

# **Synchronisation eines LTE-Empfängers mit mehreren Empfangsantennen**

Bachelorarbeit

**Kristian Maier**

Hauptreferent : Prof. Dr.rer.nat. Friedrich Jondral  
Betreuer : Dipl.-Ing. Sebastian Koslowski

Beginn : 01.05.2014  
Abgabe : 31.10.2014



# Erklärung

Ich versichere hiermit, dass ich die vorliegende Arbeit selbständig und unter Beachtung der Satzung der Universität Karlsruhe (TH) zur Sicherung guter wissenschaftlicher Praxis in der aktuellen Fassung angefertigt habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt und wörtlich oder inhaltlich übernommene Stellen als solche kenntlich gemacht.

Karlsruhe, den 31.10.2014

Kristian Maier



# Zusammenfassung

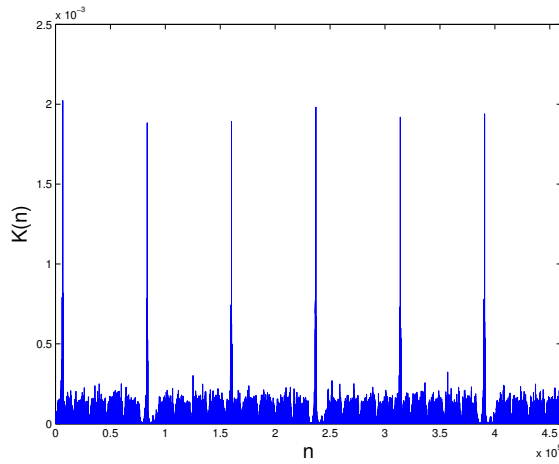
LTE ist derzeit der Mobilfunkstandard der neuesten Generation und bietet höhere Datenraten. Dabei kommt *Orthogonal Frequency-Division Multiplexing* (OFDM) und *Multiple Input Multiple Output* (MIMO) zum Einsatz. MIMO, die Verwendung mehrerer Antennen auf Sende- und Empfangsseite, ermöglicht dabei sowohl die Steigerung der Datenrate als auch ein zuverlässigeren Empfang bei gleichbleibender Bandbreite.

In dieser Arbeit wird die Erweiterung eines bestehenden Empfängers für LTE-Signale in GNU-Radio auf mehrere Empfangskanäle bzw. Empfangsantennen und deren Synchronisation vorgestellt. Mit dem bestehenden Empfänger kann bereits ein Empfangskanal in Zeit und Frequenz synchronisiert und der *Master Information Block* (MIB) ausgelesen werden, der allgemeine Zellinformationen trägt und im *Physical Broadcast Channel* (PBCH) übertragen wird. Die Schätzung des Übertragungskanals und die Entzerrung der Empfangssymbole findet im Frequenzbereich statt und wird durch gesendete Pilotsymbole realisiert.

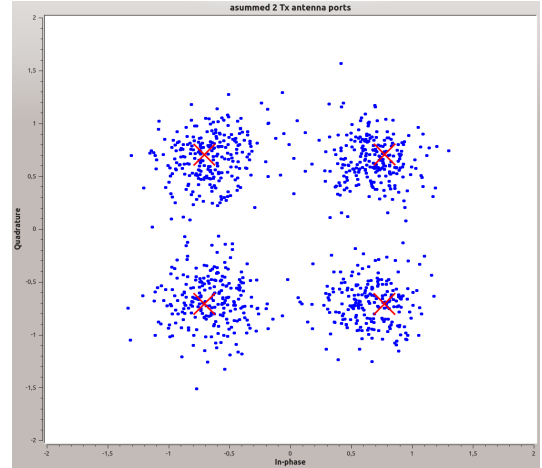
In LTE wird eine Framestruktur verwendet, welche Frames der Dauer 10 ms in Subframes und in Slots aufteilt. Bevor Daten dekodiert werden können, muss sich der Empfänger auf diese Framestruktur synchronisieren. In dieser sind 2 Synchronisationssignale eingebettet, die pro Frame 2 mal periodisch gesendet und zur Synchronisation genutzt werden können. Der bisherige Empfänger verwendete dazu das *Cyclic Prefix* und das *Primary Synchronisation Signal* (PSS). Im Rahmen dieser Arbeit wurde ein neuer Synchronisationsalgorithmus entwickelt und implementiert, welche auf Korrelation im Zeitbereich mit dem PSS beruht und mit dem bisherigen Verfahren verglichen. In Abbildung ist diese Korrelation dargestellt. Mittels Multiraten-Signalverarbeitung werden die Empfangssignale zunächst gefiltert und abwärtsgetastet. Im abwärtsgetasteten Bereich findet eine grobe Suche statt. Die so ermittelte Position wird dann auf das Signal mit der vollen Rate übertragen und in einem kleinen Suchfenster verfeinert.

GNU Radio ist eine graphische Oberfläche, dem GNU Radio Companion, erstellt und ausgeführt. In Abbildung 0.2 ist der Aufbau des Flowgraphs für den MIMO-LTE-Empfang dargestellt. Die Synchronisations- und für den Mehrkanalempfang notwendigen Algorithmen wurden in neuen GNU Radio Blöcken implementiert.

Die Funktionalität des Empfängers wurde zunächst mit berechneten Simulationsdaten verifiziert. Anschließend wurden echte LTE-Signale aufgenommen und erfolgreich dekodiert. Wichtig ist, dass die Empfangskanäle jeweils in Zeit- und Frequenz synchron zueinander sind. In Abbildung ist ein Konstellationsdiagramm der empfangenen Symbole nach der Entzerrung dargestellt. Die Messung fand im Funklabor des CEL statt, wo eine ausreichende Signalstärke vorhanden war. Der neue Synchronisationsalgorithmus funktioniert auch mit realen LTE-Signalen und die Verwendung mehrerer Empfangsantennen verbesserte die Zahl erfolgreicher Dekodierungen.



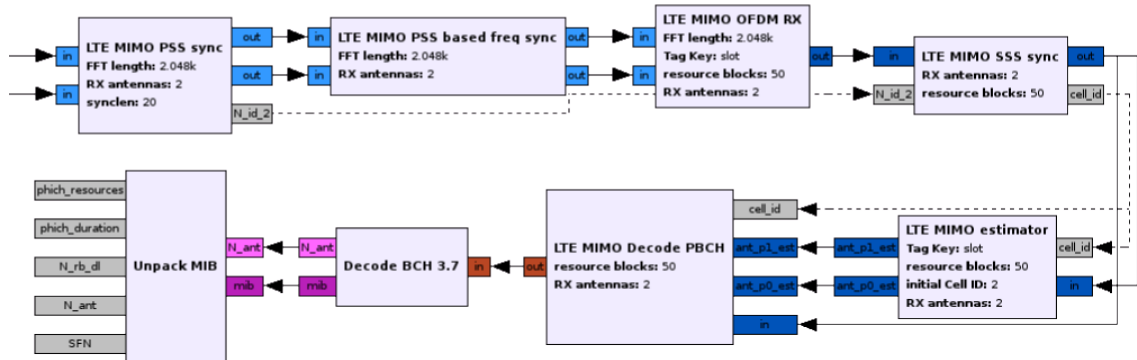
(a) Korrelation der PSS-Sequenz mit simulierten Empfangsdaten im Zeitbereich über 3 Frames



(b) Konstellationsdiagramm einer Messung nach Entzerrung (rot: ideale Punkte)

**Abbildung 0.1:** Simulations- und Messergebnisse

Auf die Dekodierung bzw. Verarbeitung der entzerrten Empfangssymbole wird nicht eingegangen, da die entsprechende Funktionalität im bestehenden Empfänger bereits vorhanden war und ohne Änderung übernommen werden konnte.



**Abbildung 0.2:** Flowgraph mit hierarchischen Blöcken

# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
<b>2</b>	<b>LTE - Physical Layer</b>	<b>3</b>
2.1	Grundlagen	3
2.1.1	Hochratige Datenübertragung im Mobilfunk	3
2.1.2	OFDM	3
2.1.3	Kanalmodell	6
2.1.4	Framestruktur	7
2.1.5	Physikalische Kanäle	8
2.1.6	Referenzsignale	9
2.2	MIMO	12
2.2.1	Antennenkonfiguration	12
2.2.2	Diversity	12
2.2.3	Spatial Multiplexing	14
<b>3</b>	<b>Synchronisation</b>	<b>17</b>
3.1	Cell-ID	17
3.2	Primary Synchronization Signal	17
3.3	Secondary Synchronization Signal	19
3.4	Zeitsynchronisation	19
3.5	Frequenzsynchronisation	25
<b>4</b>	<b>Implementierung</b>	<b>27</b>
4.1	GNU Radio	27
4.2	Übersicht	28
4.3	Synchronisation	29
4.3.1	PSS-Zeitsynchronisation	30
4.3.2	PSS-Frequenzsynchronisation	32
4.3.3	SSS-Synchronisation	33
4.4	OFDM - Operationen	34
4.5	Kanalschätzung	35
4.6	PBCH-Dekodierung	36
<b>5</b>	<b>Messung und Simulation</b>	<b>39</b>
5.1	Simulation	39
5.2	Messaufbau	39
5.3	Messergebnisse und Vergleich mit Simulation	41
5.3.1	Synchronisation	41
5.3.2	Dekodierung	42
<b>6</b>	<b>Zusammenfassung</b>	<b>45</b>





# Abbildungsverzeichnis

0.1	Simulations- und Messergebnisse . . . . .	iv
0.2	Flowgraph mit hierarischen Blöcken . . . . .	iv
2.1	Cyclic Prefix Schema (blau: Delay Spread) . . . . .	4
2.2	Aufbau eines OFDM-Senders im komplexen Basisband . . . . .	5
2.3	Aufbau eines OFDM-MIMO-Empfängers im komplexen Basisband . . . . .	6
2.4	$2 \times 2$ MIMO-System . . . . .	6
2.5	LTE - Framestruktur in Zeitrichtung [Dem12, Kap. 2.2] . . . . .	8
2.6	Datenfluss [3GP12, Kap. 6.3] . . . . .	9
2.7	Zeitfrequenzebene mit RS-Sequenzen . . . . .	11
3.1	Vergleich der AKF ( $N_{\text{ID}}^2 = 1$ ) und KKF ( $N_{\text{ID},1}^2 = 0$ , $N_{\text{ID},2}^2 = 1$ ) des PSS im Zeit- und Frequenzbereich . . . . .	18
3.2	CP - Symboltaktsynchronisation . . . . .	20
3.3	CP-Korrelation über einige simulierte OFDM-Symbole . . . . .	20
3.4	KKF der PSS-Sequenz ( $N_{\text{ID}}^2 = 0$ ) mit simulierten Empfangsdaten im Zeitbe- reich über 3 Frames . . . . .	22
3.5	PSS-Struktur im Frequenzbereich (grün: PSS, schwarz: ungenutzt, grau: Nutz- daten, rot: Tiefpassfilter) . . . . .	23
4.1	Flowgraph mit hierarischen Blöcken . . . . .	30
4.2	Flowgraph der PSS-Zeitsynchronisation . . . . .	30
4.3	Innere Struktur der PSS-Frequenzsynchronisation . . . . .	32
4.4	Innere Struktur der SSS-Framesynchronisation . . . . .	33
4.5	Innere Struktur der OFDM-Operationen . . . . .	34
4.6	Aufbau der Kanalschätzung . . . . .	35
4.7	Aufbau der PBCH-Dekodierung . . . . .	36
5.1	Spektren der Messungen (rot: Antenne 1, blau: Antenne 2) . . . . .	40
5.2	Korrelation gemittelt über 2 Frames . . . . .	42
5.3	Konstellationsdiagramme PBCH nach Entzerrung (rot: ideale Punkte) . . . .	43
5.4	Textausgabe Messung 1 . . . . .	44



# Abkürzungsverzeichnis

Akronym	Beschreibung
AKF	Autokorrelationsfunktion
CAZAC	Constant Amplitude Zero Autocorrelation
CFO	Carrier Frequency Offset
CP	Cyclic Prefix
CRC	Cyclic Redundancy Check
DFT	Diskrete Fourier Transformation
ETU	Extended Typical Urban (Kanalmodell)
FDD	Frequency Division Duplexing
FEC	Forward Error Correction
FFO	Fractional Frequency Offset
FFT	Fast Fourier Transformation
FIR	Finite Impulse Response
$\Delta f$	Unterträgerabstand
$h_{ij}$	komplexer Kanalkoeffizient
IDFT	Inverse Diskrete Fourier Transformation
IFFT	Inverse Fast Fourier Transformation
IFO	Integer Frequency Offset
ISI	Intersymbolinterferenz
KKF	Kreuzkorrelationsfunktion
LTE	Long Term Evolution
MIB	Master Information Block
MIMO	Multiple Input Multiple Output
MRC	Maximum Ratio Combining
$N_C$	Anzahl genutzter Unterträger
$N_r$	Anzahl der Empfangsantennen
$N_t$	Anzahl der Sendeantennen (Antennenports)
$N_{\text{FFT}}$	FFT-Länge
$N_{\text{ID}}$	Cell-ID
$N_{\text{ID}}^1$	Cell-ID Group
$N_{\text{ID}}^2$	Cell-ID Number
OFDM	Orthogonal Frequency-Division Multiplexing
PAPR	Peak-to-Average Power Ratio
PBCH	Physical Broadcast Channel
PSS	Primary Synchronization Signal
$P_u$	PSS-Sequenz im Frequenzbereich
$p_u$	PSS-Sequenz im Zeitbereich
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase-Shift Keying

Akronym	Beschreibung
RB	Resource Block
RE	Resource Element
$r_k$	Empfangssignal an Antenne k
RS	Referenz Signal
SC	Selection Combining
SFBC	Space Frequency Block Code
SFN	System Frame Number
SNR	Signal-to-Noise Ratio
SSS	Secondary Synchronization Signal
STBC	Space Time Block Code
TDD	Time Division Duplexing
$T_u$	OFDM-Symboldauer
UMTS	Universal Mobile Telecommunications System
ZC	Zadoff-Chu

# 1 Einleitung

Long Term Evolution (LTE) ist der Nachfolger von UMTS und der Mobilfunkstandard der vierten Generation (4G). LTE verspricht höhere Datenraten von bis zu 100 MBit/s im Downlink, mehr Nutzer pro Funkzelle und ein schnellerer Verbindungsaufbau zur Basisstation. Für LTE-Signale sind verschiedene Konfigurationen mit Bandbreiten von 1,4 MHz bis 20 MHz möglich, somit können verschiedene Spektralbereiche je nach vorhandenem Platz effizient genutzt werden. Um eine hohe spektrale Effizienz und hohe Datenraten zu erreichen, nutzt LTE im Downlink *Orthogonal Frequency-Division Multiplexing* (OFDM) und *Multiple Input Multiple Output* (MIMO) als Schlüsseltechnologien. OFDM kann hierbei durch Orthogonalität der Unterträger diese im Frequenzbereich eng aneinanderreihen und so viele Sendesymbole in einer vergleichsweise geringen Bandbreite unterbringen. Dabei garantiert OFDM ISI-Freiheit und erlaubt so die Verwendung einfacher Kanalentzerrer am Empfänger. MIMO - die Verwendung mehrerer Sende- und Empfangsantennen - kann einerseits durch *Spatial Multiplexing* die Datenrate erhöhen, andererseits ergibt sich ein Diversitätsgewinn, welcher die Robustheit des Kanals gegen Fading steigert. [DPS14, S.64f/Kap. 5.2]

GNU Radio ist ein Open-Source Software Framework zur Entwicklung von *Software Defined Radios*. Dabei werden sämtliche Signalverarbeitungsschritte in Software durchgeführt, wobei die Anbindung von Hardware zum Senden und Empfangen möglich ist. In GNU Radio sind die Signalverarbeitungsschritte in modulare Blöcke aufgeteilt, wodurch sich Abtastwerte schrittweise verarbeiten lassen. Dabei werden viele gebräuchliche Funktionen, wie z.B. Filter, von GNU Radio bereitgestellt, welche sich in verschiedenen Situationen nutzen lassen [GNU14].

Vorteile bei Verwendung von Software Defined Radios ergeben sich bei Erweiterung oder Neuentwicklung eines Übertragungssystems, da nur die Software bei gleichbleibender Hardware geändert werden muss. Mit der gleichen Hardware ist es so auch möglich, verschiedene Funkstandards zu bedienen. Durch einen modularen Aufbau der Software ist es leicht möglich, einzelne Elemente bzw. Blöcke zu ändern oder zu erweitern und schnell neue Konzepte und Algorithmen zu testen. Davon wird hier Gebrauch gemacht.

Ein bereits bestehende Empfänger kann ein empfangenes LTE-Signal durch Ausnutzen von Cyclic Prefix und im Signal integrierte Sequenzen synchronisieren. Der Kanal wird durch übertragene Pilotsymbole geschätzt und der *Master Information Block* (MIB) dekodiert, welcher im *Physical Broadcast Channel* (PBCH) übertragen wird und grundlegende Zellinformationen enthält.

In dieser Arbeit soll der bestehende Empfänger für mehrere Empfangskanäle erweitert werden. Außerdem sollen die verwendeten Synchronisationsalgorithmen überarbeitet und für den Mehrkanalempfang verbessert werden. Weiterhin umfasst dies auch die Überarbeitung von Elementen wie Kanalschätzung und OFDM-Operationen. Die Implementierung der Dekodierung des PBCH und des MIB ist bereits vorhanden und kann direkt übernommen werden. Das System wird zunächst mit simulierten Signalen verifiziert und getestet. Zum Schluss werden echte LTE-Signale aufgenommen und mit dem neuen Verfahren dekodiert.

Zunächst werden in Kapitel 2 die für LTE wichtigen theoretischen Grundlagen bzw. der Aufbau des LTE-Signals vorgestellt. Dabei wird auf OFDM, die verwendete Framestruktur und MIMO eingegangen. Die Synchronisationsmethoden und die dafür notwendigen Synchronisationssignale werden in Kapitel 3 vorgestellt. Die Implementierung in GNU Radio und die Struktur der Signalverarbeitungskette wird ausführlich in Kapitel 4 beschrieben. Abschließend werden in Kapitel 5 Simulation- und Messergebnisse präsentiert und verglichen.

## 2 LTE - Physical Layer

In diesem Kapitel werden auf die zum Verständnis der physikalischen Schicht des LTE-Standards notwendigen Grundlagen wie OFDM, der verwendeten Framestruktur und Pilotsymbole eingegangen. Danach wird MIMO und die in LTE dafür verwendeten Verfahren vorgestellt.

### 2.1 Grundlagen

#### 2.1.1 Hochratige Datenübertragung im Mobilfunk

LTE verwendet eine vergleichsweise hohe Datenrate, die bei serieller Übertragung eine kurze Symboldauer besitzt, welche meist kleiner als der *Channel Delay Spread* ist. Der *Channel Delay Spread* ist die zeitliche Differenz des längsten und kürzesten Ausbreitungspfads. Eine solche Übertragung würde deshalb aufgrund des Mehrwegeempfangs zu viel Intersymbolinterferenz (ISI) führen und den Einsatz aufwendiger Entzerrer notwendig machen. Im LTE-Downlink wird deshalb *Orthogonal Frequency-Division Multiplexing* (OFDM) genutzt, welches die Sendesymbole auf mehrere orthogonale und parallele Unterträger  $N_C$  aufteilt. Dadurch erhöht sich die effektive Symboldauer des OFDM-Systems im Vergleich zum seriellen System um den Faktor  $N_C$ , sodass die OFDM-Symboldauer  $T_u$  größer als der *Channel Delay Spread* ist und ISI verhindert wird.

