, welches in privaten Haushalten oft Anwendung findet. Hier wird in den weiter oben liegenden Schichten zwischen unterschiedlichen Anwendungsfällen unterschieden. Die nicht sicherheitsrelevanten Anwendungen setzen auf das bereits bekannte TCP-IP Protokoll auf, wohingegen die sicherheitsrelevanten Anwendungen auf das neue WSMP (WAVE Short Message Protocol) aufsetzen. Dieses Protokoll wurde entwickelt, um Priorisierungen der Nachrichten sowie Zeitkritische Kommunikation zu ermöglichen.



**Abbildung 11: LTE-V Kommunikation (3)**

Neben dem Ansatz der Kommunikation über die WLAN Schnittstelle gibt es auch den Ansatz der Nutzung des Mobilfunknetzes. Bei dieser Variante kommunizieren Fahrzeuge über die Sendemasten miteinander. Diese können auch direkt miteinander kommunizieren, wenn kein Sendemast in Reichweite ist. Das Ganze wird unter dem Standard LTE-Vehicular (LTE-V) oder auch Cellular V2X (C-V2X) entwickelt und von einigen Herstellern bereits in den Fahrzeugen verbaut.

Ein großes Problem bei LTE-V ist, dass es nicht ausreichend ist allein die Fahrzeuge mit der Technologie auszustatten, vielmehr muss auch die Infrastruktur dafür ausgelegt werden. Ein solcher Ausbau ist einerseits sehr kostspielig und andererseits benötigt es sehr viel Zeit bis die Technologie flächendeckend genutzt werden kann.

### 3.3 WAVE und VANET

Der Ansatz bei WAVE war eine Kommunikation zwischen Fahrzeugen in unmittelbarer Nähe zu ermöglichen. Trotz dessen soll dies gegen andere Funksignale störresistent sein und eine kurze Latenzzeit aufweisen. Parallel wird der Ansatz VANET entwickelt. Dieses ist für die Kommunikation zwischen Fahrzeugen untereinander gedacht, basiert jedoch auf derselben Technologie wie WAVE, nämlich der Weiterentwicklung von IEEE 802.11.

Bereits 1999 wurde vom Federal Communication Commission (FCC) in Amerika das Frequenzband zwischen 5,850 GHz und 5,925 GHz für verkehrstechnische Kommunikation reserviert. In Europa wurde nach Vorstellung des European Telecommunications Standards Institute (ETSI) das gleiche Frequenzband und die gleiche Kanaleinteilung verwendet. (3)

Folgende Tabelle zeigt einige Eckdaten des IEEE802.11p Standard, welcher die Basis für WAVE und VANET darstellt.

<b>Standard</b>	IEEE802.11p
<b>Frequenzbereich</b>	5,850 GHz – 5,925 GHz
<b>Anzahl der Kanäle</b>	sieben 10 MHz Kanäle
<b>Modulation</b>	OFDM
<b>Datenrate pro Kanal</b>	3 – 27 Mbit/s
<b>Reichweite</b>	Bis 1000m
<b>Codierung</b>	Faltungscodierung
<b>Multiplexing</b>	TDM

**Tabelle 8: Spezifikationen IEEE802.11p (4)**

Zusätzlich ist der Standard für Geschwindigkeiten bis 200 km/h ausgelegt.

### 3.4 C-V2X

Das C-V2X oder auch LTE-V2X ist eine auf LTE basierende Technik, die für den Einsatz im Fahrzeug angepasst wurde. In erster Linie ist hier eine Kommunikation über den Sendemast reguliert. Dieser reguliert zudem wann die Fahrzeuge miteinander kommunizieren können. Damit wird sichergestellt, dass die Kanäle nicht überlastet werden und eine hoch priorisierte Nachricht den Empfänger tatsächlich in der geforderten Zeit erreichen kann.

Wenn kein Sendemast in Reichweite vorhanden ist können die Fahrzeuge auch eine direkte Verbindung aufbauen und Informationen miteinander austauschen. Dafür wird dann auf das Interface PC5 umgestellt.

Ein Vorteil dieses Ansatzes ist die Aufnahme weiterer Kommunikationsteilnehmer im Netzwerk. Hier wäre die Integrierung von Fußgängern oder Fahrradfahrern über die Smartphones in das Netzwerk denkbar. Auch Warnungen könnten an diese gesendet werden.

Die Herausforderung hier ist die Definierung der Standards. Dies nimmt aufgrund der hohen und verbreiteten Nutzung mehr Zeit in Anspruch. Des Weiteren muss die Infrastruktur an



die neuen Standards angepasst werden. Damit verbunden ist die Anpassung der Hardware. Dies ist kostspielig und zeitaufwendig.

<b>Standard</b>	C-V2X
<b>Frequenzen</b>	LTE-Frequenzen + 5,9 GHz
<b>Modulation</b>	SC-FDM
<b>Codierung</b>	Turbo-Codes
<b>Multiplexing</b>	TDM und FDM

**Tabelle 9: Spezifikationen C-V2X (4)**

Die Entwicklung des C-V2X ist als Weiterentwicklung der IEEE 802.11p anzusehen. Dies wird daran deutlich, dass die Entwicklungen des 802.11p Standards in die Entwicklung des C-V2X eingeflossen sind. Die Frequenzen, die für verkehrstechnische Kommunikation reserviert worden sind, werden auch von diesem Standard unterstützt.

### 3.5 Zusammenfassung

Im Vergleich hat der C-V2X Standard einen besseren Ansatz, insbesondere bei Betrachtung der Kompatibilität zu 802.11p. Leider ist die Umsetzung mit viel Aufwand verbunden. Bis die Technologie eingesetzt werden kann wird noch einige Zeit vergehen, da für den Nutzer die Anwendung erst ab ca. 20 - 30 % Marktdurchdringung spürbar ist.

## 4 Ausarbeitung

Im Kapitel vorher wurden einige Technik vorgestellt, die im Bereich des Personal Computer schon seit längerem Stand der Technik sind. Ziel der Arbeit ist zu analysieren, wie die Technologie im Einsatzgebiet der C2C-Kommunikation angewandt werden und die Versendung größerer Datenmengen über C2C-Schnittstellen erfolgen kann.

Eine aus dem Mobiltelefonsegment bekannte Entwicklung ist die Geräte nachträglich mit den aktuellsten Softwareupdates zu versorgen, das sogenannte Update-Over-the-Air (OTA). Dieses wird auch für die Automobilhersteller immer wichtiger, da immer weitere Funktionen in das Fahrzeug eingebaut und die Funktion der Steuergeräte komplexer werden. Um die Funktionen zu gewährleisten ist eine Vernetzung der Steuergeräte notwendig, welche wiederum Fehler- und Optimierungspotential mit sich bringen. (6)

Solche Firmwareupdates für ein Steuergerät können mehrere Gigabyte groß sein. Eine solche Datei über das Mobilfunknetz vom Betreiber herunterladen kann mehrere Stunden dauern. Zusätzlich kann aufgrund der Mobilität des Fahrzeugs die Verbindung unterbrochen werden, was einen erneuten Start des Downloads zur Folge hat. Hinzu kommt, dass die Kosten für die Nutzung der Zugangspunkte des Internets bezahlt werden müssen, sei es der Kunde oder der Fahrzeughersteller. Die Kosten können sich angesichts der Anzahl der Fahrzeuge auf den Straßen schnell im sechsstelligen Bereich bewegen.

