

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY

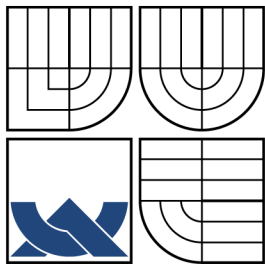
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

MODEL FYZICKÉ VRSTVY SYSTÉMU LTE

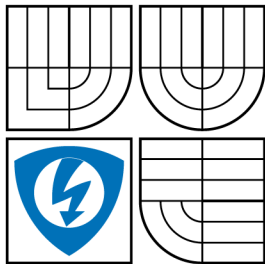
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. MILAN KOUNEK



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

MODEL FYZICKÉ VRSTVY SYSTÉMU LTE PHYSICAL LAYER MODEL OF LTE SYSTEM

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

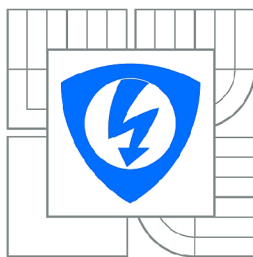
AUTOR PRÁCE
AUTHOR

Bc. MILAN KOUNEK

VEDOUCÍ PRÁCE
SUPERVISOR

prof. Ing. STANISLAV HANUS, CSc.

BRNO 2013



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav radioelektroniky

Diplomová práce

magisterský navazující studijní obor
Elektronika a sdělovací technika

Student: Bc. Milan Kounek

ID: 106551

Ročník: 2

Akademický rok: 2012/2013

NÁZEV TÉMATU:

Model fyzické vrstvy systému LTE

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte zpracování signálu na fyzické vrstvě systému LTE v uplinku i downlinku. Vytvořte blokové schéma pro zpracování signálu. V programovém prostředí MATLAB vytvořte matematický model fyzické vrstvy systému LTE pro přenos uživatelských signálů. Funkčnost modelu ověřte.

Pro vytvořený model uvažujte přenosové prostředí s nastavitelným poměrem C/I resp. S/N. Provedte simulace přenosu signálu a vyšetřete závislost chybovosti BER na poměru C/N resp. S/N.

DOPORUČENÁ LITERATURA:

[1] SESIA,, S., TOUFIK, I., BAKE, M. LTE - The UMTS Long Term Evolution. Chichester: J. Wiley & Sons, 2009.

[2] HANZO, L., AKHTMAN, J., WANG, L., JIANG, M. MIMO-OFDM for LTE, WiFi and WiMAX. Chichester: J. Wiley & Sons, 2011.

[3] Specifikace systému LTE. www.LTE.com

Termín zadání: 11.2.2013

Termín odevzdání: 24.5.2013

Vedoucí práce: prof. Ing. Stanislav Hanus, CSc.

Konzultanti diplomové práce:

prof. Dr. Ing. Zbyněk Raida

Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Diplomová práce se zabývá zpracováním signálu v systému LTE, konkrétně se zaměřuje na fyzickou vrstvu. První část práce detailně popisuje zpracování signálu na fyzické vrstvě systému LTE. Na základě získaných znalostí o zpracování signálu je vytvořeno blokové schéma, které názorně popisuje proces zpracování signálu. Další část práce se zabývá vytvořeným matematickým modelem, který je naprogramovaný v programu Matlab. V této části jsou popsány omezení modelu, možnosti nastavení simulace a zobrazení výsledků. V poslední části jsou zobrazeny výsledky simulace a proveden rozbor dosažených výsledků.

KLÍČOVÁ SLOVA

systém LTE, zpracování signálu, fyzická vrstva, Matlab

ABSTRACT

Master's thesis deals with the signal processing in LTE system, specifically focuses on the physical layer. The first part describes in detail the signal processing at the physical layer of the LTE system. Based on the acquired knowledge of signal processing is created a block diagram that graphically describes the process of the signal processing. Next part deals with the created mathematical model, which is programmed in Matlab. This section describes the limitations of the model, configuration options and view simulation results. The last part shows the results of simulations and analyzed the results.

KEYWORDS

LTE system, signal processing, physical layer, Matlab

KOUNEK, Milan *Model fyzické vrstvy systému LTE: diplomová práce*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2013. 141 s. Vedoucí práce byl prof. Ing. Stanislav Hanus, CSc.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Model fyzické vrstvy systému LTE“ jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu diplomové práce panu prof. Ing. Stanislavu Hanusovi, CSc. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

(podpis autora)

OBSAH

Úvod	14
1 Downlink	15
1.1 Transportní blok	17
1.2 Přidání CRC k Transportnímu bloku	18
1.3 Rozdělení na menší bloky a přidání CRC ke každému bloku	21
1.4 Kanálové kódování	24
1.5 Přizpůsobení datového toku	29
1.5.1 Prokládání subbloků	30
1.5.2 Sloučení subbloků	32
1.5.3 Přizpůsobení datového toku (Rate Matching)	32
1.6 Sloučení segmentů zpět do jednoho celku	33
1.7 Skramblování	33
1.8 Vnitřní modulace	36
1.9 Mapování do vrstev	38
1.10 Předkódování	40
1.11 Mapování elementů	41
1.11.1 Mapování elementů v LTE	42
1.11.2 Rádiový rámeček	42
1.11.3 Resource element	43
1.11.4 Šířka pásma v systému LTE	45
1.11.5 Referenční symboly	46
1.11.6 Synchronizační signály	49
1.12 OFDM modulace	55
1.12.1 Modulace	58
1.13 Rádiový kanál	58
1.13.1 AWGN kanál	59
1.13.2 Vícecestné šíření	59
1.14 OFDM demodulace	60
1.15 Vyrovnávání kanálu	61
1.16 Demapování elementů	61
1.17 Dekódování a demapování do vrstev	61
1.18 Vnitřní demodulace	61
1.19 Deskramblování	62
1.20 Rozdělení celku do segmentů	62
1.21 Přizpůsobení datového toku	63
1.22 Kanálové dekodování	63

1.23	Odebrání CRC a sloučení segmentů do jednoho bloku	64
1.24	Odebrání CRC	65
2	Uplink	66
2.1	Společné vlastnosti zpracování signálu s downlinkem	67
2.2	Multiplexování uživatelských a kontrolních informací	68
2.3	Kanálové prokládání	69
2.4	Skramblování	69
2.5	Mapování uživatelských dat	70
2.5.1	Referenční signály	71
2.6	SCFDMA modulace	74
2.7	SCFDMA demodulace	75
2.8	Demapování uživatelských dat	75
2.9	Deskramblování	76
3	Matematický model systému LTE	77
3.1	Požadavky modelu	77
3.2	Omezení modelu	77
3.3	Popis grafického rozhraní modelu a možnosti nastavení	78
3.3.1	Hlavní uživatelské rozhraní	79
3.3.2	Možnosti nastavení parametrů modelu	87
3.3.3	Možnosti nastavení výpisu informací o probíhajícím procesu	89
3.3.4	Zobrazení IQ diagramu	91
3.3.5	Zobrazení struktury rádiového rámce	91
3.3.6	Zobrazení závislosti chybovosti na poměru S/N (resp. C/I)	92
3.4	Blokový diagram	93
3.5	Popis blokového diagramu a vytvořených funkcí	97
3.6	Tabulka simulací	101
4	Výsledky simulací	102
4.1	Metoda testování	102
4.2	Zisk turbo dekodéru	103
4.3	Vliv použité modulace	104
4.4	Simulace poškození signálu v rádiovém prostředí	107
4.4.1	Downlink	107
4.4.2	Uplink	116
5	Závěr	120
	Literatura	121

Seznam symbolů, veličin a zkratk	123
Seznam příloh	127
A Důležité číselné hodnoty v LTE	128
A.1 Převodní tabulky pro vnitřní modulace	128
A.2 Parametry pro vnitřní prokladač turbo kodéru	131
A.3 Parametry m_0 a m_1 pro referenční symboly	133
A.4 Definice $\varphi(n)$ pro referenční symboly	134
A.5 Výpis úplného nastavení jednotlivých testů	136

SEZNAM OBRÁZKŮ

1.1	Blokové schéma zpracování signálu na fyzické vrstvě v systému LTE, downlink, zpracování uživatelských dat	15
1.2	Přidání kontrolního součtu k Transportnímu bloku	19
1.3	Rozdělení Transportního bloku na segmenty a přidání kontrolních součtů	22
1.4	Blokové schéma dělení transportního bloku na segmenty	23
1.5	Schéma konvolučního kodéru s poměrem 1/2	25
1.6	Blokové znázornění kanálového kódování	27
1.7	Vnitřní zapojení Turbo kodéru (převzato z [1])	28
1.8	Příklad funkce interního prokladače pro Turbo kodér	28
1.9	Blokové znázornění přizpůsobení datového toku	30
1.10	Vývojový diagram prokládání subbloků	31
1.11	Ukázka postupu sloučení subbloků	33
1.12	Příklad sloučení zpracovaných segmentů	33
1.13	Příklad generátoru m-sekvence	34
1.14	Příklad generátoru Gold sekvence	34
1.15	Generování PN posloupnosti pomocí posuvných registrů (převzato z [4])	36
1.16	IQ diagram modulací QPSK, 16QAM a 64QAM	39
1.17	Pozice referenčních symbolů v závislosti na použitém počtu antén pro vysílání, vlevo 2 antény, vpravo 4 antény (převzato z [4])	40
1.18	Rozlišení downlinku a uplinku ve frekvenci a v čase	41
1.19	Rádiový rámeček typ 1	42
1.20	Rádiový rámeček typ 2, Switch-point perioda 5ms	43
1.21	Časověfrekvenční struktura downlinku	44
1.22	Posun referenčních symbolů v závislosti na v_{shift} , jeden port antény, normální cyklický prefix	48
1.23	Mapování referenčních symbolů, normální cyklický prefix	48
1.24	Mapování referenčních symbolů, rozšířený cyklický prefix	49
1.25	Mapování synchronizačních signálů do struktury rádiového rámečku, normální cyklický prefix, struktura rámečku typ 1	51
1.26	Ukázka typů multiplexů	55
1.27	Rozdíl mezi FDM a OFDM	56
1.28	Ortogonální signály použité u modulace OFDM	57
1.29	Princip modulace OFDM	57
1.30	Princip přidání cyklického prefixu k OFDM symbolu	58
1.31	Princip konverze signálu do vysílacího pásma	59
1.32	Princip demodulace OFDM	60

1.33	Příklad vnitřní struktury turbo dekodéru	63
2.1	Blokové schéma zpracování signálu na fyzické vrstvě v systému LTE - Uplink	66
2.2	Časově frekvenční struktura uplinku	70
2.3	Ukázka mapování uživatelů, uplink	71
2.4	Princip modulace SC-FDMA	75
2.5	Princip demodulace SC-FDMA	75
3.1	Hlavní uživatelské rozhraní	79
3.2	Možnosti volby vstupních dat	80
3.3	Možnosti volby parametrů rádiového prostředí	81
3.4	Stav Rádiového rámce před vstupem do rádiového prostředí	84
3.5	Poškození rádiového rámce při použití modelu kanálu flat Rayleigh	84
3.6	Poškození rádiového rámce při použití modelu kanálu flat Rayleigh, detailní pohled	85
3.7	Impulsní odezva modelu rádiového kanálu PedA a PedB	85
3.8	Poškození rádiového rámce při použití modelu kanálu PedA	86
3.9	Poškození rádiového rámce při použití modelu kanálu PedA, detailní pohled	86
3.10	Poškození rádiového rámce při použití modelu kanálu PedB	87
3.11	Poškození rádiového rámce při použití modelu kanálu PedB, detailní pohled	87
3.12	Možnosti nastavení parametrů modelu systému LTE	88
3.13	Možnosti nastavení vypisování informací o postupu procesu modelu systému LTE	90
3.14	Příklad zobrazených IQ diagramů pomocí modelu systému LTE	91
3.15	Příklad zobrazení struktury rádiového rámce pomocí modelu systému LTE	92
3.16	Příklad zobrazení závislosti chybovosti na poměru S/N pomocí mo- delu systému LTE	93
3.17	Blokové schéma modelu systému LTE, část 1	94
3.18	Blokové schéma modelu systému LTE, část 2	95
3.19	Blokové schéma modelu systému LTE, část 3	96
3.20	Blokové schéma vytvoření vstupní posloupnosti	98
4.1	Závislost chybovosti na poměru S/N, při proměnném počtu iterací	103
4.2	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, Downlink	104
4.3	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, Uplink	105

4.4	Závislost chybovosti na poměru C/I, při rozdílné vnitřní modulaci, Downlink	105
4.5	Závislost chybovosti na poměru C/I, při rozdílné vnitřní modulaci, Uplink	106
4.6	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh	108
4.7	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh, vliv kanálu 75 %	108
4.8	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh, vliv kanálu 50 %	109
4.9	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh, vliv kanálu 25 %	109
4.10	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA	110
4.11	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, vliv kanálu 75 %	111
4.12	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, vliv kanálu 50 %	111
4.13	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, vliv kanálu 25 %	112
4.14	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB	113
4.15	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, vliv kanálu 75 %	114
4.16	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, vliv kanálu 50 %	114
4.17	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, vliv kanálu 25 %	115
4.18	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu Flat Rayleigh, channel seed 175	116
4.19	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu Flat Rayleigh, channel seed 9845	117
4.20	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, channel seed 175	117
4.21	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, channel seed 9845	118
4.22	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, channel seed 175	119

4.23	Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, channel seed 9845	119
------	--	-----

SEZNAM TABULEK

1.1	Mapování transportních kanálů na fyzické kanály. [1]	17
1.2	Mapování kontrolních kanálů na fyzické kanály. [1]	17
1.3	Použitý druh kódování pro transportní kanály [1]	26
1.4	Použitý druh kódování pro kontrolní kanály [1]	26
1.5	Opakující kód, kódování HI [1]	27
1.6	Blokový kód, kódování CFI [1]	27
1.7	Parametry pro interní prokladač subbloků	31
1.8	Převodní tabulka pro modulaci BPSK	37
1.9	Převodní tabulka pro modulaci QPSK	37
1.10	Rozložení subrámců pro uplink a downlink	43
1.11	Parametry resource bloku	44
1.12	Přehled pásem použitelných pro systém LTE	45
1.13	Přehled možných šířek pásma a korespondující počet RB	46
1.14	Kořenové indexy Zadoff-Chu sekvencí pro primární synchronizaci	50
1.15	Definované délky cyklického prefixu	58
2.1	Závislost pole cyklického posunu v DCI typ 0 na hodnotě $n_{DMRS}^{(2)}$	73
2.2	Závislost mezi <i>cyklickým posunem</i> a $n_{DMRS}^{(1)}$	74
A.1	Převodní tabulka pro modulaci 16QAM	128
A.2	Převodní tabulka pro modulaci 64QAM - 1. část	129
A.3	Převodní tabulka pro modulaci 64QAM - 2. část	130
A.4	Parametry pro interní prokladač turbodekodéru	131
A.5	Parametry pro interní prokladač turbodekodéru - pokračování	132
A.6	Závislost mezi $N_{ID}^{(1)}$ a proměnnými m_0 a m_1	133
A.7	Definice $\varphi(n)$ pro referenční symboly délky $M_{sc}^{RS} = N_{sc}^{RS}$	134
A.8	Definice $\varphi(n)$ pro referenční symboly délky $M_{sc}^{RS} = 2N_{sc}^{RS}$	135
A.9	Tabulka společných nastavení pro sadu simulací	136
A.10	Tabulka nastavení simulací, společné nastavení 1	137
A.11	Tabulka nastavení simulací, společné nastavení 2	138
A.12	Tabulka nastavení simulací, společné nastavení 3	139
A.13	Tabulka nastavení simulací, společné nastavení 4, část 1	140
A.14	Tabulka nastavení simulací, společné nastavení 4, část 2	141

ÚVOD

Cílem této práce je vytvořit zjednodušený matematický model systému LTE (Long Term Evolution). Tento model se skládá z většího počtu menších bloků, které tvoří ucelený systém. Pozornost je zde soustředěna na nejnižší vrstvu OSI/ISO modelu, fyzickou vrstvu. Vstupní informací modelu jsou libovolná data, která jsou předávána z vyšších vrstev. V praxi jsou to data z Transportních kanálů vrstvy MAC (Medium Access Control), která se stará o řízení přístupu na fyzické médium. Práce se zaměřuje na konkrétní nastavení, nepopisuje tedy veškeré možnosti systému LTE, které jsou velmi komplexní a proto je systém velmi univerzální. Modelována je varianta systému LTE FDD (Frequency Division Duplex), kdy je od sebe fyzicky oddělen uplink a downlink posunem ve frekvenci. Dále se tento model omezuje na Release 8, přesněji jsou bloky konstruovány podle standardu verze 8.8.0. Novější principy a postupy z Release 9 a 10 nejsou součástí této práce. Dalším zjednodušením modelu je zaměření pouze na uživatelská data, v modelu se tak nepřenáší signalizace a další signály, které v reálném systému být musí.

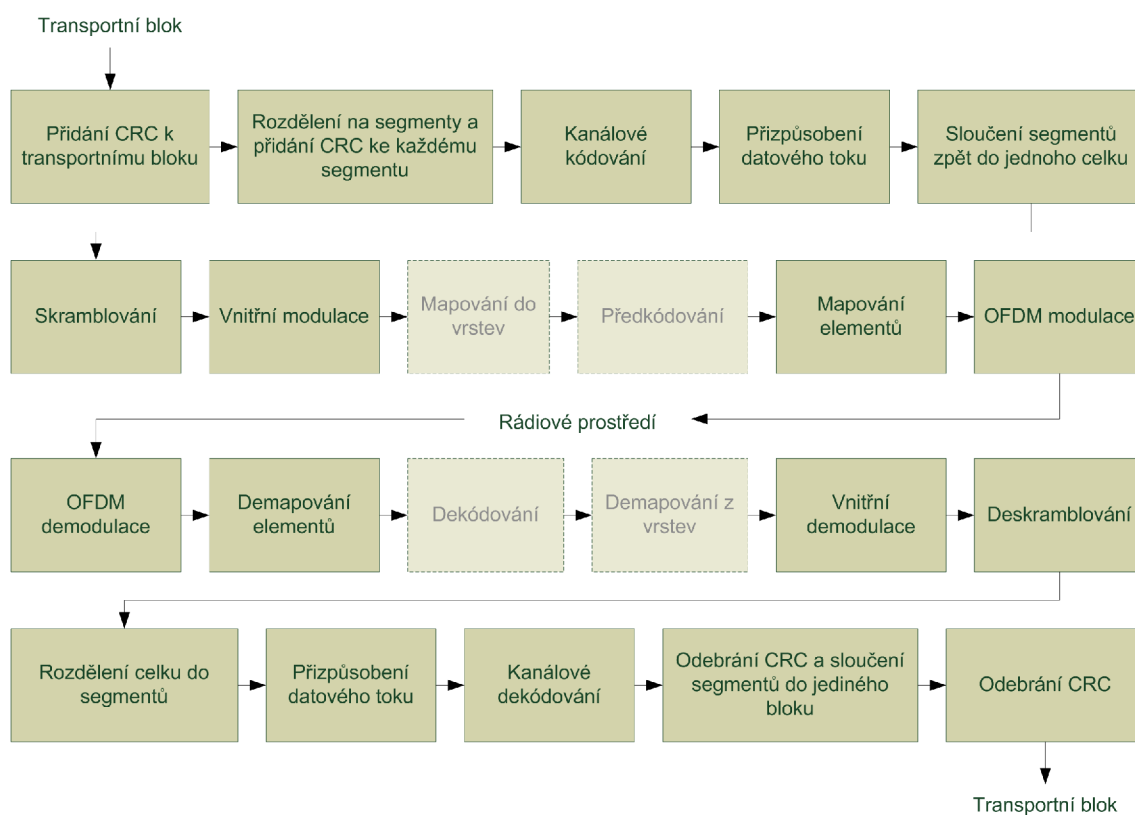
V první části je popisován systém LTE se zaměřením na fyzickou vrstvu. Je zde tedy popisován proces ochrany dat z transportního bloku, kódování dat, úprava datového toku a sloučení jednotlivých bloků zpět do celku postupujícího do dalšího zpracování, kterým je skramblování vnitřní modulace, mapování signálu do resource elementů, modulace a přidání cyklického prefixu. Následně je popisován proces příjmu, tedy operací odstranění cyklického prefixu a demodulace, demapování zakódované uživatelské informace, vnitřní demodulace, deskramblování, rozdělení toku do jednotlivých segmentů, přizpůsobení datového toku, kanálové dekódování a vyhodnocení správnosti přijatých dat. Tento popis je vytvořen pro downlink i uplink.

V další části práce je popisován vytvořený model v programu Matlab. Model se skládá z vysílače, modelu rádiového kanálu a přijímače, který přijímá signál a dekóduje z něj přenášené informace. Výsledkem modelu tak může být informace o chybovém poměru v závislosti na různém poměru S/N (resp. C/I), při použití různých typů modelů kanálu.

Poslední část práce obsahuje výsledky provedených simulací a jejich rozbor.

1 DOWNLINK

Tato kapitola se zabývá zpracováním signálu, který se šíří z eNodeB směrem k uživatelskému zařízení, tento směr je nazýván jako downlink. Pozornost je zaměřena na fyzickou vrstvu. Data, která jsou předávána z vyšší vrstvy, jsou zde zpracována, vyslána pomocí vysokofrekvenčního signálu, přijata, dekodována v přijímači a předána vyšším vrstvám v takovém stavu v jakém byla před vysláním nebo ve stavu nejvíce jim podobném. Jejich podobnost závisí na úrovni poškození signálu v rádiovém prostředí a na úrovni jejich zabezpečení, kterou určují parametry systému LTE. Všechny bloky zpracování signálu jsou zobrazeny na blokovém schématu, které je uvedeno na obrázku 1.1.



Obr. 1.1: Blokové schéma zpracování signálu na fyzické vrstvě v systému LTE, downlink, zpracování uživatelských dat

Vstupní informace, která je předávána modelu, se nazývá transportní blok, ten je představován skupinou bitů. Transportní blok je následně doplněn o kontrolní součet. O toto doplnění se stará blok Přidání CRC k transportnímu bloku. Z důvodu dalšího zpracování po blocích je transportní blok s připojeným kontrolním součtem dále zpracováván v bloku Rozdělení na segmenty a přidání CRC ke každému

segmentu. Tento blok má za úkol zpracovat vstupní data tak, aby výstupem byl segment o určité velikosti. Takto vytvořené segmenty jsou dále zpracovávány po jednom. Prvně je v bloku Kanálové kódování zabezpečena informace přidáním redundance. Následně je velikost bloku znovu upravována a to snížením velikosti přidané redundance. Tato úprava velikosti je zde z důvodu snížení objemu přenášených dat. Snížení redundance probíhá v bloku Přizpůsobení datového toku. Jednotlivě zpracované segmenty jsou nakonec sloučeny v jeden celek, toto sloučení obstarává blok Sloučení segmentů zpět do jednoho celku. Tento celek nazýváme kódové slovo. Následující operací je skramblování, které nijak neupravuje délku dat, ale upravuje data tak, aby při přenosu rádiovým prostředím byly potlačeny vzájemné interference mezi signály. V bloku Vnitřní modulace jsou jednotlivé skupiny bitů převedeny na symboly. Tímto se může dosáhnout výrazného snížení počtu přenášených vzorků, za cenu snížení jejich odolnosti vůči šumu a rušení. Bloky Mapování do vrstev a Předkódování jsou potřeba při přenosu dvou kódových slov současně nebo při použití více antén k přenosu. Jednotlivé symboly jsou zde mapovány do vrstev a připravovány k vysílání pomocí více antén. Blok Mapování elementů zajišťuje umístění jednotlivých symbolů do struktury rádiového rámce. O převod signálu na časové vzorky se stará blok OFDM modulace. Součástí bloku je přesun signálu do vysokofrekvenčního pásma.

Takto vytvořený signál je vysílán, kde v rádiovém prostředí dochází k jeho deformaci a zarušení.

Poté je přijímán přijímačem, který provede demodulaci signálu. V bloku OFDM demodulace je signál převeden do základního pásma a z časových vzorků jsou získány symboly. Blok Demapování elementů zajistí, aby byly ze struktury rádiového rámce vybrány pouze požadované symboly, například data uživatele. Bloky Dekódování a Demapování z vrstev zajišťují zpětnou konverzi symbolů získaných příjmem z více antén do toku symbolů. Tok může být jeden nebo dva, podle toho kolik kódových slov je zpracováváno současně. Po provedení vnitřní demodulace jsou ze symbolů získány bity. V bloku Deskramblování se přičte k bitové posloupnosti pseudonáhodná posloupnost a tím se získají platné bity, které představují kódové slovo. Toto kódové slovo je nejprve rozděleno do segmentů, které jsou dále zpracovávány po jednom. Nejprve se jednotlivé segmenty doplní o bity, které byly odstraněny při vysílání v bloku Přizpůsobení datového toku. Dále je v bloku Kanálové dekódování odstraněna redundance, která napomáhá při obnově informace, pokud nastala při přenosu chyba. Pokud byly při zakódování bloky rozděleny do více segmentů, nyní v bloku Odebrání CRC a sloučení segmentů do jediného bloku je vyhodnocen kontrolní součet každého segmentu a odebrány plnicí bity. Pokud transportní blok představuje pouze jediný zpracováváný blok, jsou odebrány pouze plnicí bity, pokud byly nějaké přidány. Posledním blokem je Odebrání CRC. Tento blok provede kontrolu kontrolního součtu a předá transportní blok vyšším vrstvám k jeho dalšímu zpracování.

1.1 Transportní blok

Transportní blok je kontejner obsahující informaci, která má být přenesena skrze fyzickou vrstvu. Data, která jsou předávána z vrstvy Řízení přístupu k fyzickému médiu, pomocí transportních kanálů, jsou označována jako transportní blok. Fyzická vrstva má jediný úkol, přenést tento transportní blok skrze rádiové prostředí a po dekódování ho předat vrstvě Řízení přístupu k fyzickému médiu v takovém stavu v jakém byl vyslán. Při přenosu skrze rádiové prostředí se signál deformuje a je zarušen jinými signály, které jsou skrze prostředí přenášeny. Proto je datová posloupnost na fyzické vrstvě zakódována a informace zabezpečena tak, aby při přenosu byly tyto vlivy co nejvíce potlačeny.

Transportní kanál

V systému LTE je rozlišováno několik transportních kanálů určených pro downlink, každý z těchto kanálů je mapován do fyzických kanálů. Přehled mapování kanálů je uveden v tabulce 1.1. Některé fyzické kanály dostávají informace z více transportních kanálů, příkladem je DL-SCH (Downlink Shared channel) a PCH (Paging channel), které jsou mapovány do PDSCH (Physical Downlink Shared channel). Mimo transportní kanály jsou ještě fyzickým kanálů předávány kontrolní informace z kontrolních kanálů. Přehled je uveden v tabulce 1.2. [1]

Tab. 1.1: Mapování transportních kanálů na fyzické kanály. [1]

Transportní kanály	Fyzické kanály
DL-SCH	PDSCH
BCH	PBCH
PCH	PDSCH
MCH	PMCH

Tab. 1.2: Mapování kontrolních kanálů na fyzické kanály. [1]

Kontrolní kanály	Fyzické kanály
CFI	PCFICH
HI	PHICH
DCI	PDCCH

V modelu je transportní blok představován obrázkem, náhodně generovanou posloupností dat nebo daty ze souboru. Obrázek je černobílý a každý pixel představuje jeden bit vstupní posloupnosti. K vytvoření vstupní posloupnosti, která představuje

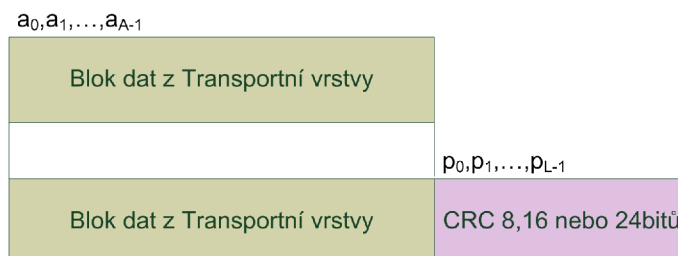
transportní blok, dochází v hlavním programu, tedy GUI.m. Dle vybraného způsobu vytvoření vstupní posloupnosti je tato posloupnost získána z testovacího obrázku nebo jsou data vygenerována a nebo načtena ze souboru. Získání vstupní posloupnosti z obrázku se provádí pomocí funkcí `imread`, která otevírá a čte soubor s obrázkem, dále funkce `rgb2gray` převádí barevný obrázek na obrázek v odstínech šedi. Posledními operacemi je zaokrouhlení hodnot na dvě hodnoty, tedy vytvoření černobílého obrázku a přeskládání dvojrozměrného obrázku do jednorozměrného pole pomocí funkce `reshape`. Generování dat se provádí pomocí funkce `randi([imin, imax], m, n)`, kde `imin` je rovno 0 a `imax` je rovno 1, `m` je rovno požadované délce generované posloupnosti a `n` je rovno 1. Získáme tak binární posloupnost požadované délky. Načtení dat ze souboru se provádí pomocí funkcí `fopen`, která soubor otevře, a `fread`, která soubor přečte a získá z něj požadovanou posloupnost. Načtená data je možné omezit na požadovanou velikost. Pokud je soubor kratší než požadovaná velikost omezení, je pro simulaci použita celá délka souboru a uživatel je informován formou informačního hlášení, že vstupní posloupnost je kratší, než požadoval. Hlášení obsahuje informaci o skutečné délce vstupní posloupnosti.

1.2 Přidání CRC k Transportnímu bloku

Kontrolní součet se používá z důvodu ověření shody přijatých dat s vysílanými. K vysílané posloupnosti dat je přidána skupina bitů, ve které je zakódována informace o této posloupnosti. Jednoduchý příklad je přidání paritního bitu, který je doplňkem vstupních dat, ke kterým je přidáván a z nichž se vypočítá. Výpočet paritního bitu je velmi jednoduchý, pouze spočítáme počet jedniček a podle typu parity doplníme paritní bit. Sudá parita představuje fakt, že skupina má sudý počet jedniček. To znamená, že pokud je vstupní posloupnost tvořena několika nulami a třemi jedničkami, tak paritní bit bude mít hodnotu jedna, aby doplnil počet jedniček na sudou hodnotu. Pokud použijeme lichou paritu, tak by paritní bit měl hodnotu nula, protože posloupnost již má lichý počet jedniček. Při příjmu je kontrolní část oddělena, znovu spočítán paritní bit a porovnán s kontrolní částí, která byla přijata. Nebo lze pro kontrolu s výhodou použít součet modulo 2, kde výsledek kontroly pro sudou paritu má být nula, v případě liché parity jedna. Nevýhodou kontroly pomocí paritních bitů je fakt, že pokud dojde ke dvěma chybám, nebo obecně k sudému počtu chyb, bude vyhodnocen kontrolní součet jako správný i když informace bude rozdílná. V dnešní době existuje mnoho způsobů detekce chyby. Některé metody dříve používané pro detekci chyby se vyvinuly v metody, které tyto chyby dokáží i opravit, příkladem jsou lineární blokové kódy. [11]

Dokonalejším algoritmem pro detekci chyby je CRC (Cyclic redundancy check).

Princip je založen na dělení polynomů a přidání zbytku po dělení jako kontrolní součet. Velmi důležitou částí tohoto algoritmu je právě polynom, kterým je vstupní posloupnost dělena. Na základě volby tohoto polynomu je možné říci, že i malá změna vstupní posloupnosti bude znamenat velkou změnu kontrolního součtu.



Obr. 1.2: Přidání kontrolního součtu k Transportnímu bloku

Přidání CRC k transportnímu bloku je první ochrana, kterou systém má. Kontrolní součet zaručuje, že přijatá data jsou s vysokou pravděpodobností shodná s vyslanými. Při příjmu je kontrolní součet spočítán pro přijatá data a je ověřena shoda s přijatým součtem, který byl přijat spolu s daty. Pokud se při příjmu dat nekorektně obnovila data nebo kontrolní součet, který je přijat spolu s daty, je velmi vysoká pravděpodobnost, že kontrolní součty nebudou souhlasit. Pak je informace o chybném příjmu ihned poskytnuta systému HARQ (Hybrid automatic repeat request), který se stará o nové přeposlání dat. Přidání kontrolního součtu k datům z transportního bloku je zobrazeno na obrázku 1.2.

Kontrolní součet, pro ochranu dat, je definován podle čtyř následujících rovnic. Rovnice, která definuje kontrolní součet o délce 24 bitů [1]:

$$g_{CRC24A}(D) = [D^{24} + D^{23} + D^{18} + D^{17} + D^{14} + D^{11} + D^{10} + D^{18} + D^7 + D^6 + D^5 + D^4 + D^3 + D + 1] \quad (1.1)$$

a

$$g_{CRC24B}(D) = [D^{24} + D^{23} + D^6 + D^5 + D + 1], \quad (1.2)$$

rovnice definující CRC o délce 16 bitů[1]:

$$g_{CRC16}(D) = [D^{16} + D^{12} + D^5 + D + 1] \quad (1.3)$$

a rovnice definující CRC o délce 8 bitů[1]:

$$g_{CRC8}(D) = [D^8 + D^7 + D^4 + D^3 + D + 1] \quad (1.4)$$

V případě přenosu uživatelských dat je použit kontrolní součet délky 24 bitů definovaný dle rovnice 1.1. Pokud se použije menší počet bitů pro kontrolní součet, je pravděpodobnost, že chyba při příjmu dat nebude zaregistrována větší. Čtvrtá rovnice 1.2 je použita v případě, že velikost transportního bloku je větší než $Z = 6144$ bitů. Pak je aplikován potup popsáný v kapitole 1.3 Rozdělení transportního bloku na segmenty a přidání CRC ke každému segmentu. Pokud je posloupnost z transportního bloku kratší, krok dalšího rozdělení na menší subbloky není vynechán. Nedochází k rozdělení na menší subbloky a nepřidává se další kontrolní součet, ale velikost bloku se upraví na konkrétní hodnotu přidáním plnicích bitů.

Kontrolní součet délky 24 bitů je vytvořen dle matematického popisu znázorněného rovnicí[1]:

$$a_0D^{A+23} + a_1D^{A+22} + \dots + a_{A-1}D^{24} + p_0D^{23} + p_1D^{22} + \dots + p_{22}D^1 + p_{23}, \quad (1.5)$$

pro vytvoření kontrolního součtu délky 16 bitů rovnicí[1]:

$$a_0D^{A+15} + a_1D^{A+14} + \dots + a_{A-1}D^{16} + p_0D^{15} + p_1D^{14} + \dots + p_{14}D^1 + p_{15} \quad (1.6)$$

a pro vytvoření kontrolního součtu délky 8 bitů rovnicí[1]:

$$a_0D^{A+7} + a_1D^{A+6} + \dots + a_{A-1}D^8 + p_0D^7 + p_1D^6 + \dots + p_6D^1 + p_7. \quad (1.7)$$

Principem je přidání určitého počtu bitů 8, 16 nebo 24 tak, aby těchto několik bitů způsobilo, že zbytek po dělení takovéto prodloužené sekvence dat byl nulový. Kontrolní součet je možno realizovat posuvným registrem a XOR členy. [1]

Pro připojení kontrolního součtu k transportnímu bloku slouží funkce $[b] = \text{add_CRC}(type, a)$. Parametr *type* očekává typ generátoru CRC, tedy 'gCRC24A', 'gCRC24B', 'gCRC16' nebo 'gCRC8'. Parametr *a* je vstupní posloupnost, ke které se přidává kontrolní součet. Pro generování CRC se v modelu používá funkce $\text{generate}(gen, msg)$, kde jako parametr *gen* je vložen objekt `crc.generator` a *msg* je zpráva, kterou chceme zakódovat. Objekt `crc.generator()` vyžaduje zadat konstrukci generátoru, která může být zadána ve formě polynomu. Na příklad, generátor pro generování CRC s délkou 8 bitů použitý v LTE je možné vygenerovat příkazem `crc.generator([1 1 0 0 1 1 0 1 1])`. Polynom je vytvořen tak, že u členů, které jsou v rovnici uvedeny, je jednička a u členů, které uvedeny v rovnici nejsou je 0, v pořadí od nejvyšší mocniny po nejnižší. Proměnná *b* je pak vstupní posloupnost opatřená kontrolním součtem, který je umístěn na konci posloupnosti.

1.3 Rozdělení na menší bloky a přidání CRC ke každému bloku

Uživatelská data v systému LTE jsou kódována pomocí blokového kódu. Ten dovoluje zpracovávat pouze bloky dat. Navíc tyto bloky musí mít určitou velikost. Jelikož je velikost transportního bloku neznámá, existuje algoritmus jak zajistit, aby blok měl tuto určitou velikost. Nejjednodušším řešením je, v případě menšího bloku, než je maximální velikost, doplnění tohoto bloku plnicími bity. V případě, že velikost Transportního bloku přesahuje maximální velikost pro zpracování, je nutné blok dělit. Lze vymyslet několik způsobů, jak blok rozdělit. Nejjednodušší je blok rozdělit na bloky o maximální velikosti a zbytek doplnit na potřebnou nejnižší velikost. Ovšem to by znamenalo, že po useknutí může zůstat i velmi malý počet bitů, který by byl třeba doplnit značným množstvím plnicích bitů. Jiným způsobem je rozdělení bloků tak, aby jejich velikosti byly co nejpodobnější a počet bloků byl nejmenší možný. Tento způsob vyžaduje komplikovanější výpočet a využívá se v LTE.

Pokud má transportní blok větší délku než $Z = 6144$ bitů, pak dochází k jeho rozdělení a přidání dalšího kontrolního součtu délky 24 bitů ke každému segmentu. Pokud je transportní blok menší, pouze se upraví velikost bloku na nejbližší větší hodnotu pomocí plnicích bitů. Přidání plnicích bitů je nutné i pokud je transportní blok větší než je definovaná hranice a transportní blok se rozděluje na segmenty. Každý z těchto segmentů má přesně definovanou velikost. Pokud je rozdílná od skutečné velikosti transportního bloku s kontrolním součtem, je přidáno několik plnicích bitů tak, aby byly velikosti všech bloků zarovnané na tabulkovou hodnotu.

Příklad: Pokud je velikost transportního bloku nižší než hodnota $Z = 6144$ bitů, pak je jeho velikost upravena na přesnou velikost. Seznam možných velikostí je uveden v příloze v tabulce A.4 a A.5. Velikost je parametr K_i . Upravení velikosti vstupní posloupnosti se provádí přidáním plnicích bitů. Pokud je ale transportní blok větší než maximální velikost, pak dochází k jeho rozdělení na segmenty. Počet segmentů je dán vztahem[1]:

$$C = \left\lceil \frac{B}{Z - L} \right\rceil, \quad (1.8)$$

kde C je výsledný počet segmentů, B je velikost transportního bloku, Z je maximální velikost segmentu a L je velikost kontrolního součtu.

Na základě znalosti počtu segmentů se pak stanoví přesná velikost všech segmentů i s přidávanými kontrolními součty, vztah[1]:

$$B' = B + C \cdot L. \quad (1.9)$$

Dále se stanoví první větší velikost segmentů. Stanoví se tak, že se vybere nejmenší možné K_+ , které splňuje podmínku $C \cdot K_+ \geq B'$. Druhá menší velikost segmentů je stanovena tak, že se vybere hned další nižší K_i , než v předchozím případě, tedy maximální K_- , které splňuje podmínku $K_- < K_+$. Zbývá určit počet menších segmentů, vztah[1]:

$$C_- = \left\lfloor \frac{C \cdot K_+ - B'}{\Delta_K} \right\rfloor, \quad (1.10)$$

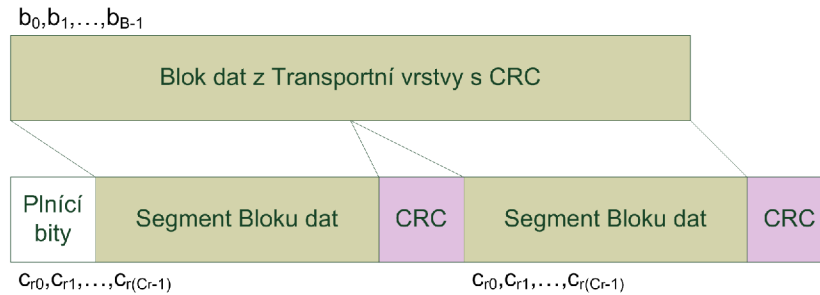
počet větších segmentů, vztah[1]:

$$C_+ = C - C_- \quad (1.11)$$

a počet plnicích bitů, které upravují velikost transportního bloku tak, aby bylo možné jeho obsah vtěsnat přesně do vytvořených subbloků, vztah[1]:

$$F = C_+ \cdot K_+ + C_- \cdot K_- - B'. \quad (1.12)$$

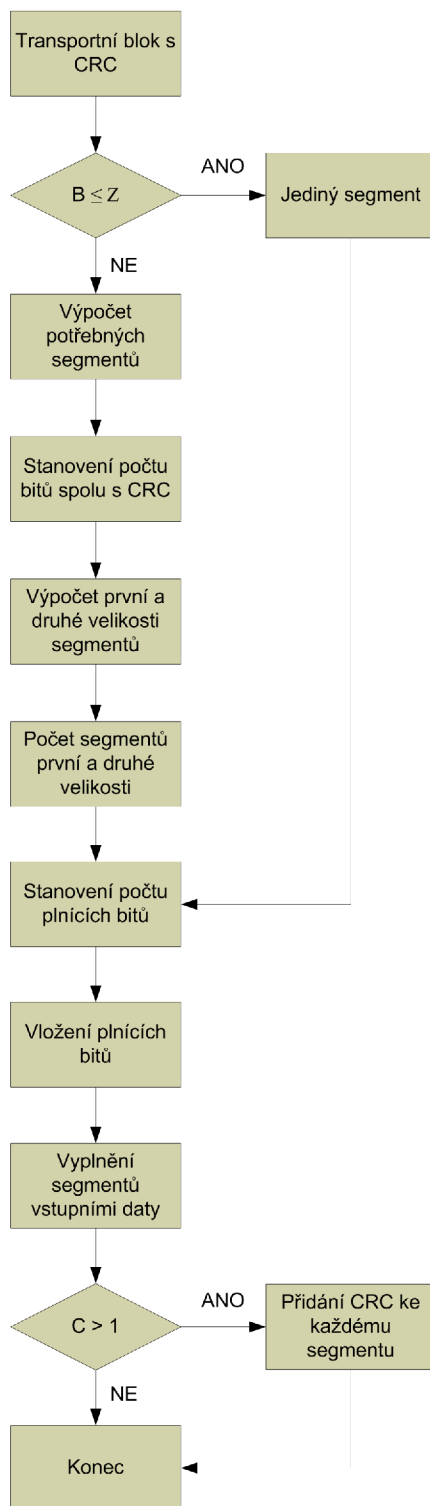
K výpočtu počtu bloků je třeba znát rozdíl velikosti menšího a většího segmentu, tedy $\Delta_K = K_+ - K_-$ [1].



Obr. 1.3: Rozdělení Transportního bloku na segmenty a přidání kontrolních součtů

Uspořádání dat v segmentech je následující. Nejprve jsou do menšího segmentu vloženy plnicí bity, pokud jsou třeba. Následně jsou naplněny všechny menší segmenty, pak teprve jsou naplněny všechny větší segmenty zbývajících daty z transportního bloku. Ke každému segmentu je nakonec připojen kontrolní součet. Operace je znázorněna na obrázku 1.3.