#### 2.1.2 OFDM

Im Folgenden wird OFDM wie in [DPS14, S.29ff/Kap. 3] eingeführt. Das OFDM-Signal  $x(t)$  wird im Zeitintervall  $mT_u \leq t \leq (m+1)T_u$  durch

$$x(t) = \sum_{k=0}^{N_C-1} x_k(t) = \sum_{k=0}^{N_C-1} a_k^{(m)} e^{j2\pi k \Delta f t} \quad (2.1)$$

dargestellt, wobei  $a_k^{(m)}$  das komplexe Modulationssymbol im  $m$ -ten OFDM-Symbol,  $x_k(t)$  der  $k$ -te modulierte Unterträger und  $\Delta f$  der Unterträgerabstand ist. Zur Pulsformung wird ein einfaches Rechteck verwendet, was im Frequenzbereich zu Sinc-förmigen Unterträgern führt. Zwei Unterträger erfüllen die Orthogonalitätsbedingung

$$\int_{mT_u}^{(m+1)T_u} x_{k_1}(t) x_{k_2}^*(t) dt = \int_{mT_u}^{(m+1)T_u} a_{k_1} a_{k_2}^* e^{j2\pi k_1 \Delta f t} e^{-j2\pi k_2 \Delta f t} dt = 0 \quad \text{für } k_1 \neq k_2 \quad (2.2)$$

dann, wenn für ihren Frequenzabstand

$$\Delta f = \frac{1}{T_u} \quad (2.3)$$

gilt. Da kein Schutzabstand zwischen den Unterträgern notwendig ist und diese überlappen, besitzt OFDM im Vergleich zu FDMA eine hohe spektrale Effizienz. Das zeitdiskrete OFDM-Signal mit einer Samplerate  $f_S = 1/T_S = N \cdot \Delta f$  ist

$$x[n] = x(nT_s) = \sum_{k=0}^{N_C-1} a_k e^{j2\pi k \Delta f n T_s} = \sum_{k=0}^{N_C-1} a_k e^{j2\pi k n / N} = \sum_{k=0}^{N-1} a'_k e^{j2\pi k n / N} \quad (2.4)$$

mit

$$a'_k = \begin{cases} a_k & 0 \leq k < N_C \\ 0 & N_C \leq k < N \end{cases} \quad (2.5)$$

und lässt sich durch eine Inverse Diskrete Fourier Transformation (IDFT) der Länge N durch

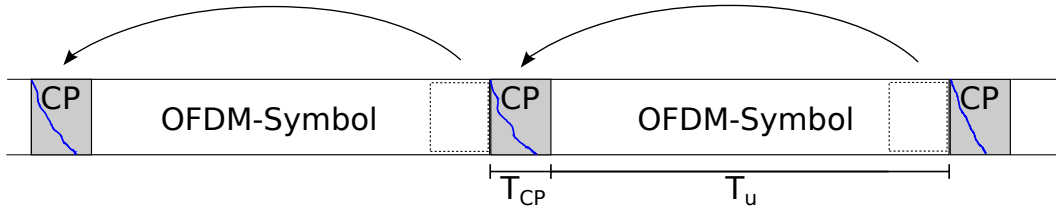
$$x[n] = N \cdot \text{IDFT}_N(a'_k) \quad (2.6)$$

ausdrücken. Fällt die Wahl von N auf eine Zweierpotenz, kann die IDFT effizient durch eine *Inverse Fast Fourier Transformation* (IFFT) berechnet werden. Die Konstruktion eines OFDM-Signals ist also mithilfe einer IFFT einfach zu realisieren. Am Empfänger kann diese Operation mit einer Fast Fourier Transformation (FFT) rückgängig gemacht werden.

Problem bei OFDM ist das hohe *Peak-to-Average Power Ratio* (PAPR). Ein OFDM-Sender muss daher auch bei hohen Ausgangsleistungen linear sein, um Verzerrungen zu vermeiden. Unter anderem wird deswegen OFDM nur im Downlink von der Basisstation genutzt.

## Cyclic Prefix

Unter Einfluss eines zeitdispersiven Kanals sind aufeinanderfolgende OFDM-Symbole nicht orthogonal. Daher wird jedem OFDM-Symbol ein *Cyclic Prefix* (CP) vorangestellt, welches eine Kopie des hinteren Teils des OFDM-Symbols ist. Die Symboldauer wird dadurch von  $T_u$  auf  $T_u + T_{CP}$  erhöht. Die Länge des CP wird so gewählt, dass jede Mehrwegeausbreitung des vorangegangenen OFDM-Symbols im CP auftritt, wodurch ISI vollständig unterdrückt werden kann. Das CP ist daher ein Schutzintervall zwischen den OFDM-Symbolen. Das CP



**Abbildung 2.1:** Cyclic Prefix Schema (blau: Delay Spread)

stellt einen Overhead dar, da sich im CP nur redundante Information befindet. Bei LTE gibt es 2 CP-Modi, welche in Tabelle 2.1 dargestellt werden.

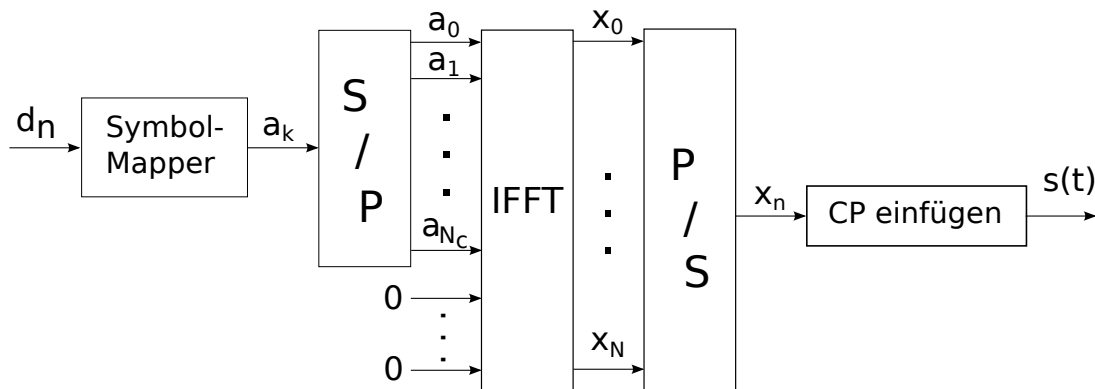


	Normal CP		Extended CP
	0. Slot	sonst	
Dauer	$5.2 \mu s$	$4.7 \mu s$	$16.7 \mu s$
Pfadlänge	1.6 km	1.4 km	5 km
Overhead	7.8 %	7.0 %	25 %
Symboldauer $T_u$	$66.7 \mu s$		

**Tabelle 2.1:** Cyclic Prefix Parameter

### Aufbau eines OFDM-Senders/Empfängers

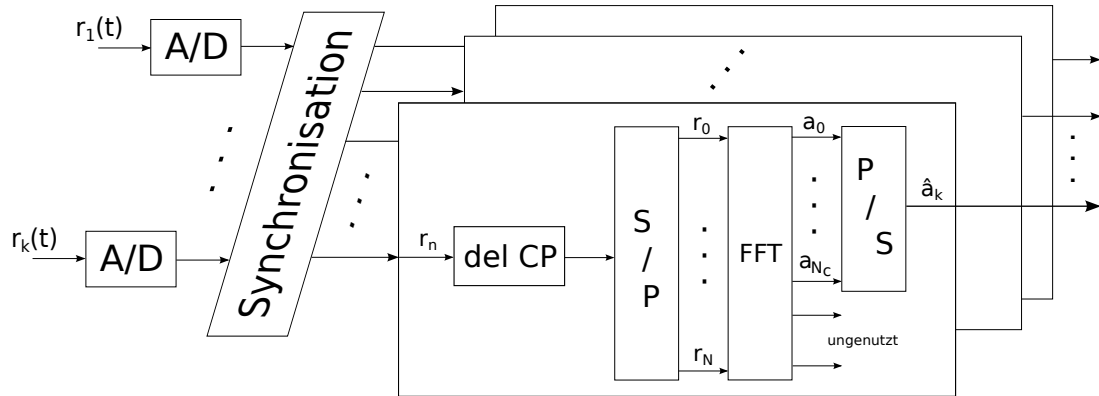
In Abbildung 2.2 ist der Aufbau eines OFDM-Senders bzw. des Sendesignals  $s(t)$  dargestellt. Die zu übertragenden Daten  $d_n$  liegen zunächst als Bits vor und werden durch Symbol-Mapping auf die komplexen Sendesymbole  $a_k$  abgebildet.  $N_C$  Sendesymbole werden dann nach seriell/parallel Wandlung einer IFFT der Länge  $N$  unterzogen, wobei die Sendesymbole um den DC-Carrier (*Direct Current*) herum angeordnet werden. Der DC-Carrier ist der mittlere Unterträger und entspricht im Basisband der Frequenz 0, bzw. einem Gleichanteil. Er wird immer zu null gesetzt, um die Analog-Digital-Wandlung zu vereinfachen. Die Ausgangsvektoren der IFFT stellen die zu sendenden Samples im Zeitbereich dar. Das kontinuierliche Zeitsignal  $s(t)$  ergibt sich nach parallel/seriell-Wandlung, Einfügen des *Cyclic Prefixes* und Digital/Analog-Wandlung. Bei Verwendung mehrerer Sendeantennen ist diese Struktur entsprechend mehrfach vorhanden, da i.A. für jede Antenne ein eigenes Signal berechnet wird.



**Abbildung 2.2:** Aufbau eines OFDM-Senders im komplexen Basisband

Am Empfänger werden obige Operationen wieder rückgängig gemacht. Der Aufbau eines solchen Empfängers ist in Abbildung 2.3 dargestellt. Bei Verwendung mehrerer Empfangsantennen sind die Empfangssignale  $r_k$  zueinander synchron und können so gemeinsam synchronisiert werden. Die nachfolgende Demodulierungsstruktur ist für jeden Empfangskanal mehrfach vorhanden, da jede Empfangsantenne eine eigene Version des gesendeten Signals empfängt. Nach der Synchronisation ist die zeitliche Position der OFDM-Symbole bekannt und das CP kann entfernt werden. Die Seriell-Parallel-Wandlung extrahiert immer ein OFDM-Symbol aus dem Eingangsdatenstrom. Diese werden dann durch eine FFT

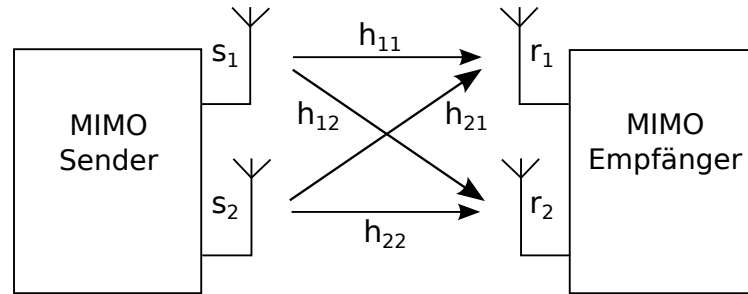
in den Frequenzbereich überführt und die genutzten Unterträger extrahiert, welche die zu dekodierenden Schätzungen für die Sendesymbole enthalten.



**Abbildung 2.3:** Aufbau eines OFDM-MIMO-Empfängers im komplexen Basisband

### 2.1.3 Kanalmodell

Ein MIMO-System besitzt allgemein  $N_t$  Sendeantennen und  $N_r$  Empfangsantennen. In Abbildung 2.4 ist beispielhaft ein solches System mit jeweils 2 Antennen auf beiden Seiten dargestellt. Bei Verwendung von OFDM beanspruchen die einzelnen Unterträger nur einen



**Abbildung 2.4:**  $2 \times 2$  MIMO-System

kleinen Frequenzbereich. Einzelne Unterträger können daher als frequenzflach und über die Dauer eines OFDM-Symbols ( $T_u = 66.7 \mu\text{s}$ ) als konstant angesehen werden. Einem Unterträger in einem OFDM-Symbol wird daher ein komplexer Kanalkoeffizient  $h_{ij}$  zwischen Empfangsantenne  $j$  und Sendeantenne  $i$  zugeordnet. Eine Empfangsantenne empfängt dann eine additive Überlagerung der mit den jeweiligen Kanalkoeffizienten gewichteten Sendesymbole. Bei einem  $2 \times 2$  System ergibt sich dabei das Übertragungsmodell

$$\mathbf{r} = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} + \begin{pmatrix} n_1 \\ n_2 \end{pmatrix} = \mathbf{H} \cdot \mathbf{s} + \mathbf{n} \quad (2.7)$$

mit dem Empfangsvektor  $\mathbf{r}$ , Sendevektor  $\mathbf{s}$  und Rauschvektor  $\mathbf{n}$ . Die Kanalmatrix  $\mathbf{H} \in \mathbb{C}^{N_r \times N_t}$  beinhaltet die Kanalkoeffizienten und beschreibt den MIMO-Kanal eines Unterträgers in einem OFDM-Symbol. Jedem dieser Unterträger wird damit eine eigene Kanalmatrix

zugeordnet, wobei in Zeit- und Frequenz benachbarte Unterträger sehr ähnliche Kanaleigenschaften besitzen. Über den Rauschvektor  $\mathbf{n}$  wird dem System komplexes gaußsches Rauschen hinzugefügt, sodass sich ein bestimmtes Signal-zu-Rauschverhältnis ergibt. Der Betrag der Kanalkoeffizienten beschreibt die jeweilige Kanaldämpfung, die Phase die verursachte Phasendrehung. Um diesen Kanaleinfluss am Empfänger rückgängig zu machen, ist die Schätzung der Kanalkoeffizienten notwendig.

Im Allgemeinen tritt in einem Mobilfunkkanal Mehrwegeausbreitung auf, d.h. dass mehrere zeitlich verschobene und skalierte Versionen eines OFDM-Symbols empfangen werden. Da hier jedoch OFDM mit einem Schutzintervall in Form des *Cyclic Prefix* verwendet wird, verursacht eine Mehrwegeausbreitung nur eine Phasendrehung der Empfangssymbole, welche vollständig durch die Kanalkoeffizienten beschrieben wird. Daher kann hier Mehrwegeausbreitung und ISI vernachlässigt werden [GZAM10, S.99ff/Kap. 3].

Die verwendete Zuordnung der Sendesymbole auf die Antennen wird in Abschnitt 2.2.2 beschrieben.

Die Distanz zwischen 2 Antennen gibt an, wie stark die Kanalkoeffizienten korreliert sind. Eine Distanz größer als die halbe Wellenlänge reicht dabei oft aus, um unabhängiges Fading beider Antennen zu erreichen. Ist die Distanz kleiner, verhalten sich beide Antennen in ihren Kanaleigenschaften ähnlich. Jede Empfangsantenne empfängt eine eigene Version der überlagerten Signalen beider Sendeantennen, somit ist auch eine destruktive Interferenz am Empfänger möglich [DPS14, S.63ff/Kap.5].

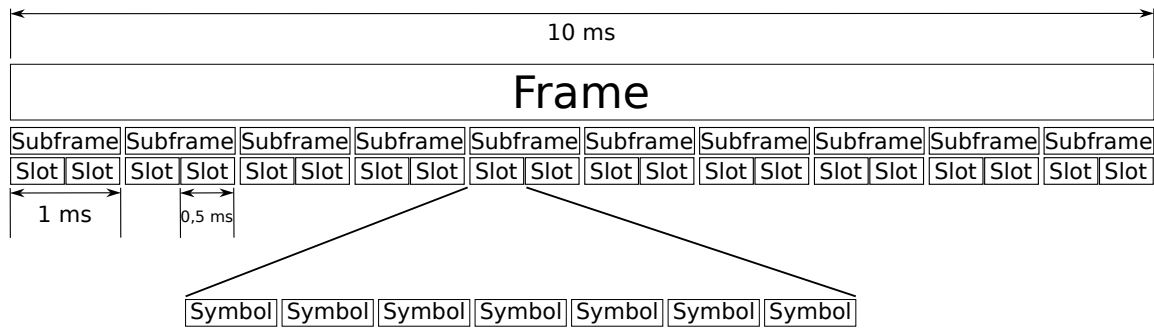
## 2.1.4 Framestruktur

Eine LTE-Basisstation kann entweder im *Frequency Division Duplexing* (FDD) oder im *Time Division Duplexing* (TDD) Modus betrieben werden. Bei Verwendung von FDD befindet sich der Uplink auf einer separaten Trägerfrequenz, während bei TDD Uplink und Downlink auf der gleichen Trägerfrequenz zeitlich getrennt werden. Da in Deutschland zurzeit nur FDD verwendet wird, wird im weiteren Verlauf der Arbeit nur FDD behandelt. Ein LTE-Downlink-Signal besteht aus Frames der Dauer 10 ms, welche in 10 Subframes der Dauer 1 ms unterteilt sind. Jeder Subframe besteht wiederum aus 2 Slots der Länge 0.5 ms, in dem 7 OFDM-Symbole bei Verwendung des *Normal CP* und 6 OFDM-Symbole bei Verwendung des *Extended CP* untergebracht sind.

Im LTE Standard ist eine Basiszeit  $T_S = 1/(15000 \cdot 2048)$  s definiert, welche der Samplezeit bei einer Implementierung mit der FFT-Länge 2048 entspricht. Alle verwendeten Zeitintervalle sind Vielfache von  $T_S$ . Ein OFDM-Symbol besitzt die Länge  $T_u = 2048T_S \approx 66.7 \mu\text{s}$ , ein Slot  $T_{\text{Slot}} = 15360T_S$ , ein Subframe  $T_{\text{Sub}} = 30720T_S$  und ein Frame  $T_{\text{Frame}} = 307200T_S$ .

Frames werden durch die *System Frame Number (SFN)* kontinuierlich von 0 bis 1023 durchnummeriert, also in einem Intervall von 10.24 s. Dies dient unter anderem dazu verschiedene Kontrolldaten, welche nicht in jedem Frame gesendet werden, zeitlich zu lokalisieren.

In Frequenzrichtung variiert die Zahl der Unterträger je nach Systembandbreite, der Unterträgerabstand beträgt jedoch immer  $\Delta f = 15 \text{ kHz}$ . Die kleinste Übertragungseinheit ist ein *Resource Element*, bestehend aus einem Unterträger in einem OFDM Symbol. 12 Unterträger in Frequenzrichtung und 1 Slot in Zeitrichtung bilden einen sogenannten *Resource Block* (RB), die kleinste Einheit welche für effizientes Scheduling einem Empfänger zur Datenübertragung zugeordnet wird [DPS14, S.141ff/Kap. 9.1]. In Abbildung 2.7 wird ein



**Abbildung 2.5:** LTE - Framestruktur in Zeitrichtung [Dem12, Kap. 2.2]

Ausschnitt der Zeitfrequenzebene gezeigt.

Ein LTE-Downlink kann die in Tabelle 2.2 dargestellten Bandbreiten verwenden. Die tatsächlich genutzte Bandbreite ist aufgrund von Schutzbändern am Rand kleiner.

Bandbreite (MHz)	1.4	3	5	10	15	20
Signalbandbreite (MHz)	1.08	2.7	4.5	9	13.5	18
Resource Blocks	6	15	25	50	75	100
Unterträger	72	180	300	600	900	1200

**Tabelle 2.2:** Systembandbreite

### 2.1.5 Physikalische Kanäle

Zur Übertragung von Daten und Kontrollinformationen werden mehrere physikalische Kanäle benutzt, welchen genau definierte *Resource Elemente* in einem Frame zugeordnet werden.

#### Physical Broadcast Channel

Über den *Physical Broadcast Channel* (PBCH) wird der *Master Information Block* (MIB) übertragen, der grundlegende Zellinformationen wie Systembandbreite, Konfigurationsdaten für andere Kanäle (Größe des PHICH) und die *System Frame Number* (SFN) - ohne die hinteren 2 Bits - des gegenwärtig übertragenen Frames. Der MIB besteht aus 14 Informationsbits, 10 reservierten Bits für spätere LTE-Releases und 16 CRC Bits (*Cyclic Redundancy Check*). Diese 40 Bits werden durch Faltungscodierung als *Forward Error Correction* (FEC) mit einer Coderate von 1/3 und Wiederholung auf 1920 Bits umgesetzt, wodurch sich eine sehr kleine Coderate von 1/48 ergibt. Damit wird ein sehr guter Fehlerschutz erreicht, um auch am Zellrand einen robusten Empfang zu gewährleisten.

Der PBCH sendet die 1920 Bits über 4 Frames verteilt in einer 40 ms Periode. Jeder Frame beinhaltet also immer nur ein Viertel der kodierten Bits, jedoch lässt sich jede dieser Teilmengen aufgrund redundanter Information getrennt dekodieren. Somit kann in jedem Frame der MIB ausgelesen werden, womit eine zeitliche Diversität realisiert wird. Da im MIB die letzten 2 Bit der SFN nicht übertragen werden, muss der Empfänger diese aus dem

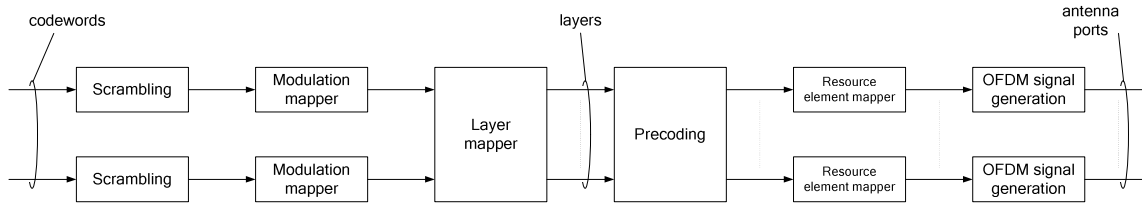


Abbildung 2.6: Datenfluss [3GP12, Kap. 6.3]

40 ms Raster des PBCH ermitteln. Z.B. übertragen die 4 Frames mit der SFN  $4 \oplus 0b100$  bis  $7 \oplus 0b111$  und einer Dauer von 40 ms jeweils nur 0b1.