Eine kostengünstigere Möglichkeit wäre es, wenn die Fahrzeuge untereinander die Softwaredateien austauschen, wenn diese sich begegnen. Dafür bietet sich die gerade in der Entwicklung befindliche C2X-Schnittstelle an.

### 4.1 Update OTA

Update-over-the-Air ist ein Softwareupdate auf ein Endgerät ohne Verbindung mit einer Programmierstation wie beispielsweise einem Computer. Im Segment der Mobiltelefone ist ein solches Verfahren seit Jahren üblich. Mit der Entwicklung des Autos vom reinen Fortbewegungsmittel zum mobilen und vernetzten Computer wird die Verarbeitung der Informationen im Steuergerät immer komplexer und größer.

Die damit verbundene Komplexität der Steuergeräte bedarf der kontinuierlichen Pflege der Software. Diese muss früher oder später auf das Steuergerät aufgespielt werden. Gegenwärtig ist die Firmware des Infotainment-Systems eines Fahrzeugs bis zu 20 Gigabyte groß. Wird die Firmware über das Mobilfunknetz mit einer konstanten Downloadgeschwindigkeit von 600 Mbit/s, welche einer maximalen Geschwindigkeit von 4G entspricht, heruntergeladen dauert dies ca. 10 Minuten.

Wenn diese Werte auf die Intervalle und Anzahl der Steuergeräte aufsummiert werden, kommt ein beachtliches Ergebnis raus. Hinzu kommt die Unterbrechung des Downloads, wenn die Verbindung zum Server getrennt wird. Ein erneuter Start des Downloads verzögert den gesamten Prozess.

Ein weiteres Problem einer serverbasierten Updateverteilung ist die Koordinierung einer sequenziellen Verteilung des Updates durch den Betreiber: Laden alle Fahrzeuge zeitgleich das Update herunter können die Server die Anfragen nicht mehr verarbeiten und das System bricht zusammen.

Eine aus dem Smartphone Segment gängige Methode für die Verteilung von Updates ist die Benutzer geografisch für das Update zuzulassen. So können Zugriffe auf den Server reduziert werden.

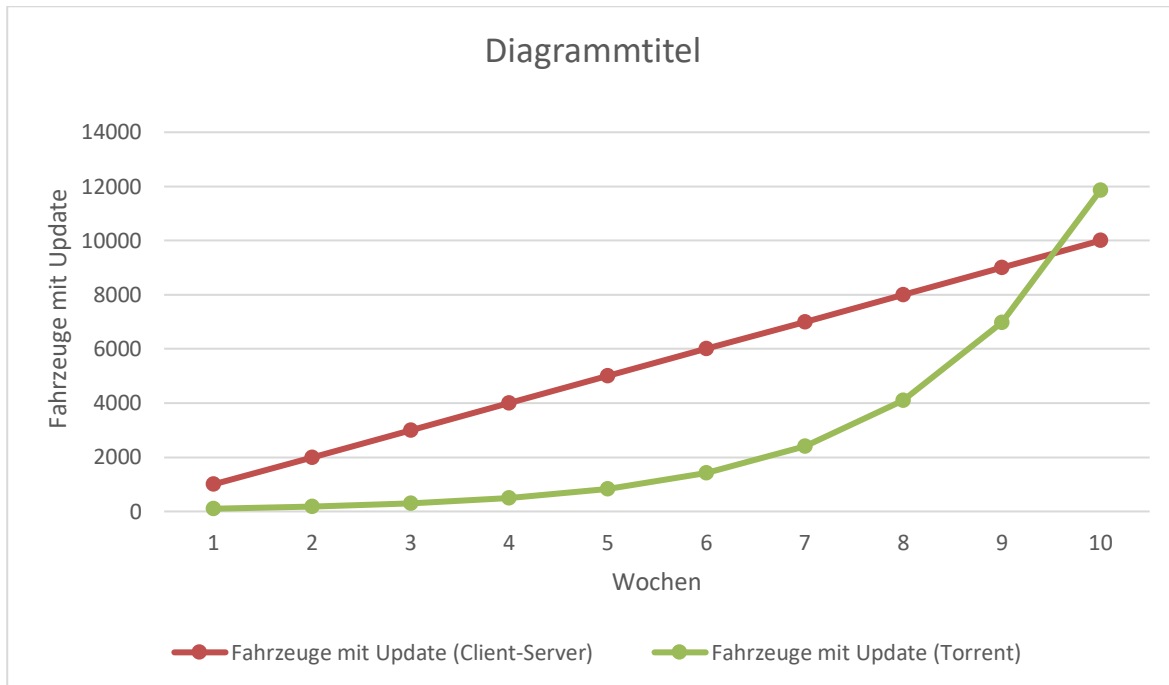
Diesem Problem kann das gängige Torrent-Protokoll Abhilfe schaffen. In weiteren Kapiteln wird analysiert, wie die Technik im Umfeld eines Fahrzeugs eingesetzt werden kann.

## 4.2 Torrent-Protokoll im Fahrzeugumfeld

Das Torrent-Protokoll wurde bereits in Kapitel 2.5 beschrieben. Das Protokoll setzt ein P2P-Netzwerk voraus. Somit ist kein Server für die Kommunikation und Verteilung von Daten mehr notwendig.

Die zu versendenden Daten werden für den Sendevorgang vorbereitet. Da im Protokoll vorgesehen ist, dass die Daten in kleinere Parts geteilt werden soll hier eine Analyse und Empfehlung zur Größe der Parts gegeben werden.

Ziel dieser Analyse ist einige wenige Fahrzeuge mit dem Update auszustatten und auf den Straßen an andere Fahrzeuge zu verteilen. Vergleicht man die Verbreitung des Updates, auf herkömmliche Weise, über die Server-Client Struktur und ein Torrent-Verfahren ist festzustellen, dass anfangs das Update über die Server-Client-Struktur mehr Fahrzeuge erreicht. Mit zunehmender Zeit wird jedoch das Verfahren vom Torrent-Protokoll überholt.



**Abbildung 12: Vergleich Verbreitung Update**

In Abbildung 12 ist ein möglicher Verlauf zur Verteilung des Updates zu sehen. Die rote Linie stellt den Verlauf über eine geregelte Server-Client-Struktur dar. Angenommen 1000 Fahrzeuge pro Woche erhalten das Update. Im Idealfall werden also 1000 Fahrzeuge pro Woche mit dem Update ausgestattet. Die grüne Linie zeigt den Verlauf über das Torrent-Protokoll. In diesem Beispiel sind 100 Fahrzeuge mit dem Update ausgestattet. Nehmen wir nun an dass jedes dieser Fahrzeuge das Update an durchschnittlich 0,7 andere Fahrzeuge pro Woche verteilt. Nach 10 Wochen erhielten mit dem Torrent-Verfahren mehr Fahrzeuge das Update als mit dem Server-Client Verfahren. Die Linie entwickelt sich weiterhin exponentiell.

Der Verlauf der grünen Linie lässt sich sehr simpel variieren, indem die Anzahl der mit dem Update ausgestatteten Fahrzeuge mit dem Update initial erhöht wird.

Bei der Torrent-Variante muss beachtet werden, dass der Datenaustausch nur durchgeführt werden kann, wenn sich die Fahrzeuge begegnen. Eine mögliche Lösung hierfür könnte die Versorgung der Fahrzeuge mit dem Update über den Server sein, jedoch nur in einer festgelegten Zeitspanne. So können auch Strukturschwache Gebiete abgedeckt und mit aktuellsten Updates versorgt werden.