Celý proces rozdělení transportního bloku na segmenty je zobrazen na obrázku 1.4. [1]



Obr. 1.4: Blokové schéma dělení transportního bloku na segmenty

O proces rozdělení Transportního bloku na dílčí segmenty se stará funkce $[c \ F \ l] = \text{cbSegmentation}(b)$. Parametr b je vstupní posloupnost, tedy Transportní blok

opatřený kontrolním součtem. Výstupní proměnná c je skupina segmentů, délka jednotlivých bloků je uchovávána v proměnné l a počet plnicích bitů je uložen v proměnné F . Jednotlivé délky segmentů a počet plnicích bitů lze spočítat z délky transportního bloku. V modelu jsou informace uchovávány pro detailnější zobrazení postupu simulace.

Funkce postupuje podle standardu. V první fázi jsou definovány proměnné Z , B a tabulka velikostí bloku K_table . Proměnná Z je maximální velikost bloku a je rovna konstantě 6144 bitů. Proměnná B označuje délku vstupní posloupnosti. Následně je vypočítáno potřebné množství segmentů C , velikost všech segmentů spolu s kontrolními součty $B_$ a stanoví se velikost segmentu K_p , která je vybrána z tabulky jako nejmenší možná. Dále se výpočet větví podle toho, jestli je zapotřebí transportní blok s kontrolním součtem dělit na více segmentů.

Pokud dělení není potřeba, vypočítá se počet potřebných plnicích bitů F , které se přidají před Transportní blok tak, aby celý segment dosahoval velikosti uvedené v tabulce A.4 a A.5.

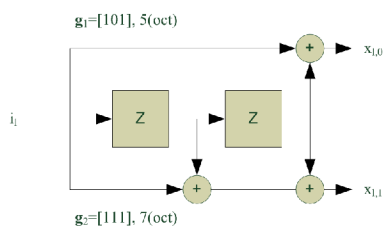
Pokud ovšem transportní blok s kontrolním součtem přesahuje maximální dovolenou velikost, bude se dělit na segmenty. Nejprve je zapotřebí nalézt obě velikosti segmentů K_p a K_m , spočítat jejich rozdíl $delta_K$ a stanovit počty segmentů obou velikostí C_m a C_p . Na základě získaných parametrů lze spočítat počet plnicích bitů F . Následně jsou vytvořeny segmenty, kdy nejprve jsou vloženy do prvního segmentu plnicí bity, následně data z transportního bloku spolu s jeho kontrolním součtem. Jakmile je první segment zaplněn, pokračuje se zaplňováním i ostatních segmentů. Každý ze segmentů je opatřen novým kontrolním součtem délky 24 bitů. Tento součet je počítán z definice generátoru uvedeného v rovnici 1.2.

1.4 Kanálové kódování

Největší vliv na kvalitu zabezpečení a odolnost signálu proti rušení má kanálové kódování. Přidáním značného množství redundance, často stejného, či dokonce dvakrát většího počtu redundantních bitů oproti informačním se dosahuje velmi silného zabezpečení přenášené informace. Kanálové kódování má za úkol ochránit informaci a v případě, že při přenosu nastane chyba, to znamená je dekodována opačná hodnota bitu než byla vyslána, algoritmus dokáže tuto chybu najít a opravit. Rozdíly jsou především v tom, jaké množství chyb a v jaký typ chyb algoritmus dokáže ještě opravit a jaký počet chyb je pro něj hraniční.

Konvoluční kódování

Konvoluční kódy jsou založeny na přidání redundance tak, že se ke provádění určité matematické operace nad posloupností a jejími zpožděnými kopiemi. Hlavní částí konvolučního kodéru je posuvný registr s M zpožďovacími členy, který je doplněn o sčítací členy, tak jak je zobrazeno na obrázku 1.5. Počet sčítacích členů a jejich umístění je dáno generujícím mnohočlenem stupně M . Pokud generující mnohočlen obsahuje prvek s danou mocninou, je signál z příslušné buňky registru přiveden na sčítací prvek.



Obr. 1.5: Schéma konvolučního kodéru s poměrem 1/2

Na obrázku je už zjednodušeně zapsána posloupnost $g_1 = [101]$, která představuje generující polynom $x^2 + 1$. Často je tento generující polynom zapsán v osmičkové soustavě. Mezi základní parametry konvolučního kodéru patří kódový poměr, který je dán počtem bitů vstupujícími do kodéru k a počtem bitů vystupujícími z kodéru n . Kódový poměr R je pak poměr mezi k a n , tedy $R = \frac{k}{n}$. Hodnota kódového poměru je často 1/2 nebo 1/3. Lze vytvořit i kodéry s kódovým poměrem 2/3, 3/4, 5/6. Ovšem častěji bývají tyto poměry dosahovány za pomoci tečkování, tedy záměrného vynechání určitých bitů z původní kódové posloupnosti, která byla vytvořena zakódováním vstupních dat kodérem s poměrem vyšším, tedy například 1/2.

V LTE je použit Tail biting konvoluční kódér. Označení Tail biting znamená, že počáteční stav hodnot registru je shodný s konečným stavem, tedy je nastaven podle posledních 6 bitů vstupní posloupnosti. Díky tomuto opatření dosáhneme snazšího dekódování na straně přijímače. [11], [10]

Turbo kódy

Turbo kódy vynikají především v oblasti nízkého poměru C/N, kde dosahují nízké úrovně chybovosti. V podstatě se jedná o dva paralelní konvoluční kodéry, které jsou shodné, přičemž na jeden kódér se přivádí posloupnost, která má být zakódována a na druhý kódér se přivádí posloupnost proložená. O proložení se stará vnitřní prokladač, který prokládá bloky dat. Proto musí být vstupem turbo kodéru blok dat

místo toku dat. Celkový zisk kódování je velmi závislý právě na typu tohoto vnitřního prokladače. Obecně může být prokladač konstruován libovolně, může se použít jednoduchý prokladač, který proloží řádky za sloupce. Ovšem nejlepších výsledků je dosahováno s prokladači, které prokládají data složitějšími způsoby, které se často blíží náhodnému prokládání. Velkou výhodou turbo kodérů je možnost iterativního dekódování, které postupně vylepšuje svůj výsledek dekódování. [10]

Kanálové kódování v LTE

V předchozích fázích systém přeměnil data ve formě transportního bloku do segmentu, či do více segmentů, o jiné určité velikosti. Toto upravení velikosti a případné rozdělení na segmenty je nutné především kvůli kanálovému kódování, které využívá Turbo kodér, jenž se řadí do kodérů zpracovávající bloky dat, na rozdíl od konvolučních kodérů, které zpracovávají proud dat. V LTE jsou použity celkem čtyři způsoby kódování informace, všechny jsou uvedeny v tabulkách 1.3 a 1.4.

Tab. 1.3: Použitý druh kódování pro transportní kanály [1]

Transportní kanály	Kódovací schéma	Kódovací poměr
DL-SCH	Turbo kodér	1/3
PCH		
MCH		
BCH	Tail biting konvoluční kodér	1/3

Tab. 1.4: Použitý druh kódování pro kontrolní kanály [1]

Kontrolní kanály	Kódovací schéma	Kódovací poměr
CFI	Blokový kód	1/16
HI	Opakující kód	1/3
DCI	Tail biting konvoluční kodér	1/3

Tail biting konvoluční kodér i turbo kodér je popisován výše. Dalšími typy kódovacích schémat je opakující kód (Repetition code), který pouze zopakuje třikrát daný symbol. Nula je tedy kódována jako tři nuly po sobě jdoucí. Blokový kód je kódování, které ze dvou bitů vytvoří skupinu 32 bitů. Oba typy kódování se řídí převodní tabulkou, která říká, který stav odpovídá jaké konečné posloupnosti. Viz. tabulky 1.5, 1.6.

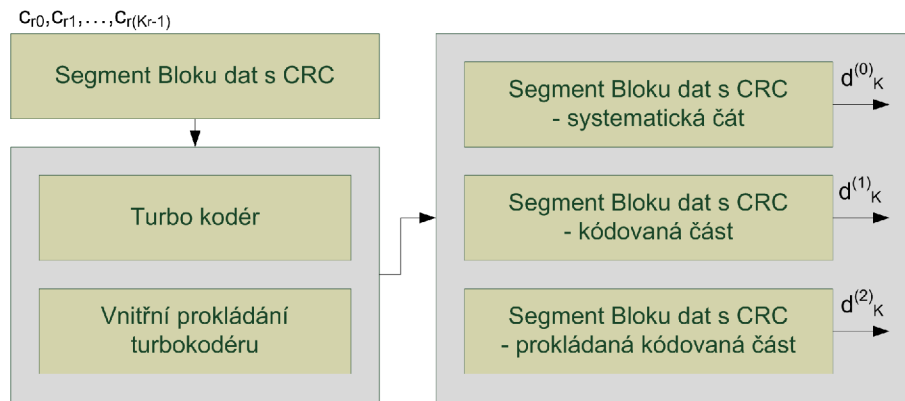
Tato práce se zaměřuje pouze na přenos uživatelských dat, která jsou kódována pomocí turbo kódů. Proto je v pozdější části kapitoly popisován jen turbo kodér, jeho vnitřní zapojení a způsob kódování.

Tab. 1.5: Opakující kód, kódování HI [1]

HI	HI kódové slovo b0, b1, b2
0	0,0,0
1	1,1,1

Tab. 1.6: Blokový kód, kódování CFI [1]

CFI	CFI kódové slovo b0, b1, ..., b31
1	0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1
2	1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0
3	1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1
4 (Rezervováno)	0,0

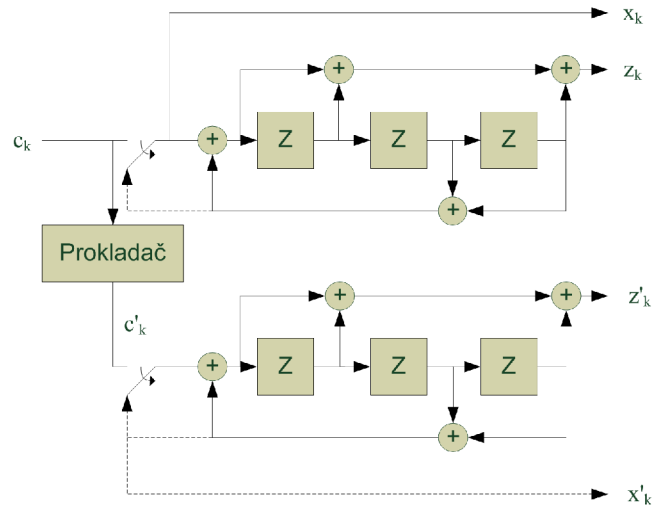


Obr. 1.6: Blokové znázornění kanálového kódování

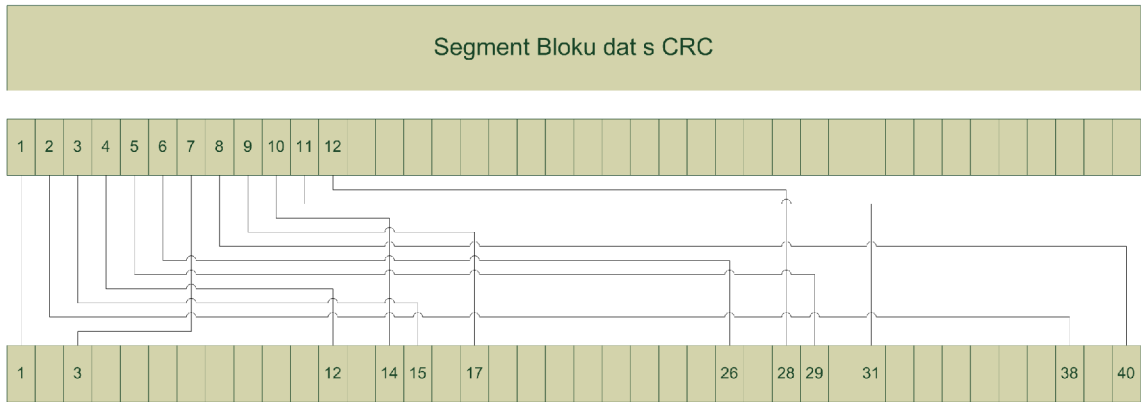
Na obrázku 1.6 je ukázáno, jak do Turbo kodéru vstupuje segment bloku dat z předchozího zpracování a vystupují tři bloky dat - systematická část a dvě zakódované části.

Přesné vnitřní zapojení Turbo kodéru je uvedeno na obrázku 1.7. Skládá se ze dvou identických konvolučních kodérů a bloku interního prokladače. Vstupní bitová posloupnost je tak zakódována dvakrát. Jednou je zakódována pomocí konvolučního kodéru a podruhé je zakódována posloupnost, která je přeskládána interním prokladačem. Navíc je přenášena i systematická část.

Prokladač je důležitou součástí Turbo kodéru. V LTE je použit prokladač typu QPP (Quadratic Permutation Polynomial). Přerozděluje vstupní sekvenci bitů tak, aby blízké vzorky ve vstupní sekvenci byly ve výstupní sekvenci hodně vzdálené.



Obr. 1.7: Vnitřní zapojení Turbo kodéru (převzato z [1])



Obr. 1.8: Příklad funkce interního prokladače pro Turbo kódér

Názorná ukázka přerozdělení vstupních 40 bitů je zobrazena na obrázku 1.8.

Přesná matematická definice je [1]:

$$c'_i = c_{\Pi(i)}, i = 0, 1, \dots, (K - 1). \quad (1.13)$$

Funkce Π je definována jako [1]:

$$\Pi(i) = (f_1 \cdot i + f_2 \cdot i^2) \bmod K. \quad (1.14)$$

Konstanty f_1 a f_2 jsou uvedeny v tabulkách A.4 a A.5. Tyto konstanty jsou vázány na určitou velikost bloku K_i .

V modelovaném systému se o zakódování dat stará funkce $[w \ l2] = \text{chCoding}(c, F, l)$. Funkce očekává blok zpracovaných segmentů s kontrolními součty jako vstupní

proměnnou c . Proměnná F obsahuje počet plnicích bitů a proměnná l seznam délek segmentů. Výstupem je proměnná w , která představuje blok segmentů po kanálovém kódování a proměnná $l2$ obsahuje seznam délek segmentů po kanálovém kódování.

Funkce pracuje se segmenty opatřenými kontrolním součtem. Každý segment zpracovává jednotlivě. Nejdříve je vytvořena struktura kodéru za pomoci funkce `poly2trellis(ConstraintLength, CodeGenerator, FeedbackConnection)`, kde parametr `ConstraintLength` určuje délku generátoru. `CodeGenerator` zapojení generátoru a `FeedbackConnection` určuje zapojení zpětných vazeb. Obě zapojení jsou zadány čísly v osmičkové soustavě. Pokud chceme toto zapojení zkonstruovat, potřebujeme znát zapojení kodéru. V LTE je použit kodér zobrazený na obrázku 1.7. Zapojení dopředné větve převedeme na binární posloupnost tak, že doplníme 1 u zpoždovacího členu, kde je sčítací člen, když zde sčítací člen není, tak doplníme 0. V případě LTE kodéru vznikne posloupnost 1101, kterou převedeme do osmičkové soustavy jako číslo 15. Obdobně u zpětné větve dostaneme 1011, což je po převedení 13. Struktura kodéru se vytvoří pomocí funkce `poly2trellis(4, [13 15], 13)`.

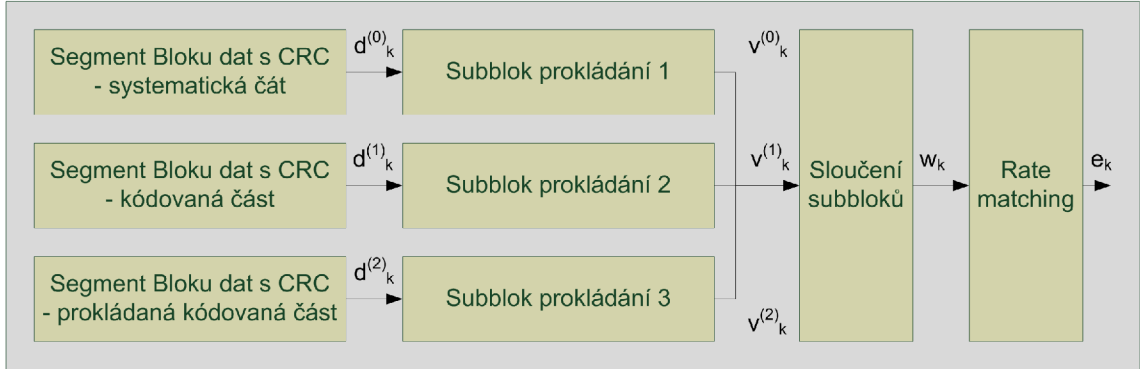
K zakódování se pak používá objekt `comm.TurboEncoder(TrellisStructure, trellis, InterleaverIndices, i_)`; kde parametr `trellis` je konstrukce kodéru, kterou jsme vytvořili v předchozím odstavci a `i_` je informace o tom, jak se mají data přeskládat v interním prokladači. Funkce `[c_ i_] = InternalInterleavingTurbo(c)`, provádí operace přeskládání podle pravidel pro interní prokladač turbo kodéru a poskytuje posloupnost pro přeskládání. Parametr c je vstupní segment dat. $c_$ je přeskládaný segment dat a $i_$ je proložená posloupnost, kterou před proložením představovala posloupnost $1,2,3,\dots,n$, kde n je délka segmentu c . Funkce provádí přeskládání na základě rovnic 1.13 a 1.14.

1.5 Přizpůsobení datového toku

V případě, že uživatelské zařízení má dostatečnou úroveň a kvalitu signálu, není nutné přenášenu informaci zabezpečovat tak důkladně, jako když dostatečnou úroveň ani kvalitu nemá. Proto je v bloku Přizpůsobení datového toku odstraňována část redundance, která byla přidána v bloku Kanálové kódování. Toto ovšem znamená, že blok Přizpůsobení datového toku musí být navržen pro konkrétní typ kanálového kódování, tedy musí přesně vědět, které bity může vynechat a které ne. Dalším důvodem zpracování signálu v tomto bloku je omezení jeho velikosti tak, aby se zakódovaná informace přesně vešla do struktury rádiového rámce a nebylo nutno nikde přidávat plnicí bity z důvodu nevhodného počtu zakódovaných bitů, které by přebývali nebo scházeli.

Z Turbo kodéru vystupují tři subbloky dat. Tyto subbloky jsou nejdříve přesklá-

dány do matice, dále sloučeny do jednoho bloku a až potom je upravována velikost tohoto bloku za pomoci Přizpůsobení datového toku. Celý proces je vyobrazen na obrázku 1.9.



Obr. 1.9: Blokové znázornění přizpůsobení datového toku

1.5.1 Prokládání subbloků

Subbloky vystupující z Turbo kodéru jsou následně naskládány do matice s počtem sloupců $R_{subblock}^{TC} = 32$. Počet řádků je stanoven dle rovnice[1]:

$$D \leq \left(R_{subblock}^{TC} \times C_{subblock}^{TC} \right), \quad (1.15)$$

kde D je počet bitů v subbloku.

Pokud je vstupní velikost subbloku nedělitelná beze zbytku 32, pak je nutný počet bitů doplněn plnicími bity, které jsou vloženy na začátek matice. Počet plnicích bitů je $N_D = \left((R_{subblock}^{TC} \times C_{subblock}^{TC} - D) \right)$. Prvky matice jsou označeny y_0 až $y_{(R_{subblock}^{TC} \times C_{subblock}^{TC})}$. Číslování probíhá od prvního sloupce do posledního, pak následně od prvního řádku do posledního.

Následující operace se liší podle výstupu z Turbo kodéru. Výstupy $d_k^{(0)}$ a $d_k^{(1)}$ jsou pouze přeskládány sloupce matice podle funkce P, hodnoty funkce P jsou uvedeny v tabulce 1.7. Data jsou potom čtena po sloupcích.

Třetí výstup $d_k^{(2)}$ je však prokládán jiným způsobem. Způsob prokládání popisuje následující rovnice[1]:

$$v_k^{(2)} = y_{\pi(k)} \quad (1.16)$$

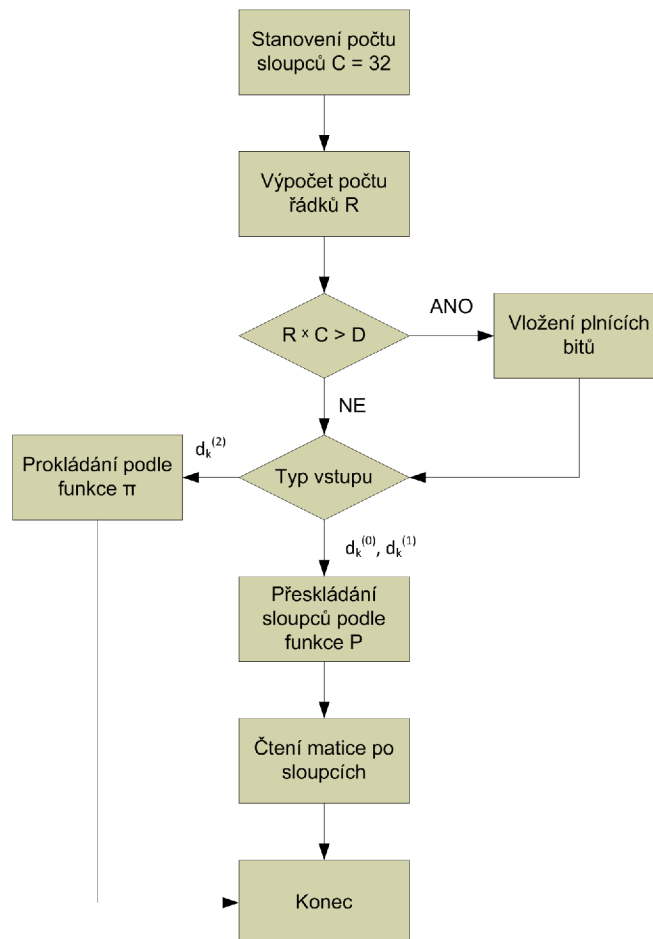
Tab. 1.7: Parametry pro interní prokladač subbloků

Počet sloupců $C_{subblock}^{TC}$	Pořadí sloupce po prokládání $\langle P(0), P(1), \dots, P(C_{subblock}^{TC} - 1) \rangle$
32	$\langle 0, 16, 8, 24, 4, 20, 12, 28, 2, 18, 10, 26, 6, 22, 14, 30, 1, 17, 9, 25, 5, 21, 13, 29, 3, 19, 11, 27, 7, 23, 15, 31 \rangle$

Funkce prokládání $\pi(k)$ je vyjádřena rovnicí[1]:

$$\pi(k) = \left(P \left(\left\lfloor \frac{k}{R_{subblock}^{TC}} \right\rfloor \right) + C_{subblock}^{TC} \times (k \bmod R_{subblock}^{TC}) + 1 \right) \bmod K_{\Pi}. \quad (1.17)$$

Celý proces prokládání subbloků vystupujících z Turbo kodéru je znázorněn na obrázku 1.10. [1]



Obr. 1.10: Vývojový diagram prokládání subbloků

V simulovaném modelu se prokládání subbloků děje přímo ve funkci `chCoding()`. Nejprve je zjištěna délka subbloku, následně je z rovnice 1.15 vypočítán počet řádků matice. Poté je stanoven počet plnicích bitů, které jsou přidány na začátek subbloku. Subblok je pak naskládán do matice o 32 sloupcích a $R_{subblock}^{TC}$ řádcích. Přeskládání se vykonává pomocí funkce `reshape(A, m, n)`, kde A představuje vstupní subblok, m počet řádků a n je počet sloupců. U subbloků $d_k^{(0)}$ a $d_k^{(1)}$ následně dochází k zaměnění sloupců matice podle funkce P . Sloupec $y_0 = y_{P(0)}$, kde y_0 představuje první sloupec matice. Následně je zaměněn další sloupec, tedy $y_1 = y_{P(1)}$, kde y_1 je druhý sloupec matice. Takto se pokračuje až do úplného přeskládání matice. Odpovídající hodnoty funkce P jsou uvedeny v tabulce 1.7.

Subblok $d_k^{(2)}$ je přeskládán dle jiného pravidla. Toto pravidlo popisují rovnice 1.16 a 1.17.

1.5.2 Sloučení subbloků

Z obrázku 1.9 je patrné, že po prokládání subbloků, které vystupují jako posloupnosti $v_k^{(0)}$ až $v_k^{(2)}$ následuje jejich sloučení do jednoho bloku dat. Postup sloučení pro $v_k^{(0)}$ je definován rovnicí[1]:

$$w_k = v_k^{(0)} \text{ pro } k = 0, \dots, K_{\Pi} - 1, \quad (1.18)$$

posloupnost $v_k^{(1)}$ je sloučena podle vztahu[1]:

$$w_{K_{\Pi}+2k} = v_k^{(1)} \text{ pro } k = 0, \dots, K_{\Pi} - 1, \quad (1.19)$$

a poslední posloupnost $v_k^{(2)}$ je sloučena podle vztahu[1]:

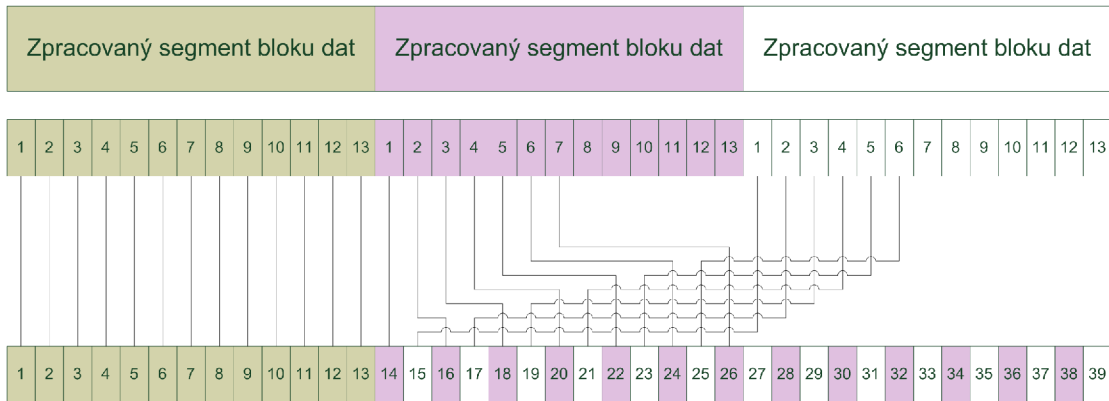
$$w_{K_{\Pi}+2k+1} = v_k^{(2)} \text{ pro } k = 0, \dots, K_{\Pi} - 1. \quad (1.20)$$

Celková délka sloučených subbloků bude $K_w = 3 \cdot K_{\Pi}$, kde K_{Π} je délka jednoho subbloku. Příklad sloučení subbloků je zobrazen na obrázku 1.11.

V simulovaném modelu se sloučení subbloků děje uvnitř funkce `chCoding()`. Sloučení probíhá podle vztahů 1.18, 1.19 a 1.20.

1.5.3 Přizpůsobení datového toku (Rate Matching)

Princip funkce a důvod, proč se provádí úprava datového toku, je uveden na začátku této sekce, v modelu je však Přizpůsobení datového toku vynecháno. Podrobnější informace lze najít v literatuře [1].

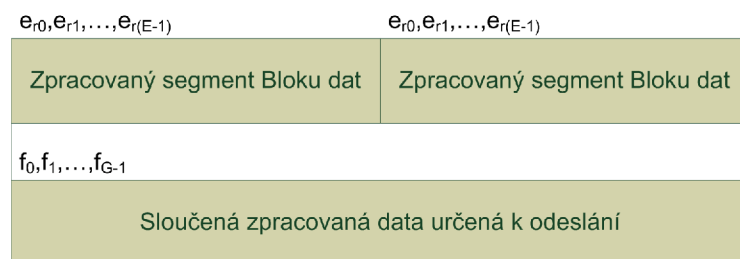


Obr. 1.11: Ukázka postupu sloučení subbloků

1.6 Sloučení segmentů zpět do jednoho celku

Sloučení segmentů zpět do jednoho bloku se provádí z důvodu přenosu Transportního bloku jako celku, zakódovaný Transportní blok se nazývá kódové slovo.

Po zpracování dat v bloku Přizpůsobení datového toku jsou jednotlivé zpracované segmenty spojeny zpět do jednoho bloku. Spojování se provádí od prvního segmentu, který vstoupil do zpracování, až do posledního segmentu. Proces spojování je zobrazen na obrázku 1.12. [1]



Obr. 1.12: Příklad sloučení zpracovaných segmentů

V modelu se sloučení segmentů provádí ve funkci `chCoding()`. Sloučení probíhá podle popisu uvedeného v předchozím odstavci.

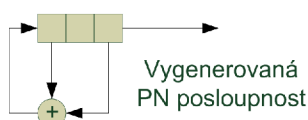
1.7 Skramblování

Operace skramblování je přičtení pseudonáhodné posloupnosti k datové posloupnosti. Tímto postupem se mohou výrazně snížit interference mezi signály.

Pseudonáhodná posloupnost

Pseudonáhodná posloupnost je taková posloupnost, která má náhodný charakter, to znamená pravděpodobnost výskytu jedničky i nuly je stejná. V případě posloupností o více stavech je pravděpodobnost každého ze stavů stejná. Dostatečně dlouhá posloupnost je tedy ideálně tvořena stejným počtem každého z možných stavů. Podmínkou pseudonáhodné posloupnosti je fakt, že musí být matematicky definovatelná, to znamená, že ji lze kdykoli zrekonstruovat.

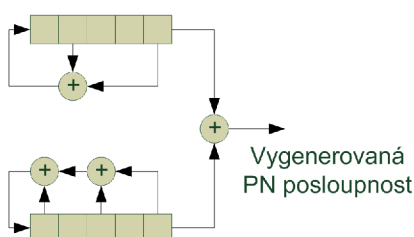
Jedním ze základních typů PN posloupností je m-sekvence. Je generována pomocí posuvného registru a sčítacích členů.



Obr. 1.13: Příklad generátoru m-sekvence

Parametrem generátoru m-sekvence je počáteční stav registru, který nesmí být nastaven na samé nuly, protože pak by výstupem byla posloupnost samých nul. Dalším důležitým parametrem je zapojení sčítacích členů a zpětných vazeb. Příklad zapojení generátoru m-sekvence je zobrazen na obrázku 1.13.

Dalším typem PN posloupností jsou Gold sekvence. Velkým přínosem Gold sekvencí je nižší vzájemná korelace než u m-sekvencí. Velikost vzájemné korelace mezi dvěma sekvencemi je důležitá neboť na ní přímo závisí velikost interferencí těchto dvou signálů v rádiovém prostředí. Generování Gold sekvencí probíhá podobným způsobem jako u m-sekvencí s tím rozdílem, že jsou použity dva posuvné registry a jejich výstup je přiveden na sčítací člen, který poskytuje výstup celého generátoru.



Obr. 1.14: Příklad generátoru Gold sekvence

Na obrázku 1.14 je zobrazen generátor Gold sekvence. [11]

Skramblování v LTE

Každé kódové slovo q se skládá z bitů $b^{(q)}(0), \dots, b^{(q)}(M_{bit}^q - 1)$, kde M_{bit}^q je počet bitů v kódovém slově q , přenášených v jednom subrámcí. Tato posloupnost je skramblována podle vztahu[1]:

$$\tilde{b}^q(i) = (b^q(i) + c^q(i)) \bmod 2. \quad (1.21)$$

Výsledkem je skramblovaná posloupnost \tilde{b}^q stejné délky, jako vstupní posloupnost.

Sekvence pro skramblování $c^q(i)$ je definována vztahem[1]:

$$c(n) = (x_1(n + N_C) + x_2(n + N_C)) \bmod 2, \quad (1.22)$$

kde $N_C = 1600$, $n = 0, 1, \dots, M_{PN} - 1$ a kde M_{PN} je délka pseudonáhodné posloupnosti.

Posloupnost x_1 je dána vztahem[1]:

$$x_1(n + 31) = (x_1(n + 3) + x_1(n)) \bmod 2 \quad (1.23)$$

x_1 musí být inicializováno sekvencí $x_1(0) = 1$ a $x_1(n) = 0$ pro $n = 1, 2, \dots, 30$. Posloupnost x_2 je dána vztahem[1]:

$$x_2(n + 31) = (x_2(n + 3) + x_2(n + 2) + x_2(n + 1) + x_2(n)) \bmod 2. \quad (1.24)$$

Posloupnost x_2 je inicializována pomocí čísla c_{init} , které je převedeno na binární číslo. $x_2(0)$ je LSB (Least significant bit).

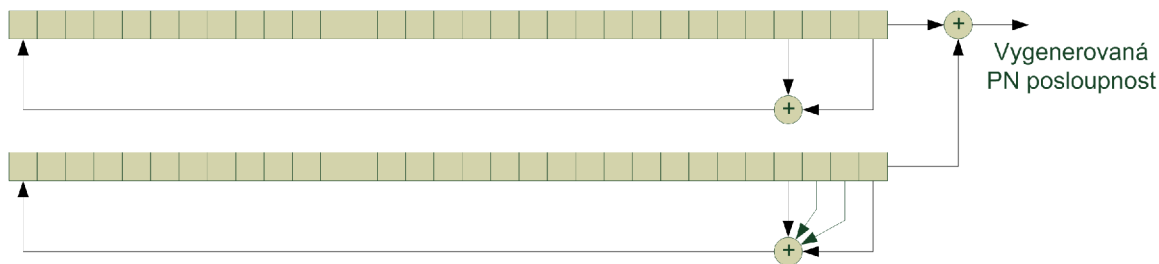
Sekvence pro zakódování musí být inicializována pro každý subrámeček jiným inicializačním číslem. Toto číslo je dáno vztahem[1]:

$$c_{init} = n_{RNTI} \cdot 2^{14} + q \cdot 2^{13} + \lfloor n_s/2 \rfloor \cdot 2^9 + N_{ID}^{cell}, \text{ pro Fyzický sdílený kanál, } (1.25)$$

kde n_{RNTI} odpovídá RNTI (Radio Network Temporary Identifier). Proměnná q může nabývat hodnot 0 a 1. Je rovna 0, pokud přenášíme pouze jedno kódové slovo v subrámcí. Proměnná n_s představuje číslo slotu v rádiovém rámcí a N_{ID}^{cell} je identifikační číslo sektoru.

Celé zapojení generátoru PN posloupnosti je zobrazeno na obrázku 1.15. Zapojení je zkonstruováno za pomoci posuvných registrů.

V modelu proces skramblování vykonává funkce $[b_] = \text{scrambling}(e, M, c_init)$. Kde proměnná e je vstupní kódové slovo, M je počet bitů v jednom subrámcí a c_init



Obr. 1.15: Generování PN posloupnosti pomocí posuvných registrů (převzato z [4])

je inicializační číslo. Výstupní proměnnou je b , kterou je zakódovaná posloupnost. Funkce nejprve rozdělí kódová slova na části, které jsou přeneseny v jednom subrámcí. Ke každému subrámcí je pak vytvořena kódovací sekvence c . Jednotlivé bity v jednotlivých subrámcích jsou zakódovány podle vzorce 1.21. Kódovací PN posloupnost je vytvořena funkcí $c = \text{scramblingCode}(M_{pn}, c_{init})$, kde M_{pn} je délka kódovací posloupnosti a c_{init} je inicializační číslo. Výstupní proměnná c je kódovací posloupnost. Funkce počítá kódovací posloupnost za pomoci vztahů 1.22, 1.23 a 1.24.

1.8 Vnitřní modulace

Hlavním úkolem vnitřní modulace je zvýšit přenosovou rychlost systému za cenu snížení odolnosti signálu vůči rušení, šumům a obecnému poškození signálu v rádiovém prostředí. Snížení odolnosti signálu vůči rušení nemusí mít na přenos žádný negativní vliv, pouze se jím poskytuje výhoda vyšších přenosových rychlostí za dobrých rádiových podmínek a nižší přenosová rychlost v případě špatných rádiových podmínek. Tato závislost přenosové rychlosti na rádiových podmínkách je zjevná v novějších komunikačních systémech. V systémech první a druhé generace nebylo z počátku možné měnit modulaci vůbec. Změna kódového poměru byla umožněna až v druhé generaci s příchodem technologie GPRS (General packet radio service) a změna kódovacího poměru spolu se změnou modulace umožňovala technologie EDGE (Enhanced Data rates for GSM Evolution). Principem vnitřní modulace je přeměna bitové posloupnosti na posloupnost komplexních symbolů. Převod bitů na symboly se řídí převodní tabulkou.

Vstupní posloupnost tvoří skupina bitů $b^{(q)}(0), \dots, b^{(q)}(M_{bit}^q - 1)$, kde q je číslo kódového slova a proměnná M_{bit}^q udává počet vstupních bitů. Pro fyzický sdílený kanál je možné použít modulace QPSK (Quadrature phase-shift keying), 16QAM (Quadrature amplitude modulation) a 64QAM. Převod se provádí na základě převodních tabulek 1.9, A.1, A.2a A.3. V tabulce je uvedena skupina 2, 4 nebo 6 bitů,

která bude přeměněna na symbol. Posloupnost je převáděna postupně po skupinách 2, 4 nebo 6 bitů.

BPSK (Binary Phase Shift Keying)

BPSK modulace převádí jediný bit, $b(i)$, na komplexní symbol $x = I + jQ$. Symbol se převádí podle tabulky 1.8.

Tab. 1.8: Převodní tabulka pro modulaci BPSK

$b(i)$	I	Q
0	$1/\sqrt{2}$	$1/\sqrt{2}$
1	$-1/\sqrt{2}$	$-1/\sqrt{2}$

QPSK (Quadrature phase-shift keying)

QPSK modulace převádí dva bity, $b(i)$, $b(i + 1)$ na komplexní symbol $x = I + jQ$. Převod se provádí podle tabulky 1.9.

Tab. 1.9: Převodní tabulka pro modulaci QPSK

$b(i), b(i + 1)$	I	Q
00	$1/\sqrt{2}$	$1/\sqrt{2}$
01	$1/\sqrt{2}$	$-1/\sqrt{2}$
10	$-1/\sqrt{2}$	$1/\sqrt{2}$
11	$-1/\sqrt{2}$	$-1/\sqrt{2}$

16QAM (Quadrature amplitude modulation)

Modulace 16QAM převádí 4 bity, $b(i)$, $b(i + 1)$, $b(i + 2)$, $b(i + 3)$ na komplexní symbol $x = I + jQ$. Převod se provádí podle tabulky A.1.

64QAM (Quadrature amplitude modulation)

Modulace 64QAM převádí 6 bitů, $b(i)$, $b(i + 1)$, $b(i + 2)$, $b(i + 3)$, $b(i + 4)$, $b(i + 5)$ na komplexní symbol $x = I + jQ$. Převod se provádí podle tabulky A.2 a A.3.

V modelu simulátoru se provádí vnitřní modulace ve funkci $[d] = \text{modulace}(b___, \text{type})$, kde vstupní proměnná $b___$ obsahuje skramblované kódové slovo, parametr

type označuje typ modulace 'BPSK', 'QPSK', '16QAM', '64QAM'. Výstupem je posloupnost symbolů d . Funkce podle zadaného typu modulace nejprve vytvoří objekt `modem.pskmod`, pro BPSK a QPSK.

Objekt `modem.pskmod` vyžaduje zadání parametrů modulátoru. V modelu je modulátor vytvořen z následujícími parametry `modem.pskmod('M', 2, 'PhaseOffset', pi/4, 'SymbolOrder', 'user-defined', 'SymbolMapping', [0 1], 'InputType', 'bit')`. Číslo za parametrem 'M' označuje počet stavů, 'PhaseOffset' fázový posun, parametry 'SymbolOrder', 'user-defined', 'SymbolMapping' značí, že modulátor bude konstruován podle zadaného mapování symbolů na bity, které se liší od výchozího nastavení. Tuto úpravu je třeba provést z důvodu shody mapování symbolů modulátoru a předpisu, který je zadán standardem. Parametr 'InputType', 'bit' určuje, jak má být vnímána vstupní posloupnost. V našem případě jde o posloupnost bitů. Modulátor pro QPSK se liší pouze v parametru 'M', které se rovná 4, a vektoru mapování symbolů [0 2 3 1].

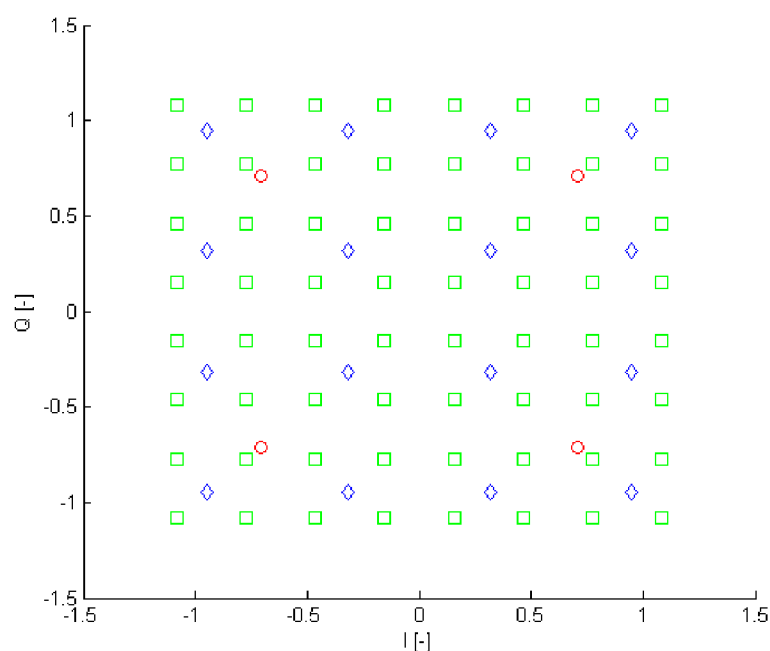
Modulace 16QAM se vytváří pomocí objektu `modem.qammod('M', 16, 'SymbolOrder', 'user-defined', 'SymbolMapping', [11 10 14 15 9 8 12 13 1 0 4 5 3 2 6 7], 'InputType', 'bit')`. Rozdíl je tedy jen v typu vytvářeného objektu, absenci fázového posunu, rozdílných hodnotách parametru 'M' a vektoru mapování symbolů. Modulace 64QAM se liší oproti 16QAM parametrem 'M' a vektorem mapování bitů na symboly. Objekt `modem.qammod` se vytváří s parametry 'M', 64, 'SymbolOrder', 'user-defined', 'SymbolMapping', [47 46 42 43 59 58 62 63 45 44 40 41 57 56 60 61 37 36 32 33 49 48 52 53 39 38 34 35 51 50 54 55 7 6 2 3 19 18 22 23 5 4 0 1 17 16 20 21 13 12 8 9 25 24 28 29 15 14 10 11 27 26 30 31], 'InputType', 'bit'.

Modulace se pak provádí pomocí funkce $y = \text{modulate}(\text{Modulator}, x)$, kde *Modulator* je vytvořený objekt pomocí funkce `modem.pskmod` nebo `modem.qammod` a x je vstupní binární posloupnost. Výstupem je y , posloupnost symbolů.

1.9 Mapování do vrstev

Pokud je systém navržen tak, že umožňuje přenos dat za pomoci více antén, je nutné rozdělit datový tok do více vrstev, kde každá vrstva představuje signál pro každou anténu. Blok Mapování do vrstev tedy definuje, jak budou kódová slova rozdělena do vrstev. Rozdělení mohou probíhat dvojího typu.

Prvním je „Spartial multiplexing“, kdy je systém MIMO používán pro zvýšení datové propustnosti systému. V tomto případě je využito i zpracovávání dvou kódových slov zároveň v jednom přenosu, kdy dvě kódová slova jsou mapovány do dvou a více vrstev.