Ein Empfänger wird bei einem Verbindungsaufbau nach der Synchronisation zunächst den MIB dekodieren. Damit der PBCH auch bei unbekannter Systembandbreite empfangen werden kann, wird er in die 72 zentralen Unterträger um den DC-Carrier herum übertragen, welches der kleinstmöglichen Systembandbreite entspricht. Ein Empfänger benötigt daher nur die Mittenfrequenz des LTE-Signals zur Dekodierung des PBCH. Der PBCH nutzt eine QPSK-Modulation und wird immer im *Transmit Diversity* Modus gesendet (siehe Abschnitt 2.2.2). Durch diese insgesamt sehr robuste Übertragung soll sichergestellt werden, dass auch am Zellrand eine Dekodierung möglich ist [STB11, S.189ff/Kap. 9.2]. Am Ende dieser Arbeit soll der PBCH und der MIB erfolgreich empfangen und dekodiert werden.

Andere physikalische Kanäle sind der *Physical Downlink Shared Channel* (PDSCH) zur Übertragung der Nutzdaten, der *Physical Downlink Control Channel* (PDCCH) zur Ressourcenzuweisung für Up- und Downlink, der *Physical Control Format Indicator Channel* (PCFICH) zur Signalisierung der Größe des PDCCH, der *Physical Hybrid ARQ Indicator Channel* (PHICH) zur Übertragung von Empfangsbestätigungen (ACKs) und der *Physical Multicast Channel* (PMCH) zur Übertragung von Multicast-Inhalten.

## Datenfluss

Der Datenfluss von Bit zu Antenne wird allgemein in Abbildung 2.6 dargestellt. Die in Codewörter eingeteilten Bits werden nach dem Scrambling in komplexe Modulationssymbole übersetzt und in ein oder mehrere Layer aufgeteilt. Als Modulationsverfahren kommt je nach Kanal BPSK, QPSK, 16QAM oder 64QAM zum Einsatz. Das Precoding und Layer Mapping ist abhängig von der verwendeten Übertragungsart, *Spatial Multiplexing* oder *Transmit Diversity* (siehe Abschnitt 2.2), und Anzahl der vorhandenen Antennenports  $N_r$ . Zum Beispiel kommt beim PBCH-Precoding hier der Alamouti-Code zum Einsatz.

### 2.1.6 Referenzsignale

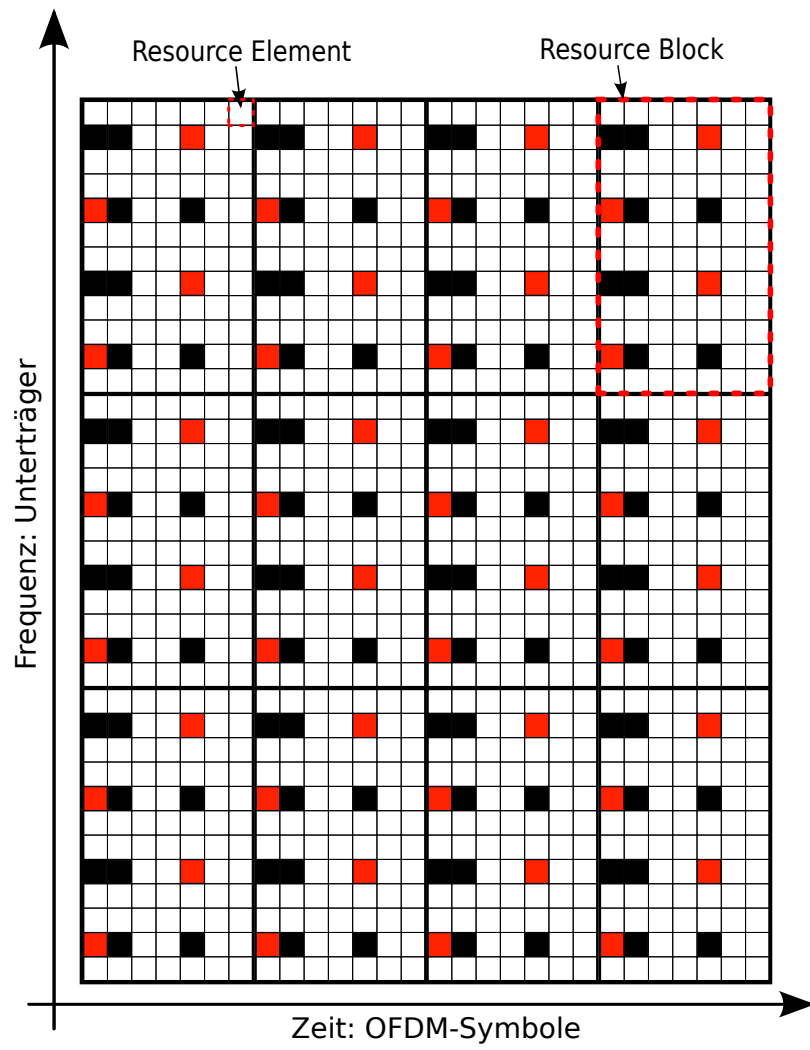
Zur Kanalschätzung werden neben denen zur Synchronisation genutzten *Primary Synchronisation Signal* und *Secondary Synchronization Signal* (siehe Abschnitt 3) zellspezifische Referenzsignale (RS) bzw. Pilotsymbole in die Zeitfrequenzebene eingebettet. Es gibt für jeden Antennenport eine eigene RS-Sequenz. Für Antennenport 0 und 1 werden die RS im ersten und drittletzten OFDM-Symbol in jedem Slot mit jeweils einem Abstand von 6 Unterträgern im Frequenzbereich übertragen. Der Wert und die genaue Position der RS-Sequenzen

in Frequenzrichtung hängt von der Cell-ID  $N_{\text{ID}}$  ab - somit gibt es insgesamt 504 verschiedene Sequenzen an unterschiedlichen Positionen. Unterträger, welche von einer Sendeantenne zur Übertragung der RS genutzt werden, werden von den anderen Sendeantennen nicht genutzt, wodurch die Messung des Kanals für jedes Sende- und Empfangsantennen-Paar möglich wird [STB11, S.167ff/Kap. 8.2].

Die RS-Sequenzen werden nach [3GP12, Kap. 6.10.1] durch

$$r_{l,n_s}(m) = \frac{1}{\sqrt{2}}(1 - 2 \cdot c(2m)) + j \cdot \frac{1}{\sqrt{2}}(1 - 2 \cdot c(2m+1)) \quad m = 0, 1, \dots, 2 \cdot N_{\text{RB}}^{\text{max,DL}} - 1 \quad (2.8)$$

berechnet. Dabei ist  $N_{\text{RB}}^{\text{max,DL}} = 110$  und  $c(n)$  eine pseudozufällige Binärsequenz - somit ergibt sich für die RS-Sequenzen eine zufällige QPSK-Modulation. In Abbildung 2.7 wird beispielhaft die Zuordnung der RS in der Zeitfrequenzebene einer Antenne gezeigt. Rot sind dabei die für das RS genutzte *Resource Elemente*, schwarz die nicht genutzten Resource Elemente, da sie von einer anderen Sendeantenne in Anspruch genommen werden. Die im zweiten OFDM-Symbol jedes RBs dargestellten RS werden von Antennenport 2 und 3 genutzt, falls diese an der Basisstation vorhanden sind (siehe Abschnitt 2.2.1), ansonsten stehen diese für andere Daten zur Verfügung.



**Abbildung 2.7:** Zeitfrequenzebene mit RS-Sequenzen

## 2.2 MIMO

Multiple Input Multiple Output (MIMO) bezeichnet allgemein die Verwendung mehrerer Sendeantennen  $N_t$  und Empfangsantennen  $N_r$ , wodurch sich im Vergleich zu einem Einantennensystem mehrere Vorteile ergeben: Zum einen wird die Kanalkapazität und somit die mögliche Nutzdatenrate erhöht (*Spatial Multiplexing*), zum anderen ergibt sich ein Diversitätsgewinn gegen das Fading des Kanals. MIMO beansprucht dabei nicht mehr Bandbreite, benötigt aber eine aufwendigere Signalverarbeitung und mehrere Antennen auf Sende- und Empfängerseite [Jan04].

### 2.2.1 Antennenkonfiguration

In LTE gibt es drei verschiedene mögliche Antennenkonfigurationen an der Basisstation, welche 1, 2 oder 4 Antennenports besitzen kann. Ein Empfänger muss die Antennenkonfiguration kennen, um die richtigen Referenzsignale und Dekodierungsschemata zu verwenden. Dies wird von der Basisstation nicht mitgeteilt, sondern muss durch ausprobieren am Empfänger ermittelt werden. Es ist nur mit genau einer Konfiguration möglich den MIB erfolgreich zu dekodieren.

Ein Antennenport ist nicht gleichbedeutend mit einer physikalischen Antenne, sondern ein in LTE verwendetes Konzept. Jedem Antennenport sind dabei eigene Referenzsignale zugewiesen, können aber von der Basisstation über dieselbe physikalische Antenne übertragen werden. Um echtes MIMO zu ermöglichen sind jedoch mehrere Antennen nötig [DPS14, S.147f/Kap.9.3].

### 2.2.2 Diversity

Bei Diversity wird zwischen *Receive* und *Transmit Diversity* unterschieden. *Transmit Diversity* bezieht sich dabei auf die Verwendung mehrerer Sendeantennen, *Receive Diversity* auf die Verwendung mehrerer Empfangsantennen. Die Datenrate steigt durch *Diversity* nicht, jedoch ergibt sich der Vorteil einer zuverlässigeren Übertragung.

#### Transmit Diversity

Bei Verwendung mehrerer Sendeantennen nutzt LTE den Alamouti-Code oder auch *Space-Time Block Code* (STBC). Dieser Code benötigt dabei keine Kenntnis des Kanals am Sender und gehört damit zu den Open-Loop Verfahren. Bei 2 Sendeantennen werden 2 Sendesymbole ( $s_1, s_2$ ) auf 2 unterschiedlichen aber benachbarten Unterträgern ( $f_1, f_2$ ) gesendet.

		Antenne	
		1	2
Unterträger	$f_1$	$s_1$	$s_2$
	$f_2$	$-s_2^*$	$s_1^*$

**Tabelle 2.3:** Alamouti - Code



Da die Symbole nicht über 2 aufeinanderfolgende zeitlich getrennte OFDM-Symbole, sondern über 2 in der Frequenz getrennten Unterträgern, übertragen werden, handelt es sich hierbei genau genommen um einen *Space-Frequency Block Code* (SFBC) anstatt eines STBC. Mathematisch besteht jedoch kein Unterschied.

Im Folgenden wird der Empfang mit einer Antenne betrachtet. Es wird angenommen, dass der Kanal frequenzflach und durch die Kanalkoeffizienten  $h_1(f)$  von Antenne 1 und durch  $h_2(f)$  von Antenne 2 zur Empfangsantenne beschrieben wird. Da weiterhin die verwendeten Unterträger benachbart sind, kann  $h_1(f_1) = h_1(f_2) = h_1$  angenommen werden, was bei ausreichend großer Kohärenzbandbreite gerechtfertigt ist. Das empfangene Signal  $r(f)$  ergibt sich dann zu

$$\begin{aligned} r(f_1) &= h_1 s_1 + h_2 s_2 + n(f_1), \\ r(f_2) &= -h_1 s_2^* + h_2 s_1^* + n(f_2). \end{aligned} \quad (2.9)$$

$n(f)$  beschreibt hierbei das Rauschen. Kombiniert man die empfangenen Samples gemäß

$$\begin{aligned} y_1 &= h_1^* r(f_1) + h_2 r^*(f_2), \\ y_2 &= h_2^* r(f_1) - h_1 r^*(f_2) \end{aligned} \quad (2.10)$$

und setzt Gleichung (2.9) ein, erhält man

$$\begin{aligned} y_1 &= (|h_1|^2 + |h_2|^2) s_1 + h_1^* n(f_1) + h_2 n^*(f_2) \\ y_2 &= (|h_1|^2 + |h_2|^2) s_2 + h_2^* n(f_1) - h_1 n^*(f_2). \end{aligned} \quad (2.11)$$

Hierbei ist  $y_i$  die Schätzung für das Sendesymbol. Es ist somit am Empfänger durch lineare Kombination leicht möglich die überlagerten Symbole zu trennen, wobei eine Schätzung des Kanals benötigt wird. Durch die Unabhängigkeit beider Antennenkanäle wird so durch diesen Code der Einfluss von schnellem Fading reduziert. Die Sendeleistung wird im Vergleich zu einem Einantennensystem auf mehrere Sendeantennen aufgeteilt, somit steigt das SNR am Empfänger nicht, sondern bleibt über die Zeit gesehen konstanter am mittleren SNR [GZAM10, S.174ff/Kap.5.3].

Der *Transmit-Diversity* Modus mit 4 Sendeantennen nutzt sowohl SFBC als auch *Frequency-Switched Transmit Diversity* (FSTD). Die mit SFBC codierten Symbole werden abwechselnd auf 2 Sendeantennenpaare übertragen. Es werden also gleichzeitig 4 Symbole auf 4 verschiedenen Unterträgern gesendet.

		Antenne			
		1	2	3	4
Unterträger	$f_1$	$s_1$	0	$-s_2^*$	0
	$f_2$	$s_2$	0	$s_1^*$	0
	$f_3$	0	$s_3$	0	$-s_4^*$
	$f_4$	0	$s_4$	0	$s_3^*$

**Tabelle 2.4:** SFBC und FSTD

## Receive Diversity

*Receive Diversity* spielt sich nur am Empfänger durch Verwendung mehrerer Empfangsantennen ab. Die parallelen Streams müssen kohärent zusammengefügt werden. Eine direkte Addition der Empfangssymbole ist aufgrund unbekannter Phasenbeziehung zwischen unterschiedlichen Empfangsantennen nicht möglich.

Zur Kombination gibt es zum einen die Möglichkeit des *Selection Combining* (SC). Hierbei wird der Stream mit der größten Empfangsstärke ausgewählt und die anderen verworfen. Dies ist zwar ein simples Verfahren, da nur ein Stream verarbeitet werden muss, jedoch suboptimal, weil die Energie der anderen Empfangsantennen ungenutzt bleibt.

Besser ist die Verwendung von *Maximum Ratio Combining* (MRC), welches alle Empfangsstreams verarbeitet und gewichtet addiert. Für die Gewichtung wird hier die Kanalschätzung herangezogen, da die Kanalkoeffizienten  $|h_{k,x}|^2$  ein Maß für die Empfangsstärke des Unterträgers  $x$  an Antenne  $k$  ist. Das in Gleichung (2.10) verwendete Kombinationsschema wird hier auf mehrere Empfangsantennen erweitert, indem die Alamouti-Dekodierung für jede einzelne Empfangsantenne kohärent addiert und anschließend durch die Gesamtenergie der verwendeten Unterträger normiert wird. Somit ergibt sich

$$\begin{aligned} y_1 &= \left( \sum_{k=0}^{N_r-1} \underbrace{h_{k,1}^* r_k(f_1) + h_{k,2} r_k^*(f_2)}_{y_{k,1}} \right) / \left( \sum_{k=0}^{N_r-1} |h_{k,1}|^2 + |h_{k,2}|^2 \right) = s_1 + n'_1 \\ y_2 &= \left( \sum_{k=0}^{N_r-1} \underbrace{h_{k,2}^* r_k(f_1) - h_{k,1} r_k^*(f_2)}_{y_{k,2}} \right) / \left( \sum_{k=0}^{N_r-1} |h_{k,1}|^2 + |h_{k,2}|^2 \right) = s_2 + n'_2 \end{aligned} \quad (2.12)$$

als Kombinationsschema für mehrere Empfangsantennen. Jedes Empfangssymbol  $r_k(f_x)$  wird so mit  $|h_{k,x}|^2$  gewichtet, wodurch der Stream mit der besseren Empfangsstärke höher in die Symbolberechnung eingeht. Dies wird für jeden Unterträger einzeln angewandt, da die Empfangsstärke auch über der Frequenz variiert [DPS14, S.65ff/Kap. 5.3].

Durch Verwendung von *Receive Diversity* in Kombination mit MRC und unkorrelierten Antennen steigt das SNR linear mit der Zahl der Empfangsantennen. Somit steigt die Kanalkapazität nach der Formel von Shannon  $C = B \log(1 + \text{SNR})$  logarithmisch [GZAM10, S.180/Kap. 5.3.3].

### 2.2.3 Spatial Multiplexing

*Spatial Multiplexing* wird im implementierten Empfänger zwar nicht verwendet, soll hier aber trotzdem kurz vorgestellt werden.

LTE verwendet *Spatial Multiplexing* zur Erhöhung der Datenrate nur bei der Übertragung von Nutzdaten im PDSCH. Die Sendedaten werden in mehrere parallele Streams  $M$  aufgeteilt und zum gleichen Zeitpunkt über verschiedene Sendeantennen übertragen. Die genutzte Bandbreite bleibt dabei konstant, während sich die spektrale Effizienz um den Faktor  $M$  erhöht.

In einem  $2 \times 2$  System ergibt sich das empfangene Signal nach Gleichung (2.7). Die 2 verschiedene Sendesymbole können am Empfänger vereinfacht (Zero-Forcing) durch

$$\begin{pmatrix} s'_1 \\ s'_2 \end{pmatrix} = \mathbf{H}^{-1} \cdot \mathbf{r} = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} + \mathbf{H}^{-1} \cdot \mathbf{n} \quad (2.13)$$

geschätzt werden. Dazu muss  $\mathbf{H}$  jedoch invertierbar sein. Die Kanaleigenschaften, definiert durch die Kanalmatrix  $\mathbf{H}$ , geben also vor, wieviele Layer M verwendet werden können und ist auf  $M \leq \min(N_r, N_t)$  beschränkt. Die Zahl der von 0 verschiedenen Eigenwerten von  $\mathbf{H}$  muss mindestens M entsprechen, also  $\text{rang}(\mathbf{H}) \geq M$ . Nur dann ist die Matrix invertierbar. Sind sich die Einzelkanäle, wie zum Beispiel in einem Line-of-sight Szenario, sehr ähnlich, ist keine Trennung der Empfangssymbole in die Sendesymbole und damit die Verwendung von *Spatial Multiplexing* möglich - die Kanalmatrix ist dann singulär, bzw. dessen Inverse besitzt sehr große Elemente und Schätzungen werden so durch Rauschen unbrauchbar gemacht.

Ist eine Verwendung von *Spatial Multiplexing* nicht möglich, so wird in LTE auf den *Diversity*-Modus umgestellt. Der Empfänger teilt dem Sender dabei im Uplink durch die sogenannte *rank indication* (RI) mit, wieviele Layer M gesendet werden sollen bzw. getrennt werden können. Ein RI von 1 entspricht dem Diversity-Modus bzw. der Übertragung nur eines Datenlayers. Ein größerer RI entspricht der Übertragung mit *Spatial Multiplexing*.

Ein weiteres Problem ist, dass die Signale an den Empfangsantennen je nach Kanal destruktiv interferieren können. Durch ein *closed-loop* Verfahren werden die am Empfänger gemessenen Kanaleigenschaften an den Sender rückgekoppelt und die Sendesignale durch eine *Precoding*-Matrix vorcodiert. Das Vorcodieren ist dabei auf den Kanal abgestimmt, so dass sich die Signale nicht mehr destruktiv überlagern. Da die Rückkopplung des gesamten Kanals zu aufwendig wäre, wird die *Precoding*-Matrix aus einem vordefinierten Codebook ausgewählt. Der Empfänger verlangt also vom Sender durch den *precoding matrix indicator* (PMI) eine bestimmte Matrix aus diesem Codebook zu verwenden [DPS14, S.77ff/Kap. 5.5, S.189f/Kap. 10.3].