Die Tatsache, dass sich Fahrzeuge begegnen müssen, birgt noch ein weiteres Problem. Begegnen sich zwei Fahrzeuge während der Bewegung so ist selten ausreichend Zeit für das Versenden von mehreren Gigabyte Daten. Es gibt mehrere Szenarien, auf welche Art sich Fahrzeuge begegnen. Die daraus resultierende Zeit für die Kommunikation ist unterschiedlich. Im nächsten Kapitel werden verschiedene Begegnungsmöglichkeiten vorgestellt.

### 4.3 Begegnungsarten

Die wohl gängigste Begegnung von Fahrzeugen ist im städtischen Verkehr in gleicher Fahrtrichtung. Die Fahrzeuge bewegen sich mit einer nahezu gleichen Geschwindigkeit in eine Richtung. Dies stellt das Beste Szenario dar, da hier die Zeit für einen Kommunikationsaufbau und den Datenaustausch miteinander optimal ist, selbst wenn beispielsweise eine Geschwindigkeitsdifferenz von 20 km/h auftritt. Angenommen Fahrzeug A fährt mit einer Geschwindigkeit von 60 km/h, Fahrzeug B mit 40 km/h.

Daraus ergibt sich eine Differenzgeschwindigkeit von

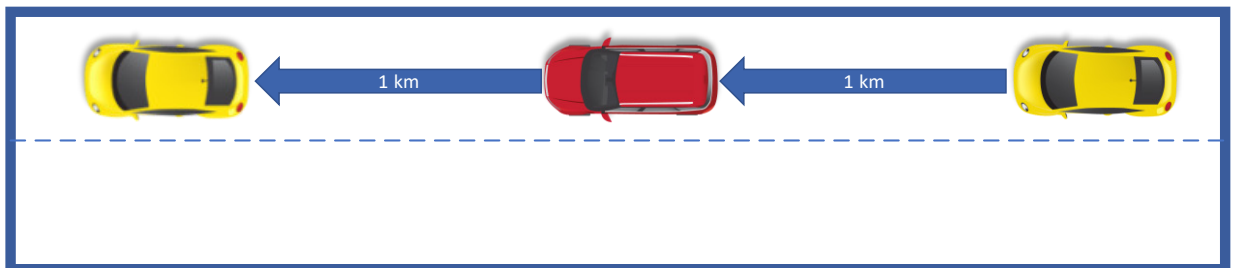
$$60 \text{ km/h} - 40 \text{ km/h} = 20 \text{ km/h}$$

Wird die Geschwindigkeit in Meter pro Sekunde umgerechnet erhält entspricht das:

$$\frac{20 \text{ km/h}}{3,6} = 5,55 \text{ m/s}$$

Der ITS-G5 Standard wird mit einer maximalen Reichweite von 1 km angegeben. Eine Kommunikation beginnt bereits, wenn Fahrzeuge sich aufeinander zubewegen und setzt sich fort, wenn diese sich wieder voneinander wegbewegen.

Die Abbildung 12 stellt beispielhaft diese Situation dar. Das gelbe Auto nähert sich mit einer Geschwindigkeit von 60 km/h dem roten Auto, welches sich mit einer Geschwindigkeit von 40 km/h bewegt. Nachdem das gelbe Auto das rote überholt hat entfernt es sich wieder von diesem.



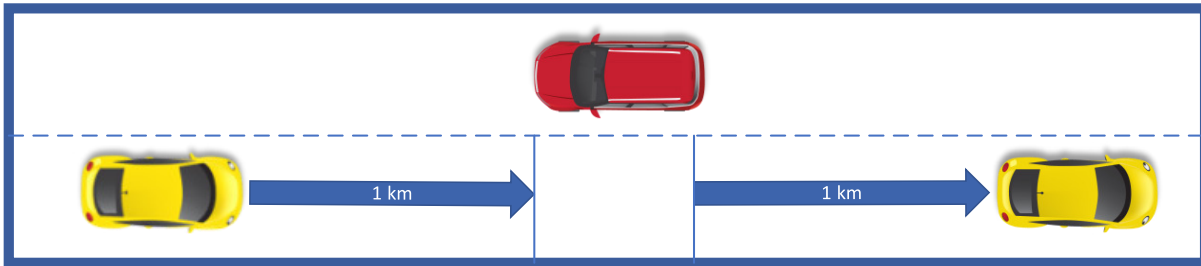
**Abbildung 13: Szenario 1 - Fahrt in gleiche Richtung**

Daraus ergibt sich ein Gesamtweg von 2 km, in dem beide Fahrzeuge miteinander kommunizieren können.

$$\frac{20000 \text{ m}}{5,55 \text{ m/s}} = 360,36 \text{ s} \equiv 6 \text{ min}$$

Die Zeit, in welcher die Fahrzeuge in Kommunikationsreichweite sind, beträgt 6 Minuten, eine ausreichende Zeit, um eine Menge Daten auszutauschen.

Eine zweite Möglichkeit der Fahrzeugkommunikation stellen zwei entgegenkommende Fahrzeuge in der Stadt dar.



**Abbildung 14: Szenario 2 - Fahrt in entgegengesetzte Richtung**

Wir nehmen eine Geschwindigkeit von 50 km/h für beide Fahrzeuge an so ergibt sich folgende Kommunikationszeit:

$$\frac{2 \text{ km}}{50 \text{ km/h} * 2} = 1,2 \text{ min}$$

Weitere Szenarien sind:

- Fahrten in gleicher Richtung auf der Autobahn
- Fahrten in entgegengesetzte Richtungen auf der Autobahn
- Warten an einer Ampel
- Vorbeifahren an einem stehenden bzw. wartenden Auto

Hier ergeben sich unterschiedliche Zeiten der Fahrzeugkommunikation. Das schlechteste eintreffende Szenario ist entgegenkommenden Fahrzeuge auf er Autobahn, da diese jeweils mit einer sehr hohen Geschwindigkeit fahren. Nehmen wir als Beispiel die Richtungsgeschwindigkeit von 130 km/h für beide Fahrzeuge. Die Kommunikationszeit beträgt hier:

$$\frac{2 \text{ km}}{130 \text{ km/h} * 2} = 0,27 \text{ min}$$

Bei einer Zeitspanne von weniger als 30 Sekunden wird es schwer eine Verbindung zwischen den Fahrzeugen aufzubauen und Nachrichten bzw. Daten auszutauschen.