Obr. 1.16: IQ diagram modulací QPSK, 16QAM a 64QAM

Druhým je „Transmit diversity“, kdy je dosahováno lepších přijímacích podmínek, protože výsledný symbol je sestaven z příspěvků signálu každé z přijímacích cest.

MIMO

Systémy využívající přenos za pomoci více antén tzv. MIMO (Multiple Input Multiple Output) vynikají vyšší spektrální účinností. Vlivem vícecestného šíření signálu je každá z cest postižena jinak. Signály lze při příjmu od sebe oddělit a získat tak více informace, nežli při přenosu s pouze jedinou anténou. Jinou možností je při příjmu zkombinovat přijatý signál a využít informaci z obou přijatých signálů k získání jediné vyslané informace.

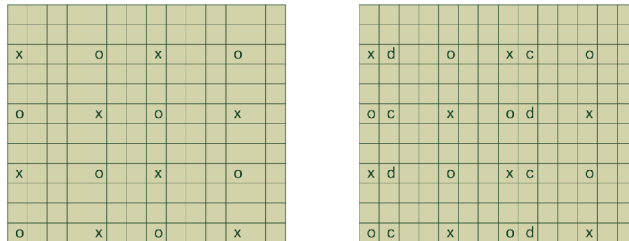
Práce se omezuje na zpracování jednoho kódového slova současně. Dalším zjednodušením je omezení se pouze na vysílání pomocí jediné antény. Toto omezení vede k zjednodušení procesu mapování do vrstev na pouhé přeznačení proměnných dle vztahu[1]:

$$x^{(0)}(i) = d^{(0)}(i) \quad \text{pro } i = 0, 1, \dots, M_{\text{sybm}}^{(0)} - 1 \quad (1.26)$$

a nijak neupravuje vstupní posloupnost symbolů.

V rovnici 1.26 je $d^{(0)}(i)$ posloupnost symbolů vystupující z modulátoru, $x^{(0)}(i)$ je posloupnost vystupující z bloku Mapování do vrstev a $M_{symb}^{(0)}$ je počet symbolů. Dále platí $M_{symb}^{(0)} = M_{symb}^{layer}$, tedy počet symbolů vstupující do bloku Mapování do vrstev je stejný, jako počet symbolů vystupujících z tohoto bloku.

Na použitém počtu vysílajících antén je závislý počet referenčních signálů umístěných v subrámcí. Umístění referenčních signálů je demonstrováno pomocí obrázku 1.17. Symbolem 'o' jsou označeny referenční symboly mapované na port první antény, symbolem 'x' referenční symboly mapované na port druhé antény, symbolem 'c' referenční symboly mapované na port třetí antény a symbolem 'd' referenční symboly mapované na port čtvrté antény. Při mapování referenčních signálů na jednu anténu jsou místa, kam mají být mapovány referenční symboly ostatních antén nulové. Není v nich tedy přenášena žádná informace. Systém umožňuje přenos MIMO za pomoci dvou antén, kdy jsou používány pouze symboly označené 'x' a 'o'. Elementy 'c' a 'd' jsou vyplněny uživatelskými nebo kontrolními daty.



Obr. 1.17: Pozice referenčních symbolů v závislosti na použitém počtu antén pro vysílání, vlevo 2 antény, vpravo 4 antény (převzato z [4])

V simulovaném modelu je toto přeznačení vynecháno, na funkci modelu nemá žádný vliv.

1.10 Předkódování

Předkódování souvisí se sekci 1.9, kde jsou jedno nebo dvě kódová slova rozdělena do vrstev. V bloku Předkódování jsou symboly v jednotlivých vrstvách od sebe odlišeny tak, aby se při vícecestném šíření co nejméně negativně ovlivňovaly.

Blok Předkódování je vlivem zjednodušení zmíněného v předchozí sekci, tedy omezení práce na vysílání signálu pouze pomocí jediné antény, značně zjednodušen. Probíhá zde obdobná operace přeznačení jako v bloku Mapování do vrstev. Operaci popisuje rovnice[1]:

$$y^{(p)}(i) = x^{(0)}(i) \tag{1.27}$$

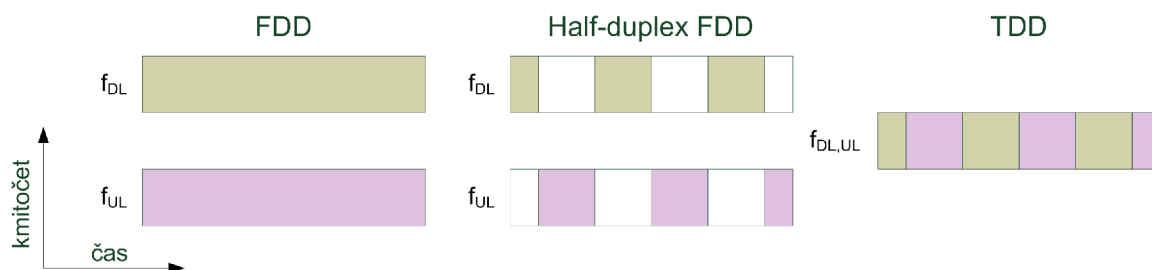
kde p označuje port antény a může nabývat pouze hodnot 0, 4 a 5. $x^{(0)}(i)$ je posloupnost po zpracování blokem Mapování do vrstev a $y^{(p)}(i)$ je výstupní posloupnost bloku Předkódování. Platí $M_{symb}^{(ap)} = M_{symb}^{layer}$, kde $M_{symb}^{(ap)}$ je počet symbolů vystupující z bloku Předkódování.

V simulovaném modelu je toto přeznačení vynecháno, na funkci modelu nemá žádný vliv.

1.11 Mapování elementů

Blok mapování elementů představuje předpis, jakým se budou do struktury radiového rámce vkládat symboly, představující jak uživatelská data, tak i kontrolní data a pomocné signály.

Systémy mobilních komunikací jsou vždy duplexní, což představuje fakt, že existují dva kanály, jeden pro uplink a druhý pro downlink. Tyto kanály mohou být odděleny ve frekvenci nebo v čase. Oddělení ve frekvenci je označováno jako FDD (Frequency Division Duplex). Oddělení v čase jako TDD (Time Division Duplex). Oddělením ve frekvenci je myšleno vysílání signálu směrem k uživateli na jiném nosném kmitočtu než vysílání signálu od uživatele k základnové stanici. Dále je možné použít nepřerušované, stálé, vysílání, tzv. Full-duplex nebo lze použít Half-duplex, kdy je v jednom časovém okamžiku signál vysílán nebo přijímán. Oddělení obou kanálů v čase znamená, že v jednom čase je vysíláno, v dalším časovém úseku je přijímáno. Není tedy možné přijímat a vysílat v jednom čase. Všechny tři varianty jsou zobrazeny na obrázku 1.18.



Obr. 1.18: Rozlišení downlinku a uplinku ve frekvenci a v čase

Oddělení kanálu v čase je výhodné zejména pro lepší odhad kvality kanálu, protože se přijímá i vysílá v jediném pásmu. Referenční signály jsou přijímány spolu s užitečnými daty a kontrolními informacemi. Jelikož rozdíl času mezi přijímáním a vysíláním není veliký, lze předpokládat, že přenosový kanál je téměř nezměněný. Další velkou výhodou je možnost asymetrického rozdělení počtu slotů pro přijímání a vysílání.

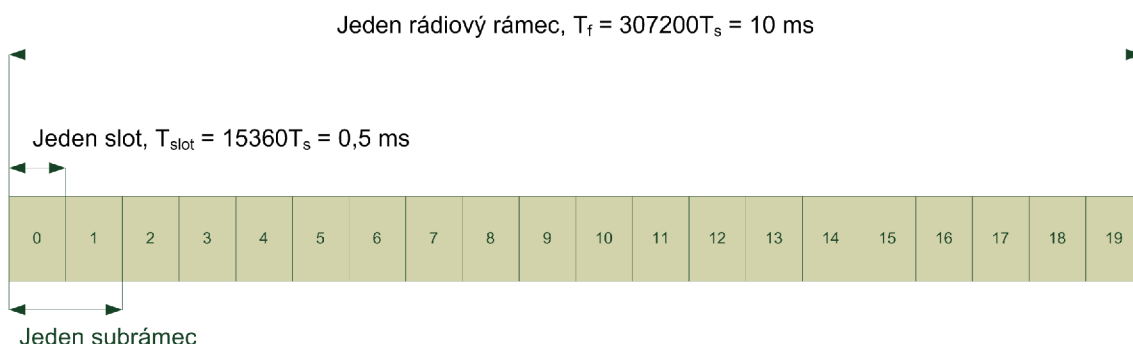
1.11.1 Mapování elementů v LTE

Tato sekce popisuje způsob, jakým jsou symboly, představující zakódovanou vstupní informaci, vkládány do časově frekvenční struktury rádiového rámce systému LTE. Mimo uživatelská data jsou ve struktuře rádiového rámce přenášeny kontrolní informace a další signály nutné ke správnému fungování systému.

1.11.2 Rádiový rámeček

Délka rádiového rámce pro downlink i uplink je 10ms, přičemž základní jednotkou času je $T_s = 1/(15000 \times 2048)$ sekundy. Velikost rádiového rámce je $T_f = 307200 \times T_s = 10ms$. Systém používá dva typy rámečků, typ 1 je využit pro FDD, a typ 2 pro TDD.

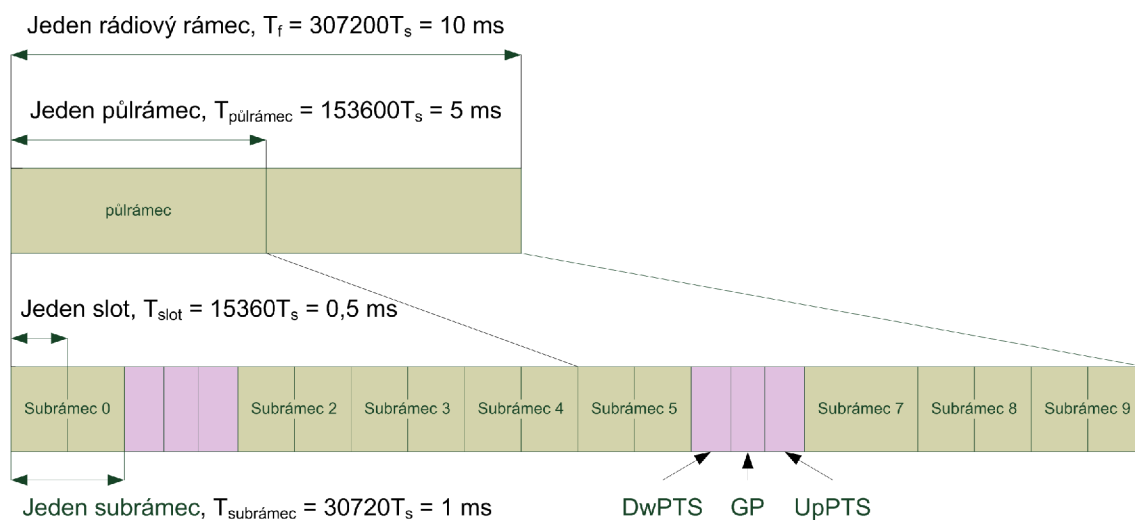
Rámeček typu 1 se skládá z 20 slotů, každý slot trvá $T_{slot} = 15360 \times T_s = 0,5ms$, číslovány jsou od 0 do 19. Subrameček je objekt skládající se ze dvou sousedících slotů, lze jej definovat jako dvojici slotů $2i$ a $2i + 1$, kde $i = 0, 1, \dots, 9$ a současně je i číslo subramečku. Trvání subramečku je tak $T_{subrameček} = 2 \times T_{slot} = 1ms$. Graficky je tento rámeček zobrazen na obrázku 1.19. [1]



Obr. 1.19: Rádiový rámeček typ 1

Rámeček typu 2 je vyhrazen pro mód TDD. Délka rámce je stejná jako u typu 1, tedy $T_f = 307200 \times T_s = 10ms$, ten je nadále dělen na dva půlramečky, každý o délce $T_{půlrameček} = 153600 \times T_s = 5ms$. Půlramečky jsou dále děleny na pět subramečků, každý o délce $T_{subrameček} = 30720 \times T_s$. V rádiovém rámci jsou dva druhy subramečků, jeden normální dělený na dva sloty, trvání slotu je $T_{slot} = 15360 \times T_s = 0,5ms$. Druhý subrameček je speciální a skládá se z DwPTS, GP, UpPTS. Každý rádiový rámeček obsahuje alespoň jeden speciální subrameček. Ten je umístěný hned za prvním subramečkem. Systém má dva módy Switch-point periody, jedna je 10ms, druhá 5ms. První subrameček v periodě je určený pro downlink, druhý je speciální subrameček,

ihned za speciálním subrámcem následuje subrámec určený pro uplink. Ostatní typy subrámců jsou určeny schématem. Schémata jsou uvedena v tabulce 1.10.



Obr. 1.20: Rádioový rámeček typ 2, Switch-point perioda 5ms

Na obrázku 1.20 je zobrazen rádioový rámeček typu 2, který je určen pro technologii TDD. Zde je použita Switch-point perioda 5ms. Toto lze poznat podle dvou speciálních subrámců v jednom rádioovém rámečku.

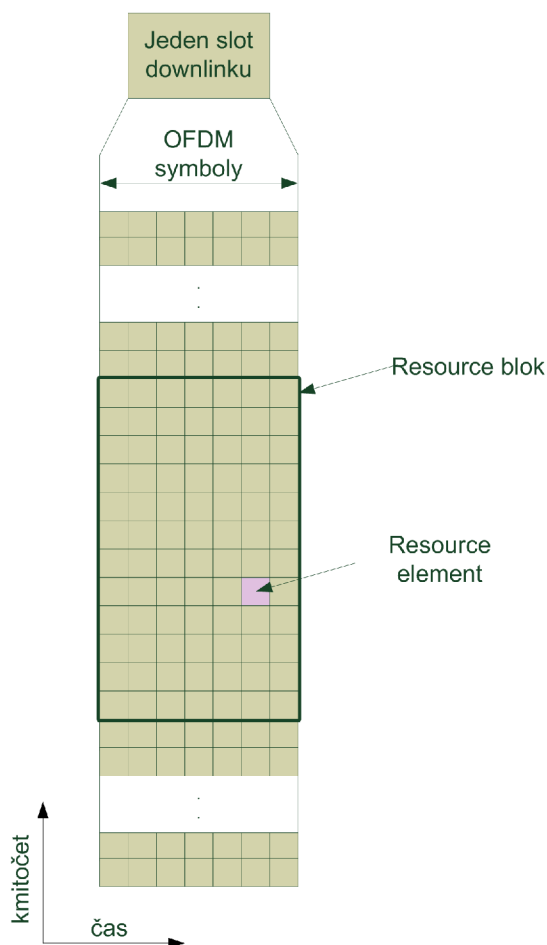
Tab. 1.10: Rozložení subrámců pro uplink a downlink

Konfigurace uplink-downlink	Switch-point perioda	Číslo subrámeče									
		0	1	2	3	4	5	6	7	8	9
0	5 ms	D	S	U	U	U	D	S	U	U	U
1	5 ms	D	S	U	U	D	D	S	U	U	D
2	5 ms	D	S	U	D	D	D	S	U	D	D
3	10 ms	D	S	U	U	U	D	D	D	D	D
4	10 ms	D	S	U	U	D	D	D	D	D	D
5	10 ms	D	S	U	D	D	D	D	D	D	D
6	5 ms	D	S	U	U	U	D	S	U	U	D

1.11.3 Resource element

Nejmenší nosnou jednotkou je resource element. Tento element je představován jednou subnosnou v jednom OFDM symbolu.

Na obrázku 1.21 je zobrazena časově-frekvenční struktura jednoho slotu rádioového rámečku. Při použití normálního cyklického prefixu má jeden slot celkem 7 OFDM



Obr. 1.21: Časověfrekvenční struktura downlinku

symbolů, každý resource blok má tedy také 7 OFDM symbolů v čase a 12 subnosných ve frekvenci. Při použití rozšířeného cyklického prefixu obsahuje jeden slot pouze 6 OFDM symbolů a 12 subnosných. Lze použít i konfiguraci, kdy je rozteč mezi subnosnými pouze $\Delta f = 7,5$ kHz, pak je resource blok představován 24 subnosnými a 3 OFDM symboly, použit je rozšířený cyklický prefix. Poslední kombinaci je možné použít jen pro MBSFN (Multicast Broadcast single frequency network).

Tab. 1.11: Parametry resource bloku

Konfigurace		N_{sc}^{RB}	N_{symp}^{DL}
Normální cyklický prefix	$\Delta f = 15$ kHz	12	7
Rozšířený cyklický prefix	$\Delta f = 15$ kHz		6
		$\Delta f = 7,5$ kHz	24

1.11.4 Šířka pásma v systému LTE

Systém LTE je navržen tak, aby byl schopen pracovat ve velkém rozsahu kmitočtů. Je definováno velké množství použitelných pásem. Přehled všech přenosových pásem je uvedený v tabulce 1.12.

Tab. 1.12: Přehled pásem použitelných pro systém LTE

E-UTRA Pásmo	Uplink (UL) BS přijíma UE vysílá	Downlink (DL) BS vysílá UE přijímá	Duplexní mód
	$f_{UL_low} - f_{UL_high}$	$f_{DL_low} - f_{DL_high}$	
1	1920 MHz – 1980 MHz	2110 MHz – 2170 MHz	FDD
2	1850 MHz – 1910 MHz	1930 MHz – 1990 MHz	FDD
3	1710 MHz – 1785 MHz	1805 MHz – 1880 MHz	FDD
4	1710 MHz – 1755 MHz	2110 MHz – 2155 MHz	FDD
5	824 MHz – 849 MHz	869 MHz – 894MHz	FDD
6	830 MHz – 840 MHz	875 MHz – 885 MHz	FDD
7	2500 MHz – 2570 MHz	2620 MHz – 2690 MHz	FDD
8	880 MHz – 915 MHz	925 MHz – 960 MHz	FDD
9	1749.9 MHz – 1784.9 MHz	1844.9 MHz – 1879.9 MHz	FDD
10	1710 MHz – 1770 MHz	2110 MHz – 2170 MHz	FDD
11	1427.9 MHz – 1447.9 MHz	1475.9 MHz – 1495.9 MHz	FDD
12	699 MHz – 716 MHz	729 MHz – 746 MHz	FDD
13	777 MHz – 787 MHz	746 MHz – 756 MHz	FDD
14	788 MHz – 798 MHz	758 MHz – 768 MHz	FDD
17	704 MHz – 716 MHz	734 MHz – 746 MHz	FDD
...			
33	1900 MHz – 1920 MHz	1900 MHz – 1920 MHz	TDD
34	2010 MHz – 2025 MHz	2010 MHz – 2025 MHz	TDD
35	1850 MHz – 1910 MHz	1850 MHz – 1910 MHz	TDD
36	1930 MHz – 1990 MHz	1930 MHz – 1990 MHz	TDD
37	1910 MHz – 1930 MHz	1910 MHz – 1930 MHz	TDD
38	2570 MHz – 2620 MHz	2570 MHz – 2620 MHz	TDD
39	1880 MHz – 1920 MHz	1880 MHz – 1920 MHz	TDD
40	2300 MHz – 2400 MHz	2300 MHz – 2400 MHz	TDD

Jak je zmíněno výše, systém LTE dovoluje volit šířku pásma, ve které systém pracuje. S ohledem na režii systému je vhodné používat největší šířky pásma, kde je režie nejnižší. Ovšem v určitých případech je nutné volit menší šířky pásma, důvody

mohou být různé. Všechny možné šířky pásma jsou popsány v tabulce 1.13. Ovšem není možné ve všech frekvenčních pásmech používat všechny šířky pásma. Povolené šířky pásma k jednotlivým frekvenčním pásmům jsou uvedeny v literatuře [1].

Šířka pásma je v systému LTE často vyjádřena počtem resource bloků. Vzájemný převod je uveden v tabulce 1.13.

Tab. 1.13: Přehled možných šířek pásma a korespondující počet RB

Šířka pásma B [MHz]	1,4	3	5	10	15	20
Počet resource bloků N_{RB}	6	15	25	50	75	100

Důvodem proč je režie největší v nejužším pásmu je fakt, že jen v šířce nejmenšího pásma, tedy 1,4 MHz je vysílána primární a sekundární synchronizace, dále je pak v této šířce pásma vysílán i broadcastový kanál. Pokud je šířka pásma větší jak nejmenší, tak na ostatních subnosných mimo toto pásmo může být vysílána informace.

1.11.5 Referenční symboly

Referenční symboly jsou v systému důležité zejména z důvodu odhadu kvality kanálu. Pomocí referenčních symbolů je měřena velikost šumu a míra poškození signálu. Dále je ze znalosti jejich velikosti je možné provádět korekce přijímaného signálu, tím je do značné míry potlačen vliv šumu a rušení.

Referenční symboly specifikující buňku

Referenční symboly mohou být vysílány pomocí jedné nebo více antén. Jsou definovány pouze pro $\Delta f = 15$ kHz.

Sekvence referenčních symbolů $r_{l,n_s}(m)$ je definována vztahem[1]:

$$r_{l,n_s}(m) = \frac{1}{\sqrt{2}} (1 - 2 \cdot c(2m)) + j \frac{1}{\sqrt{2}} (1 - 2 \cdot c(2m)), \quad (1.28)$$

$$m = 0, 1, \dots, 2N_{RB}^{max,DL} - 1$$

kde n_s je číslo slotu v rádiovém rámci a l je číslo OFDM symbolu v daném slotu. Pseudonáhodná sekvence $c(i)$ je definovaná v sekci 1.7. Pseudonáhodný generátor je inicializován číslem $c_{init} = 2^{10} \cdot (7 \cdot (n_s + 1) + l + 1) \cdot (2 \cdot N_{ID}^{cell}) + 2 \cdot N_{ID}^{cell} + N_{CP}$ v každém OFDM symbolu. $N_{CP} = 1$ při použití normálního cyklického prefixu, $N_{CP} = 0$ při použití rozšířeného cyklického prefixu.

Mapování referenčních symbolů specifikující buňku

Sekvence referenčních symbolů $r_{l,n_s}(m)$ je vkládána do časově-frekvenční struktury na pozici $a_{k,l}^p$, kde p je port antény. Mapování probíhá podle vztahu[1]:

$$a_{k,l}^p = r_{l,n_s}(m'), \quad (1.29)$$

kde

$$\begin{aligned} k &= 6m + (v + v_{shift}) \bmod 6 \\ l &= \begin{cases} 0, N_{symb}^{DL} - 3 & \text{když } p \in \{0, 1\} \\ 1 & \text{když } p \in \{2, 3\} \end{cases} \\ m &= 0, 1, \dots, 2 \cdot N_{RB}^{DL} - 1 \\ m' &= m + N_{RB}^{max,DL} - N_{RB}^{DL} \end{aligned}$$

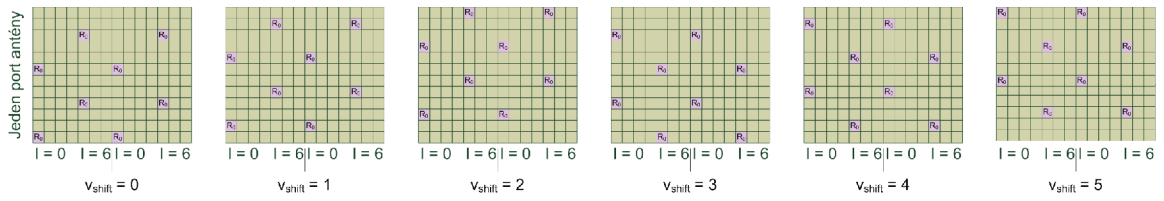
proměnné v a v_{shift} definují pozici referenčních symbolů ve frekvenční oblasti, tyto symboly jsou posunuty v závislosti na vysílací anténě a na N_{ID}^{cell} . $N_{RB}^{max,DL}$ je maximální počet resource bloků a N_{RB}^{DL} je počet resource bloků, které mohou být použity pro vysílání. Tedy počet resource bloků odpovídající použité šířce pásma.

Proměnná v je definována[1]:

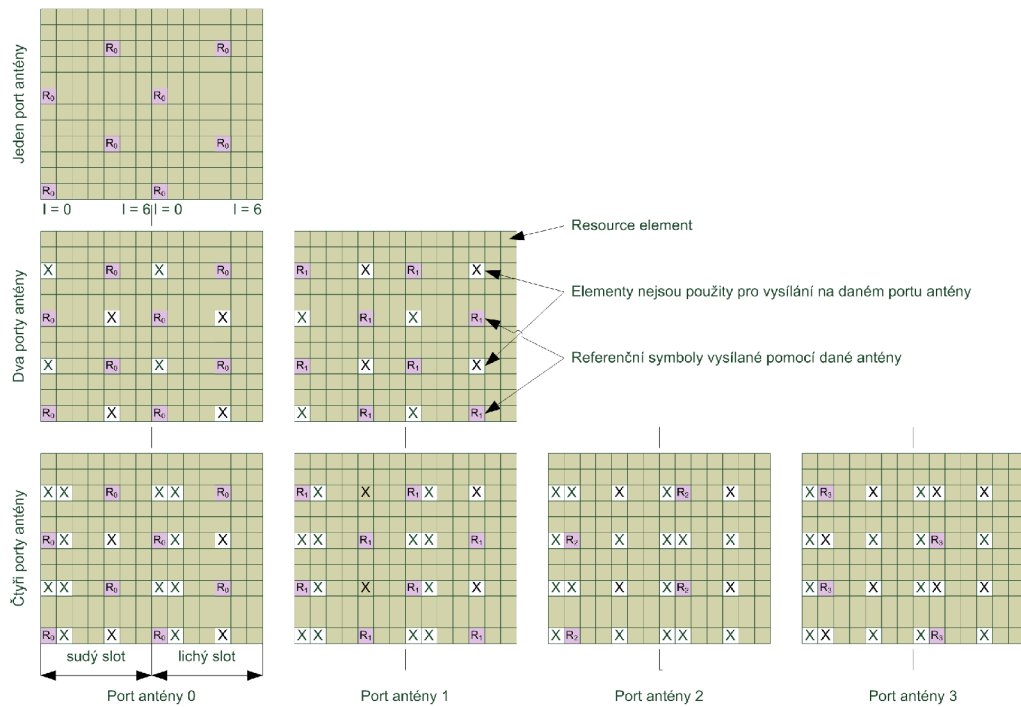
$$v = \begin{cases} 0 & \text{když } p = 0 \text{ a } l = 0 \\ 3 & \text{když } p = 0 \text{ a } l \neq 0 \\ 3 & \text{když } p = 1 \text{ a } l = 0 \\ 0 & \text{když } p = 0 \text{ a } l \neq 0 \\ 3(n_s \bmod 2) & \text{když } p = 2 \\ 3 + 3(n_s \bmod 2) & \text{když } p = 3 \end{cases}$$

Specifický posun ve frekvenci v_{shift} je dán vztahem $v_{shift} = N_{ID}^{cell} \bmod 6$. Znázornění posunu referenčních symbolů je zobrazeno na obrázku 1.22. Je zde zobrazen posun referenčních symbolů, při vysílání za pomoci jedné antény a při použití normálního cyklického prefixu.

Resource elements, které jsou použity pro vysílání referenčních symbolů na kterékoli anténě nesmí být použity pro vysílání na kterékoli jiné anténě ve stejném slotu a musí být nastaveny na nulu. Přehledně jsou tyto referenční symboly zobrazeny na obrázku 1.23, při použití normálního cyklického prefixu. Pokud je použit rozšířený cyklický prefix jsou referenční symboly mapovány podle obrázku 1.24.



Obr. 1.22: Posun referenčních symbolů v závislosti na v_{shift} , jeden port antény, normální cyklický prefix

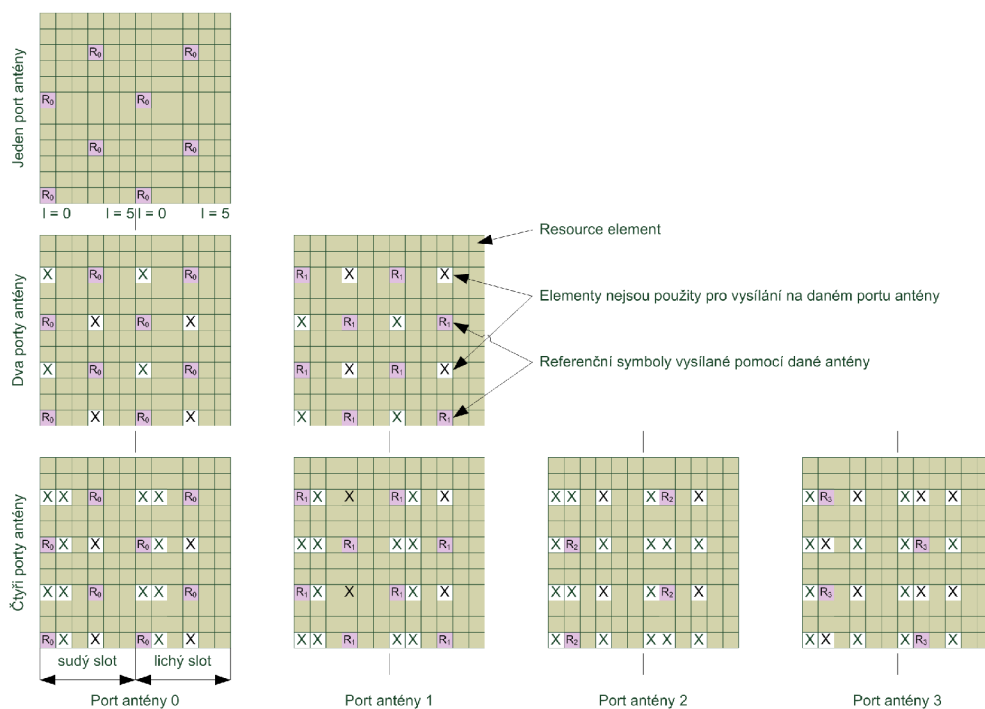


Obr. 1.23: Mapování referenčních symbolů, normální cyklický prefix

Referenční symboly pro MBSFN

Mimo referenční symboly specifikující buňku existují ještě referenční symboly pro MBSFN (Multicast Broadcast single frequency network). Tyto symboly se používají pouze při přenosu MBSFN a jsou vysílány na anténním portu číslo 4. Pro vysílání se používá pouze rozšířený cyklický prefix. Lze použít rozteč subnosných $\Delta f = 15$ kHz nebo $\Delta f = 7,5$ kHz. Více informací lze nalézt v literatuře [1].

V modelu se nepoužívají referenční symboly pro MBSFN.



Obr. 1.24: Mapování referenčních symbolů, rozšířený cyklický prefix

Referenční symboly specifikující UE

Posledním typem referenčních symbolů jsou symboly specifikující UE (User Equipment). Tyto symboly jsou vysílány pomocí jedné antény, přesněji pomocí anténního portu číslo 5. Lze použít normální i rozšířený cyklický prefix.

V modelu systému se nepoužívají referenční signály specifikující UE.

1.11.6 Synchronizační signály

Pomocí synchronizačních signálů je provedena korekce vnitřního oscilátoru přijímače, který musí být co nejpřesněji sladěn s kmitočtem vysílače. V systému používající OFDM modulaci je tato synchronizace velmi důležitá, protože ortogonality je dosaženo jen na přesně daných kmitočtech. Každá ze subnosných má své maximum a v tomto maximum jsou všechny ostatní subnosné nulové. Ovšem pokud systémy nejsou synchronizovány a kmitočet subnosné je odlišný od kmitočtu, který je měřen, pak je informace o přijímaném symbolu zkreslena ostatními subnosnými, ze kterých největší vliv na zkreslení mají nejbližší okolní subnosné. Dále je pomocí synchronizace zjišťováno číslo buňky, které obsahuje informaci o pozici referenčních symbolů. A také lze poznat číslo subrámcce, ve kterém je synchronizační signál přijat.

Primární synchronizace

Slouží k detekování začátku nultého a pátého subrámcce, přesněji nultého a desátého slotu, přičemž nelze za pomoci toho signálu rozlišit, který je nultý a který je desátý. O rozlišení konkrétního slotu se stará sekundární synchronizační signál. Primární synchronizační signál je generován na základě Zadoff-Chu sekvence, která má velmi dobrou autokorelační charakteristiku. Lze tak dobře detekovat signály ve spektru a podle nich synchronizovat oscilátor v přijímači.

Primární synchronizace v LTE

Synchronizační sekvence $d(n)$ je generována pomocí Zadoff-Chu sekvence, ta je ve frekvenční oblasti definována podle rovnice[1]:

$$d_u(n) = \begin{cases} e^{-j\frac{\pi un(n+1)}{63}} & n = 0, 1, \dots, 30 \\ e^{-j\frac{\pi u(n+1)(n+2)}{63}} & n = 31, 32, \dots, 61 \end{cases}, \quad (1.30)$$

kde u je Zadoff-Chu index kořene, definovaný v tabulce 1.14.

Tab. 1.14: Kořenové indexy Zadoff-Chu sekvencí pro primární synchronizaci

$N_{ID}^{(2)}$	Index kořene u
0	25
1	29
2	34

Mapování primárních synchronizačních symbolů

Mapování sekvence symbolů záleží na struktuře rádiového rámcce. Uživatelské zařízení nemůže rozeznat, zda signál vysílaný na určitém anténním portu patří k vysílané primární synchronizaci. Proto je primární synchronizace shodná pro všechny porty antény a je vysílána na všech anténních portech, kde jsou zároveň vysílány i referenční signály.

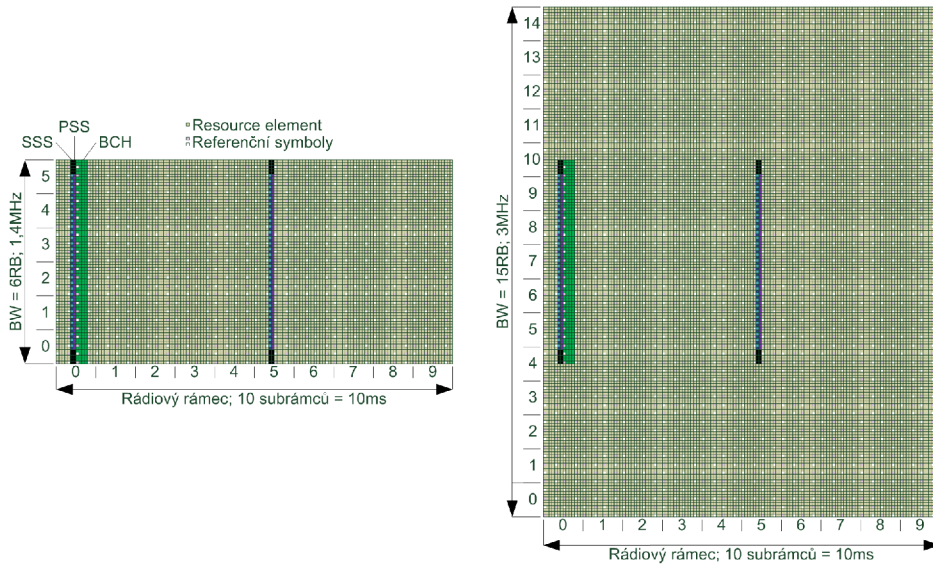
Sekvence $d(n)$ je mapována podle rovnice 1.32, kde N_{RB}^{DL} je počet resource bloků určených pro vysílání na downlinku a N_{sc}^{RB} je počet subnosných v jednom resource bloku. Proměnná l značí číslo OFDM symbolu. Ta záleží na cyklickém prefixu a typu struktury rámcce. Pokud se používá typ struktury 1, tedy struktura pro FDD a pokud je použit normální cyklický prefix, pak $l = 6$. Synchronizace je vysílána v posledním OFDM symbolu, ale pouze ve slotu číslo 0 a 10. Pokud je použit rozšířený cyklický prefix, pak $l = 5$ a opět se vysílá pouze ve slotu číslo 0 a 10. Pro strukturu rámcce

typu 2 (TDD) je $l = 2$, jedná se tedy o třetí OFDM symbol a vysílá se pouze ve slotu 2 a 12. Rovnice popisující mapování primárních synchronizačních symbolů kolem středu pásma je[1]:

$$a_{k,l} = d(n), \quad n = 0, \dots, 61 \quad (1.31)$$

$$k = n - 30 + \frac{N_{RB}^{DL}}{N_{sc}^{RB}}.$$

Příklad mapování primární synchronizace je zobrazen na obrázku 1.25.



Obr. 1.25: Mapování synchronizačních signálů do struktury rádiového rámečku, normální cyklický prefix, struktura rámečku typ 1

Na obou koncích primární synchronizační sekvence je pět rezervovaných symbolů, které nejsou určeny k vysílání primární synchronizace. Pozice těchto rezervovaných symbolů je popisována rovnicí[1]:

$$k = n - 30 + \frac{N_{RB}^{DL}}{N_{sc}^{RB}}$$

$$n = -5, -4, \dots, -1, 62, 63, \dots, 66. \quad (1.32)$$

Na obrázku 1.25 jsou tyto rezervované symboly značeny černě. Právě těchto 10 symbolů doplňuje sekvenci na rovných 72 symbolů, tedy 72 subnosných, které reprezentují jeden OFDM symbol při nejnižší šířce pásma, $1,4\text{MHz}$. Na obrázku je dále ukázáno, že při větší šířce pásma se referenční symboly neprodlužují a jsou vysílány pouze v oblasti široké 6 resource bloků.

Sekundární synchronizace

Sekundární synchronizaci tvoří 62 symbolů značených $d(0), \dots, d(61)$, tato sekvence je rozdělena do dvou navzájem se prolínajících binárních sekvencí. Každá má délku 31 symbolů a obě jsou různě definované pro vysílání v subrámcích číslo 0 a 5. Díky tomuto různému definování lze rozlišit subrámeček 0 a 5 od sebe. Navíc lze ze sekvence získat N_{ID}^{cell} , které je nutné k deskramblování signálů a ke zjištění pozice referenčních symbolů v časově frekvenční struktuře.

Sekvence je definována podle rovnic[1]:

$$d(2n) = \begin{cases} s_0^{(m_0)}(n)c_0(n) & \text{v subrámcích číslo 0} \\ s_1^{(m_1)}(n)c_0(n) & \text{v subrámcích číslo 5} \end{cases} \quad (1.33)$$

a

$$d(2n+1) = \begin{cases} s_1^{(m_1)}(n)c_1(n)z_1^{(m_0)}(n) & \text{v subrámcích číslo 0} \\ s_0^{(m_0)}(n)c_1(n)z_1^{(m_1)}(n) & \text{v subrámcích číslo 5} \end{cases}, \quad (1.34)$$

kde $n = 0, 1, \dots, 30$ a m_0 a m_1 je odvozeno od identifikace buňky $N_{ID}^{(1)}$ a jsou definovány rovnicí[1]:

$$m_0 = m' \bmod 31$$

$$m_1 = (m_0 + \lfloor m'/31 \rfloor + 1) \bmod 31$$

$$m' = N_{ID}^{(1)} + q(q+1)/2, q = \lfloor \frac{N_{ID}^{(1)} + q'(q'+1)/2}{30} \rfloor, q' = \lfloor N_{ID}^{(1)}/30 \rfloor. \quad (1.35)$$

Výsledná závislost $N_{ID}^{(1)}$ a m_0, m_1 je uvedena v tabulce A.6.

Sekvence $s_0^{(m_0)}(n)$ a $s_1^{(m_1)}(n)$ jsou definovány jako dvě cyklicky posunuté m -sekvence $\tilde{s}(n)$. Definice sekvencí je uvedena za pomoci rovnic[1]:

$$s_0^{(m_0)}(n) = \tilde{s}((n + m_0) \bmod 31) \quad (1.36)$$

a

$$s_1^{(m_1)}(n) = \tilde{s}((n + m_1) \bmod 31) \quad (1.37)$$

kde $\tilde{s}(i) = 1 - 2x(i)$, $i = 0, 1, \dots, 30$ a sekvence x je definována rovnicí[1]:

$$x(i + 5) = (x(i + 2) + x(i)) \bmod 2 \quad , \text{kde } i = 0, 1, \dots, 25. \quad (1.38)$$

Počáteční stav je $x(0) = 0, x(1) = 0, x(2) = 0, x(3) = 0, x(4) = 1$.

Obdobným způsobem jsou definovány sekvence $c_0(n)$ a $c_1(n)$, ty jsou odvozeny od $N_{ID}^{(2)}$. Jsou tvořeny jako m -sekvence $\tilde{c}(n)$ s cyklickým posunutím. Definice je uvedena v rovnicích[1]:

$$c_0(n) = \tilde{c}((n + N_{ID}^{(2)}) \bmod 31) \quad (1.39)$$

a

$$c_1(n) = \tilde{c}((n + N_{ID}^{(2)} + 3) \bmod 31), \quad (1.40)$$

kde $\tilde{c}(i) = 1 - 2x(i)$, $i = 0, 1, \dots, 30$ a sekvence x je definována rovnicí[1]:

$$x(i + 5) = (x(i + 3) + x(i)) \bmod 2 \quad , \text{kde } i = 0, 1, \dots, 25. \quad (1.41)$$

Počáteční stav je $x(0) = 0, x(1) = 0, x(2) = 0, x(3) = 0, x(4) = 1$.

Poslední potřebné sekvence jsou $z_1^{(m_0)}(n)$ a $z_0^{(m_1)}(n)$, které jsou definované jako cyklicky posunutá m -sekvence $\tilde{z}(n)$, definice uvedena rovnicemi[1]:

$$z_1^{(m_0)}(n) = \tilde{z}((n + (m_0 \bmod 8)) \bmod 31) \quad (1.42)$$

a

$$z_0^{(m_1)}(n) = \tilde{z}((n + (m_1 \bmod 8)) \bmod 31), \quad (1.43)$$

kde $\tilde{z}(i) = 1 - 2x(i)$, $i = 0, 1, \dots, 30$ a sekvence x je definována rovnicí[1]:

$$x(i + 5) = (x(i + 4) + x(i + 2) + x(i + 1) + x(i)) \bmod 2 \quad , \text{kde } i = 0, 1, \dots, 25.$$

Počáteční stav je $x(0) = 0, x(1) = 0, x(2) = 0, x(3) = 0, x(4) = 1$.

Mapování symbolů sekundární synchronizace

Sekvence $d(n)$ je mapována podle rovnice 1.45, kde N_{RB}^{DL} je počet resource bloků určených pro vysílání na downlinku a N_{sc}^{RB} je počet subnosných v jednom resource bloku. Následující rovnice popisuje, že sekundární synchronizační symboly jsou mapovány kolem středu pásma, stejně jako primární synchronizační symboly[1]:

$$a_{k,l} = d(n), \quad n = 0, \dots, 61 \quad (1.44)$$

$$k = n - 30 + \frac{N_{RB}^{DL}}{N_{sc}^{RB}}$$

$$l = \begin{cases} N_{symp}^{DL} - 2 & \text{ve slotu 0 a 10, platí pro strukturu rámce typu 1} \\ N_{symp}^{DL} - 1 & \text{ve slotu 1 a 11, platí pro strukturu rámce typu 2} \end{cases} \quad (1.45)$$

Příklad mapování sekundární synchronizace je zobrazen na obrázku 1.25.