## 3 Synchronisation

Zu Beginn einer LTE Downlink Verbindung muss zunächst eine anfängliche Zell-Suche durchgeführt werden, bevor OFDM-Symbole demoduliert werden können. Dabei muss zunächst eine Zeit- und Frequenzsynchronisation stattfinden, wozu 2 Synchronisationssignale vorgesehen sind: das *Primary Synchronization Signal* (PSS) und das *Secondary Synchronization Signal* (SSS). Weiterhin kann auch mithilfe des CP eine Synchronisation auf den Symboltakt erfolgen. In den folgenden Abschnitten werden zunächst die im LTE-Standard definierten Synchronisationssignale vorgestellt. Dann wird auf verschiedene Zeit- und Frequenzsynchronisationsverfahren eingegangen, welche getestet und verglichen wurden.

### 3.1 Cell-ID

Jede LTE-Basisstation besitzt eine von 504 möglichen Cell-IDs  $N_{\text{ID}}$ . Die Cell-ID berechnet sich durch

$$N_{\text{ID}} = 3 \cdot N_{\text{ID}}^1 + N_{\text{ID}}^2 \quad (3.1)$$

wobei die Cell-ID Group  $N_{\text{ID}}^1$  eine Zahl im Bereich von 0 bis 167 und die Cell-ID Number  $N_{\text{ID}}^2$  eine Zahl im Bereich von 0 bis 2 ist. Die Cell-ID wird verwendet, um verschiedene Basisstationen eindeutig zu unterscheiden. So hängt z.B. die Zuordnung der Referenzsignale auf die Unterträger eines OFDM-Symbols von  $N_{\text{ID}}$  ab [Dem12, Kap. 2.3].

### 3.2 Primary Synchronization Signal

Das *Primary Synchronization Signal* (PSS) besteht im Frequenzbereich aus einer Zadoff-Chu-Sequenz der Länge  $N_{\text{ZC}} = 63$ , wobei der 32. Wert der Sequenz ausgelassen wird, da dieser dem DC-Carrier zugeordnet wird. Die Sequenz wird durch

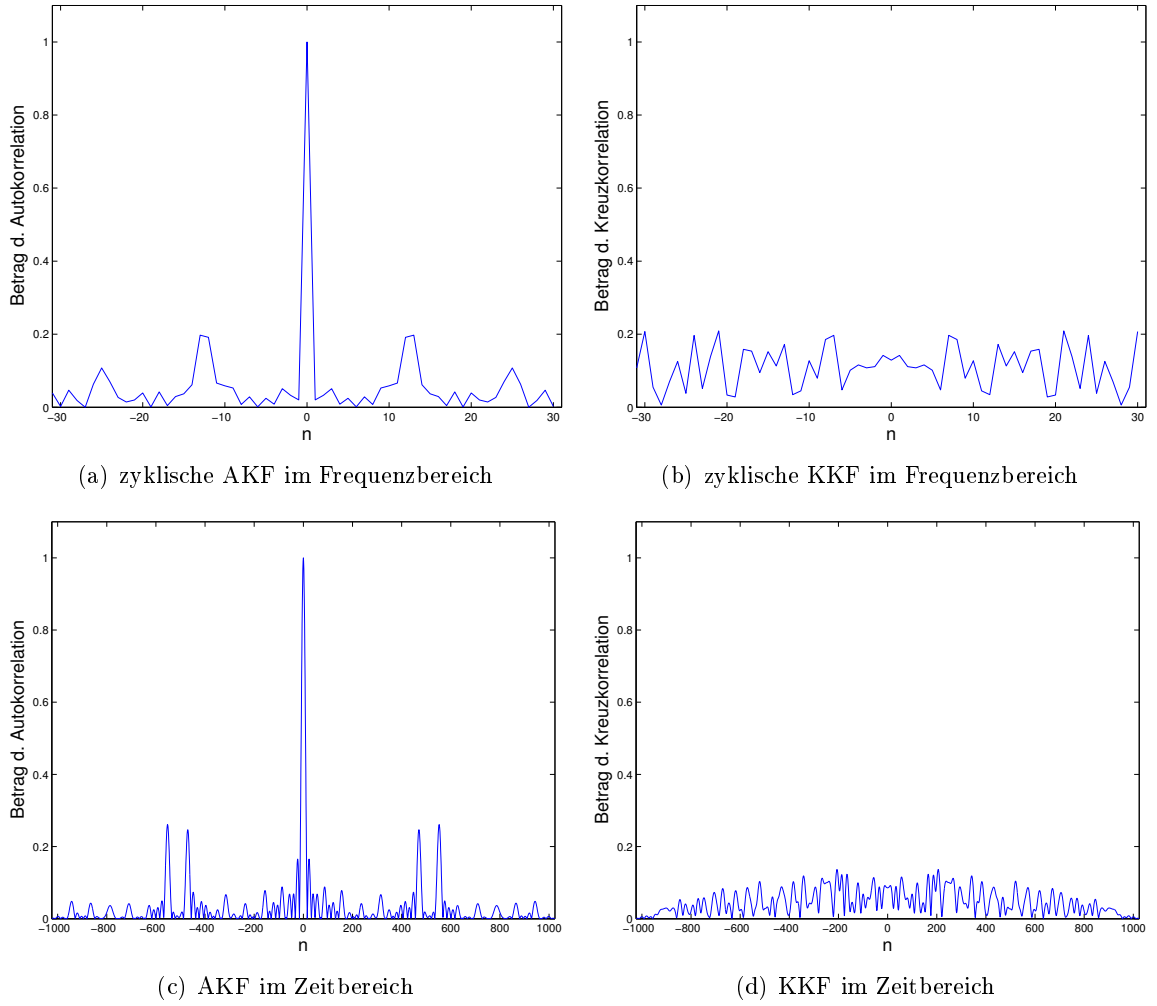
$$P_u(n) = \begin{cases} e^{-j \frac{\pi u n(n+1)}{N_{\text{ZC}}}} & n = 0..30 \\ e^{-j \frac{\pi u (n+1)(n+2)}{N_{\text{ZC}}}} & n = 31..61 \end{cases} \quad (3.2)$$

berechnet. Dabei wird  $u$  als Wurzel der Zadoff-Chu-Sequenz bezeichnet und kann je nach  $N_{\text{ID}}^2 = 0, 1, 2$  die Werte 25, 29 oder 34 annehmen [3GP12]. Das PSS ist damit als eine von drei möglichen Sequenzen für eine Basisstation fest vorgegeben und ändert sich nicht. Zadoff-Chu-Sequenzen besitzen einige wichtige Eigenschaften [STB11, S.155ff/Kap. 7.2.1]:

- Die Amplitude jedes Wertes der Sequenz ist konstant

- Zadoff-Chu-Sequenzen erfüllen die Eigenschaft eines *Constant Amplitude Zero Auto-correlation* (CAZAC) Signals, d.h. die “periodische“ AKF, also die Korrelation der Sequenz mit einer beliebig zyklisch verschobenen Version der Sequenz ist 0.
- Die “periodische“ Kreuzkorrelation zweier verschiedener ZC-Sequenzen gleicher Länge ist konstant  $\frac{1}{\sqrt{N_{\text{ZC}}}}$ , falls  $|u_1 - u_2|$  teilerfremd zu  $N_{\text{ZC}}$  ist.

Das PSS besitzt im Frequenzbereich gute Korrelationseigenschaften, welche auch im Zeitbereich existieren. Es ist deshalb nicht zwingend notwendig, das PSS im Frequenzbereich zu betrachten, wodurch eine dafür notwendige Fouriertransformation entfällt. In Abbildung 3.1 ist die AKF und KKF mit 2 unterschiedlichen PSS-Sequenzen sowohl im Frequenz- als auch im Zeitbereich nach einer IFFT der Länge 1024 dargestellt. Oben genannte Eigenschaften gelten nur näherungsweise, da in der PSS-Sequenz der DC-Carrier zu 0 gesetzt wird.



**Abbildung 3.1:** Vergleich der AKF ( $N_{\text{ID}}^2 = 1$ ) und KKF ( $N_{\text{ID},1}^2 = 0$ ,  $N_{\text{ID},2}^2 = 1$ ) des PSS im Zeit- und Frequenzbereich

Das PSS wird alle 5 ms im letzten OFDM Symbol des 0. und 10. Slots gesendet, also 2 mal pro Frame. Nach der PSS-Synchronisation ist der Empfänger folglich auf einen halben

Frame genau synchronisiert. Im Frequenzbereich befindet sich das PSS in den 62 zentralen Resource Elementen um den DC-Carrier, wodurch die Position unabhängig von der dem Empfänger noch unbekannten Systembandbreite ist und nur die Mittenfrequenz bekannt sein muss. Das PSS wird nur über einen Antennenport gesendet, um destruktive Interferenz am Empfänger zu vermeiden.

### 3.3 Secondary Synchronization Signal

Mithilfe des *Secondary Synchronization Signal* (SSS) wird auf den Frametakt synchronisiert und die Cell-ID  $N_{ID}$  ermittelt. Das SSS wird im vorletzten OFDM-Symbol des 0. und 10. Slots direkt vor dem PSS in den 6 zentralen Resource Blocks um den DC-Carrier herum gesendet. Dabei unterscheidet sich das SSS im 0. und 10. Slot, wodurch die Synchronisation auf den Frametakt möglich wird. Die SSS-Sequenz wird im Frequenzbereich aus 2 unterschiedlichen BPSK-modulierten m-Sequenzen der Länge 31 gebildet. Die Detektion findet zeitlich nach der PSS-Synchronisation im Frequenzbereich statt und stellt den letzten Synchronisationsschritt dar [STB11, S.158f/Kap. 7.2.3].

### 3.4 Zeitsynchronisation

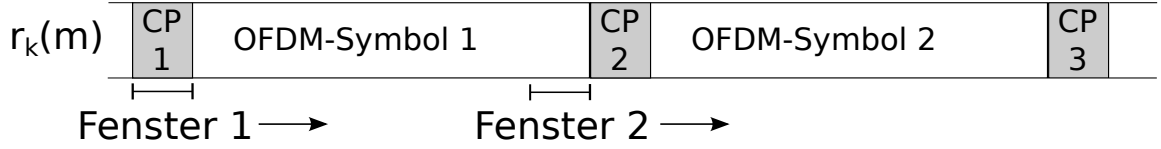
In den folgenden Abschnitten werden verschiedene Methoden zur Cell-Suche und Zeitsynchronisation vorgestellt, welche im Zuge dieser Arbeit getestet wurden. Abschließend werden die Verfahren verglichen. Dabei ist Methode 1 das bisher implementierte Verfahren. Die anderen Methoden basieren dabei auf einer direkten Synchronisation auf den Halbframetakt durch Korrelation im Zeitbereich. In der fertigen Implementation wird eine Kombination von Methode 3 und 4 verwendet (siehe Abschnitt 4.3).

Die Empfangsdaten mehrerer Empfangsantennen werden allgemein parallel verarbeitet und die verwendeten Korrelationen betragsmäßig addiert. Dadurch steigt die Zuverlässigkeit der Synchronisation. Für alle Empfangskanäle wird dabei der gleiche Zeitoffset angenommen, d.h. die Empfangskanäle werden nicht einzeln, sondern gemeinsam synchronisiert.

Die in Abbildung 3.1 gezeigten Korrelationen mit simulierter Empfangsdaten wurden mit der MATLAB-LTE-Toolbox berechnet und basieren auf dem *Extended Typical Urban* (ETU) Kanalmodell, 2 Empfangsantennen und  $SNR = 6$  dB (siehe Abschnitt 5.1).

#### Methode 1: CP Symbolsynchronisation, PSS im Frequenzbereich

Bei dieser Methode nach [Dem12, Kap. 2.3] wird zunächst mithilfe des *Cyclic Prefix* auf den Symboltakt synchronisiert. Dazu wird eine gleitende Korrelation genutzt: 2 Fenster der Länge des CP  $N_{CP}$  werden im Abstand eines OFDM-Symbols kontinuierlich über den Empfangssymbolen verschoben. Die Korrelation beider Fenster wird dann maximal, wenn diese mit dem CP und dem Ende eines OFDM-Symbols zusammenfallen, da das CP eine Kopie des letzten Teiles eines OFDM-Symbols ist.

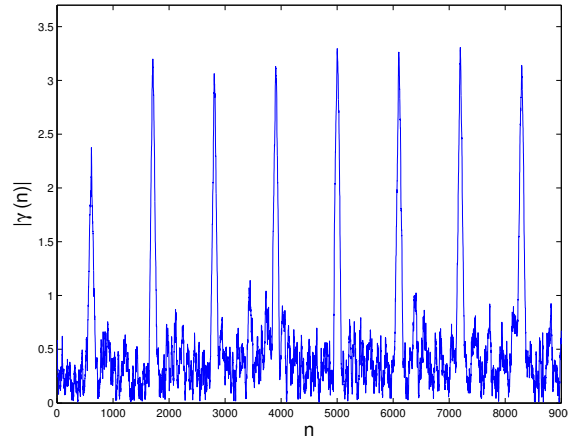


**Abbildung 3.2:** CP - Symboltaktsynchronisation

Diese Operation wird für alle Empfangsantennen parallel ausgeführt. Damit ergibt sich

$$\gamma_k(n) = \sum_{m=n}^{n+N_{\text{CP}}-1} r_k(m)r_k^*(m+N_{\text{FFT}}) \quad (3.3)$$

als Kriterium für den Symboltakt. Ein Vorteil bei diesem Verfahren ist, dass ein möglichen Frequenzoffset den Wert der Korrelation nicht ändert, da dieser den Inhalt beider Fenster gleich beeinflusst. Ein möglicher Mehrwegeempfang eines OFDM-Symbols beeinträchtigt jedoch das nachfolgende CP. In Abbildung 3.4 ist die gleitende Korrelation eines simulierten Signals über mehrere OFDM-Symbole dargestellt. Für jedes OFDM-Symbol entsteht ein Peak und kennzeichnet den Symbolbeginn. Der betragsmäßig größte Peak



**Abbildung 3.3:** CP-Korrelation über einige simulierte OFDM-Symbole

$$n_{\text{CP}} = \arg \max_n \sum_{k=0}^{N_r-1} |\gamma_k(n)| \quad (3.4)$$

definiert den OFDM-Symbolbeginn  $n_{\text{CP}}$ .  $\gamma(n)$  wird über eine definierte Länge von z.B. 200 OFDM-Symbolen berechnet. Alle weiteren OFDM-Symbole befinden sich in entsprechendem Abstand  $k \cdot (N_{\text{FFT}} + N_{\text{CP}})$  von  $n_{\text{CP}}$ .

Nach erfolgter Symbolsynchronisation wird mithilfe der PSS auf den Halbframetaktsynchronisiert. Dazu werden die OFDM-Symbole mittels einer FFT in den Frequenzbereich transformiert (inverse OFDM-Operationen) und die von der PSS beanspruchten  $N_{\text{ZC}} = 62$



Unterträger extrahiert. Durch drei KKF

$$Z_u(n) = \sum_{k=0}^{N_r-1} \left| \sum_{m=0}^{N_P-1} P_u(m) \cdot R_{k,l}^*(m+n) \right| \quad n = 0, 1, \dots, N_P - 1 \quad (3.5)$$

dieser Unterträger jedes OFDM-Symbols mit den drei möglichen PSS-Sequenzen wird die  $N_{\text{ID}}^2$  und die zeitliche Position  $n_P$  des OFDM-Symbols, welche die PSS beinhaltet, ermittelt. Die Entscheidung dafür liefert die KKF mit dem größten Maximum.  $P_u(m)$  ist die PSS-Vergleichssequenz und  $R_{k,l}(m)$  sind die extrahierten Empfangssymbole des OFDM-Symbols  $l$  im Frequenzbereich. Für jede Empfangsantenne wird eine eigene Korrelation durchgeführt und dann betragsmäßig addiert. Eine Betragsbildung ist nötig, da die verschiedenen Empfangsantennen i.A. nicht phasensynchron sind.

## **Methode 2: Korrelation des PSS im Zeitbereich**

Bei dieser Methode wird auf eine Symbolsynchronisation und die Verwendung des CP komplett verzichtet. Stattdessen erfolgt direkt mithilfe des PSS eine Synchronisation auf den Halbframetak. Wie bereits in Abschnitt 3.2 gezeigt, besitzt das PSS gute Korrelationseigenschaften im Zeitbereich. Die KKF

$$K(n) = \sum_{k=0}^{N_r-1} \left| \sum_{m=0}^{N_{\text{FFT}}-1} r_k(m+n) \cdot p^*(m) \right| \quad (3.6)$$

der Empfangssequenz  $r_k(m)$  an Antenne  $k$  und der PSS-Vergleichssequenz im Zeitbereich  $p(m) = \text{IFFT}(P(n))$  der Länge  $N_{\text{FFT}}$  zeigt einen signifikanten Peak an der entsprechenden Stelle. Der Suchbereich muss mindestens einen halben Frame lang sein, damit das Auftreten einer PSS Sequenz innerhalb  $r_k(m)$  garantiert wird. Die Zahl der zu überprüfenden Zeithypothesen beträgt also  $N_{\text{Frame}}/2$ .

Es gibt 3 verschiedene PSS-Sequenzen, daher muss obige KKF 3 mal durchgeführt werden, da nur eine Sequenz mit den Empfangsdaten korreliert. Es ist jedoch möglich, die KKF mit der Summe der PSS-Sequenzen

$$p(m) = p_0(m) + p_1(m) + p_2(m) \quad (3.7)$$

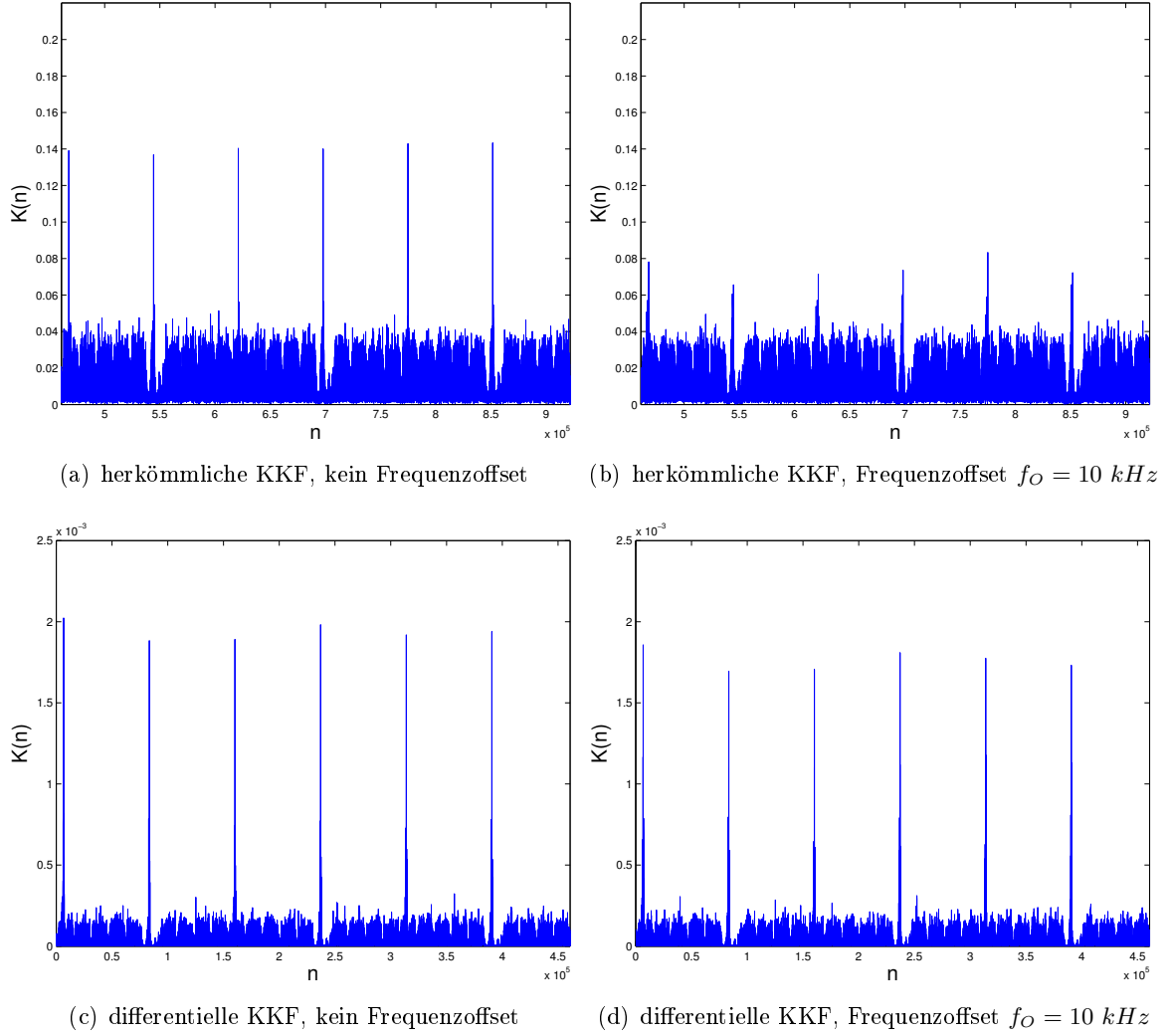
durchzuführen, da die Korrelation dem Distributivgesetz folgt:

$$A * (B + C) = A * B + A * C \quad (3.8)$$

Dabei stellt  $*$  den Korrelationsoperator dar und  $p_N(m)$  bezeichnet die PSS-Sequenz mit  $N = N_{\text{ID}}^2$ . Schlussendlich wird damit die Summe der 3 unterschiedlichen KKF Gleichung (3.6) berechnet, wobei nur eine dieser KKF und somit auch die Summe einen Peak aufweist. Durch diesen Trick kann 2/3 an Rechenoperationen gespart werden, da  $p(m)$  leicht vorberechnet werden kann. In Abbildung 3.4(a) ist die Korrelation mit simulierten Daten über die Dauer von 3 Frames dargestellt. Für die Position des PSS in der Empfangssequenz ergibt sich schließlich zu

$$n_P = \arg \max_n K(n). \quad (3.9)$$

Um die  $N_{\text{ID}}^2$  zu erhalten ist es ausreichend an ermittelter Stelle 3 Korrelationen mit den 3 unterschiedlichen PSS-Sequenzen durchzuführen [YZ13]. Mitteln über mehrere Frames verbessert die Zuverlässigkeit des Ergebnisses. Ein Nachteil dieser Methode ist, dass ein Frequenzoffset sich stark negativ auf das Ergebnis auswirkt, wie in Abbildung 3.4(b) dargestellt. Weiterhin ist diese Korrelation sehr rechenintensiv, da für jede Zeithypothese eine Korrelation der Länge  $N_{\text{FFT}}$  berechnet werden muss: z.B. müssten bei einem 10 MHz Signal so mindestens  $N_{\text{Frame}}/2 = 76800$  Korrelationen der Länge  $N_{\text{FFT}} = 1024$  berechnet werden.