Ein Sonderfall stellt das Szenario beim Warten an einer roten Ampel dar. Anders als alle anderen Szenarien bewegen sich die Fahrzeuge hier nicht. Die Rotphase einer Ampel kann nicht festgelegt werden, da diese je nach Knotenpunkt unterschiedlich ausfällt. Als Richtmaß kann die Richtlinie für Lichtsignalanlagen (RiLSA) hinzugezogen werden. In dieser ist eine maximale Umlaufzeit von 120s für die Lichtsignalanlage definiert. Bei Möglichkeit sollten 90s nicht überschritten werden. Die Mindestzeit beträgt 60s. (7)

Die folgende Tabelle zeigt die Kommunikationszeiten für die bereits beschriebenen Szenarien erneut auf:

Szenario (Geschwindigkeit Fahrzeug A / B)	Kommunikationszeit
Fahrt in gleiche Richtung (40 / 60)	6 Minuten
Fahrt in entgegengesetzte Richtung auf Autobahn (130 / 130)	0,27 Minuten
Fahrt in entgegengesetzte Richtung auf Landstraße (80 / 80)	0,45 Minuten
Fahrt in entgegengesetzte Richtung in der Stadt (50 / 50)	1,12 Minuten
Warten an einer Ampel	1,3 – 2 Minuten
Vorbeifahren an einem stehenden Fahrzeug (50 / 0)	2,44 Minuten

**Tabelle 10: Kommunikationszeiten für verschiedene Szenarien (Reichweite 1 km)**

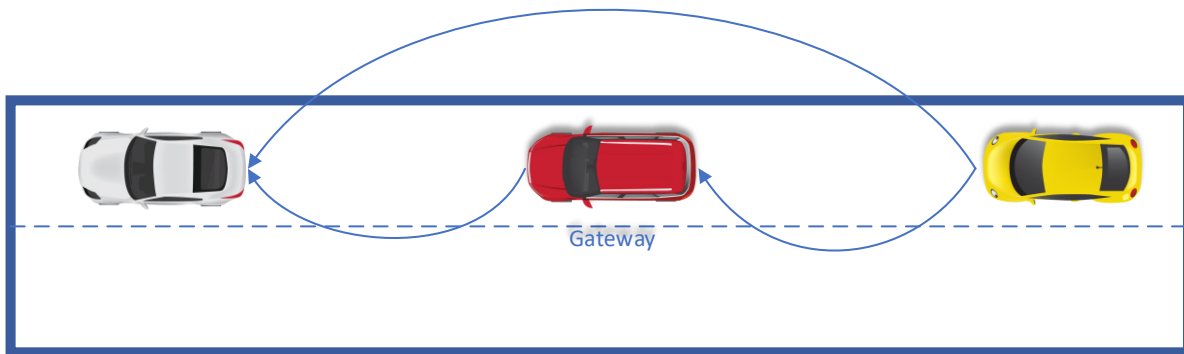
Die Kommunikation bei einer Entfernung von 1 km ist sehr optimistisch. Die Entfernung von 1 km wird in den Spezifikationen des Standards genannt, jedoch gilt dies nur bei direktem Sichtkontakt der miteinander kommunizierenden Fahrzeuge. In der Regel beträgt die Reichweite 300m.

Szenario (Geschwindigkeit Fahrzeug A / B)	Kommunikationszeit
Fahrt in gleiche Richtung (40 / 60)	1 Minute 5 Sekunden
Fahrt in entgegengesetzte Richtung auf Autobahn (130 / 130)	4,98 Sekunden
Fahrt in entgegengesetzte Richtung auf Landstraße (80 / 80)	8,1 Sekunden
Fahrt in entgegengesetzte Richtung in der Stadt (50 / 50)	12,96 Sekunden
Vorbeifahren an einem stehenden Fahrzeug (50 / 0)	25,92 Sekunden

**Tabelle 11: Kommunikationszeiten für verschiedene Szenarien (Reichweite 300 m)**

Um die Reichweite zu erhöhen sollen Fahrzeuge hier als Gateways dienen können. Dies bedeutet, dass mit einem Fahrzeug kommuniziert werden kann, auch wenn dieses nicht in

Reichweite ist. Die Kommunikation ist jedoch nur möglich solange ein drittes Fahrzeug zwischen beiden Kommunikationspartnern ist und beide in Reichweite sind.



**Abbildung 15: Szenario 3 - Kommunikation mit Gateway**

Die Gateway-Funktion muss nicht zwangsläufig von einem anderen Fahrzeug, sondern kann auch durch die RSUs übernommen werden. Auch die Sendemasten, die beim C-LTE verwendet werden können als Gateways fungieren, ohne das die Daten über das Internet ausgetauscht werden müssen.

## 4.4 Festlegung der Dateigrößen

Es wurden nun einige Szenarien definiert, in welchen Fahrzeuge miteinander kommunizieren können. Damit jedoch die Daten in dieser Zeit an das Zielfahrzeug übertragen werden können muss die Größe der einzelnen Dateipakete bestimmt werden.

Eine Möglichkeit wäre die Versendung der Datei als zum Beispiel Updatepacket. Der Nachteil dieser Methode: Bei Abbruch der Kommunikation vor Fertigstellung des Versands muss das gesamte Packet erneut gesendet werden. Andererseits können die Datenpakete in zig kleine Pakete, zu je 1 MB, aufgeteilt werden. Dies hat wiederum den Nachteil, dass die Pakete dann verwaltet werden müssen und der Overhead zu hoch wäre. Ziel ist es eine angemessene Größe für die Datenpakete sicherzustellen, sodass ein Versand während einer Kommunikation zweier Fahrzeuge abgeschlossen werden kann und kein unnötiger Überschuss an Datenpaketen erzeugt wird.

Für diese Ermittlung sollen auch die im vorherigen Kapitel beschriebenen Szenarien dienen. Mit den daraus resultierenden Zeiten sollen sinnvolle Dateigrößen konzipiert werden.

Da vor dem Senden der Datei zwischen den Fahrzeugen ein Kommunikationskanal geöffnet werden muss, und sicherheitskritische Nachrichten Priorität haben, sollen die Datenmengen nicht das maximum ausreizen.

Hierzu wurde in Kapitel 2.3 ein Verbindungsaufbau beschrieben. Bevor die Kommunikation beginnen kann und Daten ausgetauscht werden können müssen die Gesprächsteilnehmer



in drei Schleifen die Umgebungsdaten miteinander austauschen. Die theoretische Latenz für eine C-LTE-Verbindung beträgt etwa 20 Millisekunden.

Für den Aufbau der Verbindung werden somit etwa 60 Millisekunden benötigt. Die durchschnittliche Datenrate beträgt 38 Mbit/s.

Konzentriert man sich auf die Zeiten aus dem vorhergehenden Kapitel dann sind hauptsächlich die kurzen Kommunikationszeiten von Interesse. Diese ergeben sich aus dem sich entgegenkommenden Verkehr auf der Autobahn. Hier haben die Kommunikationspartner nur knapp 5 Sekunden Zeit.

Bei 20 Millisekunden Latenzzeit und dem Drei-Wege-Handschlag aus dem Verbindungsaufbau benötigen die Teilnehmer bereits 60 Millisekunden für den Verbindungsaufbau. Für eine leichtere Berechnungsgrundlage runden wir die Zeit für den Verbindungsaufbau auf 1 Sekunde auf. Wir nehmen an, dass 50 % der Bandbreite für die Sicherheitsrelevante Kommunikation reserviert ist. Somit beträgt die durchschnittliche Bandbreite 19 Mbit/s. Daraus folgt, dass in 5 Sekunden und 19 Mbit/s eine theoretische Datenmenge von 95 Mbit versandt werden kann. Dies entspricht 11,87 Mbyte an Datenmenge.

Unter Betrachtung eines anderen Szenarios wie beispielsweise die Fahrt in entgegengesetzte Richtung auf der Landstraße, welches einer Zeit von 8 Sekunden entspricht, entspricht die Datenmenge bereits 152 Mbit, was wiederum 19 Mbyte entspricht.

Folgend wurde die Tabelle aus Kapitel 4.3 um die Datenmengen erweitert, die in der Kommunikationszeit ausgetauscht werden können.