Na obou koncích sekundární synchronizační sekvence je pět rezervovaných symbolů, které nejsou určeny k vysílání sekundární synchronizace. Pozice těchto rezervovaných symbolů je popisována rovnicí[1]:

$$k = n - 30 + \frac{N_{RB}^{DL}}{N_{sc}^{RB}}$$

$$n = -5, -4, \dots, -1, 62, 63, \dots, 66 \quad (1.46)$$

$$l = \begin{cases} N_{symp}^{DL} - 2 & \text{ve slotu 0 a 10, platí pro strukturu rámce typu 1} \\ N_{symp}^{DL} - 1 & \text{ve slotu 1 a 11, platí pro strukturu rámce typu 2} \end{cases}$$

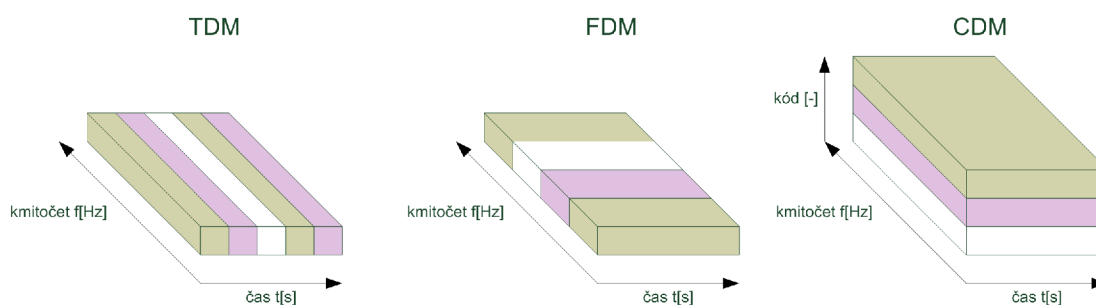
Na obrázku 1.25 jsou tyto rezervované symboly značeny černě. Právě těchto 10 symbolů doplňuje sekvenci na rovných 72 symbolů, tedy 72 subnosných, které reprezentují jeden OFDM symbol při nejnižší šířce pásma, $BW = 1,4MHz$. Na obrázku je dále ukázáno, že při větší šířce pásma se referenční symboly neprodlužují a jsou vysílány pouze v oblasti široké 6 resource bloků.

V simulovaném modelu je časověfrekvenční struktura vytvářena pomocí funkce $[D \ DMAP] = \text{mapper_dl}(d, SchMAX, dSchVec, nCellID)$. kde D je výsledná časově frekvenční struktura rádiového rámce, proměnná $DMAP$ v sobě uchovává informaci o tom, do kterých resource elementů jsou mapovány uživatelská data, kde jsou umístěny referenční symboly a další informace. Vstupem funkce je pak d , což je

posloupnost symbolů vytvořená v bloku vnitřní modulace. $SchMAX$ obsahuje informaci o maximálním počtu využitelných resource bloků, $dSchVec$ je vektor, který obsahuje informaci, do kterých resource bloků budou mapovány uživatelská data a které mají být vynechány. Proměnná $nCellID$ představuje číslo sektoru. Struktura $mapper$ není již využívána, dříve sloužila jako informace o tom, které z druhů zobrazení struktury rádiového rámce se má zobrazit při simulaci. K vytvoření subrámce se používá funkce $[B \ BMAP] = ADD_subframe_downlink(A, \ AMAP, \ nRB, \ ns, \ nCellID, \ RS, \ PSS, \ SSS, \ dispFrame)$, která vytvoří subrámec a podle proměnných RS , PSS a SSS vloží do subrámce referenční symboly ($RS=true$), primární synchronizaci ($PSS=true$) a sekundární synchronizaci ($SSS=true$). RS , PSS a RSS jsou tedy proměnné typu boolean. Je pomocí nich ovládáno, zdali budou dané symboly vloženy do rádiového rámce. A je proměnná uchovávající již vytvořené subrámce, ke kterým je nový subrámec přidán a vznikne tak proměnná B . $AMAP$ a $BMAP$ mají stejný význam jako $DMAP$, popisují jaké data jsou do rádiové struktury vloženy. nRB obsahuje informaci o používaném počtu resource bloků a ns je číslo slotu. Proměnná $nCellID$ má stejný význam jako v předchozí funkci $mapper_dl$.

1.12 OFDM modulace

Komunikační systémy zjišťují komunikaci více uživatelů současně. Tito uživatelé však musí být nějakým způsobem odděleni od sebe, pokud by se jejich signály překrývaly, vzájemně by se rušily a jejich komunikace by nebyla možná. Základní možností jak uživatele oddělit je multiplexace, tedy vzájemné střídání dvou nebo více uživatelů v používání systému. Za základní typy multiplexu jsou považovány tři typy TDM (Time Division Multiplex), FDM (Frequency Division Multiplex) a CDM (Code Division Multiplex).



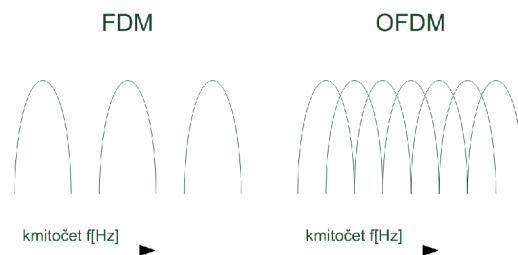
Obr. 1.26: Ukázka typů multiplexů

Základním principem u TDM je oddělení uživatelů v čase, to znamená, že uživatelé sdílí celé kmitočtové pásmo, ale průběžně se střídají ve vysílání. Toto přepínání

nemusí účastníci postřehnout. V digitálních systémech je možné toto přepínání realizovat velmi rychle, například v řádu milisekund. Pak si jednotliví účastníci po dobu kdy nevysílají, ukládají data do paměti a celou potřebnou informaci pak odešlou pouze v okamžik, kdy jsou na řadě. Pokud je toto přepínání dostatečně rychlé, účastníci hovoru nepoznají, že jsou přerušováni a zpoždění nevnímají. Oddělení uživatelů je ilustrováno na obrázku 1.26.

Mezi nejnovější typ multiplexu patří CDM, tento typ oddělení je nejhůře představitelný, protože jednotliví uživatelé sdílí celý kmitočtový i časový rozsah a jsou od sebe odděleni kódovou posloupností. Tato kódová posloupnost musí mít jisté vlastnosti, díky nimž lze jednotlivé sekvence dat od sebe po příjmu oddělit. K rozlišení uživatelů se například v systému UMTS používají Walshovy posloupnosti.

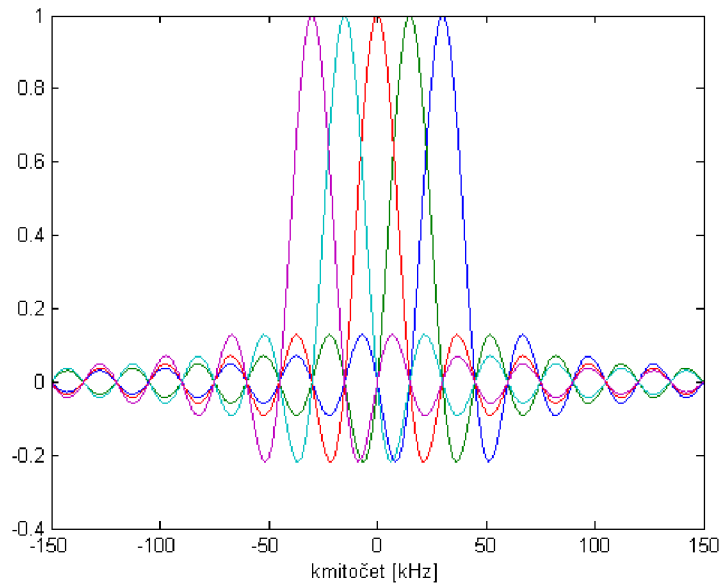
Oddělení uživatelů v kmitočtové oblasti lze realizovat dvěma základními způsoby. První je oddělení jednotlivých kanálů s přidáním ochranné mezery mezi jednotlivými kanály, každý kanál je reprezentován nosným kmitočtem. Druhý novější způsob využívá ortogonalitu a jednotlivé subnosné se tak mohou částečně překrývat, aniž by byla ztracena jakákoli informace. Díky tomuto multiplexu je systém efektivnější a má vyšší spektrální účinnost.



Obr. 1.27: Rozdíl mezi FDM a OFDM

Základním principem OFDM je použití ortogonálních signálů, které se navzájem neovlivňují. Pro ilustraci slouží obrázek 1.28, kde jednotlivé signály jsou odlišeny rozdílnou barvou. Ortogonalita je zaručena pouze v maximu signálu, kde jsou ostatní signály nulové. Rozestup nosných je na obrázku 100 Hz, systém LTE používá rozestup 15 kHz a 7,5 kHz. Takto použité signály zvětšují efektivitu využití spektra, protože není nutné dodržovat ochranné intervaly mezi jednotlivými subnosnými. Subnosné však musí být v přesně definované vzdálenosti.

Princip vytvoření OFDM symbolu je znázorněn na obrázku 1.29. Symboly jsou nejdříve převedeny ze sériového toku na paralelní a pomocí IFFT (Inverse Fast Fourier Transform) je tento signál z kmitočtové oblasti převeden do časové. Tímto je vytvořen OFDM symbol. Každý symbol OFDM má délku $2048 T_s$.



Obr. 1.28: Ortogonální signály použité u modulace OFDM

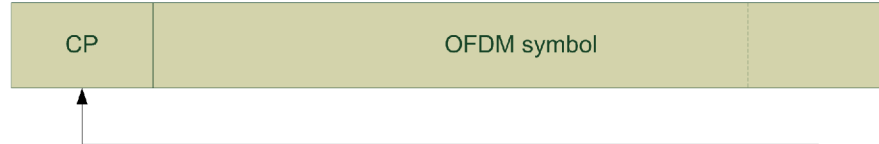


Obr. 1.29: Princip modulace OFDM

K OFDM symbolu je přidán cyklický prefix. Ten je vytvořen tak, že část konce symbolu je přidána před začátek OFDM symbolu. Cyklický prefix nahrazuje funkci ochranného intervalu, který je v systému kvůli eliminaci vlivů vícecestného šíření. Další funkcí cyklického prefixu je minimalizace mezisymbolových interferencí. V LTE je normální cyklický prefix dlouhý $160 T_s$, pro první OFDM symbol a $144 T_s$, pro druhý až sedmý symbol. Všechny délky cyklického prefixu jsou popsány v tabulce 1.15. Přidání cyklického prefixu je zobrazeno na obrázku 1.30.

Tab. 1.15: Definované délky cyklického prefixu

		Délka cyklického prefixu $N_{CP,l}$
Normální cyklický prefix	$\Delta f = 15kHz$	160 pro $l = 0$ 144 pro $l = 1, 2, \dots, 6$
	$\Delta f = 7,5kHz$	1024 pro $l = 0, 1, 2$



Obr. 1.30: Princip přidání cyklického prefixu k OFDM symbolu

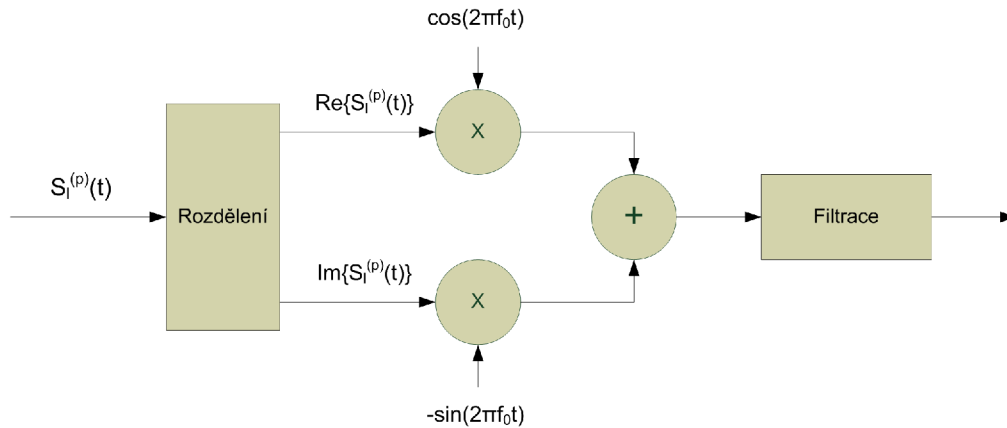
V simulovaném modelu probíhá vytvoření OFDM symbolu ve funkci $[sig] = \text{OFDMMod_dl}(B)$, kde sig je výsledná posloupnost časových vzorků, B je časově frekvenční struktura rádiového rámce vytvořena v bloku Mapování elementů. Funkce převádí jednotlivé sloupce, které představují jednotlivé OFDM symboly, pomocí IFFT na časové vzorky. Ke každé skupině vzorků představující OFDM symbol přidá cyklický prefix. Takto je vytvořena posloupnost časových vzorků. Tyto vzorky nejsou převáděny do vysílacího pásma, ale přímo předávány do bloku modelu rádiového prostředí. Zpracovává se tak komplexní obálka signálu.

1.12.1 Modulace

Po vytvoření komplexní obálky signálu pomocí IFFT a přidání cyklického prefixu je nutné přesunout signál ze základního pásma do vysílacího pásma. Proces konverze signálu je zobrazen na obrázku 1.31. Nejdříve se rozdělí komplexní signál na signál obsahující pouze reálnou část a na signál obsahující pouze imaginární část. Každá z těchto částí je násobena stejným nosným kmitočtem f_0 , ale reálná část je násobena funkcí \cos , imaginární část funkcí $-\sin$. Následně se oba signály sečtou dohromady. Výsledný signál je filtrován, tak aby splnil podmínky pro vysílání uvedené v literatuře [1].

1.13 Rádiový kanál

Rádiový kanál je prostředí, které se nachází mezi vysílačem a přijímačem. Mezi nejjednodušší typy prostředí patří AWGN (Additive white Gaussian noise) kanál.



Obr. 1.31: Princip konverze signálu do vysílacího pásma

Tento kanál je charakteristický tím, že k vysílanému signálu přidává šum a tím zhoršuje podmínky příjmu. Ovšem tento kanál je velmi ideální a reálné prostředí vystihuje jen málo, proto se nejčastěji používá v kombinaci s dokonalejšími modely, které respektují vícecestné šíření signálu a účinky rychlého a pomalého úniku signálu. Kromě těchto účinků dochází ještě ke změně kmitočtu vlivem pohybu přijímače, tuto změnu způsobuje Dopplerův jev. Tento posuv frekvence je často nezanedbatelný a je třeba jej zohlednit. Model rádiového prostředí může být velmi složitý a úprava signálu v bloku rádiového prostředí může být velmi náročná na výpočetní výkon. [11]

V simulovaném modelu se používá model rádiového prostředí, který je součástí prací [9] a [12]. Detailní popis modelu a jeho účinků na vysílaný signál je uveden v kapitole 3.3.1.

1.13.1 AWGN kanál

Tento kanál se používá velmi často, protože velmi dobře modeluje účinky šumu, který se k užitečnému signálu přidává. Výsledný signál $r(t)$ je dán rovnicí [11]:

$$r(t) = \alpha \cdot s(t) + n(t). \quad (1.47)$$

K užitečnému signálu $s(t)$, který vstupuje do rádiového prostředí, se přičítá rušivý signál $n(t)$. Velikost užitečného signálu je upravována pomocí konstanty α . [11]

1.13.2 Vícecestné šíření

Signál, který je vysílán vysílačem, se šíří volným prostředím různými směry. Směr šíření od vysílače je dán vyzařovací charakteristikou antény. Při šíření signálu v pro-

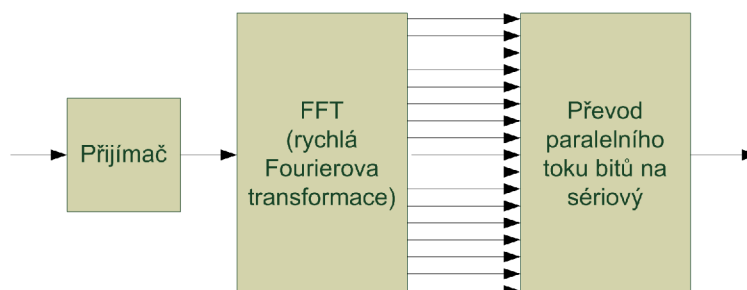
středí dochází k odrazům signálu od překážek. Dále dochází k difrakci na hranách, například budov. Dalším jevem je rozptyl signálu, který mohou způsobovat například listnaté stromy. Přijímač přijímá signál, který je poskládán z těchto druhů příspěvků. Podle typu prostředí jsou některé principy šíření dominantní a některé typy úplně chybí nebo jsou velmi silně potlačeny. Vícecestné šíření se dá popsat pomocí rovnice[11]:

$$r(t) = A \sum_n \alpha_n(t) \cos [2\pi f_c(t - \tau_n(t))], \quad (1.48)$$

kde α_n , τ_n jsou obecně časově proměnné zeslabení a zpoždění n -té cesty. Jako vysílaný signál je uvažován harmonický signál: $s(t) = A \cdot \cos(2\pi f_c t)$.

1.14 OFDM demodulace

Jakmile je signál přijat přijímačem, je nutné nejdříve tento vysokofrekvenční signál převést do základního pásma. Po převedení do základního pásma je od signálu odstraněn cyklický prefix a poté je provedena OFDM demodulace. Demodulace OFDM symbolu spočívá v provedení inverzních operací k vytvoření OFDM symbolu. Proces demodulace je zobrazen na obrázku 1.32.



Obr. 1.32: Princip demodulace OFDM

Demodulace probíhá ve funkci $[B_] = \text{OFDMDeMod_dl}(sig)$, kde proměnná sig je přijímaný signál a proměnná $B_$ je výsledná časově frekvenční struktura rádiového rámce. Funkce postupně zpracovává jednotlivé části signálu, kdy vždy od OFDM symbolu oddělí cyklický prefix a následně převede časové vzorky pomocí operace fft na symboly, které vloží do časově frekvenční struktury, kterou představuje dvourozměrné pole $B_$.

1.15 Vyrovnávání kanálu

Vyrovnávání kanálu napravuje frekvenční charakteristiku signálu tak, aby bylo eliminováno poškození, které způsobil rádiový kanál. Vyrovnávání kanálu také slouží k potlačení vlivů mezisymbolových interferencí. Často se používají adaptivní vyrovnávače, které mají časově proměnnou charakteristiku. Tato charakteristika se přizpůsobuje charakteristice kanálu a tím je vyrovnání lepší. Vyrovnávače můžeme dále dělit na lineární a nelineární. Mezi lineární vyrovnávače kanálu patří Zero forcing a MMSE (Minimum Mean Square Error) vyrovnávač. Mezi nelineární vyrovnávače patří například DFE (Decision Feedback Equalizer) vyrovnávač. Více informací je uvedeno v literatuře [11].

1.16 Demapování elementů

Po OFDM demodulaci a vyrovnání kanálu dostáváme přibližně stejný signál, jaký byl vysílán. Tato podobnost závisí na míře poškození signálu v rádiovém prostředí a dokonalosti eliminace tohoto poškození v přijímači. Nyní je nutné vyjmout data z časově frekvenční struktury rádiového rámce. K tomuto kroku je nutné znát pozice užitečných dat.

Demapování elementů vykonává funkce $[d_] = \text{Demapper_dl}(B_, BMAP)$. Proměnná $B_$ představuje časově frekvenční strukturu, která je získána z OFDM demodulátoru, proměnná $BMAP$ uchovává informace o mapování, tak lze jednoduše poznat, do kterých elementů byla uživatelská data mapována. V reálném systému se informace o umístění uživatelských dat přenáší v kanálech se signalizací, tyto kanály nejsou součástí této práce, proto je krok demapování výrazně zjednodušen. Výstupem je proměnná $d_$, která obsahuje sekvenci symbolů.

1.17 Dekódování a demapování do vrstev

Blok Dekódování a demapování do vrstev má na starosti převedení dat ze symbolů do jednotlivých vrstev. Tento proces se používá při vysílání pomocí více antén. Tato práce se omezuje pouze na vysílání pomocí jedné antény a tento proces je zjednodušen na pouhé předání dat dalšímu bloku.

1.18 Vnitřní demodulace

Přijaté symboly jsou v bloku vnitřní demodulace převedeny na bity.

V simulovaném modelu tento převod zajišťuje funkce $[b_] = \text{demodulace}(d, \text{type}, l)$, kde d je demapovaná sekvence symbolů, type je typ modulace, tedy QPSK, 16QAM, nebo 64QAM a l je proměnná, která uchovává seznam délek po kanálovém kódování. Proměnná l slouží pouze ke zkrácení sekvence, pokud byly při vnitřní modulaci přidány nějaké bity tak, aby sekvence byla dělitelná bezzbytku počtem stavů. Tento krok je v systému LTE řešen v bloku Přizpůsobení datového toku, kdy je sekvence omezena na takový počet bitů, aby byl bezzbytku dělitelný počtem stavů modulace. Tento krok je však po dohodě s vedoucím vynechán. Proměnná $b_$ obsahuje výslednou bitovou posloupnost. Funkce pracuje s objekty vytvořenými v kapitole 1.8. Konkrétně se používají objekty vytvořené funkcemi `modem.pskmod` a `modem.qammod`, z nichž jsou vytvořeny objekty demodulátoru tak, že se doplní do funkce `modem.pskdemod(demodulator)` nebo `modem.qamdemod(demodulator)` jako vstup, tedy místo proměnné *demodulator*. Demodulace se pak provádí funkcí $Y = \text{demodulate}(H, X)$, kde H je objekt demodulátoru vytvořený pomocí `modem.pskdemod` nebo `modem.qamdemod`. X je posloupnost symbolů a Y je obnovená bitová posloupnost.

1.19 Deskramblování

V bloku deskramblování je provádí úplně stejná operace jako v bloku skramblování dat. Toto je vlastnost exkluzivního(XOR) součtu, kdy $\text{xor}(x, c) = y$ a $\text{xor}(y, c) = x$, kde x je vstupní bitová posloupnost, y je výstupní bitová posloupnost a c je náhodná sekvence. Vytvoření náhodné sekvence dat je popsáno v kapitole 1.7. Funkce Deskramblování je tak úplně shodná s funkcí skramblování dat. Do funkce spolu se skramblovací sekvencí vstupuje bitová posloupnost získaná po vnitřní demodulaci a vystupuje bitová posloupnost, která je dále zpracovávána v bloku Rozdělení celku do segmentů.

1.20 Rozdělení celku do segmentů

Po deskramblování je datový tok rozdělen do segmentů, které jsou zpracovávány v bloku Přizpůsobení datového toku a bloku kanálového dekódování, nakonec je u těchto segmentů provedena kontrola kontrolního součtu, pokud je kontrola úspěšná dojde k poskládání segmentů do transportního bloku s kontrolním součtem. Nakonec je vyhodnocen kontrolní součet i u tohoto bloku. Tímto způsobem je výrazně eliminována možnost předání chybných dat vyšším vrstvám. Pokud se kontrolní součty neshodují, systém sám požádá o nový přenos dat pomocí systému HARQ. Rozdělení toku do segmentů je elementární operace, která je inverzní k operaci popisované

v kapitole 1.6.

V simulovaném modelu se tato operace provádí ve funkci `chDeCoding()`.

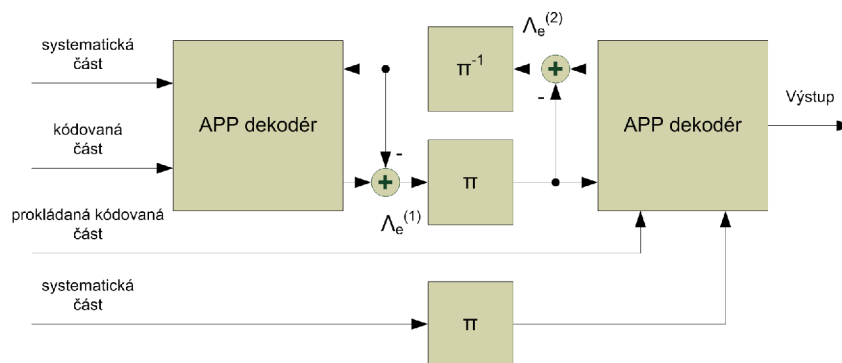
1.21 Přizpůsobení datového toku

Tato část zpracování dat má za úkol segmenty o vynechané bity, které byly vynechány z důvodu redukce datového toku. Dále má za úkol přeskládat datovou posloupnost zpět do správného pořadí tak, aby bylo možné tuto posloupnost dekodovat v bloku kanálového dekodování.

V simulovaném modelu je funkce Přizpůsobení datového toku, po dohodě s vedoucím práce, vynechána. Provádí se zde pouze operace přeskládání dat. Tyto operace jsou inverzní k operacím propisovaným v kapitole 1.5. Je zde využíváno pouze elementárních matematických operací a operací s maticemi, jako jsou přehození řádků, čtení po sloupcích, zápis po řádcích, atd.

1.22 Kanálové dekodování

Zde je za pomoci turbo dekodéru odstraněna redundance, která byla přidána v bloku kanálové kódování. Možností jak dekodovat informaci zakódovanou turbo dekodérem je více. Mezi hlavní typy turbo dekodérů patří Max-Log-MAP, Log-MAP a Soft Output Viterbi Algorithm. Výhodou turbo dekodérů je možnost iterativního dekodování, tímto způsobem je možné dosahovat nízkých chybovostí při velmi malém poměru S/N. Příklad zapojení turbo dekodéru je zobrazen na obrázku 1.33. Více informací o turbo dekodování lze nalézt v literatuře [10].



Obr. 1.33: Příklad vnitřní struktury turbo dekodéru

V simulovaném modelu systému LTE je proces turbo dekodování součástí funkce $[e_err] = \text{chDeCoding}(e, l, l2)$, kde proměnná e obsahuje deskramblovanou posloup-

nost, proměnná l obsahuje pole velikostí segmentů před kanálovým dekódováním, $l2$ obsahuje pole velikostí segmentů po kanálovém dekódování. Výstupní proměnná $c_$ obsahuje dekódované bity, proměnná err se nepoužívá. Funkce využívá turbo dekodér, který je součástí komunikačního toolboxu. Dekodér se vytvoří následujícím příkazem `comm.TurboDecoder('TrellisStructure', $trellis$, 'InterleaverIndices', $i_$, 'NumIterations', $pocetIteraci$)`, kde $trellis$ je trellis struktura, popis vytvoření této struktury naleznete v kapitole 1.4. Proměnná $i_$ je předpis pro interní prokladač, detailní informace jsou uvedeny v kapitole 1.4. Proměnná $pocetIteraci$ definuje počet iterací turbo dekodéru před výsledným rozhodnutím.

1.23 Odebrání CRC a sloučení segmentů do jednoho bloku

Kontrola kontrolního součtu probíhá tak, že je nejprve spočítán nový kontrolní součet z přijatých dat a ten je porovnán s přijatým kontrolním součtem. Pokud se kontrolní součty shodují, je kontrolní součet odebrán a data předána k dalšímu zpracování. Pokud se data neshodují, tak systém automaticky požádá o nové přeposlání chybně přijatých dat.

V modelovaném systému se operace Odebrání CRC a sloučení segmentů do jednoho bloku provádí ve funkci $[b_ b_err] = \text{cbDeSegmentation}(c, F, l)$, kde c jsou dekódované bity, F uchovává počet plnicích bitů, l je pole délek segmentů před kanálovým kódováním. Výstupní proměnná $b_$ obsahuje posloupnost bez kontrolního součtu a proměnná b_err je pole informací o úspěšnosti kontroly kontrolního součtu. V případě chyby je do pole zapsána 1, v případě správné kontroly 0. Odebrání kontrolních součtů a ověření správnosti se provádí ve funkci $[a_err] = \text{det_CRC}(type, b)$, kde $type$ je požadovaný typ detektoru kontrolního součtu. Ten se vytvoří podobně jako generátor kontrolního součtu popisovaný v kapitole 1.2, tedy zadáním polynomu, vytvořeného z rovnice, do funkce `crc.detector($polynom$)`. Příklad polynomu pro CRC délky 8 bitů použitého v LTE je $polynom = [1\ 1\ 0\ 0\ 1\ 1\ 0\ 1\ 1]$. Proměnná b je posloupnost bitů opatřená kontrolním součtem, který se má kontrolovat. Proměnná err bude po kontrole obsahovat 1 v případě nesouhlasu kontrolních součtů nebo 0 v případě shody. Proměnná a je posloupnost bez kontrolního součtu. Detekce správnosti kontrolních součtů se provádí pomocí funkce $[a_err] = \text{detect}(detektor, b)$, kde proměnné a , err a b mají stejný význam jako ve funkci `det_CRC()` a proměnná $detektor$ obsahuje objekt vytvořený funkcí `crc.detector`.

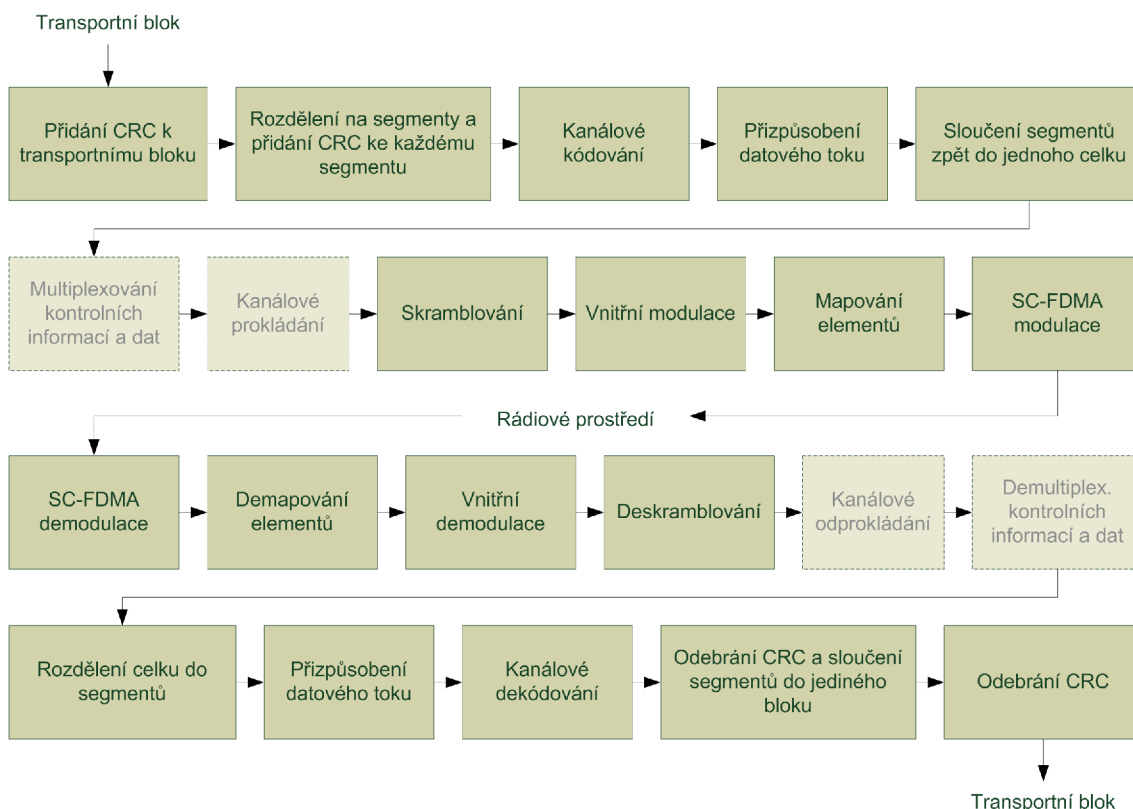
1.24 Odebrání CRC

Kontrola hlavního kontrolního součtu se provádí z důvodu možnosti chybného příjmu, ale správného vyhodnocení kontrolního součtu u segmentů. Pravděpodobnost, že i takto přijatá chybná data budou mít i tento kontrolní součet stejný je velmi malá.

V modelovaném systému se provádí odebrání kontrolního součtu od transportního bloku funkcí `det_CRC()`. Funkce je popisována v kapitole 1.23.

2 UPLINK

Zpracování a vysílání uživatelských dat probíhá velmi podobně jako v případě downlinku. Na obrázku 2.1 je zobrazeno blokové schéma zpracování signálu v případě uplinku. Z obrázku je vidět podobnost zpracování signálu s downlinkem, ovšem některé bloky mají stejný název, ale zpracování dat v nich probíhá odlišně.



Obr. 2.1: Blokové schéma zpracování signálu na fyzické vrstvě v systému LTE - Uplink

Transportní blok, který vstupuje do modelu je nejprve opatřen kontrolním součtem. Detailní popis naleznete v kapitole 1.2. Dále se podle velikosti transportního bloku, dělí tento blok na segmenty a ke každému segmentu je přidán kontrolní součet. Tento postup je popsán v kapitole 1.3. Každý ze segmentů pak vstupuje do zpracování v kanálovém kodéru a následně je upravena jeho velikost v bloku Přizpůsobení datového toku. Detailní popis je shodný s operacemi na downlinku a je popsán v kapitole 1.4 a 1.5. Tyto segmenty jsou, po zpracování, zase sloučeny do jediného bloku. Postup slučování je popsán v kapitole 1.6. Následně je prováděno multiplexování kontrolních a uživatelských informací. Tyto operace jsou prováděny jen na uplinku a jejich popis je uveden v kapitolách 2.2 a 2.3. Následující operací je skram-

blování. Operace skramblování se provádí podobně jako v případě downlinku s tím rozdílem, že kromě samotného xor součtu, jsou s posloupností prováděny ještě další operace týkající se „ACK/NAK nebo Rank Indication placeholder bits“, případně „ACK/NAK nebo Rank Indication repetition placeholder bits“. Operace skramblování je popsána v kapitole 2.4. Dalším zpracováním datové posloupnosti je vnitřní modulace, která tuto bitovou posloupnost přemění na posloupnost symbolů. Blok vnitřní modulace je shodný s blokem použitým u downlinku a je popisován v kapitole 1.8. Získaná posloupnost symbolů je mapována do časově frekvenční struktury rádiového rámce. Proces mapování je popsán v kapitole 2.5, tato operace v sobě zahrnuje i předkódování signálu. Poslední operací před vysláním signálu je SC-FDMA modulace, tato operace je popisována v kapitole 2.6. Následně vysílaný signál prochází modelem rádiového prostředí, ve kterém dochází k jeho poškození a přidání šumu. Model rádiového prostředí je stejný pro downlink a uplink a je popisován v kapitole 1.13. Takto upravený signál vstupuje do přijímače. Zde se nejprve provádí SCFDMA demodulace popsaná v kapitole 2.7. Dále se z časově frekvenční struktury rádiového rámce demapují požadované resource elementy. Proces demapování je popsán v kapitole 2.8. Získaná posloupnost symbolů je převedena na posloupnost bitů pomocí vnitřní demodulace, která je popsána v kapitole 1.18. Následuje deskramblování posloupnosti. Popis operace je uveden v kapitole 2.9. Poté jsou provedeny operace kanálového odprokládání a demultiplexace kontrolních a uživatelských informací. Zpracovaná posloupnost bitů je následně zpracovávána stejně jako v případě downlinku, tedy nejprve je rozdělena do segmentů, popis je uveden v kapitole 1.20, tyto segmenty jsou pak zpracovávány v blocích Přizpůsobení datového toku (kapitola 1.21) a Kanálové dekódování (kapitola 1.22). Následně je u bloků provedena kontrola kontrolních součtů. Popis operace je uveden v kapitole 1.23. Výsledné segmenty jsou sloučeny a provedena kontrola hlavního kontrolního součtu. Odebrání hlavního kontrolního součtu je popsáno v kapitole 1.24. Po odebrání kontrolního součtu získáváme výsledný transportní blok. Ze znalosti vyslané datové posloupnosti jsme pak schopni určit chybovost, která je důležitým výsledkem simulace.

2.1 Společné vlastnosti zpracování signálu s downlinkem

Tato kapitola obsahuje seznam společných operací, které se provádějí shodným způsobem jak na uplinku, tak i na downlinku. Všechny operace jsou vyjmenovány v následujícím seznamu. Platí pouze pro přenos uživatelských dat, tedy pro kanály DL-SCH a UL-SCH.

- Přidání CRC k transportnímu bloku

- Rozdělení transportního bloku na segmenty a přidání kontrolního součtu
- Kanálové kódování
- Přizpůsobení datového toku
- Sloučení segmentů do jednoho toku
- Vnitřní modulace
- Vnitřní demodulace
- Rozdělení toku na segmenty
- Přizpůsobení datového toku
- Kanálové dekódování
- Odstranění CRC a sloučení segmentů do jednoho bloku
- Odstranění CRC

Rozdílné zpracování signálu je především způsobeno tím, že jsou na uplinkový kanál kladeny jiné požadavky. Kvůli požadavku na nízký PAPR (Peak-to-Average Power Ratio), zejména pro možnost použití účinnějších zesilovačů v uživatelských zařízeních, systém používá jinou modulaci, tedy modulací SC-FDMA. Dále se k přenosu využívá pouze jediná anténa. Důvodem je problémové vytváření více antén na velmi malé ploše uživatelského zařízení. Dalším důvodem rozdílných operací je multiplexování uživatelských informací společně s kontrolními informacemi. Rozdílným způsobem je signál zpracováván v blocích:

- Skramblování
- Mapování elementů
- Demapování elementů
- Deskramblování

Další změnou je přidání nebo odebrání určitých operací. Blokované schéma zpracování signálu na uplinku je tak obohaceno o bloky:

- Multiplexování uživatelských a kontrolních informací
- Kanálové prokládání
- SC-FDMA modulace
- SC-FDMA demodulace
- Kanálové odprokládání
- Demultiplexování uživatelských a kontrolních informací

2.2 Multiplexování uživatelských a kontrolních informací

Do bloku Multiplexování uživatelských a kontrolních dat vstupují dvě posloupnosti. První posloupnost tvoří kontrolní data. Druhá posloupnost obsahuje zakódované bity

nesoucí uživatelská data, UL-SCH kanál (Uplink Shared channel). Data jsou rozděleny do sloupcových vektorů, kam jsou poté umístěny prvně kontrolní a následně uživatelská data.

V simulovaném modelu je tento krok vynechán. Kontrolní informace nejsou součástí modelu.

2.3 Kanálové prokládání

Po provedení operací v bloku Multiplexování uživatelských a kontrolních informací jsou takto zpracované posloupnosti prokládány s dalšími kontrolními informacemi jako jsou zakódované potvrzovací bity a zakódované RI (Rank indication) bity.

V simulovaném modelu je tento krok vynechán. Kontrolní informace nejsou součástí modelu.

2.4 Skramblování

Skramblování posloupnosti na uplinku probíhá odlišným způsobem. Odlišný způsob se používá především proto, že sekvence vstupující do bloku Skramblování obsahuje nejen uživatelská data, ale i kontrolní data, která jsou k sekvenci přidána v bloku Multiplexování uživatelských a kontrolních informací. Další kontrolní informace jsou přidány v bloku Kanálové prokládání. Do bloku Skramblování tak vstupují kromě uživatelských dat, také kontrolní informace. Tyto kontrolní informace obsahují zástupné bity 'x' a 'y'. Skramblování u zástupných bitů probíhá odlišným způsobem. Místo symbolu 'x' je do sekvence vložena 1. V případě symbolu 'y' je zopakován předchozí bit. Ostatní zakódovaná data, uživatelská i kontrolní, jsou skramblovány stejným způsobem jako na downlinku. Skramblování probíhá podle vztahu[1]:

$$\tilde{b}^q(i) = (b^q(i) + c^q(i)) \bmod 2 \quad (2.1)$$

Rozdílné je inicializační číslo, které je definováno rovnicí[1]:

$$c_{init} = n_{RNTI} \cdot 2^{14} + \lfloor n_s/2 \rfloor \cdot 2^{9aN_{ID}^{cell}}, \text{ pro PUSCH,} \quad (2.2)$$

kde n_{RNTI} odpovídá RNTI (Radio Network Temporary Identifier). Proměnná n_s představuje číslo slotu v rádiovém rámci a N_{ID}^{cell} je identifikační číslo sektoru.

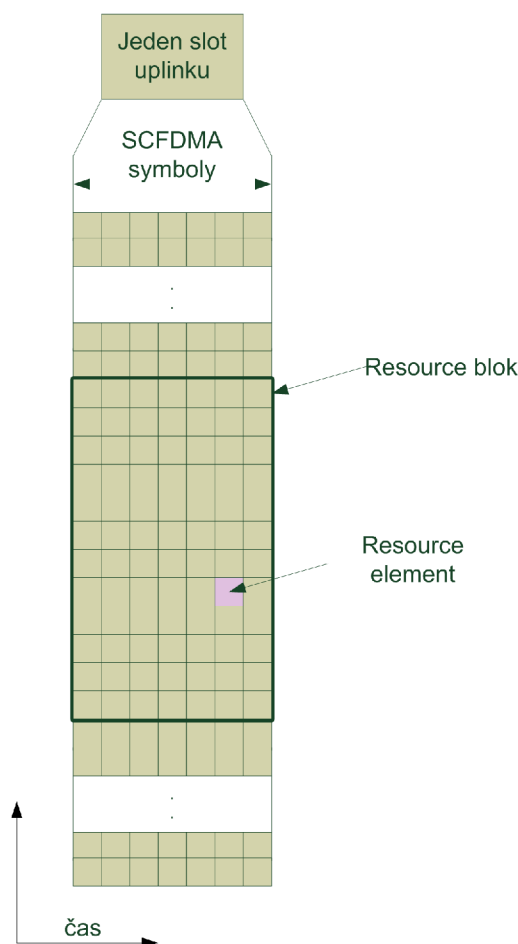
Způsob generování PN posloupnosti je však stejný a je definován vztahem 1.22

V modelovaném systému je k zakódování použita funkce $[b_] = \text{scrambling_ul}(e, uSchVec, umodBit, c_init)$, kde e je posloupnost vystupující z bloku Přizpůsobení

datového toku, $uSchVec$ je vektor popisující způsob mapování signálu, $umodBit$ je počet bitů odpovídající jednomu symbolu a c_init je inicializační číslo sekvence. Generátor pseudonáhodně sekvence je popsán v kapitole 1.7.

2.5 Mapování uživatelských dat

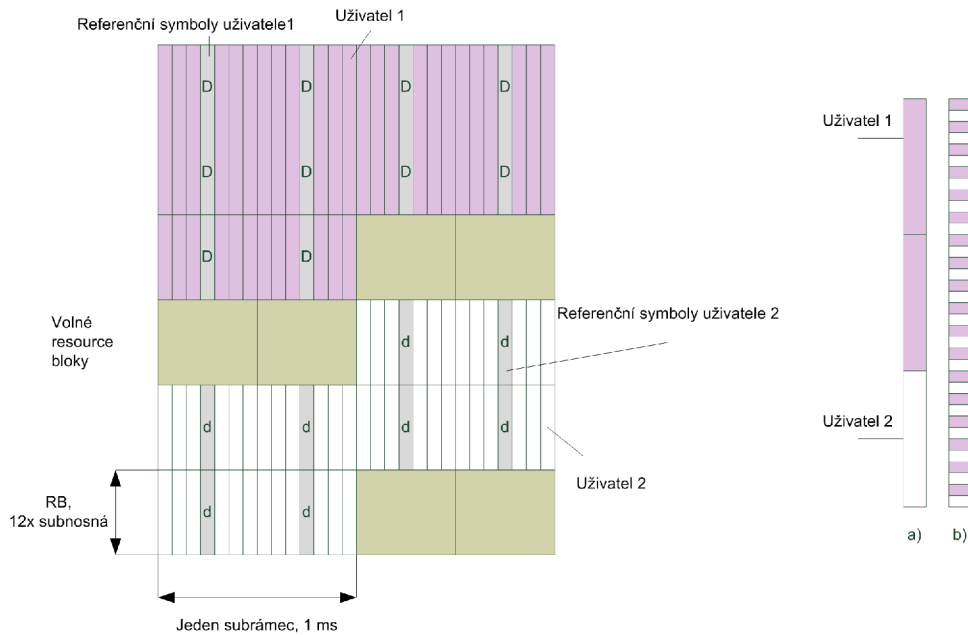
Základní pojmy jako resource blok, resource element, typy rádiových rámců, přístupové metody, pásma určená pro vysílání, možné šířky pásma a další informace o rádiovém kanálu jsou uvedeny v kapitole 1.11. Struktura uplinku je velmi podobná struktuře downlinku. Názorně je časově frekvenční struktura zobrazena na obrázku 2.2.