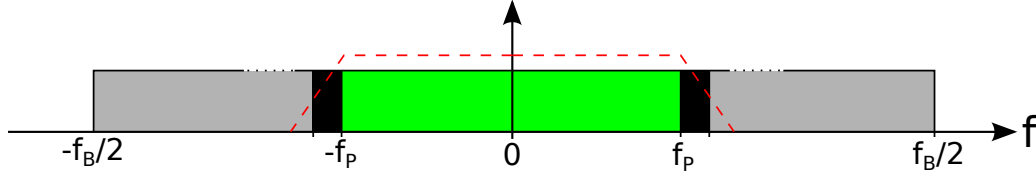
**Abbildung 3.4:** KKF der PSS-Sequenz ( $N_{\text{ID}}^2 = 0$ ) mit simulierten Empfangsdaten im Zeitbereich über 3 Frames

### Methode 3: Multiratensignalverarbeitung

Um die Rechenintensivität der KKF im Zeitbereich zu reduzieren, wird das empfangene Signal abwärtsgetastet. Vorher muss jedoch zur Vermeidung von Aliasing eine Tiefpassfil-

terung stattfinden. Nach einer groben und weniger rechenaufwändigen Suche im abwärtsgetasteten Signal wird um die grob ermittelte Position eine genau Suche bei der vollen Samplerate durchgeführt.

Wie bereits in Abschnitt 3.2 erwähnt, werden dem PSS unabhängig von der Systembandbreite 62 Unterträger um den DC-Carrier herum zugeordnet, jeweils 5 weitere Unterträger am Rand bleiben ungenutzt (siehe Abbildung 3.5).



**Abbildung 3.5:** PSS-Struktur im Frequenzbereich (grün: PSS, schwarz: ungenutzt, grau: Nutzdaten, rot: Tiefpassfilter)

Das PSS befindet sich somit in einem Frequenzbereich von  $[-f_P, f_P]$  mit  $f_P = 472.5 \text{ kHz}$ . Daher sollte ein Filter zur Bandbegrenzung eine Grenzfrequenz von  $f_P$  besitzen und sehr steilflankig sein, um innerhalb der ungenutzten 5 Unterträger eine ausreichende Dämpfung im Stopband zu erreichen. Nach der Tiefpassfilterung werden die empfangenen Samples auf eine sehr moderate Samplerate von  $0.96 \text{ MS/s}$  abwärtsgetastet, sodass die Samplerate größer als die Bandbreite des PSS mit ( $B_P = 2 \cdot f_P = 0.945 \text{ MHz} > 0.96 \text{ MHz}$ ) ist, um das Abtasttheorem nicht zu verletzen.  $0.96 \text{ MS/s}$  wird gewählt, da es durch ganzzahlige Teilung der in LTE verwendeten Samplerates zu erreichen ist. So wird z.B. bei empfangenen Samples mit einer Rate von  $15.36 \text{ MS/s}$ , welche bei einem LTE-Signal mit einer FFT der Länge 1024 verwendet wird, die geforderten  $0.96 \text{ MS/s}$  durch einen Teilungsfaktor von  $d = 16$  erreicht. Die effektive Länge eines OFDM-Symbols sinkt damit von  $N_{\text{FFT}} = 1024$  auf  $N_{\text{FFT},l} = \frac{N_{\text{FFT}}}{d} = 64$ . Weiterhin sinkt auch die Zahl der zu überprüfenden Zeithypothesen (die Länge eines halben Frames) um  $d$  von  $N_{\text{Hyp}} = 76800$  auf  $N_{\text{Hyp},l} = \frac{N_{\text{Hyp}}}{d} = 4800$ . Der Rechenaufwand für die KKF

$$K_l(n) = \sum_{k=0}^{N_r-1} \left| \sum_{m=0}^{N_{\text{FFT},l}-1} r_{k,l}(m+n) \cdot p_l^*(m) \right| \quad (3.10)$$

sinkt also um den Faktor  $d^2 = 256$ .  $r_{k,l}(m)$  ist dabei das abwärtsgetastete Empfangssignal an Antenne  $k$  und  $p_l(m)$  die PSS-Vergleichssequenz im Zeitbereich der Länge  $N_{\text{FFT},l}$ . Mittels

$$n_{P,l} = \arg \max_n K_l(n) \quad (3.11)$$

wird auf die Position des PSS im abwärtsgetasteten Signal geschlossen. Die grobe Position des PSS im Signal mit der vollen Rate ergibt sich schließlich durch  $n_P = n_{P,l} \cdot d$ . Da  $n_P$  nur in einem Raster  $d$  aufgelöst wird, muss um die grobe Position herum eine Korrelation bei der vollen Rate stattfinden. Diese Korrelation findet aber nur in einem schmalen Bereich  $n_P \pm d/2$  statt, daher ist der Rechenaufwand hierfür niedrig.

Die oben genannten Anforderungen an das Filter kann zum Beispiel mit einem elliptischen IIR-Filter der Ordnung 6 erreicht werden. Dieses hat aber einen nichtlinearen Phasenverlauf, wodurch das Ergebnis der KKF beeinträchtigt wird. Ein vergleichbares FIR-Filter mit

linearem Phasenverlauf benötigt für denselben Amplitudenverlauf wie das IIR-Filter jedoch mehrere hundert Taps. Es reicht aber aus, ein FIR-Filter mit flacherem Amplitudenverlauf zu wählen und somit in geringem Maße Aliasing zuzulassen, um den Rechenaufwand zu reduzieren. Die Korrelation liefert dann immer noch gute Ergebnisse. Weiterhin muss bei einem FIR-Filter mit gleichzeitiger Abwärtstastung im Gegensatz zum IIR-Filter nur für jedes  $d$ -te Sample eine Berechnung mit den Filterkoeffizienten stattfinden. Die verwendeten Filterparameter werden in Abschnitt 4.3.1 beschrieben.

#### Methode 4: differentielle Korrelation des PSS im Zeitbereich

Wie bereits erwähnt bewirkt ein Frequenzoffset eine starke Beeinträchtigung der KKF nach Methode 2, wobei einige Kilohertz ab etwa einem Unterträgerabstand  $f_A = 15 \text{ kHz}$  ausreichen, um eine Synchronisation auf die Weise unmöglich zu machen. Die Methode nach [YZ13] wirkt dem entgegen.

Anstatt einer herkömmlichen KKF über die Länge ( $N_{\text{FFT}}$ ) eines OFDM-Symbols durchzuführen, wird das empfangene Signal und die PSS-Vergleichssequenz in 4 Teile aufgespalten und einzeln korreliert:

$$A_{k,b}(n) = \sum_{m=0}^{N_{\text{FFT}}/4} r_k(n + m + b \cdot \frac{N_{\text{FFT}}}{4}) \cdot p_k^*(m + b \cdot \frac{N_{\text{FFT}}}{4}) \quad b = 0 \dots 3 \quad (3.12)$$

Mittels

$$K(n) = \sum_{k=0}^{N_r-1} \left| \sum_{b=0}^3 A_{k,b}(n) \cdot A_{k,b+1}^*(n) \right| \quad (3.13)$$

werden die 4 Segmente zusammengeführt und ergeben einen Korrelationswert  $K(n)$  für die Zeithypothese an der Stelle  $n$ . Die normale KKF wird also durch dieses Verfahren ersetzt, wobei der Rechenaufwand um die in Gleichung (3.13) verwendete Operationen nur wenig steigt. In Abbildung 3.4(c) und 3.4(d) ist das Ergebnis der differentiellen Korrelation mit simulierten Daten dargestellt.

Ein Frequenzoffset bewirkt allgemein eine kontinuierliche Phasendrehung der Empfangssamples. Durch das Aufspalten der Korrelation in 4 kleinere Segmente sinkt der Einfluss des Frequenzoffsets auf jedes dieser Segmente. Durch die in Gleichung (3.13) verwendete Multiplikation mit dem konjugiert Komplexen fällt die durch den Frequenzoffset verursachte Phasendrehung von  $A_{k,b}(n)$  wieder heraus.

#### Vergleich

Für die Implementierung der Zeitsynchronisation sollte ein zuverlässiges Verfahren gewählt werden, welches einen nicht zu hohen Rechenaufwand besitzt und gegenüber einem Frequenzoffset robust ist. Eine Frequenzsynchronisation ist erst nach erfolgter Zeitsynchronisation möglich, da erst dann die zeitliche Position der zur Schätzung nötigen Signale bekannt ist.

Zum Vergleich des Rechenaufwands der initialen Zeitsynchronisation kann vereinfacht die Zahl der notwendigen komplexen Multiplikationen und Additionen herangezogen werden.

Methode	Berechnung	gesamt
1	CP: $72 \cdot 153.600/20 = 552.960$	3.601.040
	FFT: $140 \cdot 2^{10} \cdot 10 = 1.433.600$	
	Korr.: $140 \cdot 3 \cdot (62 \cdot 62) = 1.614.480$	
2	Korr.: $1.024 \cdot 153.600 = 157.286.400$	157.286.400
3	Filter: $48 \cdot 153.600/16 = 460.800$	1.107.968
	Korr. 1: $4.800 \cdot 2 \cdot 64 = 614.400$	
	Korr. 2: $32 \cdot 1.024 = 32.768$	
4	wie 3	1.107.968

**Tabelle 3.1:** Zahl notwendiger komplexer Multiplikationen und Additionen

In Tabelle 3.1 ist dies über eine Synchronisationslänge von einem Frames mit  $N_{\text{FFT}} = 1024$  und einem Empfangskanal dargestellt. Ein Frame besteht aus  $N_F = 153600$  Samples und  $N_{\text{F,OFDM}} = 140$  OFDM-Symbolen. Die Zahl an komplexen Multiplikationen und Additionen für die N-Punkte-FFT wurde als  $2^N \cdot N$  angenommen.

Beim bisher implementierten Verfahren (Methode 1) findet zunächst eine Symbolsynchronisation mit dem CP statt. Diese funktioniert unabhängig von einem Frequenzoffset. Da die PSS-Synchronisation im Frequenzbereich stattfindet, muss für jedes OFDM-Symbol zuvor eine FFT durchgeführt werden.

Bei direkter KKF im Zeitbereich (Methode 2) ist keine FFT notwendig, jedoch ist die Länge der Korrelation und die Zahl der zu überprüfenden Zeithypothesen viel größer, da der Symboltakt unbekannt ist. Dies resultiert in einem vergleichbar viel höheren Rechenaufwand.

Zur Reduzierung dieses Rechenaufwands kann auf eine niedrigere Samplerate abwärtsgetastet werden (siehe Abschnitt Methode 3). Der Rechenaufwand für das Filter ist nicht besonders groß, da nur für die Samples bei niedriger Rate die Filteroperationen berechnet werden müssen. Das ist bei einem FIR-Filter möglich, weil sich hier die Samples aus vergangenen Eingangswerten berechnen. Bei niedrigerer Samplerate sinkt die Zahl der zu überprüfenden Zeithypothesen und die effektive Symbollänge in Samples im Vergleich zu Methode 2 stark. Somit sinkt auch der Rechenaufwand für die KKF um den gleichen Faktor. Im Vergleich zur bisherigen Implementierung wird keine FFT und keine CP-Synchronisation, aber die zusätzliche Filterung benötigt. Im Endeffekt resultiert für Methode 3 ein geringerer Rechenaufwand als für Methode 1.

Methode 4 ersetzt die normale Korrelation und benötigt dabei nicht mehr Rechenaufwand, liefert jedoch auch bei einem Frequenzoffset gute Ergebnisse.

Implementiert wird also eine Korrelation im Zeitbereich nach Methode 3 und 4.

### 3.5 Frequenzsynchronisation

Ein Frequenzoffset kommt durch nicht gleichlaufende Oszillatoren an Sender und Empfänger und den Dopplereffekt zustande. Mehrere Empfangskanäle haben unter Voraussetzung synchronisierter Oszillatoren auf der Empfängerseite den gleichen Frequenzoffset, welcher als Vielfaches des Unterträgerabstands  $f_A = 15 \text{ kHz}$  angegeben wird:

$$f_O = (n_I + \varepsilon_F) \cdot f_A \quad (3.14)$$

$n_I$  bezeichnet dabei den ganzzahligen Integer Frequency Offset (IFO) und  $\varepsilon_F$  den Fractional Frequency Offset (FFO) mit  $-1 < \varepsilon_F < 1$ . Der FFO führt dazu, dass die Unterträger nicht mehr orthogonal zueinander sind und bewirkt so Inter Carrier Interference (ICI). Ein IFO bewirkt kein ICI, führt aber zu einer Verschiebung der Unterträger. Eine Frequenzkorrektur findet erst nach der Zeitsynchronisation statt, da die zur Korrektur benutzten Signale zunächst zeitlich lokalisiert werden müssen. Zur Schätzung des FFOs werden nun 2 Methoden vorgestellt.

### Method 1: FFO Schätzung durch Cyclic Prefix

In Abschnitt 3.4 wird die Symbolsynchronisation mithilfe des CP vorgestellt. Die dort verwendete Korrelationsfunktion Gleichung (3.3) lässt sich auch zur Abschätzung des FFOs verwenden an Antenne  $k$  verwenden [Dem12, Kap. 2.3]:

$$\varepsilon_{F,k} = -\frac{1}{2\pi} \angle \gamma_k(n) \quad (3.15)$$

Bei Empfang mit mehreren Antennen ist es sinnvoll den FFO für jede Antenne einzeln zu berechnen und der Amplitude gemäß

$$\varepsilon_F = \left[ \sum_{k=0}^{N_r-1} \varepsilon_{F,k} \cdot |\gamma_k(n)| \right] / \left[ \sum_{k=0}^{N_r-1} |\gamma_k(n)| \right] \quad (3.16)$$

zu mitteln.

### Method 2: FFO Schätzung durch PSS

Nach [HSHT12] kann der FFO an Empfangsantenne  $k$  durch

$$\varepsilon_{F,k} = \frac{1}{\pi} \angle \left( \left[ \sum_{m=0}^{N_{\text{FFT}}/2-1} r_k(m) \cdot p^*(m) \right]^* \cdot \left[ \sum_{m=N_{\text{FFT}}/2}^{N_{\text{FFT}}} r_k(m) \cdot p^*(m) \right] \right) \quad (3.17)$$

geschätzt werden, wobei  $r_k^*(m)$  das empfangene PSS und  $p^*(m)$  die PSS-Vergleichssequenz im Zeitbereich ist. Bei exakter Gleichheit beider Sequenzen wären obige Summen rein reell. Ein FFO führt aber zu einer Phasendrehung zwischen der ersten und zweiten Hälfte der Korrelation, wodurch der FFO berechnet werden kann. Auch hier ist es sinnvoll, die FFO-Schätzung gemäß der Amplitude zu gewichten.

Eine Simulation hat gezeigt, dass die FFO-Schätzung mithilfe des PSS gegenüber des CP genauer ist. Im Gegensatz zum PSS sind im CP die störenden Mehrwegeausbreitungen des vorhergehenden OFDM-Symbols vorhanden und verschlechtern so die Schätzung. Daher wird zur Implementierung die Methode mit dem PSS gewählt.

## 4 Implementierung

Im Folgenden wird die Implementierung des MIMO-LTE-Empfängers in GNU Radio vorgestellt. Dabei wird hauptsächlich auf die Blöcke und Elemente eingegangen, welche im Rahmen dieser Arbeit neu erstellt bzw. verändert wurden.

Bei der Implementierung eines zweiten Empfangskanals wird festgestellt, dass die notwendigen Verarbeitungsschritte intern parallel für jeden Empfangskanal gleich ablaufen und sich leicht durch Schleifen realisieren lassen. Daher ist es möglich beliebig viele Empfangskanäle zu verwenden - die im folgenden Kapitel dargestellten Flowgraphs basieren jedoch alle auf einer Konfiguration mit 2 Empfangskanälen. Zunächst werden einige Grundlagen des verwendeten SDR Frameworks vorgestellt.

### 4.1 GNU Radio

GNU Radio ist ein quelloffenes und freies Software Toolkit zur Entwicklung von *Software Defined Radios* (SDR). Die gesamte Signalverarbeitung findet dabei innerhalb von GNU Radio in Software auf einem *General Purpose Processor* (GPP) statt, wobei Hardware zum Senden und Empfangen angebunden werden kann.

Ein Flowgraph definiert die Signalverarbeitungsschritte und ist in mehrere Blöcke unterteilt. Jeder Block entspricht dabei einem Knoten des Graphen und gibt die verarbeiteten Daten über Kanten an andere Blöcke weiter. GNU Radio besitzt eine Vielzahl verschiedener Blöcke für grundlegenden Verarbeitungsschritte wie Filter, einfache Rechenoperationen, Dekodierung oder Hardwarezugriff. Es ist weiterhin möglich eigene Blöcke zu schreiben und in GNU Radio zu integrieren. Durch den modularen Aufbau mit Blöcken ist das Erstellen oder Ändern einer Signalverarbeitungskette leicht möglich. Die Flowgraphs werden über eine graphische Oberfläche, dem GNU Radio Companion, erstellt und ausgeführt.

Blöcke werden über Ports verbunden. Ein Eingangsport ist dabei immer mit einem Ausgangsport verbunden. Blöcke können eine verschiedene Anzahl von Ports besitzen, ein Block ohne Ausgangsports wird Senke, ein Block ohne Eingangsports Quelle genannt. Am Anfang jedes Flowgraphs befindet sich eine Quelle und am Ende eine Senke, welche auch den Zugriff auf Hardware oder die Dateiebene ermöglichen. Mit hierarchischen Blöcken können mehrere elementare Blöcke zur Übersichtlichkeit zusammengefasst werden. Die Blöcke sind normalerweise in der Programmiersprache C++ geschrieben, möglich ist jedoch auch die Verwendung von Python.