Szenario (Geschwindigkeit Fahrzeug A / B)	Kommunikationszeit	Datenmenge
Fahrt in gleiche Richtung (40 / 60)	1,5 Minuten	261,25 Mbyte
Fahrt in entgegengesetzte Richtung auf der Autobahn (130 / 130)	4,98 Sekunden	11,87 Mbyte
Fahrt in entgegengesetzte Richtung auf der Landstraße (80 / 80)	8,1 Sekunden	19 Mbyte
Fahrt in entgegengesetzte Richtung in der Stadt (50 / 50)	12,96 Sekunden	28,41 Mbyte
Vorbeifahren an einem stehenden Fahrzeug (50 / 0)	25,92 Sekunden	59,19 Mbyte

**Tabelle 12: Datenmengen Vergleich für verschiedene Szenarien (Reichweite 300 m)**

Aus der Tabelle wird ersichtlich, dass der Austausch von Datenmengen bei Fahrzeugbegegnungen in entgegengesetzter Richtung recht gering ist.

Hinzu kommt: Je mehr Fahrzeuge die Kommunikationsstrecke nutzen, desto mehr Bandbreite muss mit diesen geteilt werden. Daraus folgt, dass die Größe der Datenmenge nicht mehr eingehalten wird.

Bevor eine Empfehlung zur Datengröße der Pakete gemacht wird sollen die Begegnungsarten nochmal analysiert und gewichtet werden.

## 4.5 Gewichtung der Begegnungsarten

In Kapitel 4.3 wurden einige Begegnungsarten aufgestellt und in Kapitel 4.4 die potenzielle Datenmenge bestimmt, die bei den jeweiligen Begegnungsarten übertragen werden kann. In diesem Kapitel sollen die Begegnungsarten bewertet und eine Gewichtung eingebracht werden. Diese soll dazu dienen eine Empfehlung der Dateigröße auszusprechen.

Um eine Gewichtung vorzunehmen werden Kriterien benötigt, um diese bei verschiedenen Szenarien anwenden zu können.

Folgende Kriterien werden für die Gewichtung betrachtet:

- Häufigkeit des Auftretens: Wie oft tritt die Begegnungsart auf und wie wahrscheinlich ist die Einbeziehung andere Verkehrsteilnehmer?
- Dauer der Begegnung: Wie viel Zeit wird benötigt, um Daten zwischen Verkehrsteilnehmern auszutauschen?
- Geschwindigkeit der Kommunikationsteilnehmer
- Entfernung bei der Begegnung: Wie nah kommen sich die Fahrzeuge maximal bei der Begegnungsart.
- Anzahl der Kommunikationsteilnehmer in Reichweite: Wie viele Teilnehmer sind in Kommunikationsreichweite gleichzeitig aktiv?

Bei den genannten Kriterien handelt es sich teilweise um Punkte, die bereits im Kapitel 4.3 beschrieben und beachtet worden sind. Nichts desto trotz werden diese für eine Gewichtung einbezogen und die verschiedenen Begegnungsarten unterteilen.

Mit den benannten Kriterien wird nun eine Gewichtung vorgenommen. Dafür werden die Kriterien miteinander verglichen. Folgende Tabelle soll bei der Gewichtung helfen. Die einzelnen Kriterien werden in die Tabelle eingefügt und paarweise verglichen also Spalte mit Zeile. Ist die Zeile Wichtiger, dann wird eine „2“ eingetragen, sind beide Kriterien gleich wichtig wird eine „1“ eingetragen ansonsten eine „0“.

Die Zahlen werden zeilenweise addiert. Die Summe ergibt die Gewichtung.

	Häufigkeit des Auftretens	Dauer der Begegnung	Geschwindigkeit	Entfernung bei der Begegnung	Anzahl der Kommunikationsteilnehmer	Summe
Häufigkeit des Auftretens	-	2	1	2	1	6
Dauer der Begegnung	0	-	1	2	0	3
Geschwindigkeit	1	1	-	0	1	3
Entfernung bei der Begegnung	0	0	2	-	2	4
Anzahl der Kommunikationsteilnehmer	1	2	1	0	-	4
Summe						20

**Tabelle 13: Gewichtung - Paarweiser Vergleich**

Nach Festlegung der Gewichtung werden als nächstes die prozentuellen Anteile der Gewichtungen berechnet und im nächsten Schritt an die Begegnungsarten angewandt.

Dafür wurde bereits in der Tabelle die Summe der einzelnen Zeilen berechnet. Zusätzlich ist auch die Summe aller Einzelsummen berechnet. Mit diesen Werten wird nun einen Prozentsatz berechnet, welcher dann auf die Begegnungsarten angewandt wird.

Häufigkeit des Auftretens	30%
Dauer der Begegnung	15%

Geschwindigkeit	15%
Entfernung bei der Begegnung	20%
Anzahl der Kommunikationsteilnehmer	20%

**Tabelle 14: Prozentuale Gewichtung**

Im nächsten Schritt werden die Begegnungsarten mit den Gewichtungen nochmal verglichen und bewertet.

	Häufigkeit (30 %)	Dauer (15 %)	Geschwindigkeit (15 %)	Entfernung (20 %)	Anzahl (20 %)	Summe
Fahrt in gleiche Richtung	9 (2,7)	10 (1,5)	8 (1,2)	10 (2)	1 (0,2)	<b>38</b> <b>(7,6)</b>
Fahrt in entgegengesetzte Richtung auf der Autobahn	5 (1,5)	2 (0,3)	1 (0,15)	2 (0,4)	7 (1,4)	<b>17</b> <b>(3,75)</b>
Fahrt in entgegengesetzte Richtung auf der Landstraße	5 (1,5)	4 (0,6)	3 (0,45)	4 (0,8)	4 (0,8)	<b>20</b> <b>(4,15)</b>
Fahrt in entgegengesetzte Richtung in der Stadt	8 (2,4)	6 (0,9)	5 (0,75)	7 (1,4)	1 (0,2)	<b>27</b> <b>(5,65)</b>
Vorbeifahren an einem stehenden Fahrzeug	6 (1,8)	8 (1,2)	7 (1,05)	7 (1,4)	2 (0,4)	<b>30</b> <b>(5,85)</b>

**Tabelle 15: Gewichtung der Begegnungsarten**

Dabei wird in einer Skala von 1 bis 10 angegeben, wie optimal die Bedingungen für die Kommunikation sind. Die 1 stellt hier eine ungünstige Bedingung dar, wohingegen die 10 die optimalste Bedingung ist.

Die Summe aller Begegnungsarten beträgt 132. Werden die gewichteten Werte zusammengerechnet ergibt dies eine Summe von 27. Dies entspricht 100 %. Mit diesem Wert nehmen wir nun eine prozentuale Unterteilung der Szenarien vor.

- Fahrt in gleiche Richtung:	28 %
- Fahrt in entgegengesetzte Richtung auf der Autobahn:	14 %
- Fahrt in entgegengesetzte Richtung auf der Landstraße:	15 %
- Fahrt in entgegengesetzte Richtung in der Stadt:	21 %
- Vorbeifahren an einem stehenden Fahrzeug:	22 %

Diese Erkenntnis sollte in die Bestimmung der Dateigröße einbezogen werden. Die Szenarien mit einem geringen Wert werden für die Bestimmung zweitrangig betrachtet, jedoch nicht ignoriert.