Obr. 2.2: Časově frekvenční struktura uplinku

Mapování uživatelských dat, tedy kanálu UL-SCH, se provádí odlišně než na downlinku. Pro jednoho uživatele je alokován souvislý blok resource bloků nebo se v souvislém bloku resource bloků střídá více uživatelů pravidelně vždy po subnosné.

Oba způsoby jsou zobrazeny na obrázku 2.3. Mapování uživatelů po souvislých blocích zobrazuje možnost a), distribuované mapování možnost b).



Obr. 2.3: Ukázka mapování uživatelů, uplink

Mapování uživatelských dat v modelu systému LTE zajišťuje funkce $[e \text{ MAP}] = \text{mapovani_uplink}(d, uSchVec, RS, N_cell_ID)$, kde d je sekvence symbolů vystupující z bloku Vnitřní modulace, $uSchVec$ je vektor definující mapování signálu do resource bloků, RS je proměnná typu boolean rozhodující o přidání(true) nebo vynechání(false) referenčních symbolů do struktury rádiového rámce. Proměnná N_cell_ID je číslo definující sektor. Výstupem jsou proměnné e , matice rádiového rámce a MAP , matice uchovávající informaci, kam byly které signály mapovány. Funkce nejdříve přidá do struktury referenční symboly a pak namapuje data do resource elementů.

2.5.1 Referenční signály

Na uplinku jsou používány dva typy referenčních signálů. Prvním typem je Demodulační referenční signál, který je spojen s přenosem informací v kanálech PUSCH (Physical Uplink Shared channel) a PUCCH (Physical Uplink Control channel). Druhým typem je „Sounding“ referenční signál, který není spojen s přenosem informací v kanálech PUSCH nebo PUCCH. „Sounding“ referenční signál slouží k odhadu stavu uplinkového kanálu v resource blocích, kde uživatel nevysílá. Oba typy referenčních signálů jsou tvořeny ze základní sekvence.

Generování základní sekvence pro referenční signály

Sekvence referenčního signálu $r_{u,v}^{(\alpha)}(n)$ je definována jako cyklicky posunutá základní sekvence $\bar{r}_{u,v}(n)$ získaná dle rovnice[1]:

$$r_{u,v}^{(\alpha)}(n) = e^{j\alpha n} \bar{r}_{u,v}(n), \quad 0 \leq n < M_{sc}^{RS}, \quad (2.3)$$

kde $M_{sc}^{RS} = mN_{sc}^{RS}$ je délka sekvence referenčního signálu. Proměnná m leží v rozsahu $1 \leq m \leq N_{RB}^{max,UL}$. Referenční signály jsou definovány jedinou základní sekvencí, liší se rozdílným α .

Sekvence $\bar{r}_{u,v}(n), \dots, \bar{r}_{u,v}(M_{sc}^{RS} - 1)$ delší jak $M_{sc}^{RS} \geq 3N_{sc}^{RS}$ je definována rovnicí[1]:

$$\bar{r}_{u,v}(n) = x_q(n \bmod N_{ZC}^{RS}), \quad 0 \leq n < M_{sc}^{RS}. \quad (2.4)$$

Sekvence kratší jak 3, tedy sekvence kde $M_{sc}^{RS} = N_{sc}^{RS}$ nebo $M_{sc}^{RS} = 2N_{sc}^{RS}$ jsou definovány rovnicí[1]:

$$\bar{r}_{u,v}(n) = e^{j\varphi(n)\pi/4}, \quad 0 \leq n \leq M_{sc}^{RS} - 1. \quad (2.5)$$

kde q -tý kořen Zadoff-Chu sekvence je definován rovnicí[1]:

$$x_q(m) = e^{-j\frac{\pi q m(m+1)}{N_{ZC}^{RS}}}, \quad 0 \leq m \leq N_{ZC}^{RS} - 1, \quad (2.6)$$

kde q je dáno rovnicí[1]:

$$\begin{aligned} q &= \lfloor \bar{q} + 1/2 \rfloor + v \cdot (-1)^{\lfloor 2\bar{q} \rfloor} \\ \bar{q} &= N_{ZC}^{RS} \cdot (u + 1) / 31, \end{aligned} \quad (2.7)$$

délka Zadoff-Chu sekvence N_{ZC}^{RS} je dána největším prvočíslem, které splňuje podmínku $N_{ZC}^{RS} < M_{sc}^{RS}$

Hodnoty $\varphi(n)$ jsou uvedeny v tabulce A.7, pro $M_{sc}^{RS} = N_{sc}^{RS}$ a v tabulce A.8, pro $M_{sc}^{RS} = 2N_{sc}^{RS}$.

Proměnná u je získána z rovnice[1]:

$$u = (f_{gh}(n_s) + f_{ss}) \bmod 30, \quad (2.8)$$

kde $f_{gh}(n_s) = 0$, pokud je deaktivované skupinové skákání a f_{ss} je definované zvláště pro PUSCH a PUCCH. Pro PUCCH je $f_{ss}^{PUCCH} = N_{ID}^{cell} \bmod 30$. Pro PUSCH je $f_{ss}^{PUSCH} = f_{ss}^{PUCCH} + \Delta_{ss} \bmod 30$, kde $\Delta_{ss} \in \{0, 1, \dots, 29\}$ je definováno vyššími vrstvami.

Demodulační referenční signál

Sekvence demodulačního referenčního signálu $r^{PUSCH}(\cdot)$ pro PUSCH je definována rovnicí[1]:

$$r^{PUSCH}(m \cdot M_{sc}^{RS} + n) = r_{u,v}^{(\alpha)}(n), \quad (2.9)$$

kde, $m = 0, 1$ a $n = 0, 1, \dots, M_{sc}^{RS}$ a $M_{sc}^{RS} = M_{sc}^{PUSCH}$. $r_{u,v}^{(\alpha)}(n)$ je definováno rovnicí 2.3.

Cyklický posun α ve slotu n_s je $\alpha = 2\pi n_{cs}/12$, kde n_{cs} je definováno rovnicí[1]:

$$n_{cs} = \left(n_{DRMS}^1 + n_{DRMS}^2 + n_{PRS}(n_s) \right) \bmod 12, \quad (2.10)$$

kde $n_{DRMS}^{(1)}$ je uvedeno v tabulce 2.2, parametr *cyklický posun* je předáván z vyšších vrstev. Pokud se používá DCI (Downlink Control Information) typ 0 pro transportní blok související s přenosem PUSCH, pak je $n_{DRMS}^{(2)}$ uvedeno v tabulce 2.1. Pokud zde není žádný PUCCH s DCI typ 0 pro stejný transportní blok, musí být $n_{DRMS}^{(2)}$ nastaveno na 0. Více informací je uvedeno v literatuře [1].

$n_{PRS}(n_s)$ je dáno rovnicí[1]:

$$n_{PRS}(n_s) = \sum_{i=0}^7 c(8N_{symb}^{UL} \cdot n_s + i) \cdot 2i, \quad (2.11)$$

kde $c(i)$ je pseudonáhodná sekvence definovaná v kapitole 1.7. Generátor je inicializován, na začátku každého rádiového rámce, číslem $c_{init} = \left\lfloor \frac{N_{ID}^{cell}}{30} \right\rfloor \cdot 2^5 + f_{ss}^{PUSCH}$.

Tab. 2.1: Závislost pole cyklického posunu v DCI typ 0 na hodnotě $n_{DMRS}^{(2)}$

Pole cyklického posunu v DCI typ 0	$n_{DMRS}^{(2)}$
0	0
1	6
10	3
11	4
100	2
101	8
110	10
111	9

Tab. 2.2: Závislost mezi *cyklickým posunem* a $n_{DMRS}^{(1)}$

<i>cyklický posun</i>	$n_{DMRS}^{(1)}$
0	0
1	2
2	3
3	4
4	6
5	8
6	9
7	10

Mapování sekvence referenčního signálu

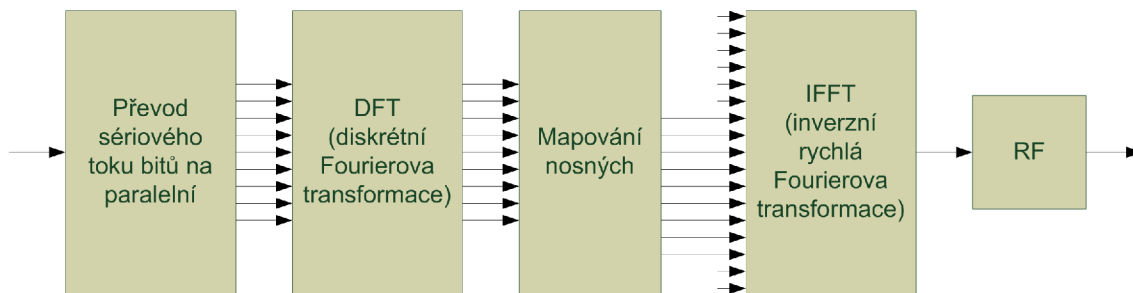
Sekvence $r^{PUSCH}(\cdot)$ musí být násobena konstantou β_{PUSCH} , která definuje amplitudové zesílení. Mapování probíhá od $r^{PUSCH}(0)$, který je umístěn do fyzického resource bloku určeného pro přenos PUSCH. Referenční signály jsou umístovány do resource elementů (k, l) kde $l = 3$, pro normální cyklický prefix a $l = 2$ v případě rozšířeného cyklického prefixu. Mapování probíhá tak, že se nejprve zvyšuje k a následně poté se zvýší číslo slotu n_s . Při procesu mapování dochází také k předkódování uživatelských dat pomocí funkce DFT.

2.6 SCFDMA modulace

Pro uplink je použita modulace SC-FDMA z důvodu nižší hodnoty PAPR (Peak-to-Average Power Ratio). Toto umožňuje použít při vysílání zesilovače s vyšší účinností a tím výrazně ušetřit na spotřebě elektrické energie.

Princip modulace SC-FDMA je zobrazen na obrázku 2.4. Nejprve je převedena sériová posloupnost symbolů na paralelní, dále je převedena pomocí DFT (Discrete Fourier Transform). Následuje mapování vzniklých symbolů a převod signálu do časové oblasti pomocí IFFT. K vzniklému symbolu je přidán cyklický prefix.

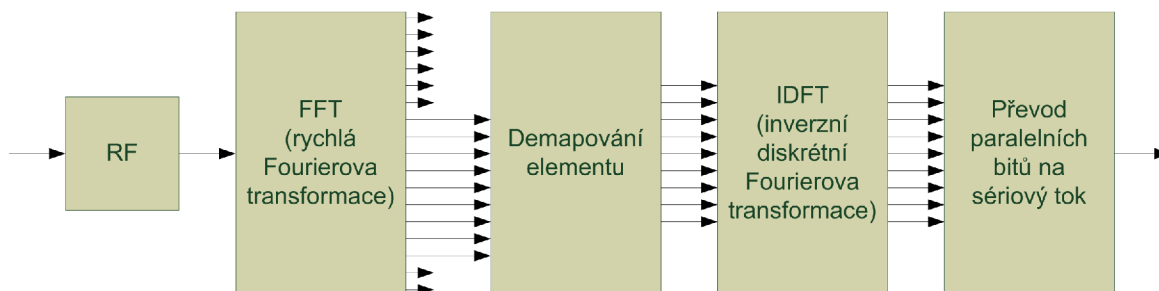
V simulovaném modelu se provádí SC-FDMA modulace pomocí funkce $[sig] = \text{SCFDMAmod_ul}(B)$, kde sig je výsledný SC-FDMA signál, B je matice rádiového rámce vystupující z bloku Mapování elementů. Funkce vykonává operace podle výše popsaného postupu.



Obr. 2.4: Princip modulace SC-FDMA

2.7 SCFDMA demodulace

Demodulace SC-FDMA se provádí na základě inverzních postupů, tedy odebrání cyklického prefixu a provedení FFT. Provedení zpětného procesu mapování symbolů a provedení inverzní DFT obstarává blok Demapování uživatelských dat. Odebrání cyklického prefixu se provádí stejným způsobem jako v případě downlinku. Přehledně je princip demodulace zobrazen na obrázku 2.5.



Obr. 2.5: Princip demodulace SC-FDMA

Demodulace probíhá ve funkci $[B_] = \text{SCFDMAdemod_ul}(sig)$, kde proměnná sig obsahuje časové vzorky signálu přijímaného přijímačem a proměnná $B_$ je obnovená matice rádiového rámce.

2.8 Demapování uživatelských dat

Proces demapování uživatelských dat má za úkol vyjmout ze struktury rádiového rámce uživatelská data a ty dále předat bloku Deskramblování.

V modelovaném systému tento proces vykonává funkce $[d_] = \text{demapovani_uplink}(d_, MAP, uSchVec)$, kde $d_$ je signál po SC-FDMA demodulaci, MAP je proměnná uchováající informaci o tom, kam jsou jaké kanály informace umístěny

a *uSchVec* je vektor popisující do jakých resource bloku budou data umístovány.

2.9 Deskramblování

Deskramblování probíhá úplně stejným způsobem jako skramblování, provádí její shodná funkce, vysvětlení naleznete v kapitole 1.19. Deskramblování provádí funkce `scrambling_ul()`, popis funkce je uveden v kapitole 2.4.

3 MATEMATICKÝ MODEL SYSTÉMU LTE

Tato kapitola se zabývá popisem vytvořeného matematického modelu systému LTE. Tento model je vytvořen v programu Matlab. V první části kapitoly jsou popsány požadavky modelu na verzi programu Matlab a jeho komponent. Další část podrobně popisuje omezení modelu jen na určité nastavení systému. Toto omezení je zde především z důvodu zjednodušení modelu, který by bez těchto omezení byl velmi komplexní a vytvoření takového modelu by přesahovalo rozsah diplomové práce. Další část je věnována popisu vytvořeného modelu, popisu možností nastavení modelu a zobrazení nasimulovaných výsledků. V poslední části je detailně popsán matematický model systému LTE. Popis má formu blokového diagramu, který popisuje naprogramovaný algoritmus, dále jsou zde popsány všechny funkce, které model využívá.

3.1 Požadavky modelu

Model systému LTE byl vytvořen v programovém prostředí Matlab a využívá mnoho jeho funkcí včetně několika funkcí, které jsou obsaženy v nástavbách Matlabu tzv. toolboxech. Ke správnému fungování modelu je zapotřebí mít správnou verzi programu Matlab a správné verze toolboxů.

Konkrétně je zapotřebí:

- Matlab 2011b (7.13.0.564)
- Communications System toolbox 5.1
- Signal Processing toolbox 6.16
- Image Processing toolbox 7.3

Funkčnost modelu na starších nebo novějších verzích programu Matlab nebo jeho komponent nelze zaručit.

3.2 Omezení modelu

Omezení modelu systému LTE je stanoveno zadáním diplomové práce, diskusí a dohodou s vedoucím práce a konkrétně zvoleným nastavením, které bylo dohodnuto s vedoucím práce. Tyto omezení jsou pro přehlednost rozděleny do tří kategorií. První kategorie zmiňuje omezení, které jsou společné pro uplink i downlink, další dvě kategorie se věnují omezením zvláště pro uplink a downlink.

1. Společná omezení modelu:

- (a) zaměření modelu pouze na přenos uživatelských dat, tedy pouze na transportní kanály DL-SCH a UL-SCH

- (b) lze simulovat přenos pouze jednoho transportního bloku
 - (c) ke generování kontrolního součtu pro ochranu transportního bloku je použit pouze generátor g_{CRC24A} . Vyplyvá z omezení přenosu pouze na uživatelská data
 - (d) kanálové kódování probíhá za pomoci turbo kodéru s poměrem 1/3
 - (e) z modelu je vynechána funkce přizpůsobování datového toku (tzv. Rate-matching)
 - (f) je možné simulovat přenos pouze jednoho uživatele
 - (g) lze simulovat přenos pouze za pomoci jedné přijímací a jedné vysílací antény (SISO), žádná z možností MIMO není implementována
 - (h) model pracuje pouze se strukturou rádiového rámce typu 1 (FDD)
 - (i) šířka přenosového kanálu je 1,4 MHz (6 RB) a nelze ji měnit
 - (j) vnitřní modulace mohou být pouze QPSK, 16QAM a 64QAM
 - (k) lze použít pouze normální cyklický prefix, tzn. přenos 7 OFDM symbolů v jednom slotu
 - (l) rozestup subnosných je 15 kHz a nelze jej měnit
 - (m) simulace je prováděna pouze v základním pásmu na tzv. komplexní obálce. Není prováděna konverze signálu do vyššího kmitočtového pásma
 - (n) model pracuje s ideální synchronizací
 - (o) v modelu není rádiovým prostředím přenášena žádná signalizace, tyto informace jsou předávány jako proměnné
2. Omezení týkající se pouze uplinku
 - (a) při zpracování dat je vynecháno Multiplexování kontrolních a uživatelských informací a Kanálové prokládání. Toto vynechání je nutné z důvodu zaměření modelu pouze na přenos uživatelských dat. Není tedy možné multiplexovat uživatelské a kontrolní informace, protože kontrolní informace nejsou nikde získávány
 - (b) v modelu se používají pouze demodulační referenční symboly. „Sounding“ referenční symboly nejsou součástí práce
 3. Omezení týkající se pouze downlinku
 - (a) přenáší se pouze jedno kódové slovo v jeden okamžik

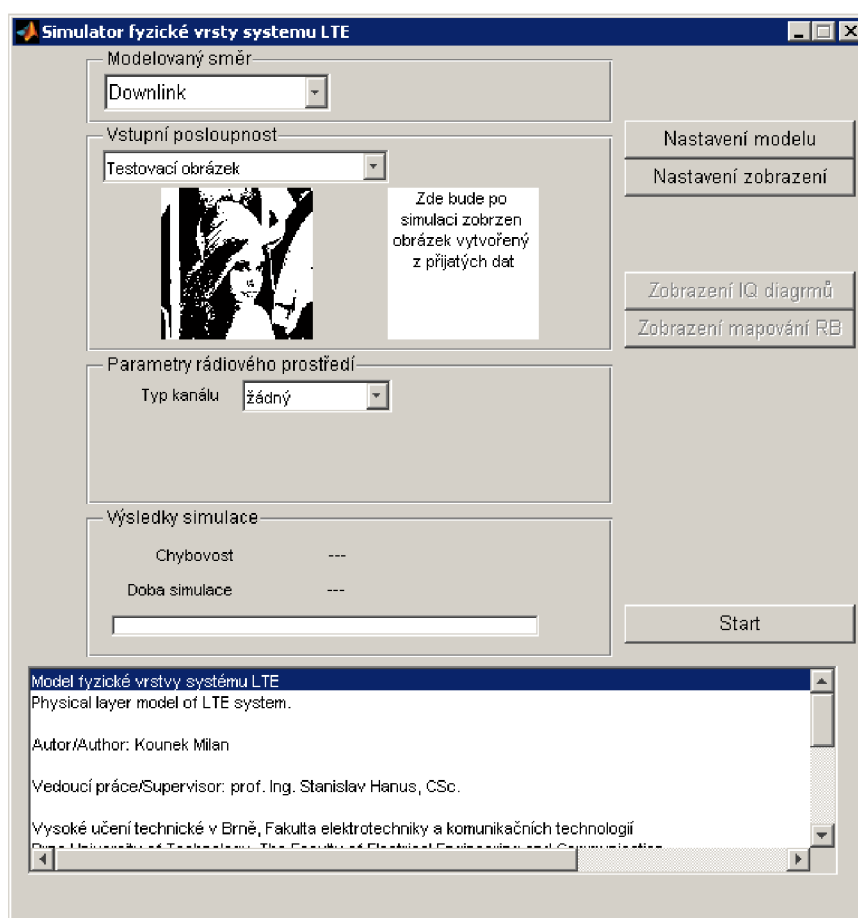
3.3 Popis grafického rozhraní modelu a možnosti nastavení

Grafické rozhraní je vytvořeno s důrazem na jednoduchost a minimalizaci chyb při nastavování parametrů simulace. Velký počet ovládacích prvků a voleb nastavení se zobrazí až při zvolení konkrétní volby simulace. Rozhraní je přehledné a jedno-

duché a je patrné, jaké volby simulace jsou aktivní. V případě nesprávně zadaných hodnot (příkladem může být zapsání řetězce do pole, kde je očekávána číselná hodnota) model upozorní uživatele a zobrazí správné možnosti zadání.

Hlavní soubor modelu se jmenuje GUI.m, informace o grafickém rozhraní jsou uloženy v souboru GUI.fig. Mimo toto rozhraní je zde ještě pět dalších doplňujících rozhraní, které slouží různým účelům, od nastavení parametrů modelu nebo zobrazení informací o postupu procesu až po vykreslení rádiového kanálu, zobrazení IQ diagramu a zobrazení závislosti chybovosti na poměru S/N (resp. C/I).

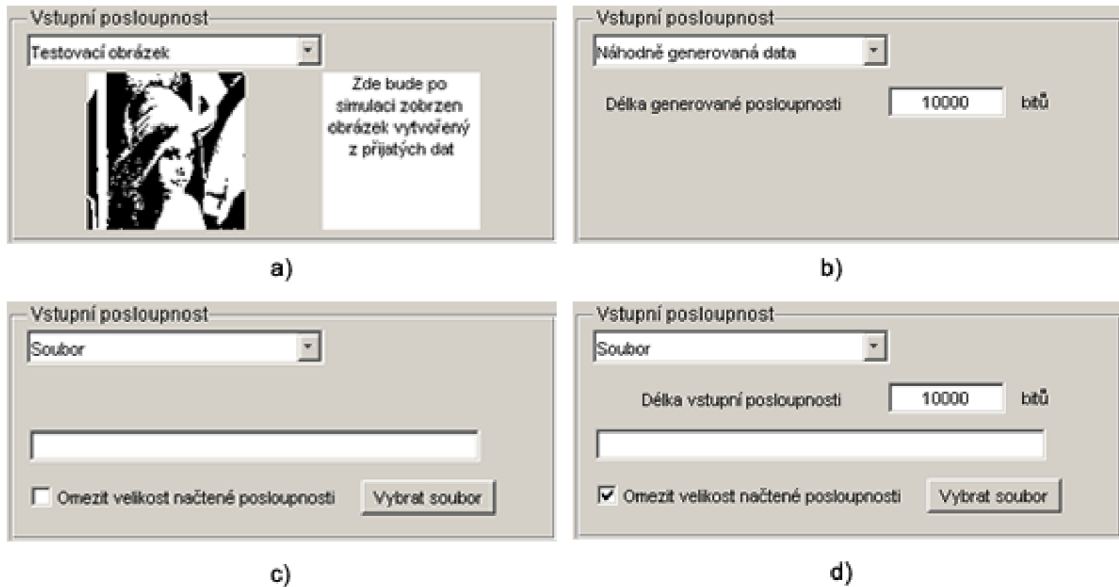
3.3.1 Hlavní uživatelské rozhraní



Obr. 3.1: Hlavní uživatelské rozhraní

Na obrázku 3.1 je zobrazeno Hlavní uživatelské rozhraní tak, jak se objeví uživateli po spuštění skriptu GUI.m. Toto rozhraní umožňuje volbu hlavních prvků simulace, jako je volba modelovaného směru, tedy uplink nebo downlink. Dále umožňuje

zvolit vstupní data, která může představovat testovací obrázek, náhodně generovaná data nebo mohou být data načtena ze souboru.



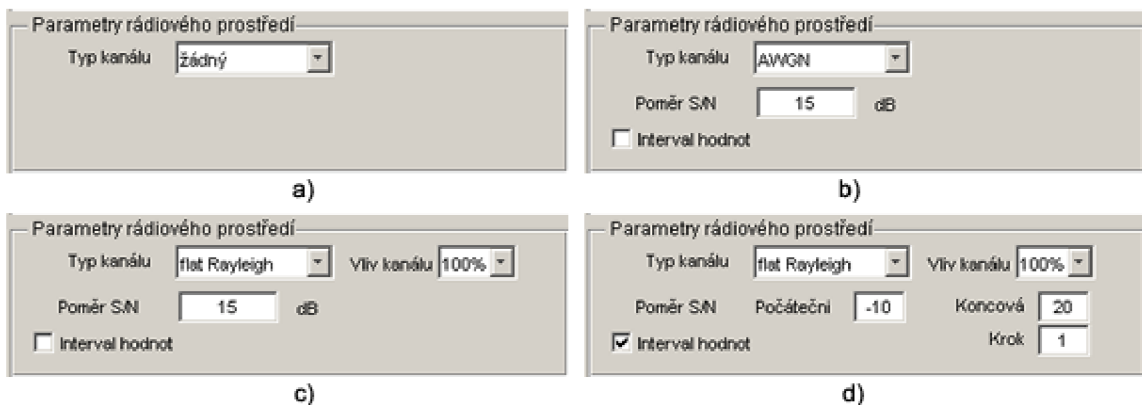
Obr. 3.2: Možnosti volby vstupních dat

Na obrázku 3.2 jsou zobrazeny všechny čtyři možnosti volby vstupních dat. Část a) představuje zvolení vstupních dat ve formě obrázku. Data reprezentují jednotlivé pixely obrázku. Tato možnost ilustruje míru poškození datové posloupnosti v kanálu, avšak není příliš vhodná pro pozorování chybovosti, protože z obrázku je patrné až poměrně vysoké poškození. Část b) Náhodně generovaná data znamená, že jako vstupní data bude zvolena náhodně vygenerovaná posloupnosti bitů „0“ a „1“. Data jsou pokaždé náhodně zvolena a tuto skutečnost nelze nijak ovlivnit. Části c) a d) společně vyžadují volbu souboru, který je načten jako bitová posloupnost. Výběr lze provést kliknutím na tlačítko Vybrat soubor nebo zadat úplnou cestu včetně názvu souboru do prázdného řádku. Při volbě c) bude celý soubor převeden na posloupnost bitů a tato pak bude považována za vstupní data modelu. Při volbě d) bude délka vstupních dat omezena na požadovanou velikost.

Maximální délka vstupní posloupnosti je omezena na 10^6 bitů a to z důvodu omezené velikosti paměti RAM. Je možné, že při používání 64 bitové verze Matlabu bude možné simulovat vstupní posloupnost větší délky, ale tuto možnost nemám vyzkoušenou.

Další volbou jsou Parametry rádiového prostředí. Zde je možné volit ze šesti možností typu kanálu:

1. žádný, tato volba nevyžaduje žádné dodatečné údaje. Signál, který je představován, po OFDM modulaci, časovými vzorky přímo vstupuje do OFDM demodulátoru a není nijak poškozen. V tomto případě musí vždy vyjít chybovost rovná 0. Tato možnost dovoluje zkontrolovat správnou funkci modelu.
2. Interferenční, v případě volby interferenčního kanálu se k časovým vzorkům signálu přičítá interferenční signál. Velikost utlumení nebo zesílení interferenčního signálu závisí na zvoleném poměru C/I, který je pro simulaci vyžadován. Popis interferenčního signálu je uveden v sekci 3.3.1.
3. AWGN, kanál typu AWGN přidává k časovým vzorkům signálu náhodný šum, jeho velikost je dána poměrem S/N. Více o tomto kanálu je zmíněno v sekci 3.3.1.
4. flat Rayleigh, popis tohoto typu kanálu je zmíněn v sekci 3.3.1, u tohoto typu kanálu lze měnit vliv kanálu v rozmezí 25 %, 50 %, 75 % nebo až 100 %. Více o vlivu kanálu je zmíněno v sekci 3.3.1. K poškozenému signálu se dále přičítá šum. Velikost šumu je definována poměrem S/N.
5. PedA, neboli Pedestrian A, popis kanálu je zmíněn v sekci 3.3.1, míru poškození signálu je možné měnit volbou Vlivu kanálu, dále je k signálu přidáván šum, který je definován poměrem S/N.
6. PedB, neboli Pedestrian B, popis kanálu je zmíněn v sekci 3.3.1, míru poškození signálu je možné měnit volbou Vlivu kanálu, dále je k signálu přidáván šum, který je definován poměrem S/N.



Obr. 3.3: Možnosti volby parametrů rádiového prostředí

Na obrázku 3.3 jsou zobrazeny možnosti volby parametrů rádiového prostředí.

Část a) typ kanálu žádný nevyžaduje žádné doplňující údaje. Část b) je stejná pro AWGN a Interferenční kanál, zde je potřeba zvolit i poměr S/N (resp. C/I). Namísto jedné hodnoty je možné zvolit i krokování poměru S/N (resp. C/I), tuto možnost vyobrazuje část d). Počáteční hodnota je hodnota od které je poměr S/N měněn do koncové hodnoty zadané v poli Koncová, s krokem udaným v poli Krok. Na obrázku u volby d) je zobrazeno nastavení simulace, kdy se postupně provede simulace s prvotním nastavením poměru S/N na hodnotu -10 dB, následně je tato hodnota zvětšována o 1 dB, simulace skončí po odsimulování poslední hodnoty S/N = 20 dB. Při volbě kanálu flat Rayleigh, PedA nebo PedB je navíc možné volit i vliv kanálu, tato možnost je zobrazena v části c).

Interferenční signál

Interferenční signál je definován v modelu jako časové vzorky signálu vystupujícího z vytvořeného modelu o určitém nastavení. Toto nastavení je:

- Vnitřní modulace: 64QAM
- N_CELL_ID: 5
- n_RNTI: 489
- Data jsou přenášena v celé šířce pásma, SchVec: = [1, 1, 1, 1, 1, 1] ;
- Vstupní data jsou načtena ze souboru __test_file.JPG, velikost souboru je omezena v případě uplinku na 17040 bitů, v případě downlinku na 18360 bitů.

AWGN

Při použití kanálu tohoto typu je k signálu přičten náhodný šum o velikosti, která je dána hodnotou poměru S/N. Přidání AWGN šumu, s poměrem S/N, k signálu probíhá ve funkci LTE_channel_model.m, která byla převzata z modelu, který je součástí prací [9] a [12].

Vliv kanálu

U typů kanálů flat Rayleigh, PedA a PedB je možnost snížit účinky kanálu na signál volbou vlivu kanálu. To znamená, že výsledný signál vystupující z modelu rádiového prostředí je kombinací nepoškozeného signálu a signálu, který byl poškozen v daném typu rádiového kanálu. Tento vliv je nastavitelný v rozmezí 25 % až 100 % s krokem 25 %. V případě volby 25 % je výsledný signál výsledkem součtu jedné čtvrtiny poškozeného signálu a tří čtvrtin nepoškozeného signálu. Velikost šumu odpovídá nastavené hodnotě poměru S/N.

Společná nastavení modelů rádiových kanálů flat Rayleigh, PedA a PedB

Modely rádiového kanálu flat Rayleigh, PedA a PedB jsou převzaty z literatury [9], [12]. Pro zakomponování modelů do mého systému bylo potřeba nastavit značné množství parametrů, tak, aby odpovídali konkrétnímu nastavení shodnému s nastavením mého modelu.

Dále bylo vybráno jen několik nastavení rádiového prostředí ze všech možných. Vybrané typy modelu rádiového kanálu jsou AWGN, flat Rayleigh, PedA a PedB. Společné nastavení těchto kanálů je:

- ChanMod.time_correlation = independent;
- ChanMod.filtering = BlockFading;

Více o významu těchto nastavení se dozvíte v literatuře [9], [12].

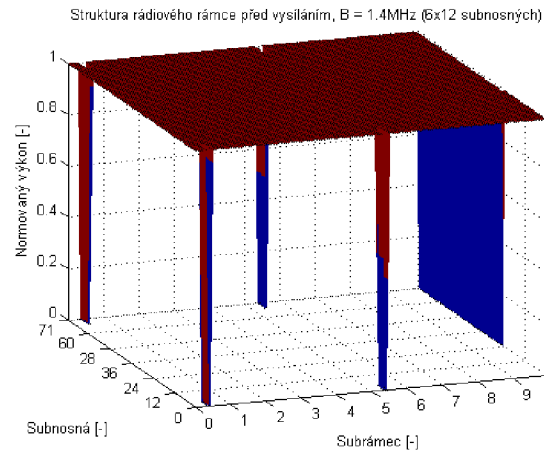
Flat Rayleigh

Při zvolení ChanMod.filtering = BlockFading, je účinek tohoto kanálu následující. Z časově frekvenční struktury je část signálu, odpovídající celé šířce pásma ve frekvenční oblasti a šířce jednoho subrámece, tedy časového úseku o délce 1 ms, v časové oblasti, vynásobena náhodnou komplexní hodnotou, která je vygenerována na základě LTE_params.channel_param_RandStream, tedy náhodné sekvence. Tato sekvence je určena číslem channel_rand_seed. Názorně je vliv tohoto kanálu zobrazen na obrázcích 3.4, 3.5 a 3.6. Všechny tři obrázky zobrazují strukturu rádiového rámce, první obrázek 3.4 zobrazuje strukturu rádiového rámce před vysláním, stejná struktura rádiového rámce je použita i při demonstraci účinků modelů kanálů PedA a PedB. Jelikož je použita modulace QPSK jsou všechny symboly stejně velké mají hodnotu 1, nejsou zaplněny pouze resource elementy, ve kterých není vysíláno, tedy krajní části synchronizačních signálů a nevyužitá část posledního subrámece. Obrázek 3.5 zobrazuje strukturu rádiového rámce po průchodu kanálem bez přidaného šumu. Z obrázku je patrný vliv kanálu popsáný výše. Obrázek 3.6 zobrazuje detail subrámece číslo 1 a 2.

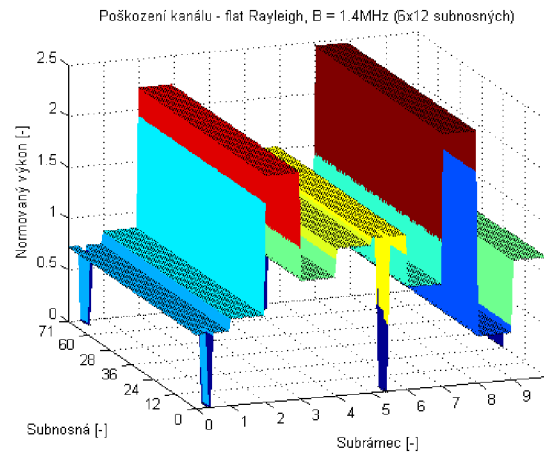
PedA

Model kanálu PedA, tedy Pedestrian A definovaný ITU, je dán jeho impulsní odezvou, která je zobrazena modře na obrázku 3.7.

Dále je, vlivem zvoleného parametru ChanMod.filtering = BlockFading, aplikována matice definující kanál na celou šířku pásma ve frekvenční oblasti a jeden subrámece v časové oblasti. Vliv poškození struktury rádiového rámce je patrný z obrázku 3.8, detail poškození je zobrazen na obrázku 3.9.



Obr. 3.4: Stav Rádiového rámce před vstupem do rádiového prostředí



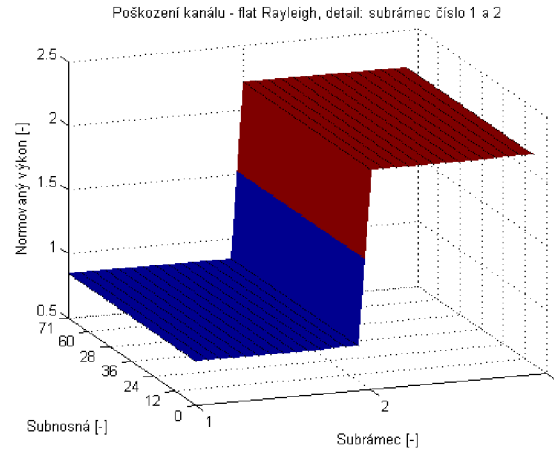
Obr. 3.5: Poškození rádiového rámce při použití modelu kanálu flat Rayleigh

Rozdíl účinků modelu rádiového kanálu oproti modelu flat Rayleigh je především v proměnném zesílení subnosných v celé šířce pásma. Zesílení se mění poměrně málo.

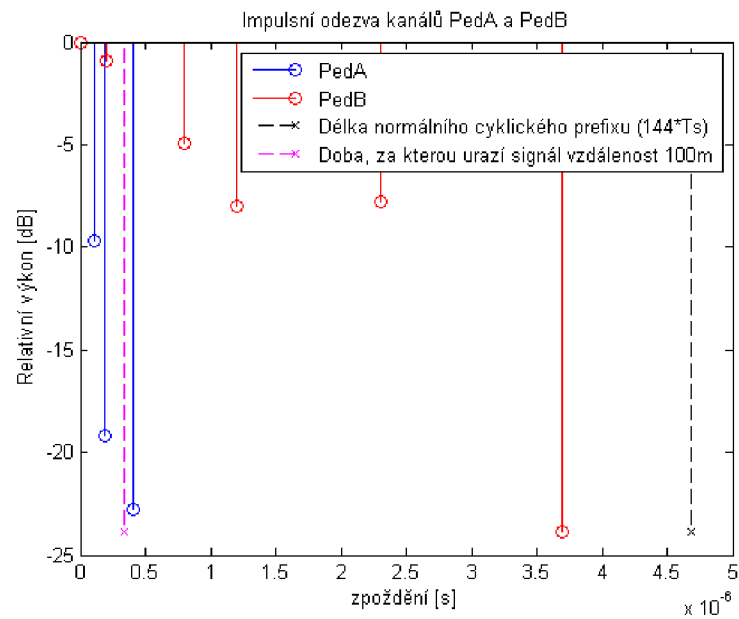
PedB

Model rádiového kanálu PedB, tedy Pedestrian B definovaný ITU, je velmi podobný modelu rádiového kanálu PedA, rozdílná je pouze Impulzní odezva kanálu. Tato odezva je zobrazena na obrázku 3.7 červeně.

Vliv poškození struktury rádiového rámce při průchodu signálu moelem rádiového kanálu typu PedB je zobrazen na obrázku 3.10, detail poškození je zobrazen na obrázku 3.11. Na obrázku 3.4 je vyobrazena struktura rádiového rámce před vstupem signálu do modelu rádiového prostředí.



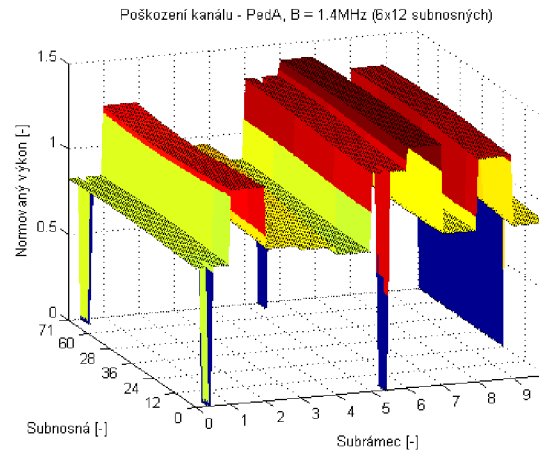
Obr. 3.6: Poškození rádiového rámce při použití modelu kanálu flat Rayleigh, detailní pohled



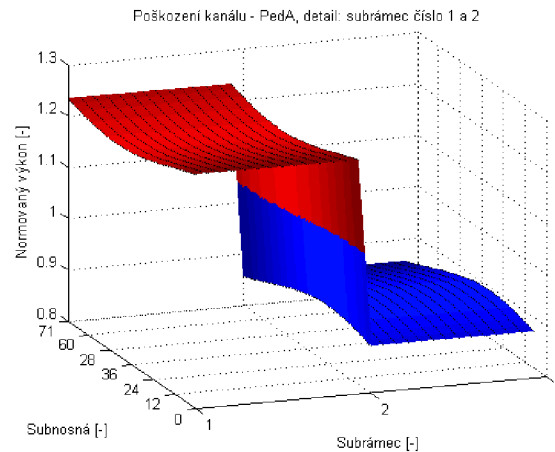
Obr. 3.7: Impulsní odezva modelu rádiového kanálu PedA a PedB

Popis jednotlivých tlačítek a start simulace

Na obrázku 3.1 je v pravé části zobrazeno 5 tlačítek. První dvě tlačítka Nastavení modelu a Nastavení zobrazení slouží k nastavování vlastností modelu a nastavení zobrazení informací o procesu simulace. Další skupina tlačítek Zobrazení IQ diagramů a Zobrazení mapování RB slouží ke zobrazení informací o signálu před vstupem do modelu rádiového kanálu a po výstupu z modelu rádiového kanálu. Poslední tla-



Obr. 3.8: Poškození rádiového rámce při použití modelu kanálu PedA

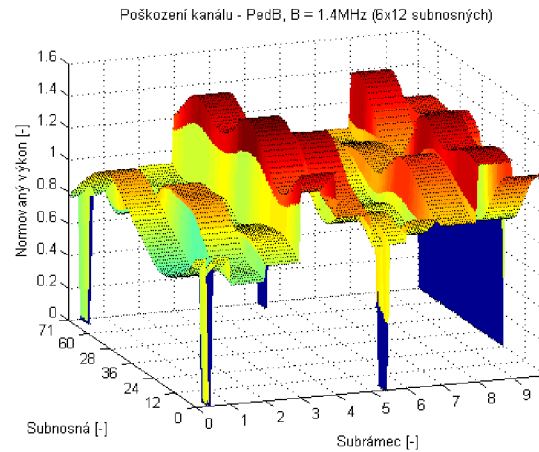


Obr. 3.9: Poškození rádiového rámce při použití modelu kanálu PedA, detailní pohled

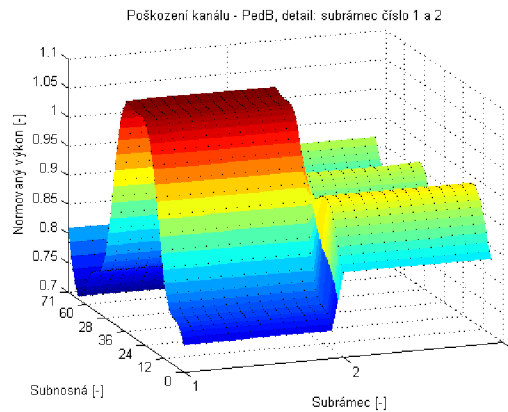
čítka Start spouští proces simulace. Pokud je zvolena volba Interval hodnot, v části Parametry rádiového kanálu, objeví se tlačítko Závislost BER, S/N. Toto tlačítko slouží na zobrazení výsledků simulace, konkrétně závislosti chybovosti na poměru S/N (resp. C/I). Více informací o jednotlivých tlačítkách je uvedeno v následujících kapitolách 3.3.2 až 3.3.6.