Zwischen den Blöcken fließen sogenannte Items auf den Kanten in Streams. Diese Items können verschiedene Datentypen wie reelle Werte, komplexe Werte, Integer Werte und Vektoren davon annehmen. Das Scheduling und die Weitergabe der Items wird dabei von GNU Radio selbst übernommen.

GNU Radio arbeitet nicht mit einer festen Samplerate im Sinne von Samples pro Sekunde, sondern mit relativen Samplerrates zwischen Blöcken. Befindet sich im Flowgraph kein Hardware- oder Throttle-Block, welche nur eine gewisse Zahl von Samples pro Sekunde generieren bzw. verarbeiten, wird GNU Radio versuchen Samples schnellstmöglich zu verarbeiten und den Prozessor komplett auslasten [GNU14].

## Stream Tags

Ein Stream Tag dient zum Austausch von Metadaten zwischen Blöcken. Tags werden innerhalb eines Blockes generiert und fest mit einem Sample verbunden. Nachfolgende Blöcke können die Tags lesen und festlegen, ob sie weiterpropagiert werden sollen. Ein Tag besteht aus 4 Feldern: *value*, *key*, *offset* und *srcid*. Der zu übertragene Wert *value* kann mithilfe von PMTs (Polymorphic Types) beliebige Datentypen annehmen, wobei der Datentyp durch das Feld *key* bestimmt wird. *offset* gibt die absolute Position des Tags im Stream an, das optionale Feld *srcid* den Block, welcher den Tag generiert hat [GNU14].

## Messages

Messages dienen dem asynchronen Datenaustausch zwischen Blöcken und werden im Flowgraph über Ports verbunden. Messages sind im Gegensatz zu den Stream Tags nicht an die Daten gebunden und es ist möglich Informationen an vorhergehende Blöcke zu übergeben, wodurch eine Art Rückkopplung realisiert werden kann. Messages basieren auf *Polymorphic Types* und können so jeden beliebigen Datentyp annehmen [GNU14].

## VOLK

*Vector-Optimized Library of Kernels* (VOLK) bietet Funktionen für Vektoroperationen mit einem großen Geschwindigkeitsvorteil gegenüber manuell programmierten Funktionen. Dazu nutzt VOLK mehrere spezielle prozessorspezifische Instruktionen (SIMD). Der implementierte Empfänger nutzt VOLK an verschiedenen Stellen zur Geschwindigkeitssteigerung [GNU14].

## 4.2 Übersicht

Der LTE-Flowgraph besteht aus vielen elementaren Blöcken, welche der Übersichtlichkeit wegen in mehrere hierarchischen Blöcken zusammengefasst werden. In Abbildung 4.1 ist dieser Flowgraph für 2 Empfangsantennen dargestellt.

Zunächst wird mit den Eingangsdaten eine Zeit- und Frequenzsynchronisation durchgeführt. Dazu gehören die hierarchischen Blöcke *LTE MIMO PSS sync*, *LTE MIMO PSS based freq sync* und *LTE MIMO SSS Sync*. Mit dem PSS wird eine initiale Zellsuche durchgeführt, der Halbframetak und die Cell-ID Number  $N_{ID}^2$  ermittelt. Die Position des Halbframe-taktes wird mittels Tags an nachfolgende Blöcke weitergegeben, indem auf den Sample zu Beginn jedes Halbframes ein Tag gesetzt wird. Das SSS dient der Synchronisation auf den



Frametakt. Dies geschieht im Frequenzbereich und befindet sich daher hinter dem OFDM-Block.

Zu Beginn werden die komplexen Samples jeder Empfangsantenne über separate Verbindungen übertragen. GNU Radio speichert die Samples verschiedener Verbindungen in verschiedenen Puffern, was zunächst günstig ist, da die PSS-Synchronisation die Samples verschiedener Empfangsantennen getrennt verarbeitet. Es wäre auch möglich Samples verschiedener Empfangskanäle in Vektoren über eine Verbindung zu übertragen - dann aber müssten intern in jedem Block zunächst Kopieroperationen zur Trennung stattfinden.

Als Quelle kann im Flowgraph je nach Anforderung ein Hardwareblock oder eine Dateiquelle dienen. Unerlässlich ist die Synchronizität in Zeit- und Frequenzoffset der Samples verschiedener Empfangskanäle, denn nur dann kann eine gemeinsame Synchronisation stattfinden. Über den Parameter *FFT length* wird die Länge eines OFDM-Symbols in Samples angegeben, über den sich auch die CP-Länge berechnen lässt. Am Eingang muss daher die zur FFT-Länge gehörende Samplerate anliegen - dies kann auch durch einen Resampling-Block erreicht werden.

Der Block *LTE MIMO OFDM RX* überführt das Empfangssignal in den Frequenzbereich. Dazu wird das Cyclic Prefix entfernt, eine FFT durchgeführt und anschließend die Unterträger extrahiert. Für jeden Empfangsstream geschieht dies zunächst separat, am Ausgang werden jedoch Vektoren mit den Unterträgern ausgegeben. Ein Vektor beinhaltet dabei die OFDM-Symbole aller Empfangsantennen zu einem Zeitpunkt.

Die lineare Kanalschätzung in *LTE MIMO estimator* ermittelt für die Unterträger die Kanalkoeffizienten, wobei jedes Sende- und Empfangsantennenpaar berücksichtigt werden muss. Vor der Dekodierung des MIB ist die Antennenkonfiguration auf der Sendeseite unbekannt, daher muss zunächst für alle möglichen Sendeantennen eine Kanalschätzung durchgeführt werden. Im Flowgraph 4.1 wird in 2 separaten Vektoren die Schätzung für 2 Sendeantennen weitergegeben - jeder Vektor beinhaltet dabei die Schätzung aller Empfangsantennen zur jeweiligen Sendeantenne. Bei einer  $2 \times 2$  Konfiguration z.B. werden dann 4 Kanäle geschätzt.

Im Block *LTE MIMO Decode PBCH* werden die vom PBCH genutzten Unterträger extrahiert und durch die geschätzten Kanalkoeffizienten korrigiert. Hier werden auch die Streams von verschiedenen Empfangsantennen zusammengeführt. In *Decode BCH* und *Unpack MIB* wird der PBCH dekodiert und der MIB ausgegeben. Diese Blöcke konnten ohne Änderung von der bisherigen Implementierung übernommen werden.

Im Vergleich zur bisherigen Implementierung wurde die komplette Zeitsynchronisation geändert und alle Blöcke, die für den Empfang mehrerer Kanäle geändert werden mussten, wurden dementsprechend erweitert. Weiterhin wurde die Dekodierung für den Übertragungsmodus mit 4 Antennenports implementiert, sodass z.B. ein  $4 \times 2$  System denkbar ist.

## 4.3 Synchronisation

Im Folgenden werden die verschiedenen Synchronisationsschritte beschrieben. Die PSS-Synchronisation, basierend auf dem *Primary Synchronization Signal*, synchronisiert auf den

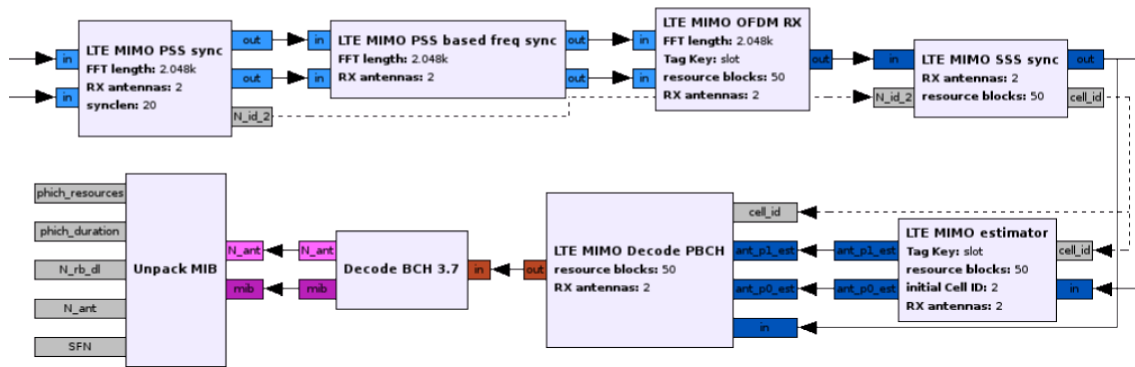


Abbildung 4.1: Flowgraph mit hierarischen Blöcken

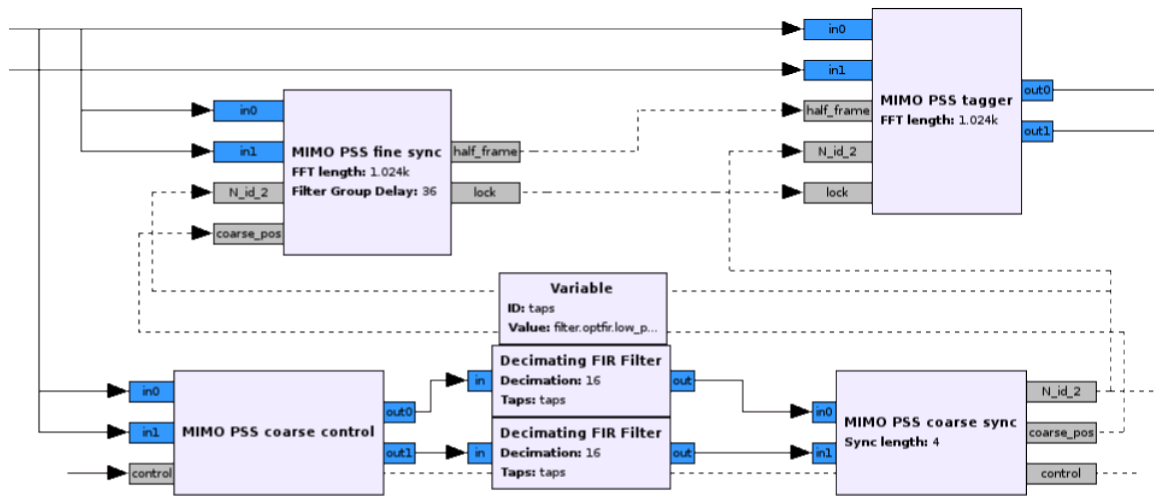


Abbildung 4.2: Flowgraph der PSS-Zeitsynchronisation

Halbframetak. Anschließend findet eine FFO Korrektor statt, welche auch auf dem PSS basiert. Der Aufbau des PSS und Synchronisationsmethoden wurden bereits in Abschnitt 3 eingeführt. Im Frequenzbereich wird mit dem *Secondary Synchronisation Signal* zuletzt einmalig auf den Frametak synchronisiert.

### 4.3.1 PSS-Zeitsynchronisation

Um den Rechenaufwand der Korrelationsberechnungen zu reduzieren ist die PSS-Synchronisation in 2 Schritte unterteilt und arbeitet bei verschiedenen Samplerraten. Zuerst werden die Eingangssignale wie in Abschnitt 3.4 abwärtsgetastet und der grobe Offset des PSS ermittelt. Für den genauen Offset wird bei der vollen Rate eine feine Korrelation um den groben Offset herum in einem kleineren Suchbereich durchgeführt. Alle Korrelationen finden im Zeitbereich statt. In Abbildung 4.2 ist der innere Aufbau des PSS-Synchronisations-Blocks dargestellt.

Im ersten Schritt wird, um Aliasing zu vermeiden bzw. zu reduzieren, das Eingangssignal vor dem Abwärtsasten mit einem Tiefpass gefiltert. Dies geschieht für jeden Empfangsstream separat durch den von GNU Radio nativ bereitgestellten Block *Decimating FIR Filter*. Die Taps des FIR-Filters werden mit dem Equiripple-Verfahren mit den in Tabelle 4.1 vorgegeben Parameter berechnet. Die genaue Wahl der Parameter ist jedoch unkritisch - auch weniger restriktive Parameter führen zum Ziel.

Parameter	Wert
Verstärkung	1
Grenzfrequenz $f_{pass}$	472.5 kHz
Stopfrequenz $f_{stop}$	960 kHz
Passband Ripple	1.0 dB
Stopband Dämpfung	40 dB
Taps ( $N_{FFT} = 1024$ )	48

**Tabelle 4.1:** Tiefpassfilter Parameter

Die Eingangssignale mit dem einen Faktor  $d = N_{FFT}/64$  auf eine Abtastrate  $f_A = 0.96$  MS/s abwärtsgetastet. Ein halber Frame besitzt dann immer die Länge von  $n_{Hyp,l} = 4800$  Samples - dies entspricht der Zahl der zu überprüfenden Zeithypothesen. Im Block *MIMO PSS coarse sync* wird mittels Korrelation

$$n_{p,l} = \arg \max_n K(n) \quad 0 < n < n_{Hyp,l} - 1 \quad (4.1)$$

auf den groben Offset des PSS geschlossen, wobei  $K(n)$  nach Gleichung 3.6 berechnet wird. Der übergebene Parameter *Sync length* gibt an, über wieviele Halbframes gemittelt wird. Für zuverlässigere Ergebnisse kann dieser erhöht werden, jedoch steigt dann auch der Rechenaufwand und die Synchronisationsdauer.

Danach wird die Cell-ID Number  $N_{ID}^2$  berechnet, indem am ermittelten Offset mit den 3 unterschiedlichen PSS-Sequenzen korreliert wird. Der größte Korrelationswert bestimmt dabei die jeweilige Cell-ID Number.

Um Rechenkapazität zu sparen wird nach erfolgter Berechnung dem Block *MIMO PSS coarse control* über eine Message mitgeteilt, keine Samples mehr an das FIR-Filter weiterzugeben. Die initiale Zellsuche findet nur einmal zu Beginn der Programmlaufzeit statt und wird deshalb nach erfolgter Synchronisation abgeschaltet. Der untere Teil des in Abbildung 4.2 dargestellten Flowgraphs hat dann keine Funktion mehr.

Die errechneten Werte  $n_{p,l}$  und  $N_{ID}^2$  werden über Messages an den Block *MIMO PSS fine sync* weitergegeben, welcher bis dahin inaktiv ist. Der grobe Offset bei der vollen Rate ergibt sich durch

$$n'_p = n_{p,l} \cdot d - g \quad (4.2)$$

wobei  $g$  die Gruppenlaufzeit des FIR-Filters in Samples bezeichnet. Die Korrektur mit  $g$  ist notwendig, da das FIR-Filter das Signal um  $g$  verzögert. Bei einem FIR-Filter ergibt sich  $g = N_{TAB}/2$  direkt als die Hälfte der Anzahl der Tabs. Diesen Wert erhält der Block als Parameter. Die feine Korrelation findet im Bereich  $n'_p \pm d/2$  statt und liefert eine sehr genaue Zeitposition.

Da in GNU Radio allen Blöcken die Anzahl ihrer bearbeiteten Samples bekannt ist, beschreibt der Wert

$$n_{P,mod} = n_P \bmod N_{HF} \quad (4.3)$$

den Halbframetakts vollständig.  $N_{HF}$  ist die Länge eines halben Frames.  $n_{P,mod}$  gibt den Abstand zwischen dem ersten Sample und der ersten auftretenden PSS-Sequenz an. Alle weiteren PSS-Sequenzen befinden sich dann in einem Abstand der halben Framelänge.

Nach der initialen Berechnung des Offsets geht der Block *MIMO PSS fine sync* in einen Tracking Modus über, welcher den vermuteten Offset kontinuierlich überprüft und gegebenenfalls korrigiert. Dies ist aufgrund nicht perfekter Oszillatoren am Empfänger nötig, wodurch die geforderte Abtastrate in sehr geringem Maße von der tatsächlichen abweicht. Es werden über einem gewissen Zeitraum also zuviel bzw. zu wenig Samples produziert und der PSS-Offset dadurch langsam verschoben. Bei jeder auftretenden PSS-Sequenz wird dabei um den vermuteten Offset herum 3 Korrelationen berechnet, also im Bereich  $n_{PSS} \pm 1$ . Ergibt sich für benachbarte Stellen ein höherer Korrelationswert, wird der Offset in diese Richtung verschoben.

Sobald die initiale Zeitsynchronisation abgeschlossen ist, erhält der Block *MIMO PSS tagger* durch Messages kontinuierlich den aktuellen Offset. Zum Start jedes Slots wird ein Tag mit der Slotnummer in einem Halbframe und auf den Start jedes Halbframes ein Tag mit der  $N_{ID}^2$  gesetzt, da der nachfolgende Block zur Frequenzkorrektur diesen benötigt. Es werden keine Daten verändert, sondern nur Metadaten hinzugefügt, mit denen nachfolgende Blöcke so auf der Framestruktur arbeiten können.

### 4.3.2 PSS-Frequenzsynchronisation

Das PSS wird wie in Abschnitt 3.5 beschrieben zur Schätzung des FFO genutzt. Eine Frequenzsynchronisation ist erst nach erfolgter Zeitsynchronisation möglich, da erst dann der Offset des PSS bekannt ist. In Abbildung 4.3 ist der Aufbau des hierarchischen Blocks *LTE MIMO PSS based freq sync* dargestellt.

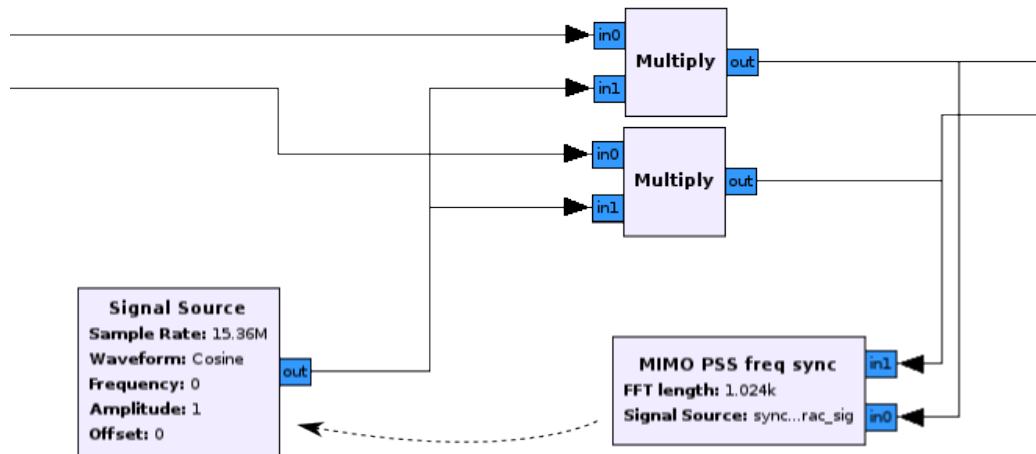


Abbildung 4.3: Innere Struktur der PSS-Frequenzsynchronisation

Zur Kompensation des Frequenzoffsets  $f_O$  werden die Empfangsströme aller Antennen mit einer komplexen Schwingung der Frequenz  $-\bar{f}_O$  und Amplitude 1 multipliziert. Der Block *MIMO PSS freq sync* buffert zunächst eine vollständige PSS-Sequenz und führt damit eine Schätzung  $\hat{f}_O$  des Offsets durch. Dabei werden auch vorhergehende Schätzungen berücksichtigt, indem die FFO-Schätzung

$$\bar{f}_{O,neu} = \bar{f}_{O,alt} + \alpha \cdot \hat{f}_O \quad (4.4)$$

kontinuierlich angepasst wird. Die Berechnung von  $\hat{f}_O$  befindet sich dabei hinter den Multiplikationsblöcken und erfasst immer nur die verbleibende Differenz zwischen  $f_O$  und  $\bar{f}_O$ . Eine Schrittweite wird durch  $\alpha$  festgelegt, welche zu Beginn höher ausfällt und mit der Zahl der Schätzungen abnimmt. Durch diese adaptive Anpassung soll sich die Schätzung schnell den korrekten Wert erreichen und dann mit nur kleinem Gradientenrauschen um den wahren Frequenzoffset springen.

Der Block *MIMO PSS freq sync* erhält eine Referenz auf die *Signal Source* und kann so  $\bar{f}_O$  ständig anpassen. Durch die gestrichelte Linie wird diese Rückkopplung angedeutet.

### 4.3.3 SSS-Synchronisation

Der letzte Synchronisationsschritt erfolgt im Frequenzbereich, synchronisiert auf den Frametakt und ermittelt die Cell-ID  $N_{ID}$ . In Abbildung 4.4 ist der Aufbau dargestellt. Die Eingangsdaten sind Vektoren, welche als Elemente die Schätzungen für die Symbole aller Unterträger eines OFDM-Symbols beinhalten. Der Block *MIMO SSS symbol selector* extrahiert aus diesen Vektoren die vom SSS genutzten Unterträger, wobei der zeitliche Offset durch Stream Tags auf den Vektoren bekannt ist.