## 4.6 Bestimmung der Dateigröße

In Kapitel 4.4 wurde die Dauer für den Austausch der Daten zwischen Kommunikationspartnern bei den jeweiligen Begegnungsarten berechnet. In Kapitel 4.5 wurden die Begegnungsarten gewichtet. Mit diesen Informationen werden nun Überlegungen über die potenzielle Größe für ein Datenpaket gemacht, sodass dieses während einer Begegnung erfolgreich übertragen wird. Dabei ist die Gewichtung insofern wichtig, als dass die Begegnungen, die eher seltener eintreffen, teilweise ignorieren und die Dateigröße nicht vollständig angepasst auf diese bestimmen können.

Basierend auf die berechneten Datenmengen aus Kapitel 4.4 soll nun ein Mittelwert errechnet werden.

$$\frac{261,25 + 11,87 + 19 + 28,41 + 59,19}{5} = 75,944$$

Aufgerundet kann hier von 76 Mbyte ausgegangen werden. Da bei dieser Rechnung die Gewichtung nicht einbezogen wird soll folgend nochmal der Mittelwert berechnet werden. Diesmal unter Berücksichtigung der zuvor bestimmten Gewichtungen:

$$261,25 * 0,28 + 11,87 * 0,14 + 19 * 0,15 + 28,41 * 0,21 + 59,19 * 0,22 = 96,65$$

Daraus folgt, dass die Größe des Datenpaketes, bei den getroffenen Annahmen, sich bei ungefähr 96 Mbyte befindet. Da dieser Wert eher unüblich ist empfiehlt es sich hier zu runden und Datenpakete zu 100 Mbyte pro Sendevorgang zu versenden. Da am Anfang, bezüglich der Reservierung der Bandbreite für die sicherheitskritische Kommunikation, eine

recht großzügige Annahme getroffen wurde ist hier eine leicht optimistische Annahme zu treffen und die Paketgröße leicht zu der berechneten zu erhöhen.

Außerdem wird gerade die 5G Technologie stark vorangetrieben. Es ist zu erwarten das in wenigen Jahren diese Technologie verbreitet ist das mit der damit verbundenen erhöhten Datenraten auch zwischen den Fahrzeugen kommuniziert werden kann.

Die gesamte Technik funktioniert jedoch erst wenn genug Kommunikationsteilnehmer auf den Straßen vorhanden sind, mit welchen kommuniziert werden kann. Hier wird ein Verbreitungsgrad von 20 % aller Fahrzeuge auf den Straßen erwartet bis bei einem Kunden ein spürbarer Vorteil durch diese Technologie merkbar ist. Dem zugute kommt, dass aufgrund der sicherheitskritischen Funktion eine solche Technologie durch den Gesetzgeber beschleunigt werden kann. Der Hersteller kann dazu aufgefordert werden alle neu zugelassenen Fahrzeuge mit einer solchen Technologie auszustatten. Dies hätte den Vorteil, dass ein Kunde, nicht bewusst und gegen Aufpreis, eine Sonderausstattung zum Fahrzeug bestellen müsste.

Dennoch wird eine gewisse Zeit benötigt bis die 20 % der Fahrzeuge auf den Straßen erreicht werden. Mit dieser Frage beschäftigt sich das nächste Kapitel.

## **4.7 Fahrzeugbestand und Neuzulassungen**

Für eine Prognose der Dauer bis zum Erreichen der 20 % Marktdurchdringung mit dieser Technik kann vorerst nur von vergangenen Werten ausgegangen werden. Bevor eine Prognose erstellt wird werden die Zulassungszahlen der vergangenen Jahre angeschaut. Mithilfe dieser Werte kann, mit bestimmten Annahmen, eine Prognose angefertigt werden. Daraus resultierend kann ermittelt werden wann voraussichtlich die bereits erwähnte 20 % Marktdurchdringung erreicht ist.

Bei Betrachtung des Bestands der Kraftfahrzeuge in Deutschland stellt man fest, dass dieser in den letzten vier Jahren stetig gewachsen ist. Die Zuwachsrate beträgt pro Jahr zwischen 1,3 % und 1,6 %. (8)

2015:	45 Millionen
2016:	45,8 Millionen
2017:	46,5 Millionen
2018:	47,1 Millionen

**Tabelle 16: Fahrzeugbestand in Deutschland**

Die Tabelle 16 zeigt konkrete Zahlen zum Fahrzeugbestand in Deutschland. Die Daten werden im Zuge von Pressemitteilungen durch das Kraftfahrt-Bundesamt (KBA) öffentlich zur Verfügung gestellt.

Selbstverständlich ist der jährliche Zuwachs an dieser Stelle nicht identisch mit den jährlich neu zugelassenen Fahrzeugen, da auch Fahrzeuge stillgelegt werden. Die Daten werden vom KBA in monatlichen Berichten zur Verfügung gestellt.

In Abbildung 16 sind die monatlichen Zulassungszahlen der letzten drei Jahre ersichtlich. Hier ist erkennbar, dass die Kurven jedes Jahr ähnlich verlaufen. Die Zahlen bewegen sich im Bereich zwischen 200.134 und 359.683 neu zugelassener Fahrzeuge pro Monat. Auch die Summe der neu zugelassenen Fahrzeuge pro Jahr lässt sich aus diesen Berichten leicht berechnen. Diese sind:

- 2015: 3.206.042 Fahrzeuge
- 2016: 3.351.607 Fahrzeuge
- 2017: 3.441.262 Fahrzeuge
- 2018: 3.435.778 Fahrzeuge
- 2019: 880.092 Fahrzeuge

Die Jahre 2015 und 2016 sind aufgrund der Übersichtlichkeit nicht in der Abbildung 16 aufgeführt. Für das Jahr 2019 sind nur Daten des ersten Quartalls berücksichtigt. Es ist zu erwarten, dass sich der Graph hier in ähnlicher Weise entwickelt wie in den Jahren zuvor.