Start simulace

Tlačítko Start spouští proces simulace. Popis procesu je uveden v kapitole 3.4. Po provedení simulace se zobrazí okno s informací o úspěšném provedení simulace. Ve spodní části hlavního okna, v sekci Výsledky simulace, se po provedení simulace



Obr. 3.10: Poškození rádiového rámce při použití modelu kanálu PedB



Obr. 3.11: Poškození rádiového rámce při použití modelu kanálu PedB, detailní pohled

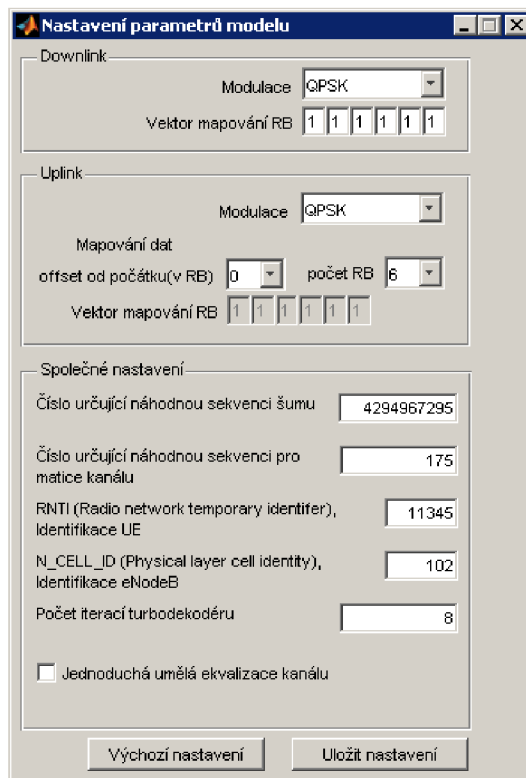
zobrazí chybovost a celková doba simulace. Během trvání procesu se plní proces bar a dává tím uživateli informaci o průběhu simulace.

Pokud bylo nastaveno zobrazování výpisu informací o procesu simulace, jsou tyto informace vypisovány ve spodní části hlavního okna.

3.3.2 Možnosti nastavení parametrů modelu

Na obrázku 3.12 je zobrazeno rozhraní pro změnu parametrů modelu systému LTE. Jednotlivé položky mají význam:

1. Downlink
 - (a) Modulační technice, která bude použita pro vnitřní modulaci



Obr. 3.12: Možnosti nastavení parametrů modelu systému LTE

- (b) Vektor mapování resource bloků, podle zadaného vektoru budou mapovány uživatelská data do struktury rádiového rámce. První pole označuje resource blok 0, poslední resource blok 5. Do polí je možné zadat čísla 0 nebo 1. Hodnota 1 značí, že do daného resource bloku budou symboly umístovány. Hodnota 0 značí, že do daného resource bloku nebudou vkládány symboly přenášející zakódovanou informaci.
2. Uplink
- (a) Modulační technice, odpovídá modulační technice, která bude použita pro vnitřní modulaci
- (b) Mapování dat, offset od počátku (v RB) znamená, že data budou mapována až od zvoleného resource bloku. Počet RB je počet resource bloků, které budou využity pro mapování symbolů. To znamená, že mapování začne od resource bloku uvedeného v poli offset od počátku (v RB) a bude mapováno postupně do následujících resource bloků, počet těchto resource bloků je dán polem Počet RB. Po stisku tlačítka Uložit nastavení se objeví aktuální Vektor mapování RB, který má stejný význam jako u downlinku
3. Společné nastavení

- (a) Číslo určující náhodnou sekvenci šumu, je číslo, které je použito jako seed (inicializační číslo) pro náhodný generátor. Tento generátor generuje sekvenci dat, která je použita pro vytváření šumu. To znamená, že toto číslo přímo určuje, jak bude vypadat šum v rádiovém kanálu. Pokud bude toto číslo stejné pro rozdílná nastavení kanálu, tak výsledkem bude stejný šum, který se bude přičítat k časovým vzorkům v modelu rádiového kanálu. Povolené hodnoty jsou od 0 do 4 294 967 295.
- (b) Číslo určující náhodnou sekvenci pro matice kanálu, toto číslo má obdobou funkci jako Číslo určující náhodnou sekvenci šumu, s tím rozdílem, že toto číslo je použito jako seed pro generátor náhodných hodnot, které se používají pro matice kanálu typu flat Rayleigh, PedA, PedB. Povolené hodnoty jsou od 0 do 4 294 967 295.
- (c) RNTI (Radio network temporary identifier), Identifikace UE. Toto číslo identifikuje UE a je významné pro skramblování. Je možné volit hodnoty od 0 do 65535.
- (d) N_CELL_ID (Physical layer cell identity), Identifikace eNodeB. Slouží k identifikaci eNodeB, toto číslo je důležité hlavně pro skramblování na downlinku a pro mapování dat, kdy je z něho odvozovány referenční symboly, primární a sekundární synchronizace. Je možné volit čísla od 0 do 503.
- (e) Počet iterací turbo dekodéru, nastaví počet iterací turbo dekodéru. Možné hodnoty jsou 1 až 64. Horní mez je nastavena z důvodu zdlouhavosti procesu.
- (f) Jednoduchá umělá ekvalizace kanálu, zaškrtnutím volby je povolena jednoduchá ekvalizace popsaná v kapitole 3.5.

Tlačítkem Uložit nastavení se provede kontrola správnosti zadaných hodnot. V případě správnosti bude nastavení uloženo a zobrazena informace o uložení. V případě chyby bude zobrazena chybová hláška a pole se špatnou informací bude zvýrazněno červeně.

Výchozí nastavení, které je definováno autorem lze nastavit stiskem tlačítka Výchozí nastavení.

3.3.3 Možnosti nastavení výpisu informací o probíhajícím procesu

Obrázek 3.13 zobrazuje veškeré možnosti nastavení výpisu informací, které je možné vypsat do hlavního okna. Nastavení je rozděleno do několika skupin.

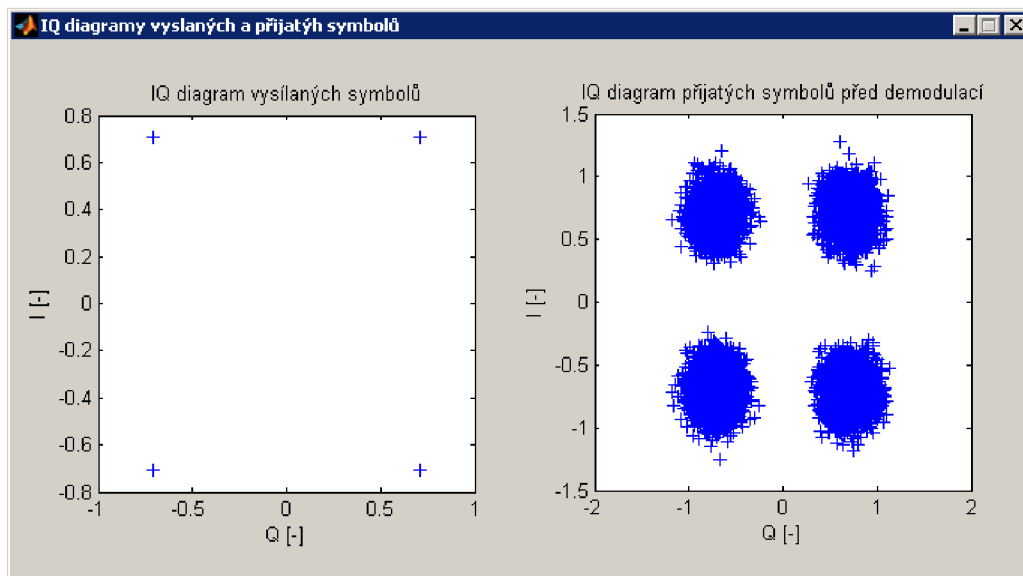
1. v prvním sloupci je nejvýše zobrazena skupina výpisu o chybách.
2. v dolní části jsou samostatně uvedeny tři ovládací prvky



Obr. 3.13: Možnosti nastavení vypisování informací o postupu procesu modelu systému LTE

- (a) Celkový čas trvání všech procesů, vypíše sumu trvání jednotlivých procesů. Tato suma je rozdílná od času simulace, protože zahrnuje pouze časy vykonávání funkcí.
 - (b) Zobrazení souhrnu o trvání simulace, zobrazuje informace o stavu simulace s odhadem času, kdy bude dokončena.
 - (c) Zobrazení oddělovacích řádků, zobrazí oddělovací řádky, které umožňují větší přehlednost výpisu.
3. druhý sloupec je věnován Zobrazení informací o postupu procesu, zobrazuje základní informace o tom, že daný proces byl dokončen.
 4. Třetí sloupec umožňuje zobrazit detailní informace o procesech, příkladem může být počet bitů vstupujících a vystupujících z turbo kodéru.
 5. Poslední sloupec umožňuje zobrazit časy trvání jednotlivých procesů.

Každá skupina je ovládána nejvyšším polem, které aktivuje nebo deaktivuje celou skupinu. Při aktivované skupině je pak možné zvolit jaké informace nás zajímají a mají být zobrazeny.



Obr. 3.14: Příklad zobrazených IQ diagramů pomocí modelu systému LTE

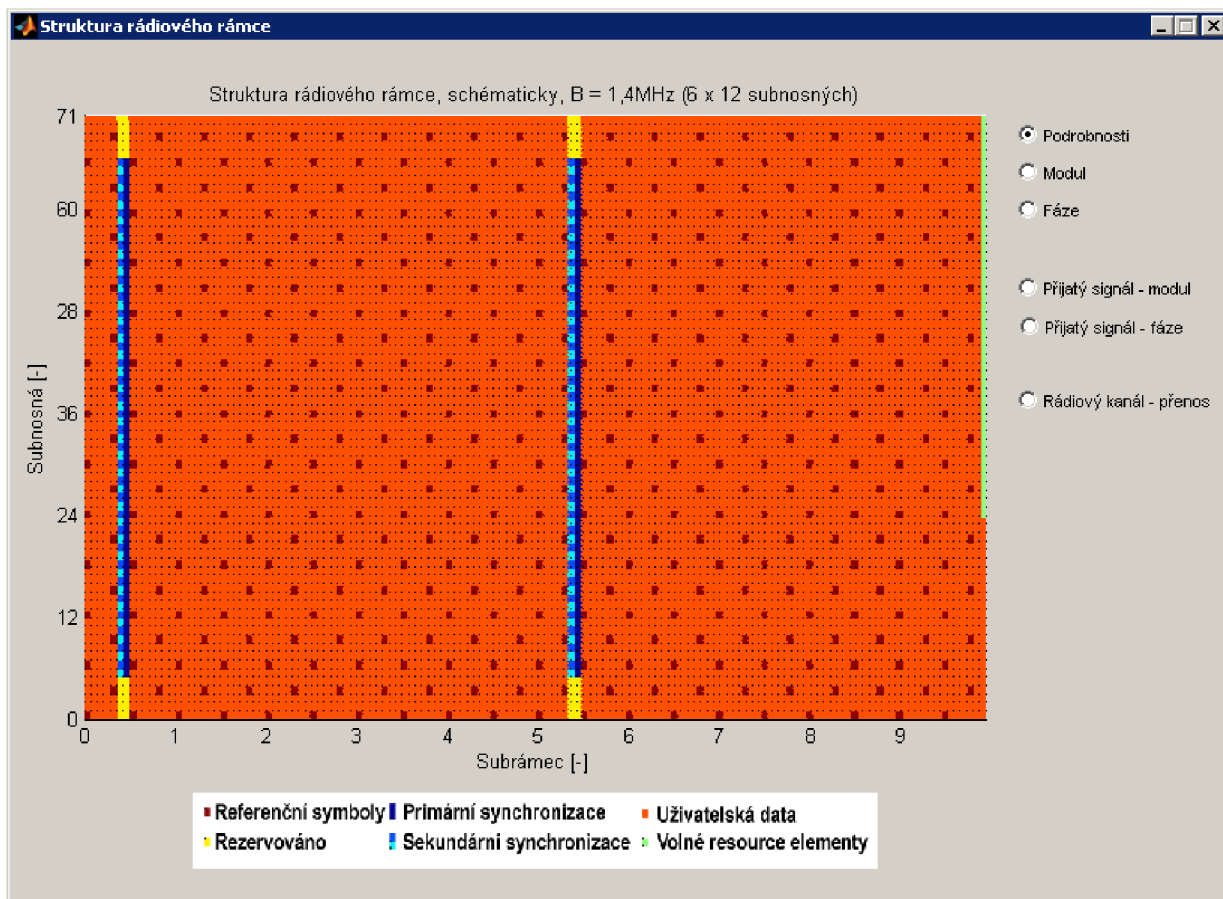
3.3.4 Zobrazení IQ diagramu

Po kliknutí na tlačítko Zobrazení IQ diagramů, které je umístěno na hlavním rozhraní, se zobrazí dvojice IQ diagramů, levý diagram odpovídá stavu před poškozením v modelu rádiového kanálu. Pravý diagram zobrazuje symboly, které byly přijaty v přijímači. Příklad zobrazených IQ diagramů je uveden na obrázku 3.14.

3.3.5 Zobrazení struktury rádiového rámce

Na obrázku 3.15 je zobrazena struktura rádiového kanálu pomocí modelu systému LTE. Model umožňuje zobrazit strukturu rádiového rámce v takové podobě, v jaké ji uživatel simuluje. Celkem je možné zobrazit 6 typů zobrazení struktury rádiového kanálu:

1. Podrobnosti, zobrazuje informace o tom, jaký resource element je zaplněn konkrétním symbolem. Z tohoto přehledu je možné vyčíst pozice referenčních symbolů, dále symbolů, které nesou zakódovanou informaci. V případě downlinku je zde uvedena primární a sekundární synchronizace.
2. Modul, zobrazuje modul vysílaných symbolů před průchodem blokem rádiového kanálu.
3. Fáze, zobrazuje fázi vysílaných symbolů před průchodem blokem rádiového kanálu.
4. Přijatý signál - modul, zobrazuje modul symbolů, které jsou přijímány přijímačem.



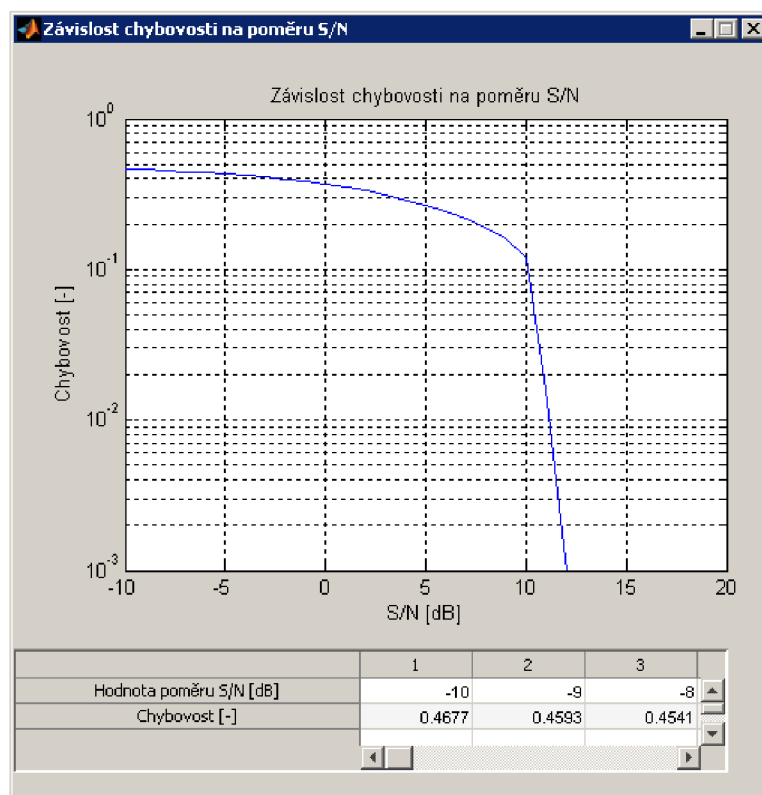
Obr. 3.15: Příklad zobrazení struktury rádiového rámce pomocí modelu systému LTE

5. Přijatý signál - fáze, zobrazuje fázi symbolů, které jsou přijímány přijímačem. Fáze je uvedena v radiánech.
6. Rádiový kanál - přenos, zobrazuje přenos rádiového kanálu, ten je definován jako $10 \cdot \log(B'/B)$, kde B' je Struktura rádiového rámce rekonstruovaná v přijímači a B je struktura rádiového rámce před vysláním.

3.3.6 Zobrazení závislosti chybovosti na poměru S/N (resp. C/I)

Nejdůležitějším výstupem modelu je informace o chybovosti. Pokud je simulován interval hodnot S/N (resp. C/I) je možné zobrazit závislost chybovosti na poměru S/N (resp. C/I). Příklad zobrazení je uveden na obrázku 3.16. V horní části okna je uveden přehledný graf, který zobrazuje závislost chybovosti na poměru S/N (resp. C/I), v dolní části okna je uvedena tabulka s hodnotami poměru S/N (resp. C/I)

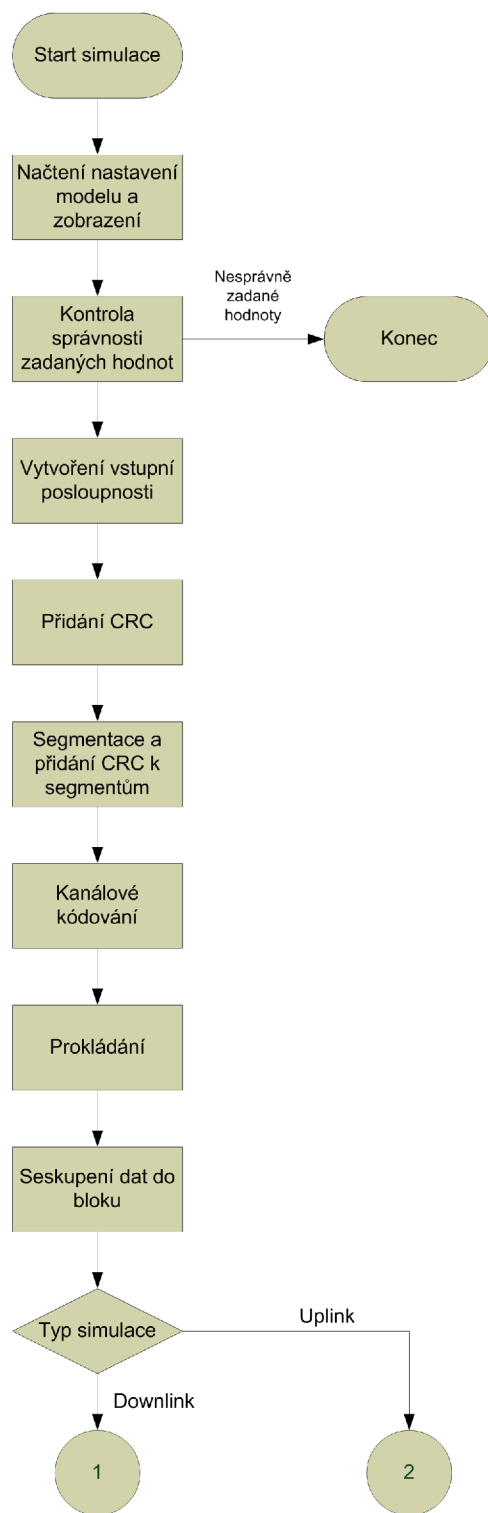
a chybovosti, tak jak byly naměřeny. Hodnoty v tabulce lze označit a zkopírovat, následně je možné zpracovávat hodnoty v jiném programu.



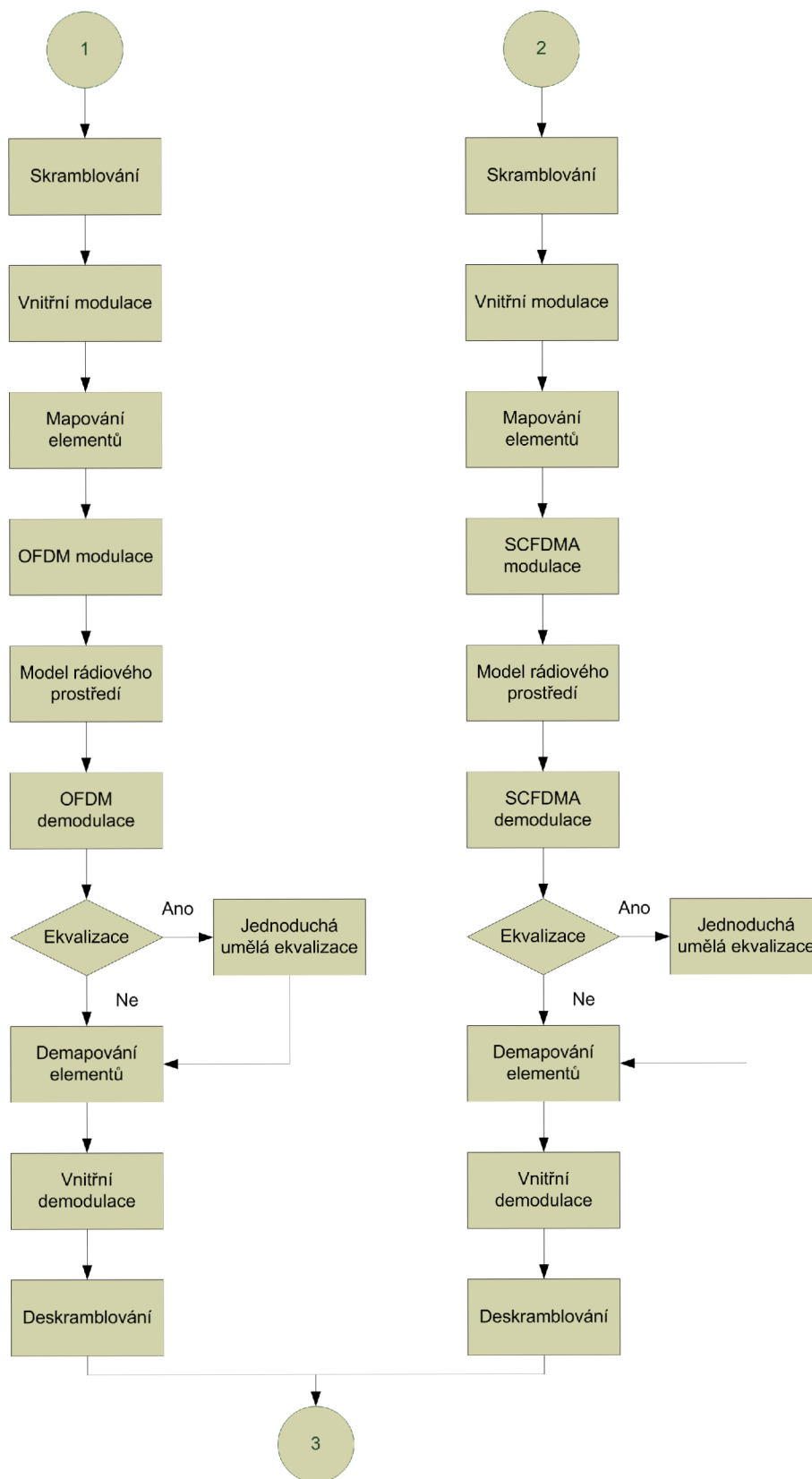
Obr. 3.16: Příklad zobrazení závislosti chybovosti na poměru S/N pomocí modelu systému LTE

3.4 Blokový diagram

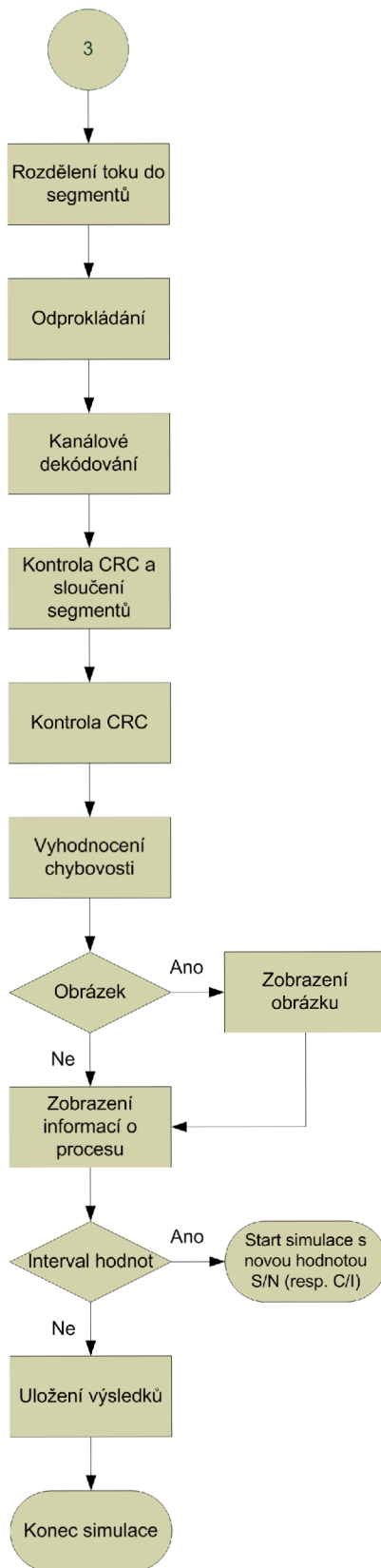
Tato sekce se zaměřuje na grafické znázornění funkce modelu systému LTE. Na obrázcích 3.17, 3.18 a 3.19 je znázorněn průběh procesu simulace.



Obr. 3.17: Blokové schéma modelu systému LTE, část 1



Obr. 3.18: Blokové schéma modelu systému LTE, část 2



Obr. 3.19: Blokové schéma modelu systému LTE, část 3

3.5 Popis blokového diagramu a vytvořených funkcí

Simulace začíná zmáčknutím tlačítka Start v hlavním okně programu.

Načtení nastavení modelu a zobrazení

Prvním krokem je načtení nastavení modelu a načtení nastavení zobrazení modelu. Tento krok se vykonává částečně při spuštění skriptu GUI.m, kdy je kontrolováno, jestli existují soubory s nastavením, pokud ano, načte se nastavení z nich. Pokud ne, použije se výchozí nastavení. Jestliže dojde ke změně parametrů při spuštěném skriptu GUI.m, bude toto nastavení předáno hlavnímu rozhraní ihned po zmáčknutí tlačítka uložit nastavení nebo tlačítka výchozí nastavení. Po zmáčknutí tlačítka start se toto nastavení přepíše do proměnných, podle kterých se řídí běh programu.

Kontrola správnosti zadaných hodnot

V tomto kroku je kontrolováno, zda byly hodnoty v zadávacích polích zadány ve správném formátu a jestli jsou v povoleném rozsahu. Pokud ano, skript pokračuje dál, pokud ne, špatně zadané pole zčervená, objeví se varovná informace a simulace končí. Uživatel musí opravit hodnotu a spustit znovu simulaci.

Vytvoření vstupní posloupnosti

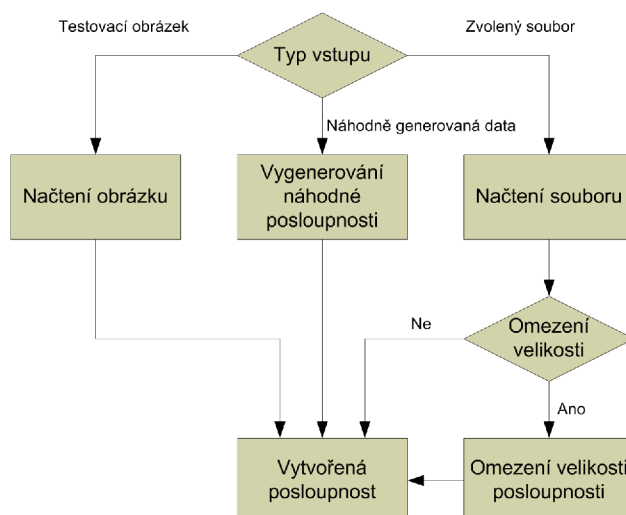
Na obrázku 3.20 je znázorněn proces vytvoření vstupní posloupnosti. Podle zvoleného typu vstupní posloupnosti jsou data načtena jako jednotlivé pixely testovacího obrázku. Nebo jsou data generována pomocí generátoru náhodné posloupnosti. Poslední možností je použít jako vstupní data soubor, který je reprezentován posloupností bitů. Pokud je jako vstupní datová posloupnost použit soubor, je možné načíst celý soubor nebo jeho velikost následně omezit.

Přidání CRC

Přidání kontrolního součtu se provádí podle postupu popsáného v kapitole 1.2. V modelu tento krok provádí funkce `add_CRC.m`.

Segmentace a přidání CRC k segmentům

Rozdělení transportního bloku na segmenty a přidání kontrolního součtu ke všem segmentům je popisováno v kapitole 1.3. V modelu tento krok provádí funkce `cb-Segmentation.m`.



Obr. 3.20: Blokové schéma vytvoření vstupní posloupnosti

Kanálové kódování

V bloku Kanálové kódování je vytvořená posloupnost zakódována pomocí turbo kodéru. Proces je popisován v kapitole 1.4. Operace provádí funkce `chCoding.m`.

Prokládání

Data jsou následně přeskládána podle definice v kapitole 1.5, neprovádí se pouze přizpůsobení datového toku. Tuto část provádí také funkce `chCoding.m`.

Sloučení dat do bloku

následně jsou data sloučena podle definice uvedené v kapitole 1.6. Operace se vykonává ve funkci `chCoding.m`.

Skramblování

Proces skramblování se liší v závislosti na použitém směru simulace. Skramblování na downlinku je popisováno v kapitole 1.7, provádí jej funkce `scrambling_dl.m`. V případě uplinku je proces popisován v kapitole 2.4 a provádí jej funkce `scrambling_ul.m`.

Vnitřní modulace

Proces vnitřní modulace je stejný jak pro downlink, tak pro uplink. Popis procesu naleznete v kapitole 1.8. Proces vykonává funkce `modulace.m`

Mapování elementů

Proces mapování elementů je popisován v kapitole 1.11 pro downlink a v kapitole 2.5 pro uplink. Proces zajišťuje umístění zakódovaných uživatelských dat do struktury radiového rámce a přidání pomocných signálů, jako jsou synchronizační signály a referenční signály. Mapování signálu na downlinku zajišťuje funkce `mapper_dl.m`, na uplinku funkce `mapovani_uplink.m`.

OFDM a SC-FDMA modulace

Proces OFDM modulace je popsán v kapitole 1.12 v případě downlinku, vykonává jej funkce `OFDMMod_dl.m`. V případě uplinku je SC-FDMA modulace popsána v kapitole 2.6, proces vykonává funkce `SCFDMAmod_ul.m`. Proces má za úkol převést symboly na časové vzorky signálu. V modelu se používá komplexní obálka signálu, nedochází tedy ke konverzi signálu do vysílacího pásma.

Model rádiového prostředí

Model rádiového prostředí je stejný pro uplink i downlink, jednotlivé modely kanálu jsou popsány v kapitole 1.13.

OFDM a SCFDMA demodulace

Proces OFDM demodulace je popisován v kapitole 1.14, vykonává jej funkce `OFDMDeMod_dl.m`. Proces SC-FDMA demodulace je popisován v kapitole 2.7, vykonává jej funkce `SCFDMAdemod_ul.m`. Proces převádí časové vzorky signálu na symboly, které jsou ukládány do matice reprezentující časově frekvenční strukturu radiového rámce.

Ekvalizace

Proces jednoduché umělé ekvalizace provádí napravení frekvenční charakteristiky přijatého signálu. Tento proces je však velmi zjednodušen a provádí korekci frekvenční charakteristiky tím, že část časově frekvenční struktury radiového rámce vynásobí převrácenou hodnotou podílu přijatých vzorků referenčních symbolů a rekonstruovaných ideálních referenčních symbolů v oblasti jednoho resource bloku, tzn. oblast o velikosti 12 subnosných ve frekvenční oblasti a 7 OFDM symbolů v časové oblasti. Ekvalizaci provádí funkce `vyrovnani_Kanalu_dl.m` v případě downlinku a `vyrovnani_Kanalu_ul.m` v případě uplinku.

Demapování elementů

Proces demapování elementů je popsán v kapitole 1.16 v případě downlinku a vykonává jej funkce Demapper_dl.m. V případě uplinku je proces popsán v kapitole 2.8 a vykonává jej funkce demapovani_uplink.m.

Vnitřní demodulace

Proces vnitřní demodulace probíhá podle popisu, který je uveden v kapitole 1.18. Tento proces je stejný pro downlink i uplink. Jeho hlavní úlohou je převést posloupnost symbolů na posloupnost bitů. Proces vykonává funkce demodulace.m.

Deskramblování

Proces deskramblování je popisován v kapitole 1.19 v případě downlinku a v kapitole 2.9 v případě uplinku. Hlavním úkolem procesu je obnovit zakódované bity, které byly skramblovány. Tento proces je zde kvůli snížení interferencí.

Rozdělení toku do segmentů

Proces Rozdělení toku do segmentů je popisován v kapitole 1.20. Tento proces je stejný pro downlink i uplink a má za úkol rozdělit zpracovávaná data na segmenty, které jsou dále zpracovávány v bloku Přizpůsobení datového toku a Kanálové dekódování. Proces je vykonáván ve funkci chCoding.m.

Odprokládání

Proces Odprokládání je součástí operace přizpůsobení datového toku, kde je vynecháno samotné přizpůsobování toku, ale ostatní části procesu jsou provedeny. Proces je popisován v kapitole 1.21. Proces je vykonáván ve funkci chCoding.m.

Kanálové dekódování

Proces kanálového dekódování je popsán v kapitole 1.22. Je shodný pro downlink i uplink a jeho hlavní funkcí je obnovit zakódovanou informaci tak, aby byla co nejvíce shodná s původní vysílanou sekvencí, před kanálovým kódováním. Proces je vykonáván ve funkci chCoding.m.

Kontrola CRC a sloučení segmentů

Proces kontroly kontrolních součtů a sloučení segmentů do bloku je popsán v kapitole 1.23. Tento proces je stejný pro downlink i uplink a má za úkol informovat systém o chybném přenosu. Proces vykonává funkce cbDeSegmentation.m.

Kontrola CRC

Proces kontroly kontrolního součtu je popisován v kapitole 1.24. Tento proces je sekundární ochranou proti chybnému vyhodnocení příjmu. Pokud by první kontrola u segmentů vyšla správně i s chybně přijatými daty, tak tato druhá kontrola by měla chybu naleznout. Pokud se přenáší malé bloky dat, tak k nim není přikládán další kontrolní součet u segmentů. V tomto případě je toto jediný proces, který se stará o odhalení chyby přenosu. Proces vykonává funkce `det_CRC.m`.

Vyhodnocení chybovosti

Vyhodnocení chybovosti je závěrečný krok simulace, kdy je spočítán chybový poměr, tedy poměr počtu chybných bitů ku přeneseným bitům. Mimo tento poměr lze zobrazit počet chybně detekovaných bitů a výsledky kontroly kontrolních součtů.

O toto vyhodnocení se stará funkce `[err nErr] = noOfErr(a, a_)`, kde a je původní sekvence, $a_$ je dekodovaná sekvence, $nErr$ je počet chyb a err je chybový poměr.

V grafu závislosti chybovosti na poměru S/N (resp. C/I) je pak výsledná chybovost upravena tak, aby byla splněna podmínka stanovení chybovosti s několikanásobně vyššího počtu bitů. Konkrétně vytvořený model stanovuje chybovost z 10x většího počtu bitů. To znamená, že pokud je spočítána chybovost 10^{-5} pro sekvenci dat dlouhou právě 10^5 bitů bude výsledná chybovost stanovena na hodnotu 10^{-4} .

3.6 Tabulka simulací

V této sekci je vypsán seznam simulací. Výsledky simulací jsou detailněji popsány v kapitole 4. Seznam je rozdělen do pěti částí. V první části je uvedena tabulka se společným nastavením pro celou sadu simulací, tabulka A.9. Tato tabulka slouží ke zpřehlednění tabulek následujících, které obsahují pouze pole, které se mění. Tabulky obsahují název testu a úplné nastavení simulace. Důvodem je opakovatelnost simulace. Pokud se s daným nastavením spustí simulace podruhé, výsledky musí být shodné. Tabulky s nastavením pro jednotlivé testy jsou: A.10, A.11, A.12, A.13 a A.14.

4 VÝSLEDKY SIMULACÍ

Tato kapitola se zabývá dosaženými výsledky simulací modelu fyzické vrstvy systému LTE a popisem metody testování. Všechny simulované testy jsou přehledně uvedeny v tabulkách A.9, A.10, A.11, A.12, A.13 a A.14. Testování je rozděleno do několika sad, kdy každá sada má společné nastavení a mění se tak pouze jen část parametrů.

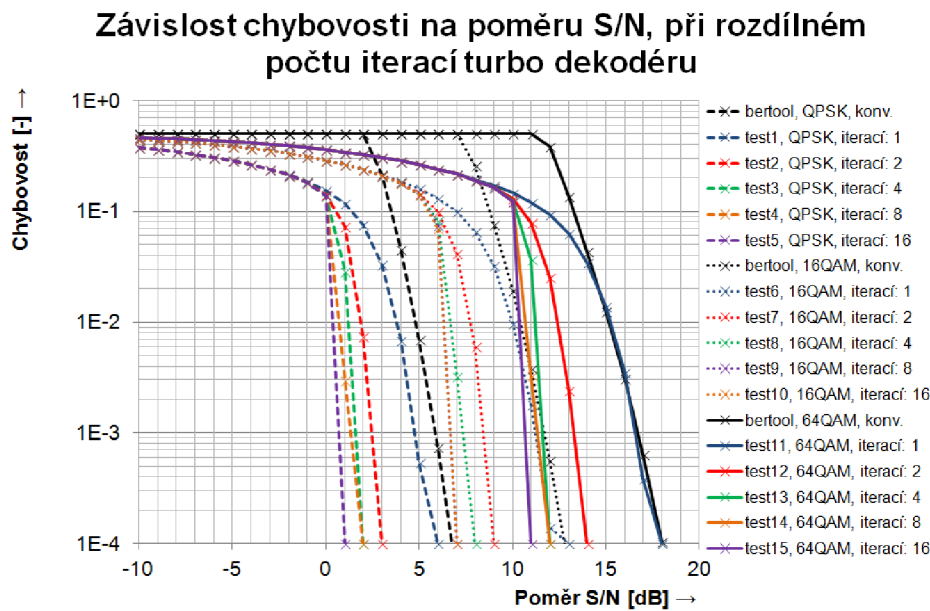
První sada testů se zaměřuje na zisk turbo dekodéru. Závislost chybového poměru na poměru S/N, případně C/I, kde dalším parametrem je počet iterací turbo dekodéru. Druhý sada se zaměřuje na prozkoumání vlivu použité modulace v závislosti na použití interferenčního nebo AWGN kanálu. Tento test je simulován pro uplink i downlink. Poslední sada testů je nejobsáhlejší a zabývá se testy s modelem rádiového prostředí. Je zde pozorován vliv rádiového kanálu. Sada testů obsahuje i testy se zapnutou jednoduchou umělou ekvalizací.

4.1 Metoda testování

Každý z testů používá jiné nastavení simulace, avšak velká část nastavení je stejná. Jednotlivé testy označeny číslem a v přehledné tabulce je vypsáno úplné nastavení simulace. Pokud jsou navíc zvolena stejná vstupní data, musí simulace pokaždé proběhnout se stejným výsledkem. Jako vstupní posloupnost je zvolen soubor s názvem `_test_file.JPG`, délka tohoto souboru je následně omezena na hodnotu uvedenou v tabulce. Vlastnost dosažení shodného výsledku je dosažena použitím prvku `seed` u generátoru náhodných čísel pro rádiový kanál a šum, tedy AWGN kanál. Tímto je zaručena opakovatelnost testu. Pro všechny testy platí omezení, která jsou uvedena v kapitole 3.2.

4.2 Zisk turbo dekodéru

Závislost chybovosti turbo kodéru na poměru S/N se mění s několika parametry. Prvním parametrem je velikost bloku dat, který je zakódován. Dalším parametrem je počet iterací, které turbo kodér provádí před výsledným rozhodnutím. Posledním parametrem je typ použitého interního prokladače, který má také vliv na výslednou chybovost. Z důvodu existence mnoha parametrů se špatně vyhledávají křivky závislosti chybovosti na poměru S/N , které by byly vhodné pro porovnání s dosaženými výsledky. Proto pro porovnání uvádím teoretickou chybovost konvolučního kodéru, která byla získána pomocí nástroje bertool, který je součástí komunikačního toolboxu programu Matlab. Konvoluční kodér je definován strukturou `poly2trellis(7, [133 171 165])`, který je používán v LTE, s tvrdým rozhodováním. Výsledkem je však závislost chybovosti na poměru E_b/N_0 , ten je přepočítán pomocí vztahu $\frac{E_s}{N_0} = \frac{SB}{NR_s}$ uvedený v literatuře [11]. Úpravou dostáváme vztah $\frac{S}{N} = \frac{E_s}{N_0} \cdot \frac{R_s}{B}$, symbolová rychlost se dá vyjádřit jako $R_s = R_b/m$, pro energii symbolu platí $E_s = E_b \cdot m$. Úpravou dostáváme výsledný vztah $\frac{S}{N} = \frac{E_b}{N_0} + 10 \cdot \log \frac{R_b}{B}$, kde hodnoty poměru S/N a E_b/N_0 jsou zadány v decibelové míře. Pak jsou výsledné křivky vloženy do grafu k porovnání a zobrazeny na obrázku 4.1.



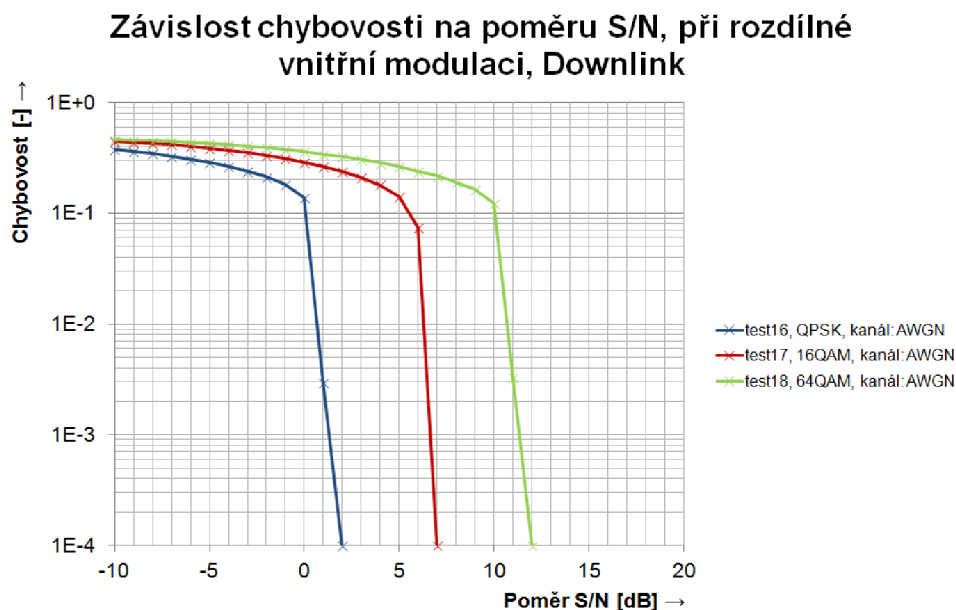
Obr. 4.1: Závislost chybovosti na poměru S/N , při proměnném počtu iterací

Výsledný teoretický průběh závislosti chybovosti na poměru S/N konvolučního kodéru se velmi přibližuje závislosti turbo kodéru při jedné iteraci. Toto platí pro všechny modulace. Dále lze pozorovat zisk turbo kodéru, který je pro modulaci

QPSK přibližně roven hodnotě 5 dB, pro modulace 16QAM asi 6 dB a pro modulaci 64QAM přibližně 7 dB, což je velmi výrazné zlepšení. Také je vidět, že největší rozdíl je mezi první a druhou iterací. Rozdíl mezi 8 a 16 iteracemi je asi 1 dB, pro všechny modulace. Délka simulace, při použití 16 iterací turbo dekodéru místo 8 iterací, je téměř dvojnásobná.