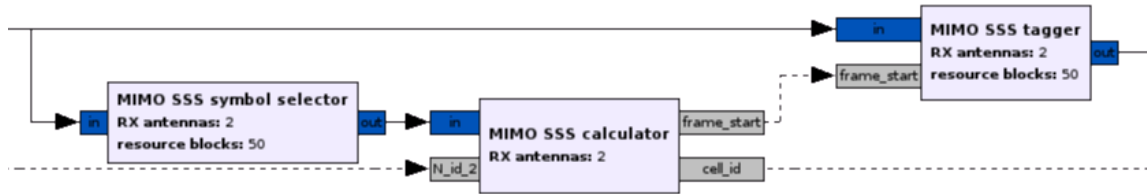


Abbildung 4.4: Innere Struktur der SSS-Framesynchronisation

Im Block *MIMO SSS calculator* wird wie in [Dem12, Kap. 3.3.4] die  $N_{ID}$  und der Frametakt ermittelt. Der zugrundeliegende Algorithmus wurde dabei im Vergleich zur vorhergehenden Implementierung nicht wesentlich geändert, sondern nur für mehrere Empfangsantennen erweitert, indem die verwendeten Korrelationen nun mehrere Empfangsströme kohärent verarbeiten. Daher wird hier nicht näher auf die einzelnen Berechnungsschritte eingegangen.

Weiterhin wurden bisher die OFDM-Symbole mit dem SSS im Zeitbereich extrahiert und durch eine gesonderte FFT in den Frequenzbereich überführt. Da das komplette Eingangssignal jedoch auch ohne SSS-Synchronisation in den Frequenzbereich überführt werden kann, wurde diese hinter die OFDM-Operationen verschoben. Somit kann auf die zusätzliche FFT-Operation verzichtet und damit Rechenoperationen eingespart werden.

Die  $N_{ID}$  wird über einen Messageport für andere Blöcke veröffentlicht. Diese wird an mehreren Stellen gebraucht, um z.B die Position der Referenz Signale zu bestimmen. Der Block *MIMO SSS tagger* erhält den Frameoffset durch eine Message und setzt auf jeden Vektor mit dem ersten OFDM-Symbol jedes Slots ein Tag mit der jeweiligen Slotnummer. Damit ist das System vollständig synchronisiert.

## 4.4 OFDM - Operationen

In Abschnitt 2.1.2 wurde bereits allgemein die OFDM-Empfangsstruktur vorgestellt. Abbildung 4.5 zeigt die Implementierung des hierarchischen Blocks *LTE MIMO OFDM RX*. Bevor mittels einer FFT die Empfangsdaten in den Frequenzbereich überführt werden, wird das CP entfernt. Anschließend werden die genutzten Unterträger extrahiert und als Vektoren weitergegeben.

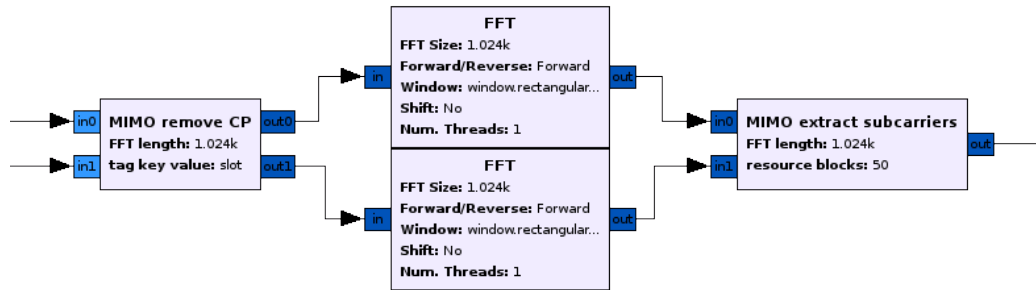


Abbildung 4.5: Innere Struktur der OFDM-Operationen

Im Block *MIMO remove CP* wird bei jedem Empfangsstream synchron das CP entfernt. Dies kann erst nach erfolgter Zeitsynchronisation geschehen, da erst dann die OFDM-Symbole zeitlich lokalisierbar sind. Alle Samples, die den Block erreichen, bevor die initiale Zeitsynchronisation beendet ist, werden verworfen. Erst wenn die Tags, welche den Framestart kennzeichnen, den Block erreichen, werden die Samples eines OFDM-Symbol zu Vektoren zusammengefasst und an die nachfolgende FFT weitergegeben. Um die Vektorlänge und die Länge des CP zu bestimmen, wird daher als Parameter die FFT-Länge  $N_{FFT}$  benötigt. Vektoren, welche das erste OFDM-Symbol eines Slots beinhalten, erhalten ein Tag mit der Nummer dieses Symbols im aktuellen Halbframe. Nachfolgende Blöcke erhalten so Zugriff auf die Framestruktur.

Die Vektoren mit den extrahierten Samples im Zeitbereich werden für jeden Stream durch einen FFT-Block in den Frequenzbereich überführt. Diese sind bereits in GNU Radio enthalten und können ohne Änderung verwendet werden. Wichtiger Parameter ist dabei die FFT-Länge  $N_{FFT}$ , mit der auch die Länge der CPs berechnet wurde.

Zuletzt werden im Block *MIMO extract subcarriers* die genutzten Unterträger extrahiert und in die richtige Reihenfolge gebracht. LTE-Systeme verwenden immer eine kleinere Zahl an Unterträger als die minimal nutzbare Länge FFT-Länge, daher werden einige Elemente der Eingangsvektoren verworfen. Beachtet werden muss auch, dass der DC-Carrier nicht genutzt

wird. Als Parameter erhält der Block die Anzahl der angenommenen Resource Blocks  $N_{RB}^{DL}$  und die Zahl der Empfangsantennen  $N_r$ , um die Größe des Ausgangsvektors  $12 \cdot N_{RB}^{DL} \cdot N_r$  zu bestimmen. Die OFDM-Symbole verschiedener Empfangskanäle und gleichen Zeitpunkts werden hier in einem Vektor zusammengefasst und hintereinander übertragen. Nachfolgende Blöcke erhalten so synchron Zugriff auf alle Empfangsdaten zeitgleich gesendeter OFDM-Symbole.

## 4.5 Kanalschätzung

Im hierarchischen Block *LTE MIMO estimator*, dessen innerer Aufbau in Abbildung 4.6 dargestellt ist, findet die Kanalschätzung statt, welche zur Entzerrung der Unterträger benötigt wird.

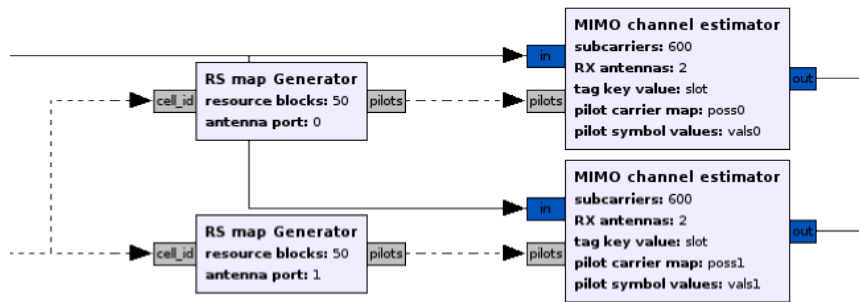


Abbildung 4.6: Aufbau der Kanalschätzung

Für jedes Sende- und Empfangsantennenpaar wird eine eigene Kanalschätzung durchgeführt. Der zugrundeliegende Algorithmus war dabei schon vorhanden und wurde nur für den Mehrkanalempfang erweitert. Zur Schätzung werden die in der Zeitfrequenzebene eingebetteten Pilotsymbole verwendet. Zunächst wird dabei die Differenz in Amplitude und Phase zwischen den empfangenen Symbolen und den Piloten berechnet. Dann findet in Frequenzrichtung zwischen REs mit Referenzsymbolen und in Zeitrichtung zwischen OFDM-Symbolen mit Referenzsignalen eine lineare Interpolation statt [Dem12, Kap. 3.4.3].

Der Block *RS map Generator* berechnet die Position und die Werte der Pilotsymbole für einen Antennenport. Diese sind abhängig von der Cell-ID und können erst zur Laufzeit nach erfolgter SSS-Synchronisation berechnet werden.

Für jeden Antennenport auf Sendeseite gibt es einen Block *MIMO channel estimator*, welcher die Schätzung für alle Empfangsantennen von diesem Antennenport berechnet. Als Eingangsvektor erhält der Block dazu die Schätzungen der Symbole aller Unterträger eines OFDM-Symbols und die Pilotsymbole. Um die Größe des Eingangsvektors zu bestimmen, wird dem Block als Parameter die Anzahl der Unterträger  $12 \cdot N_{RB}^{DL}$  und die Zahl der Empfangsantennen  $N_r$  übergeben. Die Ausgangsdaten beinhalten die geschätzten Kanalkoeffizienten für jedes RE und können zur Entzerrung genutzt werden.

Zu Beginn ist die Zahl der vom Sender genutzten Antennenports nicht bekannt und kann erst nach Dekodierung des MIB bestimmt werden. Wie in Abschnitt 2.2.1 beschrieben, gibt es bis zu 4 Antennenports. In Abbildung 4.6 ist nur die Schätzung für Port 0 und 1 berücksichtigt, da die meisten Basisstation bisher eine Konfiguration mit 2 Antennenports verwenden.

## 4.6 PBCH-Dekodierung

Im Folgenden wird der Aufbau des Flowgraphs zur PBCH-Dekodierung beschrieben, welcher in Abbildung 4.7 dargestellt ist. Der grundlegende Aufbau des PBCH wurde bereits in Abschnitt 2.1.5 vorgestellt. Die PBCH-Dekodierung beginnt, sobald das System synchronisiert ist und der Kanal geschätzt wurde.

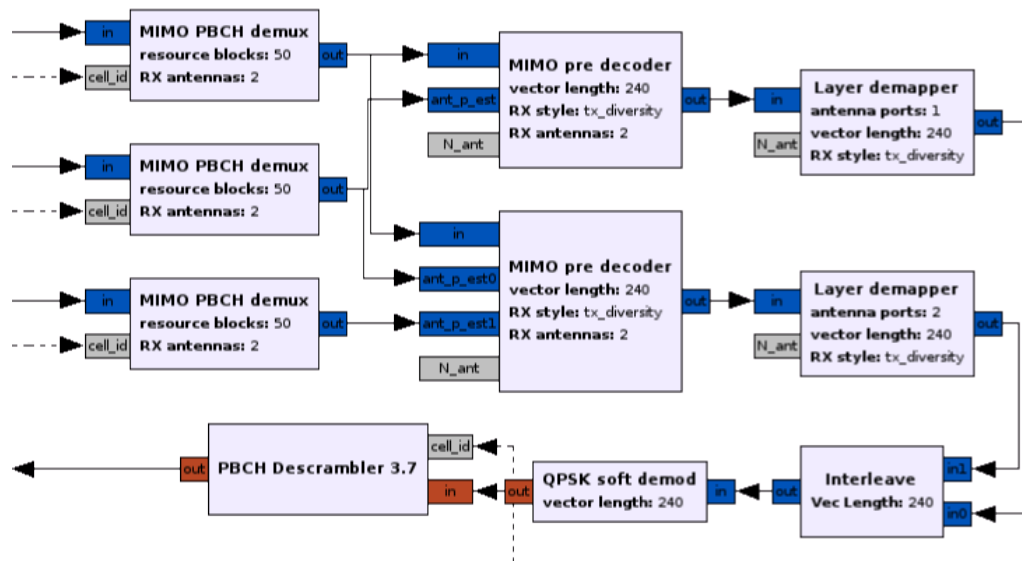


Abbildung 4.7: Aufbau der PBCH-Dekodierung

Zunächst werden die vom PBCH genutzten Unterträger und deren Kanalkoeffizienten aus den empfangenen Unterträgern und den berechneten Kanalkoeffizienten extrahiert. Dies geschieht in den Blöcken *MIMO PBCH demux*. Um die Position der Pilotsymbolen, welche sich zwischen den vom PBCH genutzten Unterträgern befinden, wird als Parameter die Cell-ID benötigt. In jedem Frame werden vom PBCH 240 QPSK-modulierte Unterträger genutzt, die Größe der Ausgangsvektoren ergibt sich daher zu  $240 \cdot N_r$ . Auch hier werden die Daten jeder Empfangsantenne hintereinander im Vektor abgelegt. Die ersten 240 Elemente des Vektors gehören also zur ersten Empfangsantenne. Obwohl der vollständige PBCH über 4 Frames verteilt ist, erhalten die Daten in einem Frame genug Information zur Dekodierung des *Master Information Blocks*. Daher werden immer nur  $1/4$  der PBCH-Daten weitergegeben.

Der Block *MIMO pre decoder* führt die in Abschnitt 2.2.2 vorgestellten Berechnungen zur Entzerrung durch. Der PBCH nutzt immer den Transmit Diversity Modus für eine zuver-



lässigere Übertragung - der Block erhält dazu den Parameter (*tx\_diversity*). Da zu Beginn die Antennenkonfiguration auf Sendeseite unbekannt ist, werden zunächst 2 unterschiedlich parametrisierte Blöcke verwendet. Einer zur Dekodierung der Konfiguration mit einem Antennenport, der nur die Kanalschätzung für diesen Port erhält. Für die Konfiguration mit 2 Antennenports benötigt der andere Block zusätzlich die Kanalkoeffizienten für den zweiten Antennenport. Die Dekodierung mit 4 Antennenports ist in Abbildung 4.7 der Übersichtlichkeit wegen nicht dargestellt, ist jedoch mit Verwendung eines weiteren Decoders möglich. Erst in diesem Block werden die Daten von mehreren Empfangsantennen kohärent addiert. Die Größe des Ausgangsvektors (240) mit den entzerrten komplexen Symbolen ist hier daher unabhängig von der Zahl der Empfangsantennen.

Alle weiteren Blöcke wurden ohne Änderung von der bisherigen Implementierung übernommen. Im Block *Layer demapper* werden die verschiedenen Layer in einen Datenfluss zusammengeführt. Der *Interleave*-Block verschachtelt die Vektoren der unterschiedlichen Antennenkonfigurationen, womit alle Empfangsvektoren in einem einzelnen Datenfluss weiter bearbeitet werden können [Dem12, Kap. 3.5].



## 5 Messung und Simulation

In diesem Kapitel werden Mess- und Dekodierungsergebnisse von aufgenommenen Signalen mit einem *Universal Software Radio Peripheral* (USRP) vorgestellt.

### 5.1 Simulation

Die in Abschnitt 4.3 vorgestellten Algorithmen zur Synchronisation werden zunächst in MATLAB getestet und dann in GNU Radio implementiert. Damit diese Tests möglichst realitätsnah sind, werden Sende- und Empfangsdaten zur Simulation mit der MATLAB-LTE-Toolbox generiert. Diese Toolbox bietet dabei auch die Funktion einen Mehrwege-MIMO-Fading-Kanal dem Standard nach zu simulieren [3GP13, Ch. B.2]. Dazu sind verschiedene Kanalprofile definiert. Die zur Simulation verwendeten Empfangsdaten wurden mit dem *Extended Typical Urban Model* (ETU) erstellt, welches die in Tabelle 5.1 dargestellten Mehrwegepfade besitzt. Weiterhin wurden den Daten weißes Rauschen hinzugefügt, sodass sich ein Signal-zu-Rauschverhältnis von 6 dB ergibt.

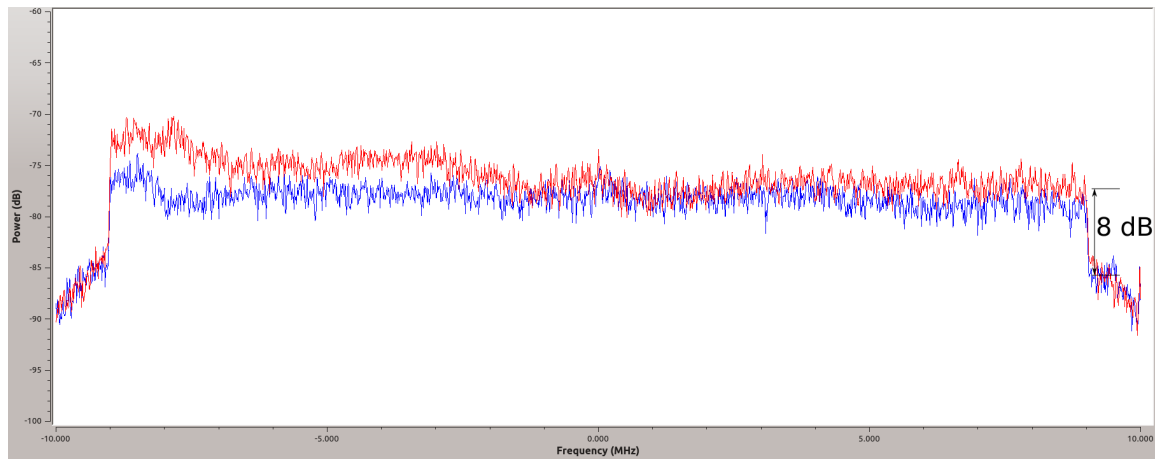
tap delay (ns)	0	50	120	200	230	500	1600	2300	5000
rel. power (db)	-1	-1	-1	0	0	0	-3	-5	-7

**Tabelle 5.1:** Extended Typical Urban Model - Mehrwegeausbreitung

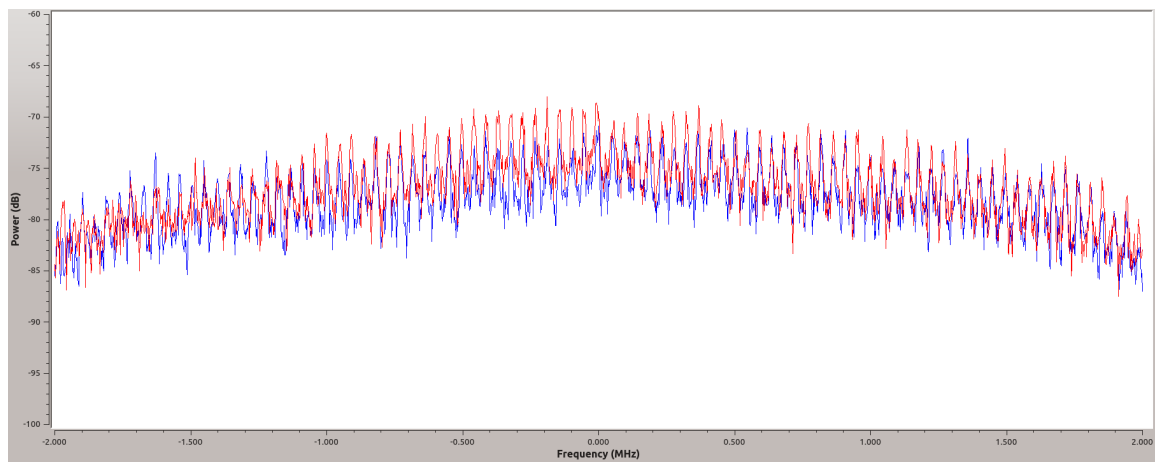
### 5.2 Messaufbau

LTE wird in Deutschland in verschiedenen Funkbändern betrieben. Im Folgenden werden die Ergebnisse von 2 Messungen im 1800 MHz-Frequenzband (E-UTRA Band 3, 1805-1880 MHz) von 2 verschiedenen Downlink-Kanälen vorgestellt. Die Parameter der Messungen sind in Tabelle 5.2 beschrieben und fanden im Funklabor des CEL bei gutem Wetter statt. Als Empfangshardware diente das USRP B210, welches zwei in Zeit und Frequenz synchronisierte Empfangskanäle besitzt. Die Empfangsantennen wurden ca. 1 m voneinander entfernt platziert, um unabhängiges Fading zu gewährleisten.