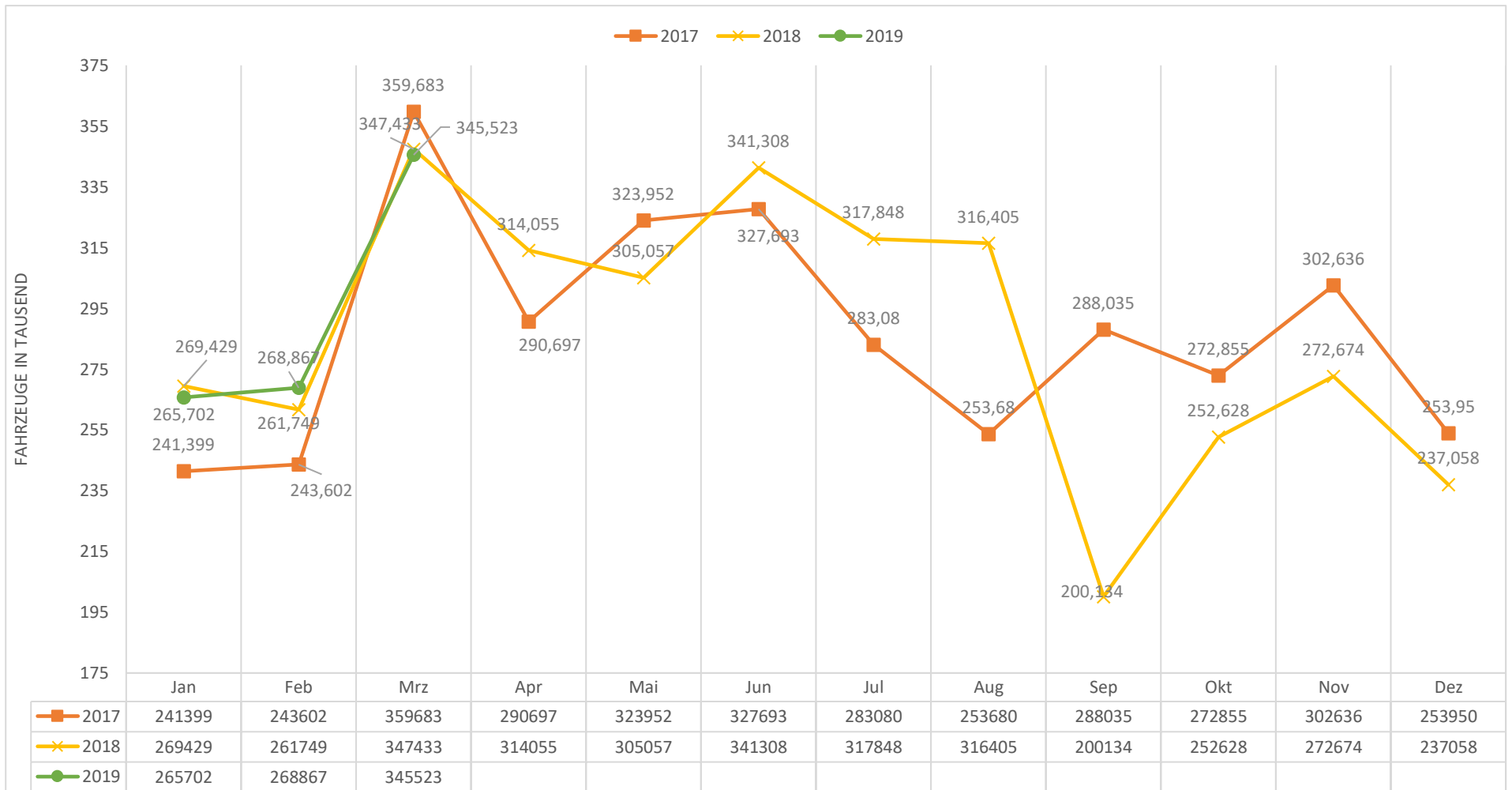


Abbildung 16: Kraftwagen Neuzulassungen pro Monat

Die Anzahl der Bestandsfahrzeuge kann nun in Relation mit den neu zugelassenen Fahrzeugen gesetzt werden. Das Ergebnis daraus ist der prozentuelle Anteil an jährlich neu zugelassenen Fahrzeugen.

Jahr	Bestandsfahrzeuge	Neu zugelassene Fahrzeuge	Prozentueller Anteil
2015	45 Millionen	3,2 Millionen	7,11 %
2016	45,8 Millionen	3,3 Millionen	7,21 %
2017	46,5 Millionen	3,4 Millionen	7,31 %
2018	47,1 Millionen	3,4 Millionen	7,16 %

**Tabelle 17: Prozentueller Vergleich neu Zugelassener Fahrzeuge**

Knapp über 7 % des gesamten deutschen Fahrzeugbestandes wird jährlich ausgetauscht. Diese schwankt leicht. Eine abnehmende oder steigende Entwicklung ist nicht zu erkennen.

## 4.8 Prognose

Um eine Prognose aus den Daten des vorhergehenden Kapitels erstellen zu können wird noch eine Schätzung benötigt, wie viele Fahrzeuge mit der Technologie ausgestattet werden. Die Durchdringung der Technologie kann insbesondere damit beschleunigt werden, dass sich die Hersteller darauf einlassen alle neu zugelassenen Fahrzeuge ab einem bestimmten Zeitpunkt mit der Technologie auszustatten. Eine andere Möglichkeit wäre, wenn der Gesetzgeber die Hersteller dazu verpflichtet alle Fahrzeuge mit einer Kommunikationseinheit auszustatten.

Mit dieser Annahme, und einer gleichbleibenden Entwicklung von knapp 7 % pro Jahr neu zugelassener Fahrzeuge wird die 20 % Marktdurchdringung nach knapp 3 Jahren erreicht. Sollten also alle Fahrzeuge ab 2020 nur mit der Technologie ausgestattet zugelassen werden würden die Kunden erst 2023 davon profitieren.

Diese Rechnung setzt voraus, dass die Fahrzeuge verschiedener Hersteller miteinander kommunizieren und Daten austauschen. Das ist für die sicherheitskritischen Nachrichten im Rahmen des Möglichen. Der Datenaustausch von zum Beispiel Update-Dateien zwischen Fahrzeugen unterschiedlicher Hersteller ist zum aktuellen Zeitpunkt eher weniger denkbar.

Wenn die Daten bezogen auf einzelne Hersteller analysiert werden sollten verlässlichere Zahlen herauskommen, um eine Prognose zu erstellen.

Da eine Vielzahl von Fahrzeugherstellern existiert werden hier die sechs größten und am meisten vertretenen in Deutschland analysiert. Diese sind:

- VW
- Opel
- Mercedes
- Ford
- BMW
- Audi

Zunächst wird ermittelt, wie viele Fahrzeuge dieser Hersteller auf dem Markt bereits besitzt.

	2015	2016	2017	2018
VW	9 740 692 (21,6 %)	9 887 132 (21,6 %)	9 982 099 (21,5 %)	10 039 389 (21,3 %)
Opel	4 687 804 (10,4 %)	4 607 857 (10,1 %)	4 534 639 (9,8 %)	4 455 662 (9,5 %)
Mercedes	4 223 587 (9,4 %)	4 307 158 (9,4 %)	4 378 615 (9,4 %)	4 434 329 (9,4 %)
Ford	3 352 642 (7,4 %)	3 380 357 (7,4 %)	3 406 624 (7,3 %)	3 438 207 (7,3 %)
BMW	3 082 243 (6,8 %)	3 154 576 (6,9 %)	3 203 507 (6,9 %)	3 256 884 (6,9 %)
Audi	3 091 807 (6,9 %)	3 173 116 (6,9 %)	3 224 007 (6,9 %)	3 242 838 (6,9 %)

**Tabelle 18: Fahrzeugbestand nach Hersteller**

Die in Klammern stehenden Prozentwerte sind der Anteil verglichen mit dem Gesamtbestand aller Hersteller.

	VW	Opel	Mercedes	Ford	BMW	Audi
Jan	54 348	17 364	26 419	18 387	20 759	21 032
Feb	47 822	19 218	21 910	19 258	17 664	23 510
März	64 496	21 147	29 583	26 890	26 278	25 673
Apr	64 769	19 970	29 112	24 571	21 079	26 392
Mai	63 835	18 761	27 214	22 050	19 027	26 971
Jun	66 389	22 066	28 376	24 525	25 325	27 775
Jul	66 364	17 404	25 377	20 829	18 250	33 562
Aug	63 623	17 609	23 764	17 810	19 289	28 198
Sep	20 278	20 337	22 379	15 006	24 524	4 596
Okt	40 811	19 545	31 737	22 346	23 521	7785
Nov	49 253	18 952	30 416	22 940	24 476	13 170
Dez	41 527	15 594	22 876	17 711	24 859	16 636
Summe	643 515	227 967	319 163	252 323	265 051	255 300

**Tabelle 19: Neuzulassungen nach Hersteller – 2018**

	VW	Opel	Mercedes	Ford	BMW	Audi
Jan	50 823	16 225	25 292	21 064	19 198	21 814
Feb	49 965	17 677	24 380	21 264	17 825	24 050
März	62 627	22 505	28 054	25 613	25 458	29 751
Summe	163 415	56 407	77 726	67 941	62 481	75 615

**Tabelle 20: Neuzulassungen nach Hersteller - 2019 1. Quartal**

Mit diesen Zahlen kann nun der prozentuelle Anteil der Fahrzeuge im Vergleich zum Gesamtbestand betrachtet werden. Dazu wird die Summe der neu zugelassenen Fahrzeuge über das gesamte Jahr verwendet. Diese wird mit zwei Werten verglichen, einmal der Gesamtbestand des Herstellers und einmal der Gesamtbestand über alle Hersteller hinweg.