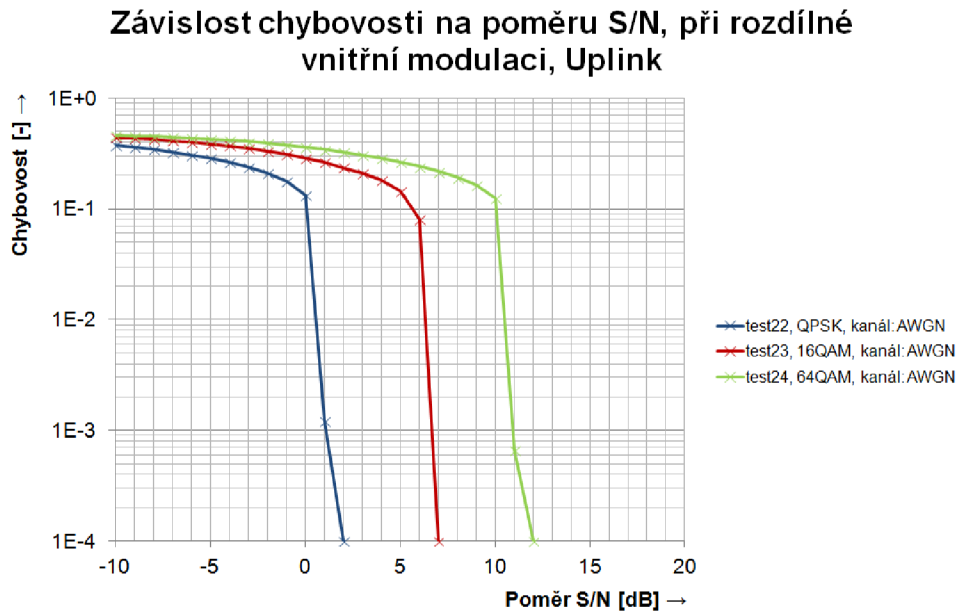
4.3 Vliv použité modulace

Je předpokládáno, že modulace s menším počtem stavů bude dosahovat menší chybovosti při konkrétním poměru S/N (resp C/I), jak modulace s vyšším počtem stavů. Tento předpoklad je založen na znalosti faktu, že vzdálenost symbolů u modulace s vyšším počtem stavů je menší, jak u modulace s vyšším počtem stavů. Šum nebo interference tak více znesnadňují detekci správného symbolu. Vliv použité modulace je sledován na downlinku i uplinku.

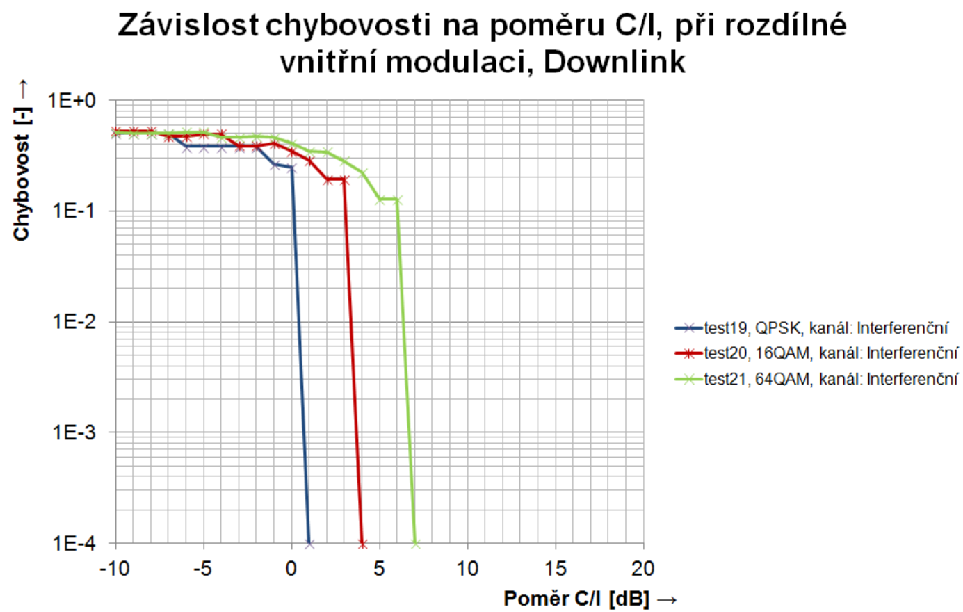


Obr. 4.2: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, Downlink

Pokud srovnáme průběhy na obrázcích 4.2 a 4.3 zjistíme, že rozdíl v průbězích závislosti chybovosti na poměru S/N na downlinku a uplinku není nijak významný, oba směry dosahují stejných parametrů. Rozdíl mezi modulacemi QPSK a 16QAM je 5 dB, rozdíl mezi 16QAM a 64QAM je také 5 dB.

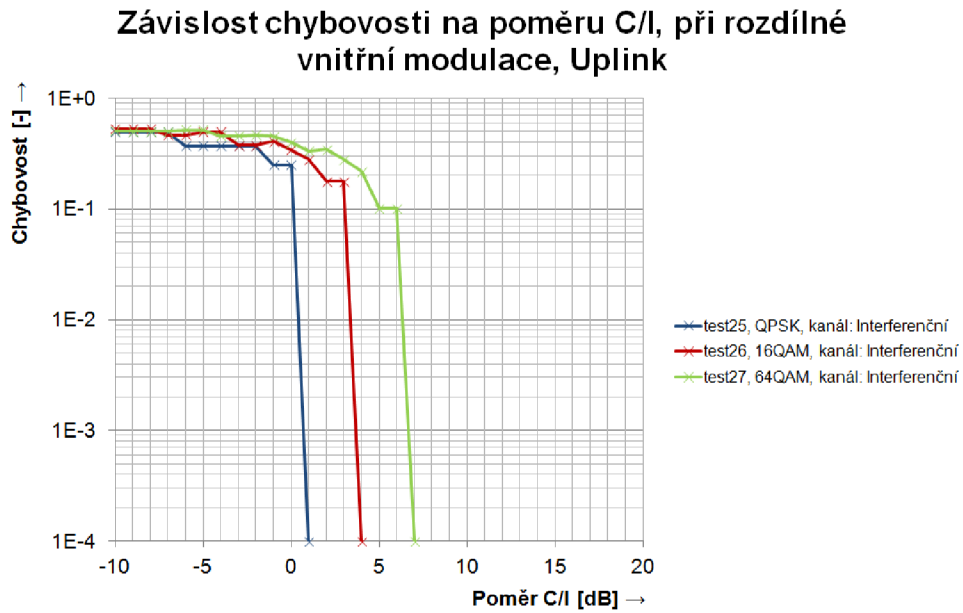


Obr. 4.3: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, Uplink



Obr. 4.4: Závislost chybovosti na poměru C/I, při rozdílné vnitřní modulaci, Downlink

U průběhů závislosti chybovosti na poměru C/I, kde se používá interferenční kanál je pozorován schodovitý tvar křivky (viz. obr. 4.4). Toto je pravděpodobně způsobeno tím, že interferenční signál používá modulaci 64QAM.



Obr. 4.5: Závislost chybovosti na poměru C/I, při rozdílné vnitřní modulaci, Uplink

Pokud porovnáváme průběhy na obrázcích 4.4 a 4.5, tak rozdíl downlinku a uplinku, při použití interferenčního kanálu není patrný. Oba simulované směry mají téměř shodné charakteristiky. Pokud srovnáme kanály AWGN a Interferenční, zjistíme, že při modulaci QPSK nedochází k výraznému rozdílu. Rozdíl při chybovosti 10^{-4} je o 1 dB nižší poměr C/I u interferenčního kanálu. Velký rozdíl je patrný u modulace 16QAM, kde je rozdíl už 3 dB a u modulace 64QAM dokonce 5 dB. Interferenční kanál s interferenčním signálem, který používá modulaci 64QAM, dosahuje výrazně nižších chybovostí než v případě AWGN kanálu.

4.4 Simulace poškození signálu v rádiovém prostředí

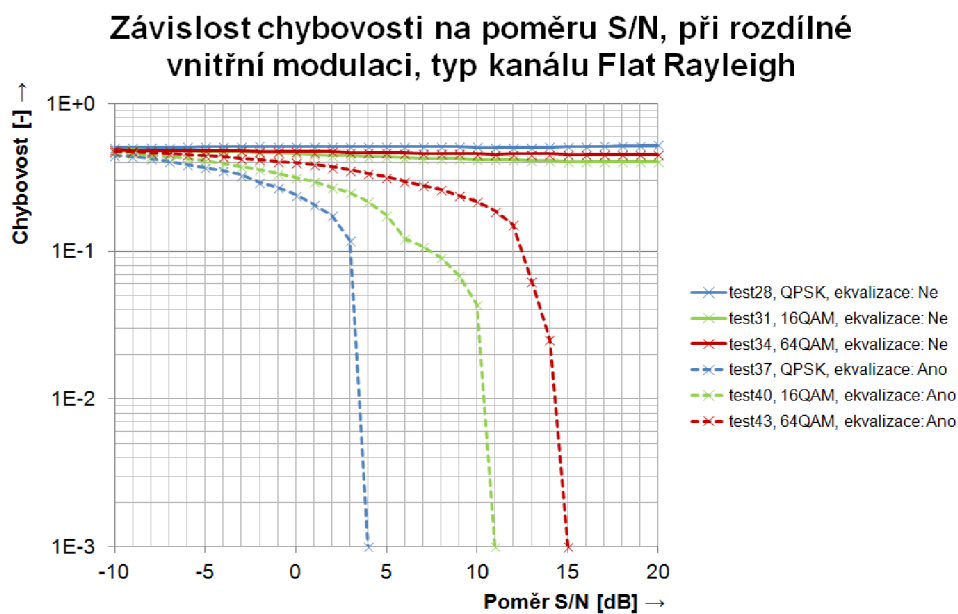
V této kapitole je porovnávána závislost chybovosti na poměru S/N, při použití různých modelů rádiového kanálu. Vliv jednotlivých modelů rádiových kanálů na přenášený signál je popsán v kapitole 3.3.1. Ze závislostí vyplývá, že pokud není vliv kanálu uměle snižován a není použita žádná metoda vyrovnání kanálu, tak model není schopen dosáhnout ani chybovosti 10^{-1} , čímž se stává naprosto nepoužitelný pro jakýkoli přenos. Pro kvalitní přenos dat je potřebné dosáhnout chybovosti alespoň 10^{-6} . Chybovost 10^{-9} je pak hranicí pro bezproblémovou komunikaci.

4.4.1 Downlink

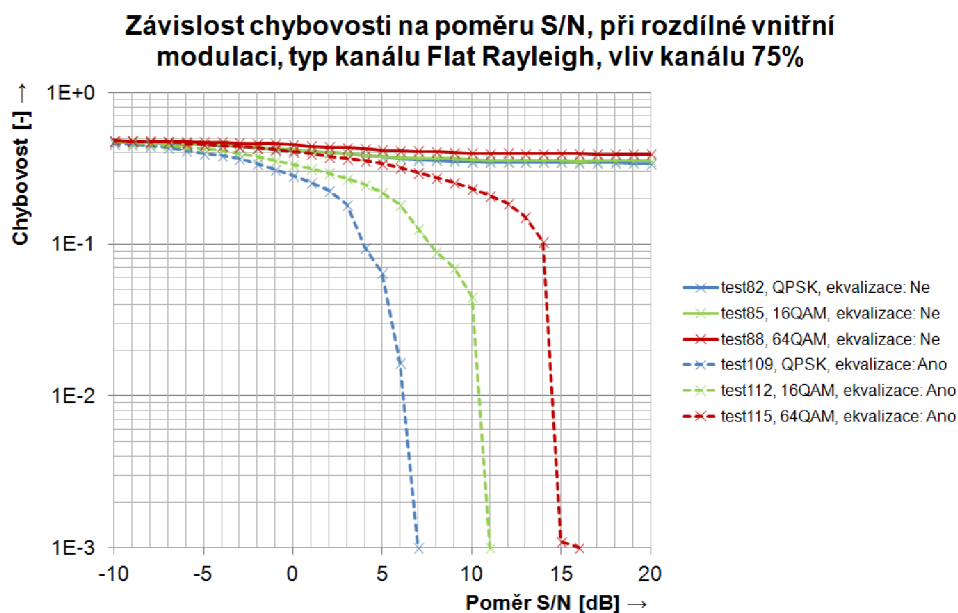
Simulace prováděné na downlinku se zaměřují na porovnání jednotlivých modulací v závislosti na použitém modelu prostředí, dále je zkoumán vliv kanálu na výslednou chybovost s použitím jednoduché umělé ekvalizace a bez ní.

kanál flat Rayleigh

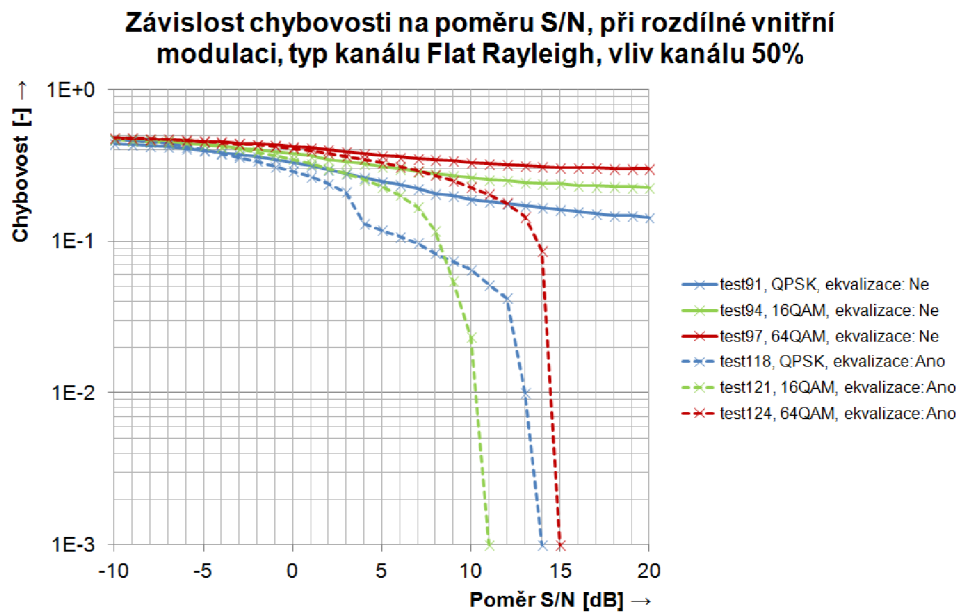
Účinky kanálu typu flat Rayleigh jsou dobře potlačitelné i jednoduchou umělou ekvalizací, jak je vidět na obrázku 4.6. Z obrázků 4.7, 4.8 a 4.7 je patrné, že pokud se snižuje vliv kanálu, tak dochází ke zlepšení chybovosti. Při 25% vlivu kanálu je poškození signálu tak slabé, že modulace QPSK dosahuje chybovosti 10^{-3} při poměru S/N = 5 dB. Na obrázku 4.8 dochází k tomu, že chybovost při použití modulace QPSK, při poměru S/N = 10dB, je horší, než chybovost v případě použití modulace 16QAM. K takovýmto chybám dochází vlivem nedokonalosti jednoduché ekvalizace a její velké náchylnosti na velikost šumu, který je přidán k referenčním symbolům. I při použití umělé ekvalizace je však výsledný poměr S/N průměrně o 3 dB horší než u kanálu AWGN, při chybovosti 10^{-3} .



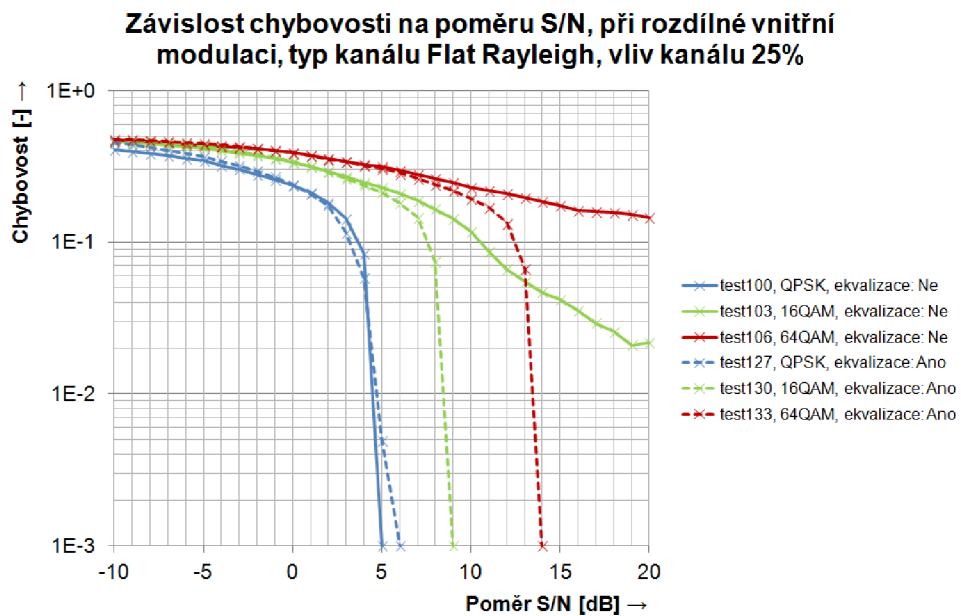
Obr. 4.6: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh



Obr. 4.7: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh, vliv kanálu 75 %



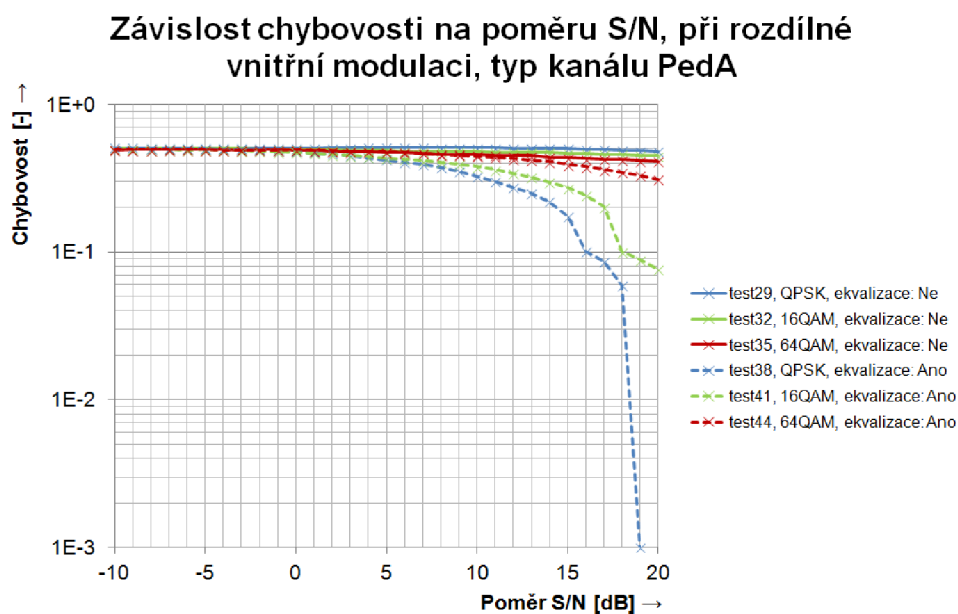
Obr. 4.8: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh, vliv kanálu 50 %



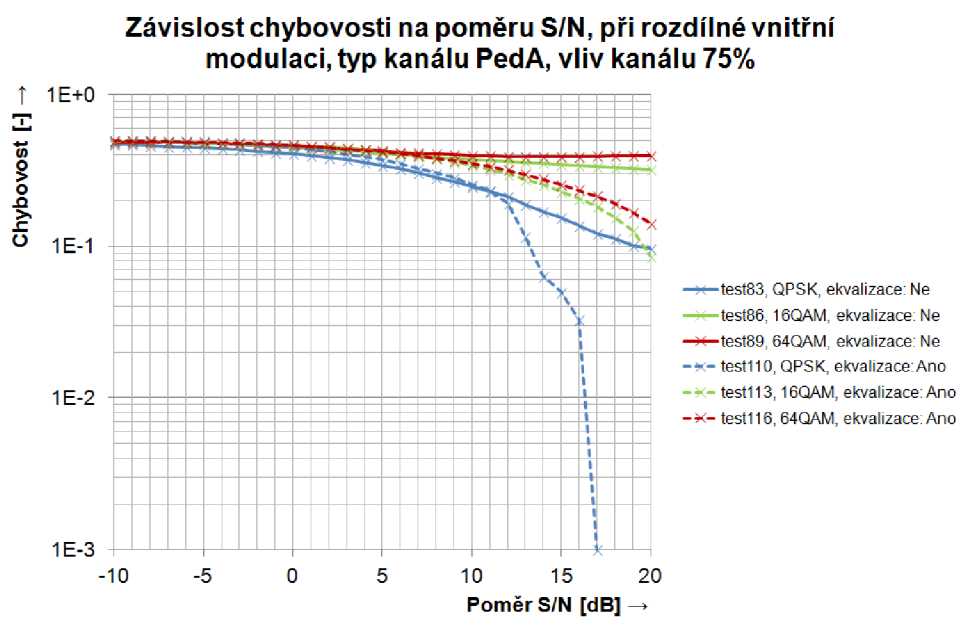
Obr. 4.9: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu flat Rayleigh, vliv kanálu 25 %

kanál PedA

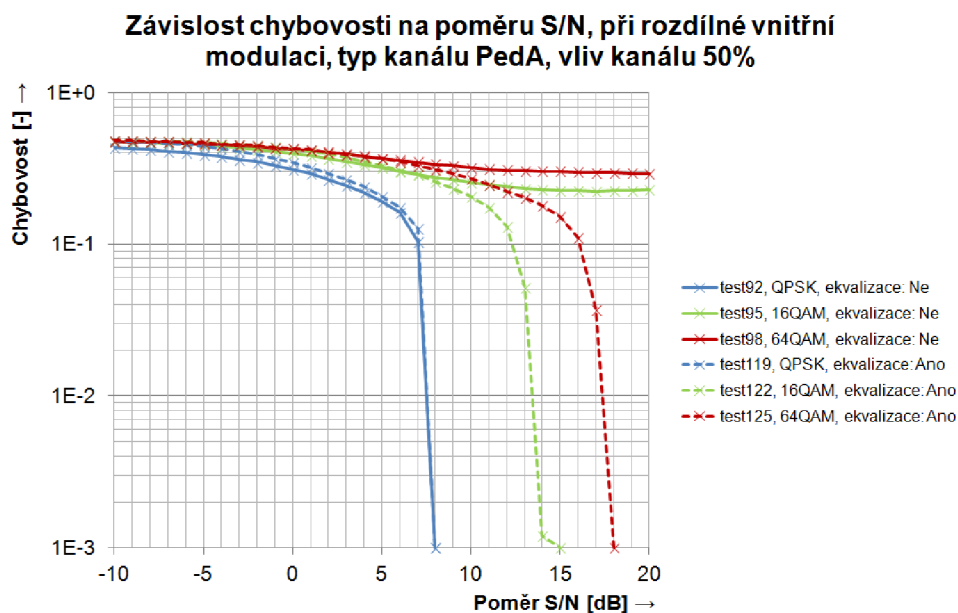
Při použití modelu rádiového kanálu Pedestrian A je umělá ekvalizace jen velmi málo účinná. Pokud je vliv kanálu snižován (viz. obr. 4.11, 4.12 a 4.13, tak dochází ke zlepšení chybovosti, toto zlepšení je nejvíce patrné na obrázku 4.13, kdy modulace 16QAM dosahuje lepšího poměru S/N bez jednoduché umělé ekvalizace, než v případě kanálu flat Rayleigh. Při použití jednoduché umělé ekvalizace je výsledná chybovost při 25% vlivu kanálu srovnatelná s chybovostí kanálu flat Rayleigh s 25% vlivem kanálu.



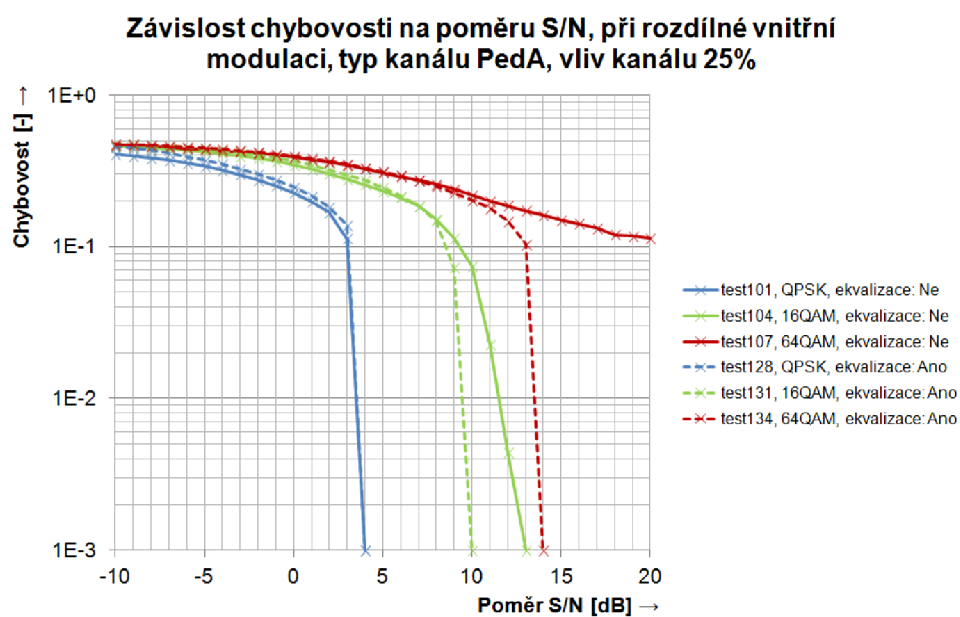
Obr. 4.10: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA



Obr. 4.11: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, vliv kanálu 75 %



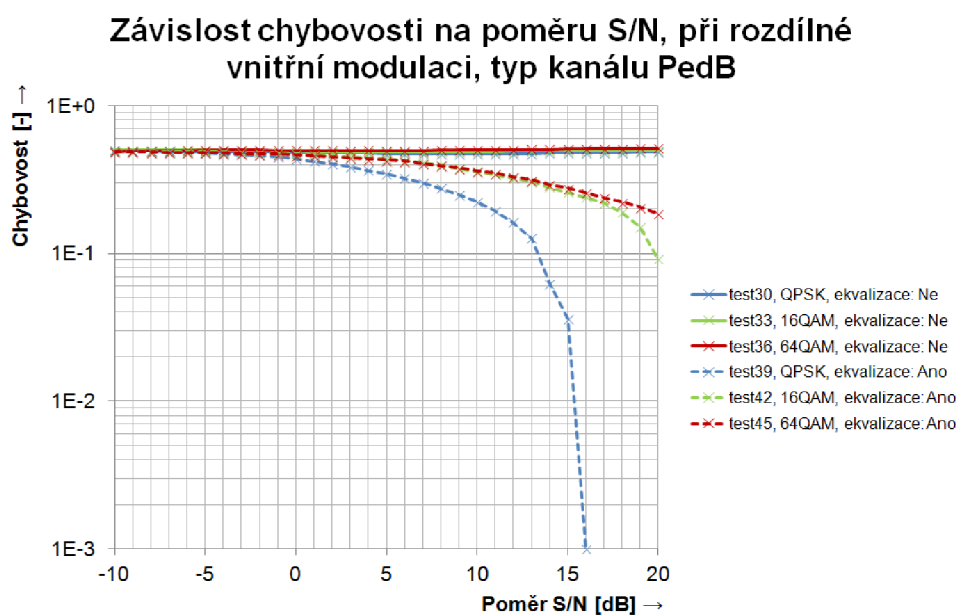
Obr. 4.12: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, vliv kanálu 50 %



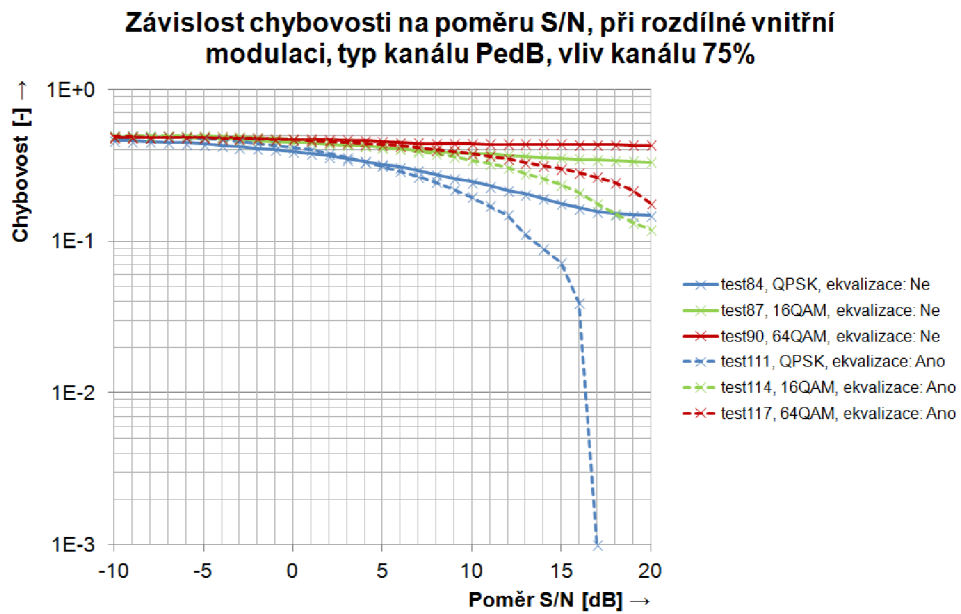
Obr. 4.13: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, vliv kanálu 25 %

kanál PedB

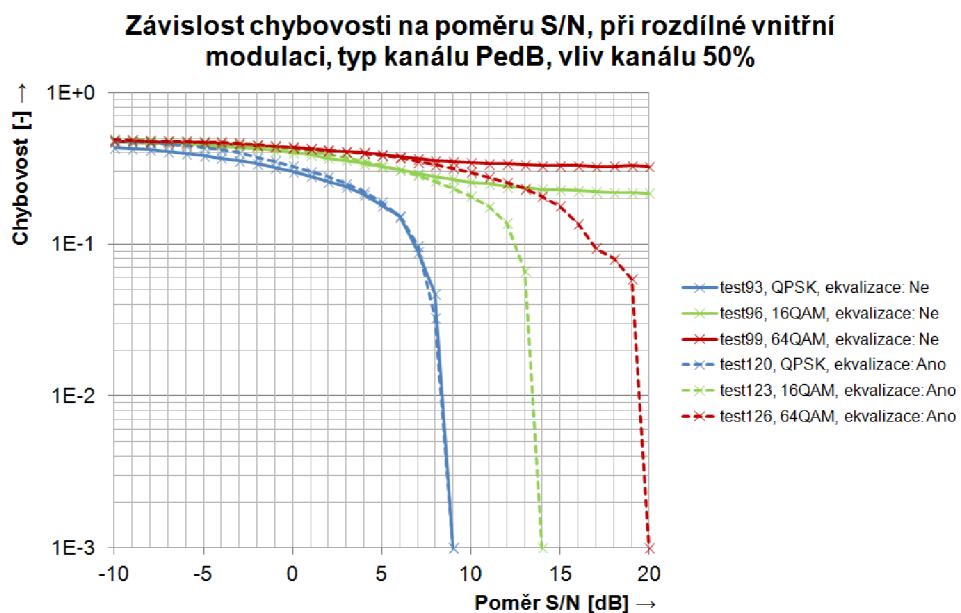
Výsledky simulací kanálu Pedestrian B se velmi podobají výsledkům simulací modelu Pedestrian A. Jediný větší rozdíl je při použití jednoduché umělé ekvalizace a modulace QPSK, při 100% vlivu kanálu, kdy k dosažení chybovosti 10^{-3} je třeba zvýšit poměr S/N u kanálu typu Pedestrian A o 3 dB (viz. obr. 4.14). Jinak nejsou výsledky simulací rozdílnější o více než 1 dB. Také zde platí, že jednoduchá umělá ekvalizace není pro tento typ kanálu účinná. Její účinnost se zvyšuje až s umělým snižováním vlivu kanálu. Změna parametru *vlivkanálu* je demonstrována obrázky 4.15, 4.16 a 4.17.



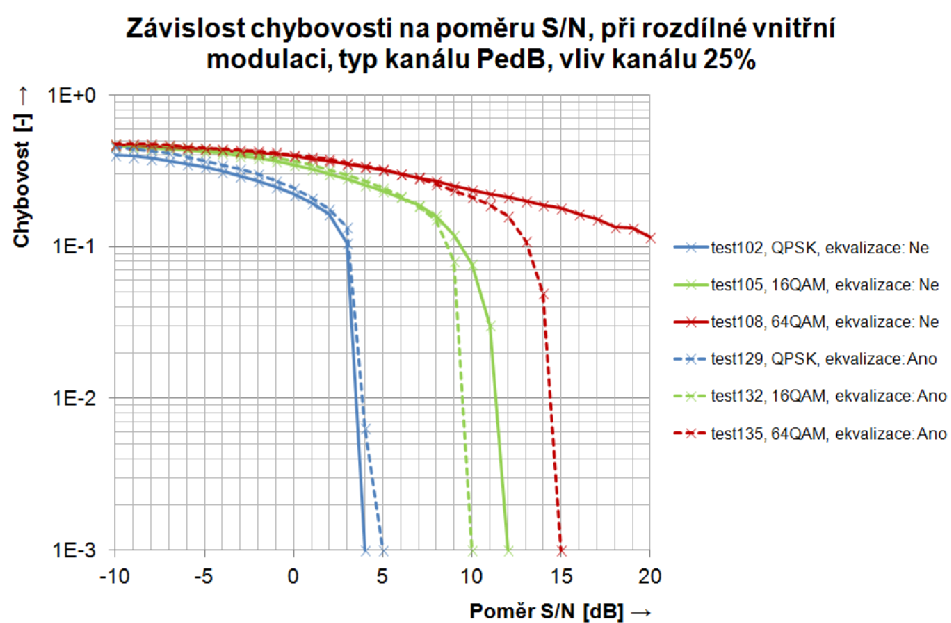
Obr. 4.14: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB



Obr. 4.15: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, vliv kanálu 75 %



Obr. 4.16: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, vliv kanálu 50 %



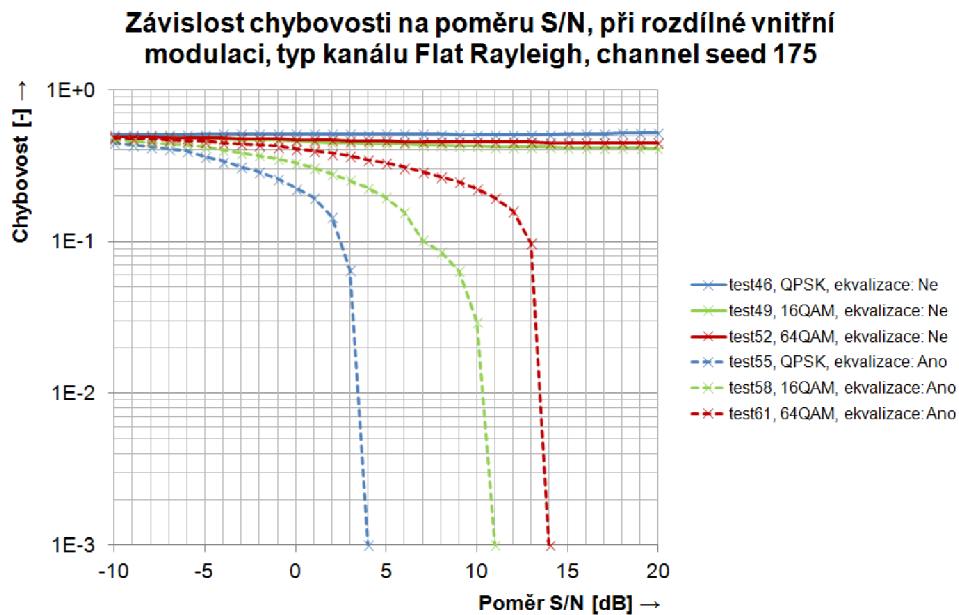
Obr. 4.17: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, vliv kanálu 25 %

4.4.2 Uplink

Simulace prováděné na uplinku se soustřeďují k vyšetření vlivu použití různého parametru *channelseed*, více informací o tomto parametru je uvedeno v kapitole 3.3.2. Při dostatečně dlouhé simulaci by měl být rozdíl různého 'channel seed' nepatrný. Stejně jako v případě downlinku platí, že pokud jednoduchá umělá ekvalizace není používána, je chybovost velmi vysoká. Při poměru $S/N = 20$ dB je chybovost stále blízká hodnotě 0,5.

kanál flat Rayleigh

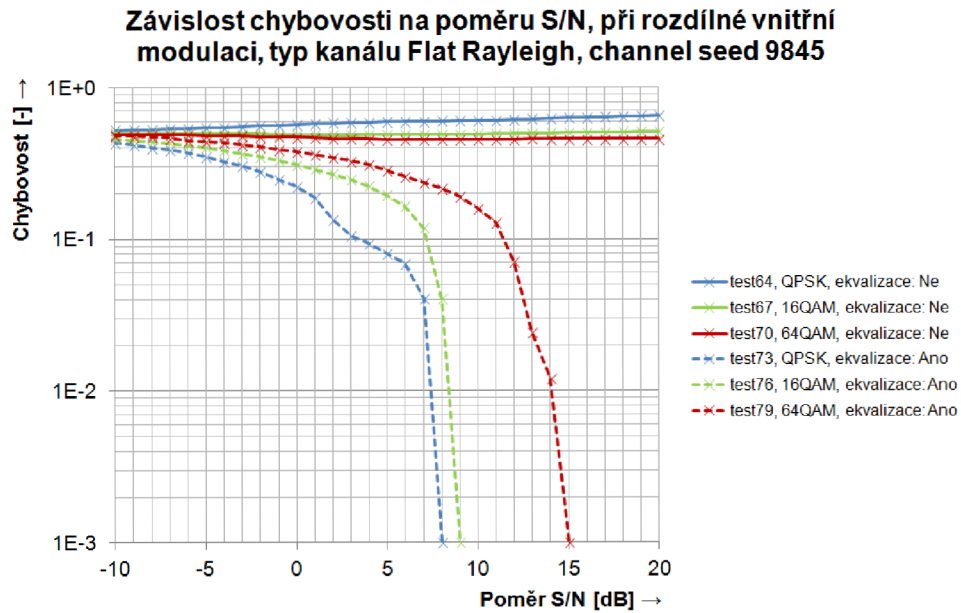
Vliv různého *channelseed* u kanálu typu flat Rayleigh je velice významný především pro modulaci QPSK, při použití jednoduché umělé ekvalizace (viz. obr. 4.18 a 4.19). Zde je rozdíl poměru S/N , kdy chybovost dosahuje hodnoty 10^{-3} , 4 dB. U modulace 16QAM jsou to už pouze 2 dB a u modulace 64QAM pouze 1 dB. I tak jsou průběhy chybovostí značně rozdílné.



Obr. 4.18: Závislost chybovosti na poměru S/N , při rozdílné vnitřní modulaci, typ kanálu Flat Rayleigh, channel seed 175

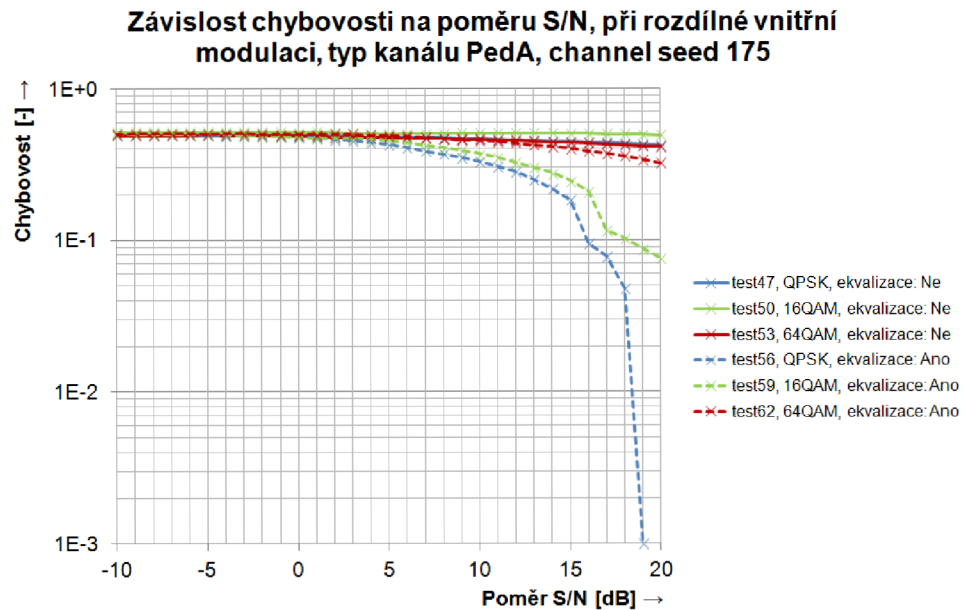
kanál PedA

Závislost parametru *channelseed* je patrná i při použití kanálu Pedestrian A. Rozdíl poměru S/N , při modulaci QPSK a stejné chybovosti 10^{-3} , je 5dB (viz. obr. 4.20 a

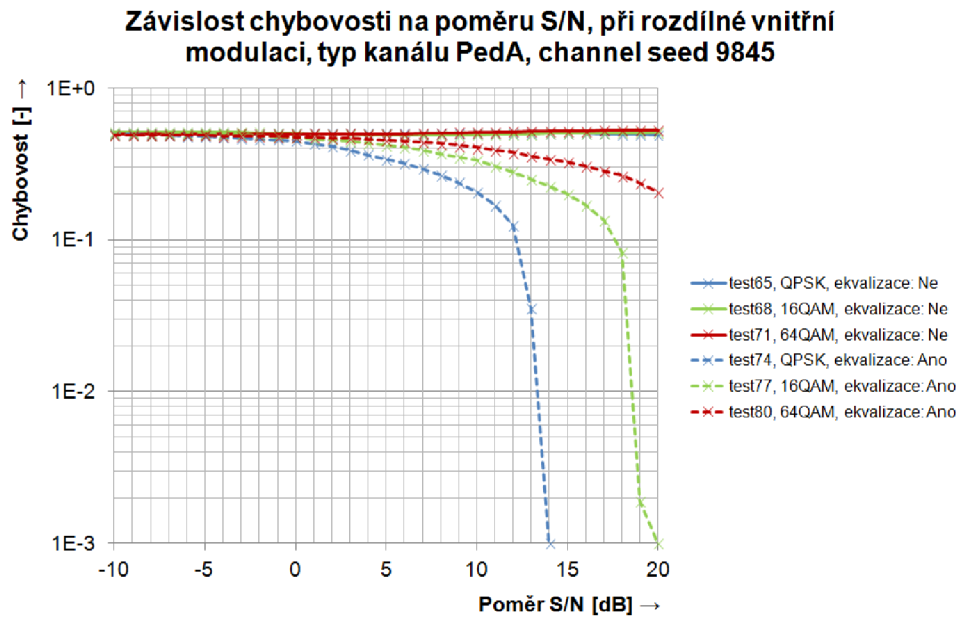


Obr. 4.19: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu Flat Rayleigh, channel seed 9845

4.21). I při takovémto rozdílu je chybovost stále velmi vysoká a model systému je pro přenos dat, bez dokonalejší ekvalizace, nepoužitelný.