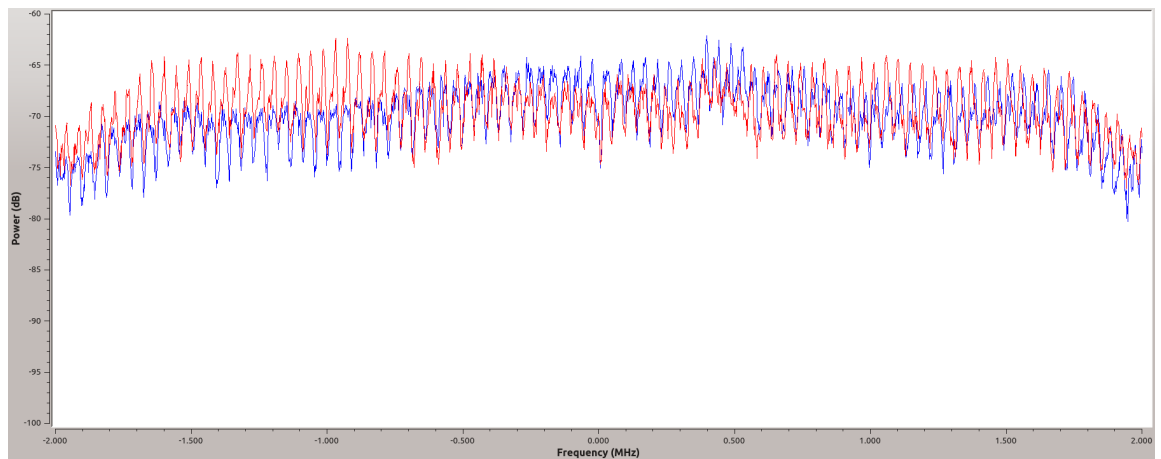
Zunächst wurden die Rohdaten in einer Datei aufgezeichnet, damit sie später dekodiert werden können. Eine Echtzeit-Dekodierung ist zu rechenintensiv - für 10 s LTE-Daten werden auf der verfügbaren Hardware ca. 35 s an Rechenzeit benötigt. Da zur Dekodierung des PBCH und zur Synchronisation nicht die volle Bandbreite des LTE-Downlinks benötigt wird, wird nur ein Bereich von 4 MHz um den DC-Carrier aufgenommen.



(a) Gesamtspektrum Messung 1



(b) Messung 1



(c) Messung 2

**Abbildung 5.1:** Spektren der Messungen (rot: Antenne 1, blau: Antenne 2)

USRP	B210
Abtastrate	4 MS/s
Anzahl Antennen $N_r$	2
Frequenz	1815 MHz u. 1870.5 MHz
Ort	Funklabor CEL
Anbieter	Telekom u. Eplus
Datum	13.08.2014

**Tabelle 5.2:** Messaufbau

In Abbildung 5.1(a) ist das Gesamtspektrum des in Messung 1 aufgenommenen Downlink-Signals dargestellt. Erkennbar ist die typische OFDM-Signalförm innerhalb einer Bandbreite von 20 MHz. Für das Signal-zu-Rauschverhältnis kann der 8 dB Abstand vom Grundrauschen zum Signal herangezogen werden. Bei Messung 2 ergibt sich so ein Signal-zu-Rauschverhältnis von ca. 12 dB. Erkennbar ist weiterhin die frequenzabhängige unterschiedliche Empfangsleistung beider Empfangskanäle.

Die Spektren der aufgenommenen Signale sind in Abbildung 5.1(a) und 5.1(b) dargestellt. Das Signal von Messung 2 hat eine ca. 5 dB größere Leistung als das Signal von Messung 1.

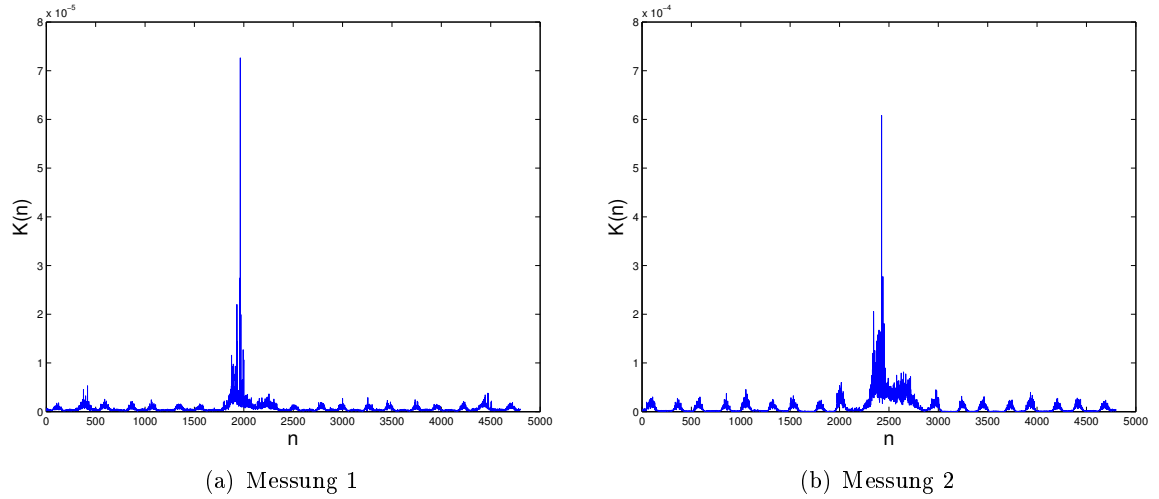
## 5.3 Messergebnisse und Vergleich mit Simulation

### 5.3.1 Synchronisation

Das System kann beide Messungen zu Beginn problemlos synchronisieren. In Abbildung 5.2 sind dazu die Korrelationen mit dem PSS im runtergetasteten Bereich über 2 Frames gemittelt dargestellt. Die Peaks spiegeln dabei die Position des PSS wider und sind wie in der Simulation eindeutig. Die feine Korrelation bei der vollen Rate liefert eine sehr genau Zeitsynchronisation. Dass im Bereich außerhalb der Peaks nur sehr geringe Korrelationswerte auftreten, ist entweder auf eine niedrige Zellennutzung zum Messzeitpunkt zurückzuführen oder die Synchronisationssymbole erhalten gegenüber den Nutzdaten mehr Sendeleistung. Weiterhin ist eine kontinuierliche Zeitsynchronisation wichtig, da der Abtasttakt des USRP nicht ganz genau ist. Durch das implementierte Tracking verschiebt sich der Zeitoffset bei beiden Messungen ca. alle 20-30ms um 1 Sample. Ohne Tracking wäre der Empfänger daher schon kurze Zeit nach der initialen Zeitsynchronisation wieder asynchron. Bei den Simulationsdaten gibt es diesen Effekt nicht.

Eine zu frühe Synchronisation um nur wenige Samples wäre weniger kritisch, da sich dann das OFDM-Symbol ins CP verschiebt und nur eine konstante Phasendrehung bewirkt. Eine zu späte Synchronisation dagegen führt zu ISI. Das manuelle Verschieben der Synchronisation um nur wenige Samples verhindert dabei das erfolgreiche Dekodieren.

Die Frequenzsynchronisation ist gegenüber Fehlern toleranter, da auch mit einem Restoffset von wenigen hundert Hertz eine Dekodierung möglich ist. Durch die zeitlich oft auftretenden Pilotsymbole und die Kanalschätzung wird dieser Restoffset herausgerechnet. Bei beiden Messungen ergab sich ein Frequenzoffset von  $f_O \approx 5.2 \text{ kHz}$  - das entspricht einer Frequenzabweichung des Empfängers von ca. 2.9 ppm.



**Abbildung 5.2:** Korrelation gemittelt über 2 Frames

### 5.3.2 Dekodierung

Nach der Entzerrung ergeben sich die in Abbildung 5.3 dargestellten Konstellationsdiagramme. Während bei Messung 1 das zu erwartende QPSK-Muster eindeutig zu erkennen ist, sind die Messwerte bei Messung 2 stark verzerrt. Außer der Mittenfrequenz wurde jedoch am Messaufbau zwischen beiden Messungen nichts geändert und die Synchronisation scheint auch bei Messung 2 gute Ergebnisse zu liefern. Weiterhin besitzt das Signal von Messung 2 eine höheres Signal-zu-Rauschverhältnis. Möglicherweise wird der Empfang durch eine andere Basisstation oder durch starke Mehrwegeausbreitung gestört.

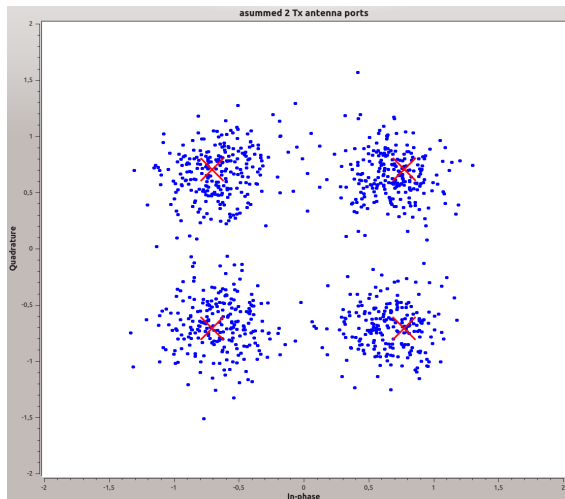
Trotzdem ist bei beiden Messungen eine Dekodierung des PBCH und des MIB zuverlässig möglich. Messung 2 profitiert dabei sicherlich durch die sehr niedrige Coderate von  $1/48$ . In Tabelle 5.3 sind die gemessenen Zellinformationen aufgelistet.

	Messung 1	Messung 2
$N_{ID}$	215	486
$N_t$	2	2
$N_{RB}^{DL}$	100	50
PHICH Duration	Normal	Normal
PHICH Resources	1	1/6

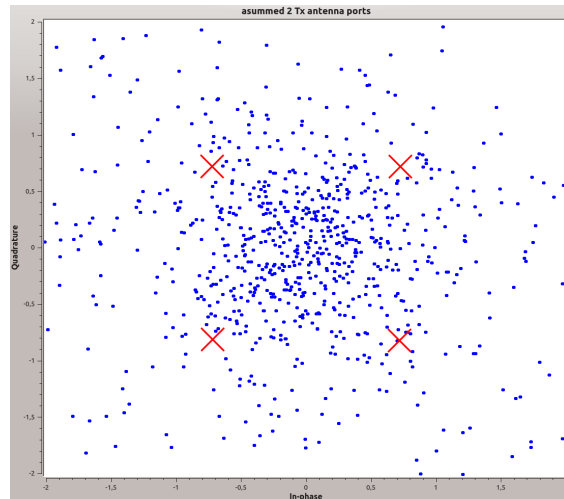
**Tabelle 5.3:** Dekodierungsergebnisse MIB der Zellen

Bei Messung 1 wird in nahezu jedem Frame der MIB erfolgreich ausgelesen. Dies funktioniert auch, wenn beide Empfangskanäle nach dem gleichen Schema einzeln dekodiert werden - so als wäre nur eine Empfangsantenne vorhanden. Im Flowgraph kann dafür die Zahl der Empfangsantennen auf 1 gestellt und die Daten eines Empfangskanals verworfen werden.

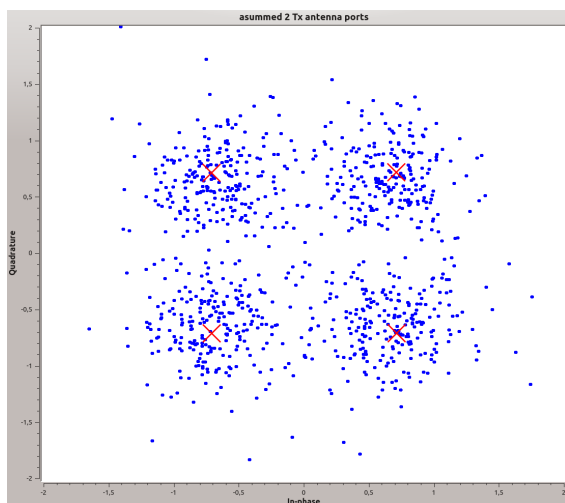
Die genauere Betrachtung von Messung 2 ist hierbei interessanter. In Tabelle 5.4 sind die Dekodierungsraten des MIB bei verschiedener Verwendung der Empfangskanäle aufgelistet.



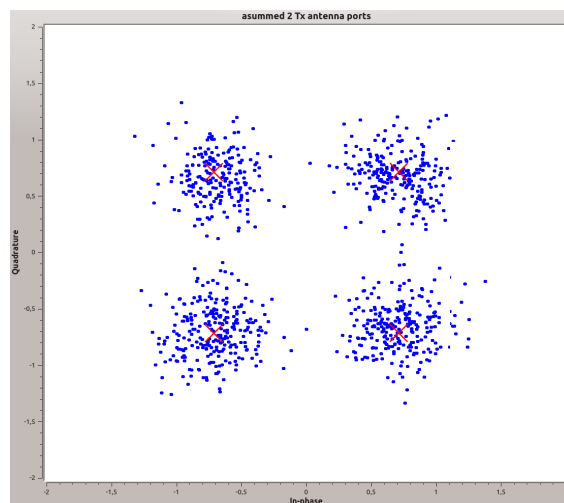
(a) Messung 1



(b) Messung 2



(c) simulierte Werte, ETU-Kanalmodell,  $\text{SNR} = 6\text{dB}$



(d) simulierte Werte, konstanter Kanal,  $\text{SNR} = 6\text{dB}$

**Abbildung 5.3:** Konstellationsdiagramme PBCH nach Entzerrung (rot: ideale Punkte)

Ob eine Dekodierung erfolgreich war, ist an dem Abstand der im MIB gespeicherten *System Frame Number* erkennbar. Die Verwendung mehrerer Empfangsantennen verbessert den Empfang durch die nach Abschnitt 2.2.2 implementierte Receive Diversity merklich. Eine

nur Antenne 1	nur Antenne 2	Antenne 1 u. 2
67,8 %	72,8%	96,1%

**Tabelle 5.4:** Decodingrate des MIB für Messung 2

gekürzte Textausgabe des Flowgraphs in Abbildung 5.4 zeigt die Synchronisationsschritte und die gelungenen Decodierungen des MIB. *diff* gibt dabei den Abstand der *System Frame Number* zweier folgender decodierter MIBs an. 1 heißt keinen Verlust. *half frame start* beschreibt den Offset des ersten Samples mit Daten eines Halbframes nach erfolgter Synchronisation und *mod start* den Offset des ersten Samples mit Daten eines Halbframes.

```
mimo_pss_coarse_sync:new coarse timing max found: val=66,609009 pos=1962 syncloop=4
Nid2 correlation values: id0:3,807259 id1:0,730321 id2:30,802231

mimo_pss_coarse_sync:found N_id_2=2
mimo_pss_coarse_sync:coarse pss-pos=1962
fine timing is locked to mod pss_pos:31352
now tracking

mimo_remove_cp
half frame_start = 408696
mod start      = 24696

mimo_sss_calculator publish_cell_id 215

INFO: MIB changed! (N_ant=2 N_rb_dl=100 PHICH: dur=0 res=1.00)
INFO: SFN = 363 diff = 1 (N_ant=2 N_rb_dl=100 PHICH: dur=0 res=1.00)
INFO: SFN = 364 diff = 1 (N_ant=2 N_rb_dl=100 PHICH: dur=0 res=1.00)
INFO: SFN = 365 diff = 1 (N_ant=2 N_rb_dl=100 PHICH: dur=0 res=1.00)
...
INFO: SFN = 421 diff = 1 (N_ant=2 N_rb_dl=100 PHICH: dur=0 res=1.00)
INFO: SFN = 422 diff = 1 (N_ant=2 N_rb_dl=100 PHICH: dur=0 res=1.00)
...
```

**Abbildung 5.4:** Textausgabe Messung 1



## 6 Zusammenfassung

In dieser Arbeit wurde ein bestehender LTE-Empfänger für den Empfang mit mehreren Antennen erweitert und ein neues Synchronisationsverfahren eingeführt. Viele Elemente konnten dabei direkt bzw. leicht verändert übernommen werden.

LTE nutzt OFDM im Downlink - ein Verfahren, bei dem die Sendesymbole auf mehrere Unterträger aufgeteilt werden. Die Zahl der Unterträger ist je nach Systembandbreite unterschiedlich. Das Downlink-Signal verwendet eine Framestruktur und ist zeitlich in Frames der Dauer 10 ms unterteilt, welche wiederum in Subframes und Slots eingeteilt werden. Der zu dekodierende Kanal ist der PBCH, in welchem den *Master Information Block* mit grundlegenden Zellkonfigurationsdaten übertragen wird. Im Allgemeinen nutzt LTE zur Übertragung des PBCH den Alamouti-Code. Zur Kanalschätzung werden Pilotsymbole an definierten Stellen in die Zeitfrequenzebene eingebettet. Das *Primary Synchronisation Signal* (PSS) wird zweimal pro Frame übertragen und dient zur Ermittlung der Cell-ID und des Halbframetaktes.

Es gibt grundsätzlich mehrere Möglichkeiten zur Synchronisation. Die neu implementierte Variante zur Zeitsynchronisation nutzt allein das PSS durch Korrelation im Zeitbereich. Da eine Korrelation bei der vollen Rate einen hohen Rechenaufwand besitzt, werden die Empfangssignale gefiltert und abwärtsgetastet. Es findet also eine Multirate-Signalverarbeitung statt. Im abwärtsgetasteten Bereich findet zunächst eine grobe Suche des PSS statt. Die so ermittelte Position wird dann auf das Signal mit der vollen Rate übertragen und in einem kleinen Suchfenster verfeinert. Im Vergleich zum Zeitsynchronisationsalgorithmus des bestehenden Empfängers ergibt sich so ein geringerer Rechenaufwand. Auch die Frequenzsynchronisation wird nun durch einen neuen Algorithmus mit dem PSS berechnet.

Durch den modularen Aufbau der Empfängerstruktur war es leicht möglich, einzelne Blöcke und Elemente zu ersetzen bzw. zu erweitern. Die bestehenden Synchronisationsblöcke wurden so größtenteils durch das neue Verfahren ersetzt und mehrkanalfähig implementiert. Weiterhin wurden viele Blöcke für die Verarbeitung mehrerer Empfangskanäle erweitert und in ihrer grundsätzlichen Funktion nicht geändert. Dies gilt für die OFDM-Operationen, die SSS-Synchronisation und die Kanalschätzung.

Die Funktionalität des Empfängers wurde mit Simulationsdaten, welche in MATLAB generiert wurden, getestet. Weiterhin wurden auch reale Signale aufgezeichnet und dekodiert. Dabei ergaben sich große Unterschiede bei verschiedenen Downlink-Signalen. Die Verwendung mehrerer Empfangsantennen brachte im Gegensatz zum Einantennenempfänger eine Verbesserung.



# Literaturverzeichnis

- [3GP12] 3GPP (3rd Generation Partnership Project). *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Physical channels and modulation*, Dezember 2012. TSS 36.211 version 11.5.0 Release 11.
- [3GP13] 3GPP (3rd Generation Partnership Project). *LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Base Station (BS) radio transmission and reception*, July 2013. TSS 36.104 version 11.5.0 Release 11.
- [Dem12] Johannes Demel. Empfang von LTE-Signalen in GNU Radio. Bachelor's thesis, Karlsruhe Institute of Technology (KIT), 2012.
- [DPS14] Erik Dahlman, Stefan Parkvall, and Johan Sköld. *4G: LTE/LTE-Advanced for Mobile Broadband*. Elsevier Ltd., 2nd edition, 2014.
- [GNU14] *GNU Radio Website*, 2014 (Zugriff am August 22, 2014). <http://gnuradio.org>.
- [GZAM10] Arunabha Ghosh, Jun Zhang, Jeffrey G. Andrews, and Rias Muhamed. *Fundamentals of LTE*. Prentice Hall Communications Engineering and Emerging Technologies Series. Prentice Hall, 1st edition, 2010.
- [HSHT12] Shoujun Huang, Yongtao Su, Ying He, and Shan Tang. Joint time and frequency offset estimation in LTE downlink. In *Communications and Networking in China (CHINACOM), 2012 7th International ICST Conference on*, pages 394–398, Aug 2012.
- [Jan04] Mohinder Jankiraman. *Space-Time codes and MIMO Systems*. Artech House Publisher, 1st edition, 2004.
- [STB11] Stefania Sesia, Issam Toufik, and Mathew Baker. *LTE - The UMTS Long Term Evolution*. John Wiley and Sons Ltd., 2nd edition, 2011.
- [YZ13] Yongzhi Yu and Qidan Zhu. A novel Time Synchronization for 3GPP LTE cell search. In *Communications and Networking in China (CHINACOM), 2013 8th International ICST Conference on*, pages 328–331, Aug 2013.