	2015			2016		
		Hersteller	Gesamt		Hersteller	Gesamt
VW	685 669	7,04 %	1,52 %	656 025	6,64 %	1,43 %
Opel	229 352	4,89 %	0,51 %	243 792	5,29 %	0,53 %
Mercedes	286 883	6,79 %	0,64 %	311 286	7,23 %	0,68 %
Ford	224 579	6,70 %	0,50 %	239 766	7,09 %	0,52 %
BMW	248 565	8,06 %	0,55 %	262 083	8,31 %	0,57 %
Audi	269 047	8,70 %	0,60 %	289 617	9,13 %	0,63 %

**Tabelle 21: Vergleich neu Zugelassener Fahrzeuge - 2015 & 2016**

	2017			2018		
		Hersteller	Gesamt		Hersteller	Gesamt
VW	634 270	6,35 %	1,36 %	643 515	6,41 %	1,37 %
Opel	243 715	5,37 %	0,52 %	227 967	5,12 %	0,48 %
Mercedes	326 188	7,45 %	0,70 %	319 163	7,20 %	0,68 %
Ford	246 589	7,24 %	0,53 %	252 323	7,34 %	0,54 %
BMW	261 864	8,17 %	0,56 %	365 051	11,21 %	0,78 %
Audi	283 169	8,78 %	0,61 %	255 300	7,87 %	0,54 %

**Tabelle 22: Vergleich neu zugelassener Fahrzeuge - 2017 & 2018**

Diese Werte für die Prognose ändern nicht viel gegenüber der Prognose über alle Hersteller. Wie allerdings bereits erwähnt ist zum aktuellen Zeitpunkt eine Kommunikation zwischen Herstellern übergreifend kaum vorzustellen. Unter Betrachtung der Zulassungszahlen der Fahrzeuge auf die Hersteller verteilt und verglichen mit den Werten des Gesamtbestands aller Fahrzeuge dann bewegt sich diese bei nicht mal 1 % pro Jahr.

Damit würde eine Marktdurchdringung von 20 % auch bei über 20 Jahren dauern. Dies ist einfach nicht realistisch.

Daran zeigt sich die Notwendigkeit das die Hersteller in der Entwicklung näher zusammenrücken, um eine solche Kommunikation für den Kunden erlebbar zu machen.

## 5 Fazit

Zusammenfassend eignet sich das Car2Car für den Austausch von sicherheitsrelevanten Nachrichten gut.

Im ersten Schritt sollte sich auf eine Technologie geeinigt werden, welche dann verfolgt wird, sei es die auf WLAN basierende IEEE802.11p oder die auf LTE basierende C-V2X. Beide Technologien haben, für sich gesehen, ihre Vorteile. Aus persönlicher Sicht ist die C-V2X die zukunftsorientiertere Variante.

Weiter sollten sich sowohl herstellerübergreifende als auch landesunabhängige Standards etablieren. Es sollten möglichst wenig Unterschiede der Standards zwischen Fahrzeugen internationaler Märkte vorhanden sein.

Positiv anzumerken ist, dass bereits Fountain-Codes als Codierungsalgorithmus beim Standard verwendet wird. Damit wird die Nachrichtenübertragung mit aktuellem Stand der Technik übertragen und insbesondere die stör anfällige Wireless-Verbindung maximal abgesichert. Aus diesem Aspekt kann in dieser Arbeit keine Empfehlung ausgesprochen werden, um eine Verbesserung zu erhalten.

In den nächsten Jahren zeigt sich, ob sich diese Technologie durchsetzen kann. Diese sollte, gerade in den ersten Jahren, in neu zugelassenen Fahrzeugen ohne Aufpreis für den Kunden verbaut werden. Um dies zu gewährleisten sollte der Gesetzgeber die Hersteller verpflichten diese Technologie in die Fahrzeuge einzubauen. Dennoch wird es, auch mit diesem Zwang, noch ca. 3 Jahre dauern, bis die Technologie beim Kunden merkbar Anwendung findet.

Der Austausch von herstellerbezogenen Daten ist ein weites Zukunftsthema, da hier die Marktvielfalt zu hoch ist und sich Fahrzeuge gleicher Hersteller zu langsam verbreiten. Eine mögliche Lösung wäre, dass solche Daten auch von anderen Herstellern empfangen und verbreitet werden. Dies ist, aufgrund der Konkurrenzgedanken zwischen den Herstellern, eher nicht denkbar. Für Ballungsgebiete könnte es eventuell funktionieren. In Gebieten, in welchen eine hohe Vielfalt von Fahrzeugen verschiedener Hersteller herrscht, wird diese Art kaum Kommunikationspartner finden, mit denen die Daten ausgetauscht werden können.





## 6 Literatur

1. **Wikipedia.** <https://de.wikipedia.org/wiki/Peer-to-Peer>. [Online] <https://de.wikipedia.org/wiki/Peer-to-Peer>.
2. **Yao, Yuan.** Semantic Scholar. [Online] <https://www.semanticscholar.org/paper/Delay-analysis-and-study-of-IEEE-802.11p-based-DSRC-Yao-Rao/de31c8ced09cfa83abbcabab1fad217f0cdc0e22>.
3. **Gozalvez, Rafael Molina-Masegosa and Javier.** <https://vtsociety.org>. *LTE-V for Sidelink 5G V2X Vehicular Communications: A New 5G Technology for Short-Range Vehicle-to-Everything Communications*. [Online] <https://vtsociety.org/2018/02/vehicles-february-2018/>.
4. **ITWissen.info.** [Online] <https://www.itwissen.info/WAVE-wireless-access-for-vehicular-environment.html>.
5. **aieacts.** [Online] <http://www.aieacts.com/nd.jsp?id=78>.
6. **Wolf, Marko.** all-electronics.de. [Online] <https://www.all-electronics.de/sota-software-updates-im-automobil/>.
7. **Fachverband Fußverkehr Deutschland.** Neue Richtlinien für Lichtsignalanlagen. [Online] <http://www.geh-recht.info/planungsgrundlagen/40-planungsgrundlagen/planungsgrundlagen/155-pg-rilsa-richtlinien-fuer-lichtsignalanlagen-2010-kritische-einschaetzung.html>.
8. **Bundesministerium für Verkehr und Digitale Infrastruktur.** KBA. [Online] [https://www.kba.de/DE/Presse/Pressemitteilungen/2018/Fahrzeugzulassungen/fahrzeugzulassungen\\_gesamtTabelle.html?nn=2145958](https://www.kba.de/DE/Presse/Pressemitteilungen/2018/Fahrzeugzulassungen/fahrzeugzulassungen_gesamtTabelle.html?nn=2145958).

## Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Wolfsburg, den 12.05.2019

<unterschrift>

Alexander Kinder