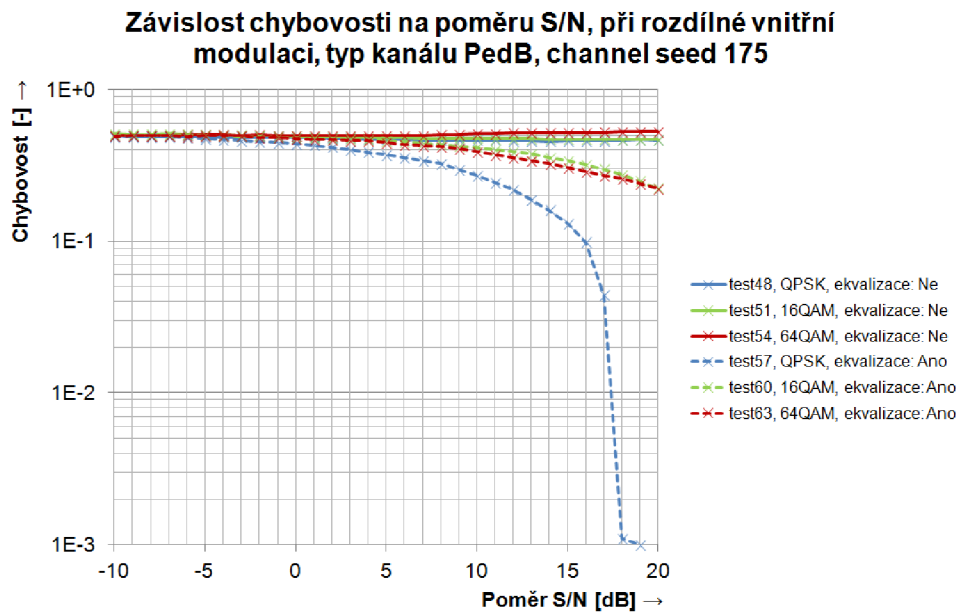
Obr. 4.20: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, channel seed 175



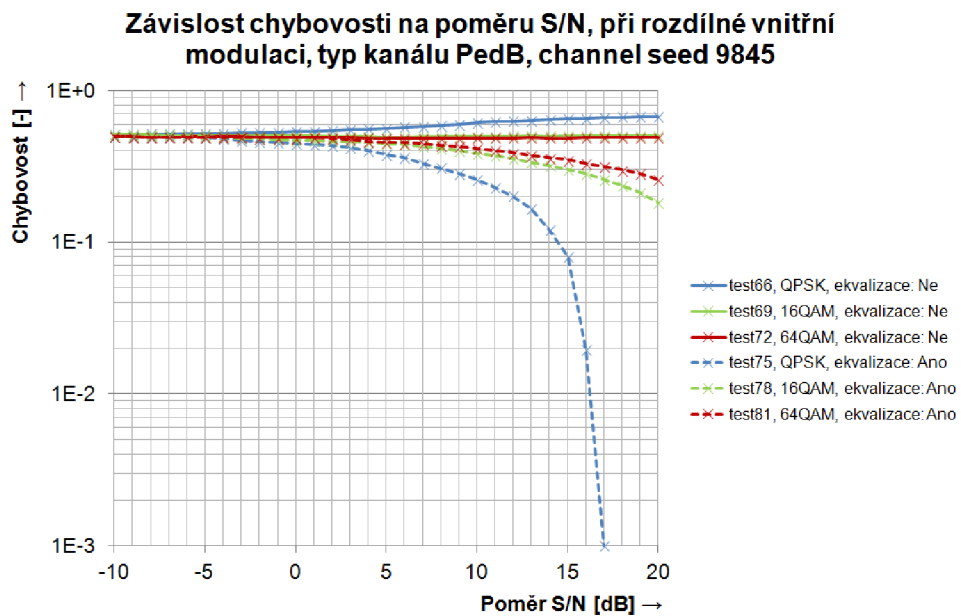
Obr. 4.21: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedA, channel seed 9845

kanál PedB

Výsledky simulací, při použití kanálu typu Pedestrian B, jsou podobné výsledkům simulace při použití kanálu typu Pedestrian A, s tím rozdílem, že vliv parametru *channelseed* je zde jen velmi malý (viz. obr. 4.22 a 4.23). Chybovost je u obou zvolených hodnot parametru velmi vysoká. Pokles chybovosti je pozorován pouze při použití jednoduché umělé ekvalizace a vysokém poměru S/N = 19 dB.



Obr. 4.22: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, channel seed 175



Obr. 4.23: Závislost chybovosti na poměru S/N, při rozdílné vnitřní modulaci, typ kanálu PedB, channel seed 9845

5 ZÁVĚR

Diplomová práce se zabývá komunikačním systémem LTE. Pozornost je zaměřena na detailní popsání zpracování signálu na fyzické vrstvě.

Nejprve je podle specifikace systému LTE vytvořeno blokové schéma zpracování signálu na downlinku, které je pak postupně, blok po bloku, popisováno. Poté je vytvořeno blokové schéma zpracování signálu na uplinku. Následně jsou popsány rozdíly zpracování signálu na downlinku a uplinku a popsány bloky, které se na downlinku nevyskytují nebo se v nich signál zpracovává rozdílným způsobem.

Na základě blokových schémat a uvedeného popisu je vytvořen matematický model systému LTE. Model obsahuje řadu zjednodušení, které spočívají v zaměření se na přenos uživatelských dat, další omezení byly udělány na základě dohody s vedoucím diplomové práce. Všechny omezení jsou zmíněny v části popisující vytvořený model.

Funkčnost modelu je ověřena dosažením nulové chybovosti, tedy stavu, kdy přijatá posloupnost je shodná s vyslanou. Dále je možné zobrazit IQ diagram, ze kterého je patrné, že přijaté symboly nejsou nijak zkreslené a shodují se s vyslanými.

V poslední části práce je zkoumán vliv rádiového prostředí na chybovost v závislosti na poměru S/N (resp. C/I). Nejprve je zkoumán vliv nastavení počtu iterací turbo dekodéru, při použití AWGN kanálu, v závislosti na poměru S/N. Výsledná chybovost je pro názornost porovnávána s teoretickým průběhem konvolučního kodéru. V případě AWGN kanálu pracuje systém velmi dobře. Zisk turbo kodéru oproti konvolučnímu kodéru je 5 dB při chybovosti 10^{-4} .

V dalším kroku jsou porovnávány závislosti chybovosti na poměru S/N u interferenčního a AWGN kanálu. Především při použití interferenčního kanálu model dosahuje velmi nízkých chybovostí, tuto skutečnost nejspíše ovlivňuje zvolená modulace interferenčního signálu 64QAM.

V posledním kroku jsou pozorovány vlivy různých prostředí na chybovost v závislosti na poměru S/N. Pokud není použita jednoduchá umělá ekvalizace a vliv kanálu není uměle snižován, tak model nedokáže obnovit zakódovanou informaci ani při poměru S/N = 20 dB. Chybovost při tomto poměru, tedy S/N = 20 dB, se pohybuje kolem hodnoty 0,5. Při použití jednoduché umělé ekvalizace se v závislosti na použitém typu modelu rádiového prostředí situace zlepšuje. Taktéž dochází ke zlepšení při umělém snižování vlivu kanálu.

Modelovaný systém se zaměřuje na přenos pomocí jedné vysílací a jedné přijímací antény, proto by bylo velkým přínosem vylepšit zejména model přijímače. Zejména použití vyrovnávače typu MMSE by mělo mít velký vliv na zlepšení dekódování při použití různých modelů rádiového prostředí. K dalšímu výraznému zlepšení by přispělo využití systému HARQ.

LITERATURA

- [1] *3GPP Specification series: 36series* [online]. 2012 [cit. 2012-12-10]. Dostupné z: <http://www.3gpp.org/ftp/Specs/html-info/36-series.htm>
- [2] MISHRA, Ajay R. *Advanced Cellular Network Planning and Optimisation: 2G/2.5G/3G...Evolution to 4G* Chichester: John Wiley, 2007, 521 s. ISBN 04-700-1471-7.
- [3] SAUTER, M. *From GSM to LTE: an introduction to mobile networks and mobile broadband* Chichester: John Wiley, 2011, 414 s. ISBN 978-0-470-66711-8.
- [4] SESIA, S., TOUFIK, I., BAKER, M. *LTE-the UMTS long term evolution: from theory to practise* Chichester: John Wiley, 2009, 611 s. ISBN 978-047-0697-160.
- [5] *MATLAB Documentation* [online]. 2013 [cit. 2013-05-21]. Dostupné z: <http://www.mathworks.com/help/matlab/index.html>
- [6] WESOŁOWSKI, K. *Mobile communication systems* Chichester: John Wiley, 2001, 449 s. ISBN 04-714-9837-8.
- [7] STASIAK, M., GŁABOWSKI, M., WISNIEWSKI, A. *Modeling and dimensioning of mobile networks: from GSM to LTE* Chichester, West Sussex: John Wiley, 2011, 315 s. ISBN 978-047-066586-2.
- [8] KROUK, E., SEMENOV, S. *Modulation and coding techniques in wireless communications* Chichester, West Sussex: John Wiley, 2011, 662 s. ISBN 978-0-470-74505-2.
- [9] Mehlführer C., Wrulich M., Ikuno J. C., Bosanska D., and Rupp M., *Simulating the Long Term Evolution Physical Layer*, in Proc. of the 17th European Signal Processing Conference (EUSIPCO 2009), Aug. 2009, Glasgow, Scotland. [Online]. Dostupné z: http://publik.tuwien.ac.at/files/PubDat_175708.pdf
- [10] PROKOPEC J. *Systémy mobilních komunikací: sítě pro mobilní datové služby* Vyd. 1. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2012, 186 s. ISBN 978-80-214-4498-0.
- [11] MARŠÁLEK R. *Teorie rádiové komunikace* Vyd. 1. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2012, 143 s. ISBN 978-80-214-4503-1.

- [12] Mehlführer C., Wrulich M., Ikuno J. C., Simko M., Schwarz S. and Rupp M., *The Vienna LTE Simulators - Enabling Reproducibility in Wireless Communications Research*, in EURASIP Journal on Advances in Signal Processing, Vol. 2011. [Online]. Dostupné z: http://publik.tuwien.ac.at/files/PubDat_199104.pdf
- [13] SCHLEGEL, C. B., PÉREZ, L. C. *Trellis and turbo coding* New Jersey : IEEE Press, 2004. 386 s. ISBN 04-712-2755-2.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

(k,l) Index resource elementu, k ve frekvenční oblasti, l v časové oblasti

$a_{k,l}^{(p)}$ Hodnota resource elementu (k,l) , platí pro anténní port p

Δf Frekvenční rozestup mezi subnosnými

f_{UL_low} Spodní hraniční kmitočet pásma pro uplink

f_{UL_high} Horní hraniční kmitočet pásma pro uplink

f_{DL_low} Spodní hraniční kmitočet pásma pro downlink

f_{DL_high} Horní hraniční kmitočet pásma pro downlink

p Číslo portu antény

N_{ID}^{cell} Identifikační číslo sektoru

N_{sc}^{RB} Počet subnosných v jednom resource bloku

N_{symp}^{RB} Počet symbolů v jednom resource bloku

N_{RB}^{DL} Počet resource bloků pro downlink, aktuální konfigurace

$N_{RB}^{max,DL}$ Počet resource bloků pro downlink, maximální konfigurace

N_{RB}^{UL} Počet resource bloků pro uplink, aktuální konfigurace

$N_{RB}^{max,UL}$ Počet resource bloků pro uplink, maximální konfigurace

n_{RNTI} číslo RNTI (Radio Network Temporary Identifier)

n_s číslo slotu v rádiovém rámci

T_f Doba trvání jednoho rámce

T_f Doba trvání jednoho rámce

$T_{půlrámec}$ Doba trvání jednoho půlrámce

T_s Základní časová jednotka

T_{slot} Doba trvání jednoho slotu

$T_{subrámec}$ Doba trvání jednoho subrámce

AMR Adaptive Multi-Rate Speech Codec

AWGN Additive white Gaussian noise

BPSK Binary Phase Shift Keying

BCH Broadcast channel

CDM Code Division Multiplex

CDMA Code Division Multiple Access

CFI Control Format Indicator

CS Circuit switched

CSD Circuit switched data

CRC Cyclic redundancy check

DCI Downlink Control Information

DFE Decision Feedback Equalizer

DFT Discrete Fourier Transform

DL-SCH Downlink Shared channel

EDGE Enhanced Data rates for GSM Evolution

FDD Frequency Division Duplex

FDM Frequency Division Multiplex

FDMA Frequency Division Multiple Access

FFT Fast Fourier Transform

GMSK Gaussian minimum shift keying

GPRS General packet radio service

GSM Global System for Mobile Communications

HARQ Hybrid automatic repeat request

HI HARQ indicator

IFFT Inverse Fast Fourier Transform

LSB Least significant bit

LTE Long Term Evolution

MAC Medium Access Control

MBSFN Multicast Broadcast single frequency network

MCH Multicast channel

MIMO Multiple Input Multiple Output

MMSE Minimum Mean Square Error

MS Mobile station

OFDM Orthogonal Frequency Division Multiplex

OFDMA Orthogonal Frequency Division Multiple Access

PAPR Peak-to-Average Power Ratio

PBCH Physical Broadcast channel

PCH Physical Control Format Indicator channel

PDSCH Physical Downlink Shared channel

PDCCH Physical Downlink Control channel

PCH Paging channel

PMCH Physical Multicast channel

PHICH Physical HARQ indicator channel

PN Pseudonáhodná posloupnost

PS Packet switched

PSK Phase-shift keying

PUSCH Physical Uplink Shared channel

PUCCH Physical Uplink Control channel

QPP Quadratic Permutation Polynomial

RI Rank indication

RPE-LTP Regular Pulse Excitation Long Time Prediction

RNC Radio Network Controller

RNTI Radio Network Temporary Identifier

QAM Quadrature amplitude modulation

QoS Quality of Service

QPSK Quadrature phase-shift keying

SC-FDMA Single Carrier - Frequency Division Multiple Access

SON Self-Organizing Network

TDD Time Division Duplex

TDM Time Division Multiplex

TDMA Time Division Multiple Access

TTI Transmission Time Interval

UE User Equipment

UL-SCH Uplink Shared channel

UMTS Universal Mobile Telecommunication System

WCDMA Wideband Code Division Multiple Access

WiMAX Worldwide Interoperability for Microwave Access

SEZNAM PŘÍLOH

A	Důležité číselné hodnoty v LTE	128
A.1	Převodní tabulky pro vnitřní modulace	128
A.2	Parametry pro vnitřní prokladač turbo kodéru	131
A.3	Parametry m_0 a m_1 pro referenční symboly	133
A.4	Definice $\varphi(n)$ pro referenční symboly	134
A.5	Výpis úplného nastavení jednotlivých testů	136

A DŮLEŽITÉ ČÍSELNÉ HODNOTY V LTE

A.1 Převodní tabulky pro vnitřní modulace

Tab. A.1: Převodní tabulka pro modulaci 16QAM

$b(i0), b(i+1), b(i+2), b(i+3)$	I	Q
0000	$1/\sqrt{10}$	$1/\sqrt{10}$
0001	$1/\sqrt{10}$	$3/\sqrt{10}$
0010	$3/\sqrt{10}$	$1/\sqrt{10}$
0011	$3/\sqrt{10}$	$3/\sqrt{10}$
0100	$1/\sqrt{10}$	$-1/\sqrt{10}$
0101	$1/\sqrt{10}$	$-3/\sqrt{10}$
0110	$3/\sqrt{10}$	$-1/\sqrt{10}$
0111	$3/\sqrt{10}$	$-3/\sqrt{10}$
1000	$-1/\sqrt{10}$	$1/\sqrt{10}$
1001	$-1/\sqrt{10}$	$3/\sqrt{10}$
1010	$-3/\sqrt{10}$	$1/\sqrt{10}$
1011	$-3/\sqrt{10}$	$3/\sqrt{10}$
1100	$1/\sqrt{10}$	$-1/\sqrt{10}$
1101	$1/\sqrt{10}$	$-3/\sqrt{10}$
1110	$3/\sqrt{10}$	$-1/\sqrt{10}$
1111	$3/\sqrt{10}$	$-3/\sqrt{10}$

$b(i0), b(i+1), b(i+2), b(i+3), b(i+4), b(i+5)$	I	\bar{Q}
000000	$3/\sqrt{42}$	$3/\sqrt{42}$
000001	$3/\sqrt{42}$	$1/\sqrt{42}$
000010	$1/\sqrt{42}$	$3/\sqrt{42}$
000011	$1/\sqrt{42}$	$7/\sqrt{42}$
001000	$5/\sqrt{42}$	$3/\sqrt{42}$
001001	$5/\sqrt{42}$	$1/\sqrt{42}$
001010	$7/\sqrt{42}$	$3/\sqrt{42}$
001011	$7/\sqrt{42}$	$1/\sqrt{42}$
001100	$5/\sqrt{42}$	$5/\sqrt{42}$
001101	$5/\sqrt{42}$	$7/\sqrt{42}$
001110	$7/\sqrt{42}$	$5/\sqrt{42}$
001111	$7/\sqrt{42}$	$7/\sqrt{24}$
010000	$3/\sqrt{42}$	$3/\sqrt{42}$
010001	$3/\sqrt{42}$	$-1/\sqrt{42}$
010010	$1/\sqrt{42}$	$-3/\sqrt{42}$
010011	$1/\sqrt{42}$	$-1/\sqrt{42}$
010100	$3/\sqrt{42}$	$-5/\sqrt{42}$
010101	$3/\sqrt{42}$	$-7/\sqrt{42}$
010110	$1/\sqrt{42}$	$-5/\sqrt{42}$
010111	$1/\sqrt{42}$	$-7/\sqrt{42}$
011000	$5/\sqrt{42}$	$-3/\sqrt{42}$
011001	$5/\sqrt{42}$	$-1/\sqrt{42}$
011010	$7/\sqrt{42}$	$-3/\sqrt{42}$
011011	$7/\sqrt{42}$	$-1/\sqrt{42}$
011100	$5/\sqrt{42}$	$-5/\sqrt{42}$
011101	$5/\sqrt{42}$	$-7/\sqrt{42}$
011110	$7/\sqrt{42}$	$-5/\sqrt{42}$
011111	$7/\sqrt{42}$	$-7/\sqrt{24}$

Tab. A.2: Převodní tabulka pro modulaci 64QAM - 1. část

Tab. A.3: Převodní tabulka pro modulaci 64QAM - 2. část

$b(i0), b(i + 1), b(i + 2), b(i + 3), b(i + 4), b(i + 5)$	I	Q
100000	$-3/\sqrt{42}$	$3/\sqrt{42}$
100001	$-3/\sqrt{42}$	$1/\sqrt{42}$
100010	$-1/\sqrt{42}$	$3/\sqrt{42}$
100011	$-1/\sqrt{42}$	$1/\sqrt{42}$
100100	$-3/\sqrt{42}$	$5/\sqrt{42}$
100101	$-3/\sqrt{42}$	$7/\sqrt{42}$
100110	$-1/\sqrt{42}$	$5/\sqrt{42}$
100111	$-1/\sqrt{42}$	$7/\sqrt{42}$
101000	$-5/\sqrt{42}$	$3/\sqrt{42}$
101001	$-5/\sqrt{42}$	$1/\sqrt{42}$
101010	$-7/\sqrt{42}$	$3/\sqrt{42}$
101011	$-7/\sqrt{42}$	$1/\sqrt{42}$
101100	$-5/\sqrt{42}$	$5/\sqrt{42}$
101101	$-5/\sqrt{42}$	$7/\sqrt{42}$
101110	$-7/\sqrt{42}$	$5/\sqrt{42}$
101111	$-7/\sqrt{42}$	$7/\sqrt{24}$
110000	$-3/\sqrt{42}$	$-3/\sqrt{42}$
110001	$-3/\sqrt{42}$	$-1/\sqrt{42}$
110010	$-1/\sqrt{42}$	$-3/\sqrt{42}$
110011	$-1/\sqrt{42}$	$-1/\sqrt{42}$
110100	$-3/\sqrt{42}$	$-5/\sqrt{42}$
110101	$-3/\sqrt{42}$	$-7/\sqrt{42}$
110110	$-1/\sqrt{42}$	$-5/\sqrt{42}$
110111	$-1/\sqrt{42}$	$-7/\sqrt{42}$
111000	$-5/\sqrt{42}$	$-3/\sqrt{42}$
111001	$-5/\sqrt{42}$	$-1/\sqrt{42}$
111010	$-7/\sqrt{42}$	$-3/\sqrt{42}$
111011	$-7/\sqrt{42}$	$-1/\sqrt{42}$
111100	$-5/\sqrt{42}$	$-5/\sqrt{42}$
111101	$-5/\sqrt{42}$	$-7/\sqrt{42}$
111110	$-7/\sqrt{42}$	$-5/\sqrt{42}$
111111	$-7/\sqrt{42}$	$-7/\sqrt{24}$

A.2 Parametry pro vnitřní prokladač turbo kodéru

Tab. A.4: Parametry pro interní prokladač turbodekodéru

i	K_i	f_1	f_2	i	K_i	f_1	f_2	i	K_i	f_1	f_2
1	40	3	10	33	296	19	74	65	592	19	74
2	48	7	12	34	304	37	76	66	608	37	76
3	56	19	42	35	312	19	78	67	624	41	234
4	64	7	16	36	320	21	120	68	640	39	80
5	72	7	18	37	328	21	82	69	656	185	82
6	80	11	20	38	336	115	84	70	672	43	252
7	88	5	22	39	344	193	86	71	688	21	86
8	96	11	24	40	352	21	44	72	704	155	44
9	104	7	26	41	360	133	90	73	720	79	120
10	112	41	84	42	368	81	46	74	736	139	92
11	120	103	90	43	376	45	94	75	752	23	94
12	128	15	32	44	384	23	48	76	768	217	48
13	136	9	34	45	392	243	98	77	784	25	98
14	144	17	108	46	400	151	40	78	800	17	80
15	152	9	38	47	408	155	102	79	816	127	102
16	160	21	120	48	416	25	52	80	832	25	52
17	168	101	84	49	424	51	106	81	848	239	106
18	176	21	44	50	432	47	72	82	864	17	48
19	184	57	46	51	440	91	110	83	880	137	110
20	192	23	48	52	448	29	168	84	896	215	112
21	200	13	50	53	456	29	114	85	912	29	114
22	208	27	52	54	464	247	58	86	928	15	58
23	216	11	36	55	472	29	118	87	944	147	118
24	224	27	56	56	480	89	180	88	960	29	60
25	232	85	58	57	488	91	122	89	976	59	122
26	240	29	60	58	496	157	62	90	992	65	124
27	248	33	62	59	504	55	84	91	1008	55	84
28	256	15	32	60	512	31	64	92	1024	31	64
29	264	17	198	61	528	17	66	93	1056	17	66
30	272	33	68	62	544	35	68	94	1088	171	204
31	280	103	210	63	560	227	420	95	1120	67	140
32	288	19	36	64	576	65	96	96	1152	35	72

Tab. A.5: Parametry pro interní prokladač turbodekodéru - pokračování

i	K_i	f_1	f_2	i	K_i	f_1	f_2	i	K_i	f_1	f_2
97	1184	19	74	129	2368	367	444	161	4416	35	138
98	1216	39	76	130	2432	265	456	162	4480	233	280
99	1248	19	78	131	2496	181	468	163	4544	357	142
100	1280	199	240	132	2560	39	80	164	4608	337	480
101	1312	21	82	133	2624	27	164	165	4672	37	146
102	1344	211	252	134	2688	127	504	166	4736	71	444
103	1376	21	86	135	2752	143	172	167	4800	71	120
104	1408	43	88	136	2816	43	88	168	4864	37	152
105	1440	149	60	137	2880	29	300	169	4928	39	462
106	1472	45	92	138	2944	45	92	170	4992	127	234
107	1504	49	846	139	3008	157	188	171	5056	39	158
108	1536	71	48	140	3072	47	96	172	5120	39	80
109	1568	13	28	141	3136	13	28	173	5184	31	96
110	1600	17	80	142	3200	111	240	174	5248	113	902
111	1632	25	102	143	3264	443	204	175	5312	41	166
112	1664	183	104	144	3328	51	104	176	5376	251	336
113	1696	55	954	145	3392	51	212	177	5440	43	170
114	1728	127	96	146	3456	451	192	178	5504	21	86
115	1760	27	110	147	3520	257	220	179	5568	43	174
116	1792	29	112	148	3584	57	336	180	5632	45	176
117	1824	29	114	149	3648	313	228	181	5696	45	178
118	1856	57	116	150	3712	271	232	182	5760	161	120
119	1888	45	354	151	3776	179	236	183	5824	89	182
120	1920	31	120	152	3840	331	120	184	5888	323	184
121	1952	59	610	153	3904	363	244	185	5952	47	186
122	1984	185	124	154	3968	375	248	186	6016	23	94
123	2016	113	420	155	4032	127	168	187	6080	47	190
124	2048	31	64	156	4096	31	64	188	6144	263	480
125	2112	17	66	157	4160	33	130				
126	2176	171	136	158	4224	43	264				
127	2240	209	420	159	4288	33	134				
128	2304	253	216	160	4352	477	408				

A.3 Parametry m_0 a m_1 pro referenční symboly

Tab. A.6: Závislost mezi $N_{ID}^{(1)}$ a proměnnými m_0 a m_1

$N_{ID}^{(1)}$	m_0	m_1	$N_{ID}^{(1)}$	m_0	m_1	$N_{ID}^{(1)}$	m_0	m_1	$N_{ID}^{(1)}$	m_0	m_1	$N_{ID}^{(1)}$	m_0	m_1
0	0	1	34	4	6	68	9	12	102	15	19	136	22	27
1	1	2	35	5	7	69	10	13	103	16	20	137	23	28
2	2	3	36	6	8	70	11	14	104	17	21	138	24	29
3	3	4	37	7	9	71	12	15	105	18	22	139	25	30
4	4	5	38	8	10	72	13	16	106	19	23	140	0	6
5	5	6	39	9	11	73	14	17	107	20	24	141	1	7
6	6	7	40	10	12	74	15	18	108	21	25	142	2	8
7	7	8	41	11	13	75	16	19	109	22	26	143	3	9
8	8	9	42	12	14	76	17	20	110	23	27	144	4	10
9	9	10	43	13	15	77	18	21	111	24	28	145	5	11
10	10	11	44	14	16	78	19	22	112	25	29	146	6	12
11	11	12	45	15	17	79	20	23	113	26	30	147	7	13
12	12	13	46	16	18	80	21	24	114	0	5	148	8	14
13	13	14	47	17	19	81	22	25	115	1	6	149	9	15
14	14	15	48	18	20	82	23	26	116	2	7	150	10	16
15	15	16	49	19	21	83	24	27	117	3	8	151	11	17
16	16	17	50	20	22	84	25	28	118	4	9	152	12	18
17	17	18	51	21	23	85	26	29	119	5	10	153	13	19
18	18	19	52	22	24	86	27	30	120	6	11	154	14	20
19	19	20	53	23	25	87	0	4	121	7	12	155	15	21
20	20	21	54	24	26	88	1	5	122	8	13	156	16	22
21	21	22	55	25	27	89	2	6	123	9	14	157	17	23
22	22	23	56	26	28	90	3	7	124	10	15	158	18	24
23	23	24	57	27	29	91	4	8	125	11	16	159	19	25
24	24	25	58	28	30	92	5	9	126	12	17	160	20	26
25	25	26	59	0	3	93	6	10	127	13	18	161	21	27
26	26	27	60	1	4	94	7	11	128	14	19	162	22	28
27	27	28	61	2	5	95	8	12	129	15	20	163	23	29
28	28	29	62	3	6	96	9	13	130	16	21	164	24	30
29	29	30	63	4	7	97	10	14	131	17	22	165	0	7
30	0	2	64	5	8	98	11	15	132	18	23	166	1	8
31	1	3	65	6	9	99	12	16	133	19	24	167	2	9
32	2	4	66	7	10	100	13	17	134	20	25	-	-	-
33	3	5	67	8	11	101	14	18	135	21	26	-	-	-

A.4 Definice $\varphi(n)$ pro referenční symboly

Tab. A.7: Definice $\varphi(n)$ pro referenční symboly délky $M_{sc}^{RS} = N_{sc}^{RS}$

u	$\varphi(0), \dots, \varphi(11)$											
0	-1	1	3	-3	3	3	1	1	3	1	-3	3
1	1	1	3	3	3	-1	1	-3	-3	1	-3	3
2	1	1	-3	-3	-3	-1	-3	-3	1	-3	1	-1
3	-1	1	1	1	1	-1	-3	-3	1	-3	3	-1
4	-1	3	1	-1	1	-1	-3	-1	1	-1	1	3
5	1	-3	3	-1	-1	1	1	-1	-1	3	-3	1
6	-1	3	-3	-3	-3	3	1	-1	3	3	-3	1
7	-3	-1	-1	-1	1	-3	3	-1	1	-3	3	1
8	1	-3	3	1	-1	-1	-1	1	1	3	-1	1
9	1	-3	-1	3	3	-1	-3	1	1	1	1	1
10	-1	3	-1	1	1	-3	-3	-1	-3	-3	3	-1
11	3	1	-1	-1	3	3	-3	1	3	1	3	3
12	1	-3	1	1	-3	1	1	1	-3	-3	-3	1
13	3	3	-3	3	-3	1	1	3	-1	-3	3	3
14	-3	1	-1	-3	-1	3	1	3	3	3	-1	1
15	3	-1	1	-3	-1	-1	1	1	3	1	-1	-3
16	1	3	1	-1	1	3	3	3	-1	-1	3	-1
17	-3	1	1	3	-3	3	-3	-3	3	1	3	-1
18	-3	3	1	1	-3	1	-3	-3	-1	-1	1	-3
19	-1	3	1	3	1	-1	-1	3	-3	-1	-3	-1
20	-1	-3	1	1	1	1	3	1	-1	1	-3	-1
21	-1	3	-1	1	-3	-3	-3	-3	-3	1	-1	-3
22	1	1	-3	-3	-3	-3	-1	3	-3	1	-3	3
23	1	1	-1	-3	-1	-3	1	-1	1	3	-1	1
24	1	1	3	1	3	3	-1	1	-1	-3	-3	1
25	1	-3	3	3	1	3	3	1	-3	-1	-1	3
26	1	3	-3	-3	3	-3	1	-1	-1	3	-1	-3
27	-3	-1	-3	-1	-3	3	1	-1	1	3	-3	-3
28	-1	3	-3	3	-1	3	3	-3	3	3	-1	-1
29	3	-3	-3	-1	-1	-3	-1	3	-3	3	1	-1

Tab. A.8: Definice $\varphi(n)$ pro referenční symboly délky $M_{sc}^{RS} = 2N_{sc}^{RS}$

u	$\varphi(0), \dots, \varphi(23)$																							
0	-1	3	1	-3	3	-1	1	3	-3	3	1	3	-3	3	1	1	-1	1	3	-3	3	-3	-1	-3
1	-3	3	-3	-3	-3	1	-3	-3	3	-1	1	1	1	3	1	-1	3	-3	-3	1	3	1	1	-3
2	3	-1	3	3	1	1	-3	3	3	3	3	1	-1	3	-1	1	1	-1	-3	-1	-1	1	3	3
3	-1	-3	1	1	3	-3	1	1	-3	-1	-1	1	3	1	3	1	-1	3	1	1	-3	-1	-3	-1
4	-1	-1	-1	-3	-3	-1	1	1	3	3	-1	3	-1	1	-1	-3	1	-1	-3	-3	1	-3	-1	-1
5	-3	1	1	3	-1	1	3	1	-3	1	-3	1	1	-1	-1	3	-1	-3	3	-3	-3	-3	1	1
6	1	1	-1	-1	3	-3	-3	3	-3	1	-1	-1	1	-1	1	1	-1	-3	-1	1	-1	3	-1	-3
7	-3	3	3	-1	-1	-3	-1	3	1	3	1	3	1	1	-1	3	1	-1	1	3	-3	-1	-1	1
8	-3	1	3	-3	1	-1	-3	3	-3	3	-1	-1	-1	-1	1	-3	-3	-3	1	-3	-3	-3	1	-3
9	1	1	-3	3	3	-1	-3	-1	3	-3	3	3	3	-1	1	1	-3	1	-1	1	1	-3	1	1
10	-1	1	-3	-3	3	-1	3	-1	-1	-3	-3	-3	-1	-3	-3	1	-1	1	3	3	-1	1	-1	3
11	1	3	3	-3	-3	1	3	1	-1	-3	-3	-3	3	3	-3	3	3	-1	-3	3	-1	1	-3	1
12	1	3	3	1	1	1	-1	-1	1	-3	3	-1	1	1	-3	3	3	-1	-3	3	-3	-1	-3	-1
13	3	-1	-1	-1	-1	-3	-1	3	3	1	-1	1	3	3	3	-1	1	1	-3	1	3	-1	-3	3
14	-3	-3	3	1	3	1	-3	3	1	3	1	1	3	3	-1	-1	-3	1	-3	-1	3	1	1	3
15	-1	-1	1	-3	1	3	-3	1	-1	-3	-1	3	1	3	1	-1	-3	-3	-1	-1	-3	-3	-3	-1
16	-1	-3	3	-1	-1	-1	-1	1	1	-3	3	1	3	3	1	-1	1	-3	1	-3	1	1	-3	-1
17	1	3	-1	3	3	-1	-3	1	-1	-3	3	3	3	-1	1	1	3	-1	-3	-1	3	-1	-1	-1
18	1	1	1	1	1	-1	3	-1	-3	1	1	3	-3	1	-3	-1	1	1	-3	-3	3	1	1	-3
19	1	3	3	1	-1	-3	3	-1	3	3	3	-3	1	-1	1	-1	-3	-1	1	3	-1	3	-3	-3
20	-1	-3	3	-3	-3	-3	-1	-1	-3	-1	-3	3	1	3	-3	-1	3	-1	1	-1	3	-3	1	-1
21	-3	-3	1	1	-1	1	-1	1	-1	3	1	-3	-1	1	-1	1	-1	-1	3	3	-3	-1	1	-3
22	-3	-1	-3	3	1	-1	-3	-1	-3	-3	3	-3	3	-3	-1	1	3	1	-3	1	3	3	-1	-3
23	-1	-1	-1	-1	3	3	3	1	3	3	-3	1	3	-1	3	-1	3	3	-3	3	1	-1	3	3
24	1	-1	3	3	-1	-3	3	-3	-1	-1	3	-1	3	-1	-1	1	1	1	1	-1	-1	-3	-1	3
25	1	-1	1	-1	3	-1	3	1	1	-1	-1	-3	1	1	-3	1	3	-3	1	1	-3	-3	-1	-1
26	-3	-1	1	3	1	1	-3	-1	-1	-3	3	-3	3	1	-3	3	-3	1	-1	1	-3	1	1	1
27	-1	-3	3	3	1	1	3	-1	-3	-1	-1	-1	3	1	-3	-3	-1	3	-3	-1	-3	-1	-3	-1
28	-1	-3	-1	-1	1	-3	-1	-1	1	-1	-3	1	1	-3	1	-3	-3	3	1	1	-1	3	-1	-1
29	1	1	-1	-1	-3	-1	3	-1	3	-1	1	3	1	-1	3	1	3	-3	-3	1	-1	-1	1	3

A.5 Výpis úplného nastavení jednotlivých testů

Tab. A.9: Tabulka společných nastavení pro sadu simulací

Společné nastvení					
Název	Společné vlastnosti nastavení				
Nastavení 1	SchVec	n_RNTI	N_CELL	noise seed	channel seed
	[1,1,1,1,1,1]	11345	102	1945173372	175
	Vliv kanálu	ekvalizace	typ vstupu	délka vstupu	
	-	Ne	soubor	1,00E+06	
Nastavení 2	SchVec	n_RNTI	N_CELL	noise seed	Směr
	[1,1,1,1,1,1]	11345	102	1945173372	DL
	Vliv kanálu	typ vstupu	délka vstupu	Počet iterací	
	100%	soubor	1,00E+06	8	
Nastavení 3	SchVec	n_RNTI	N_CELL	noise seed	Směr
	[1,1,1,1,1,1]	11345	102	1945173372	UL
	Vliv kanálu	typ vstupu	délka vstupu	Počet iterací	
	100%	soubor	1,00E+06	8	
Nastavení 4	SchVec	n_RNTI	N_CELL	noise seed	channel seed
	[1,1,1,1,1,1]	11345	102	1945173372	175
	Směr	typ vstupu	délka vstupu	Počet iterací	
	DL	soubor	1,00E+06	8	

Tab. A.10: Tabulka nastavení simulací, společné nastavení 1

Název	Nastavení	Směr	Typ prostředí	Modulace	Počet iterací
test1	Nastavení 1	DL	AWGN	QPSK	1
test2	Nastavení 1	DL	AWGN	QPSK	2
test3	Nastavení 1	DL	AWGN	QPSK	4
test4	Nastavení 1	DL	AWGN	QPSK	8
test5	Nastavení 1	DL	AWGN	QPSK	16
test6	Nastavení 1	DL	AWGN	16QAM	1
test7	Nastavení 1	DL	AWGN	16QAM	2
test8	Nastavení 1	DL	AWGN	16QAM	4
test9	Nastavení 1	DL	AWGN	16QAM	8
test10	Nastavení 1	DL	AWGN	16QAM	16
test11	Nastavení 1	DL	AWGN	64QAM	1
test12	Nastavení 1	DL	AWGN	64QAM	2
test13	Nastavení 1	DL	AWGN	64QAM	4
test14	Nastavení 1	DL	AWGN	64QAM	8
test15	Nastavení 1	DL	AWGN	64QAM	16
test16	Nastavení 1	DL	AWGN	QPSK	8
test17	Nastavení 1	DL	AWGN	16QAM	8
test18	Nastavení 1	DL	AWGN	64QAM	8
test19	Nastavení 1	DL	Interferenční	QPSK	8
test20	Nastavení 1	DL	Interferenční	16QAM	8
test21	Nastavení 1	DL	Interferenční	64QAM	8
test22	Nastavení 1	UL	AWGN	QPSK	8
test23	Nastavení 1	UL	AWGN	16QAM	8
test24	Nastavení 1	UL	AWGN	64QAM	8
test25	Nastavení 1	UL	Interferenční	QPSK	8
test26	Nastavení 1	UL	Interferenční	16QAM	8
test27	Nastavení 1	UL	Interferenční	64QAM	8

Tab. A.11: Tabulka nastavení simulací, společné nastavení 2

Název	Nastavení	Typ prostředí	Modulace	chanel seed	ekvalizace
test28	Nastavení 2	flat Rayleigh	QPSK	175	Ne
test29	Nastavení 2	PedA	QPSK	175	Ne
test30	Nastavení 2	PedB	QPSK	175	Ne
test31	Nastavení 2	flat Rayleigh	16QAM	175	Ne
test32	Nastavení 2	PedA	16QAM	175	Ne
test33	Nastavení 2	PedB	16QAM	175	Ne
test34	Nastavení 2	flat Rayleigh	64QAM	175	Ne
test35	Nastavení 2	PedA	64QAM	175	Ne
test36	Nastavení 2	PedB	64QAM	175	Ne
test37	Nastavení 2	flat Rayleigh	QPSK	175	Ano
test38	Nastavení 2	PedA	QPSK	175	Ano
test39	Nastavení 2	PedB	QPSK	175	Ano
test40	Nastavení 2	flat Rayleigh	16QAM	175	Ano
test41	Nastavení 2	PedA	16QAM	175	Ano
test42	Nastavení 2	PedB	16QAM	175	Ano
test43	Nastavení 2	flat Rayleigh	64QAM	175	Ano
test44	Nastavení 2	PedA	64QAM	175	Ano
test45	Nastavení 2	PedB	64QAM	175	Ano

Tab. A.12: Tabulka nastavení simulací, společné nastavení 3

Název	Nastavení	Typ prostředí	Modulace	chanel seed	ekvalizace
test46	Nastavení 3	flat Rayleigh	QPSK	175	Ne
test47	Nastavení 3	PedA	QPSK	175	Ne
test48	Nastavení 3	PedB	QPSK	175	Ne
test49	Nastavení 3	flat Rayleigh	16QAM	175	Ne
test50	Nastavení 3	PedA	16QAM	175	Ne
test51	Nastavení 3	PedB	16QAM	175	Ne
test52	Nastavení 3	flat Rayleigh	64QAM	175	Ne
test53	Nastavení 3	PedA	64QAM	175	Ne
test54	Nastavení 3	PedB	64QAM	175	Ne
test55	Nastavení 3	flat Rayleigh	QPSK	175	Ano
test56	Nastavení 3	PedA	QPSK	175	Ano
test57	Nastavení 3	PedB	QPSK	175	Ano
test58	Nastavení 3	flat Rayleigh	16QAM	175	Ano
test59	Nastavení 3	PedA	16QAM	175	Ano
test60	Nastavení 3	PedB	16QAM	175	Ano
test61	Nastavení 3	flat Rayleigh	64QAM	175	Ano
test62	Nastavení 3	PedA	64QAM	175	Ano
test63	Nastavení 3	PedB	64QAM	175	Ano
test64	Nastavení 3	flat Rayleigh	QPSK	9845	Ne
test65	Nastavení 3	PedA	QPSK	9845	Ne
test66	Nastavení 3	PedB	QPSK	9845	Ne
test67	Nastavení 3	flat Rayleigh	16QAM	9845	Ne
test68	Nastavení 3	PedA	16QAM	9845	Ne
test69	Nastavení 3	PedB	16QAM	9845	Ne
test70	Nastavení 3	flat Rayleigh	64QAM	9845	Ne
test71	Nastavení 3	PedA	64QAM	9845	Ne
test72	Nastavení 3	PedB	64QAM	9845	Ne
test73	Nastavení 3	flat Rayleigh	QPSK	9845	Ano
test74	Nastavení 3	PedA	QPSK	9845	Ano
test75	Nastavení 3	PedB	QPSK	9845	Ano
test76	Nastavení 3	flat Rayleigh	16QAM	9845	Ano
test77	Nastavení 3	PedA	16QAM	9845	Ano
test78	Nastavení 3	PedB	16QAM	9845	Ano
test79	Nastavení 3	flat Rayleigh	64QAM	9845	Ano
test80	Nastavení 3	PedA	64QAM	9845	Ano
test81	Nastavení 3	PedB	64QAM	9845	Ano

Tab. A.13: Tabulka nastavení simulací, společné nastavení 4, část 1

Název	Nastavení	Typ prostředí	Modulace	vliv kanálu	ekvalizace
test82	Nastavení 4	flat Rayleigh	QPSK	75%	Ne
test83	Nastavení 4	PedA	QPSK	75%	Ne
test84	Nastavení 4	PedB	QPSK	75%	Ne
test85	Nastavení 4	flat Rayleigh	16QAM	75%	Ne
test86	Nastavení 4	PedA	16QAM	75%	Ne
test87	Nastavení 4	PedB	16QAM	75%	Ne
test88	Nastavení 4	flat Rayleigh	64QAM	75%	Ne
test89	Nastavení 4	PedA	64QAM	75%	Ne
test90	Nastavení 4	PedB	64QAM	75%	Ne
test91	Nastavení 4	flat Rayleigh	QPSK	50%	Ne
test92	Nastavení 4	PedA	QPSK	50%	Ne
test93	Nastavení 4	PedB	QPSK	50%	Ne
test94	Nastavení 4	flat Rayleigh	16QAM	50%	Ne
test95	Nastavení 4	PedA	16QAM	50%	Ne
test96	Nastavení 4	PedB	16QAM	50%	Ne
test97	Nastavení 4	flat Rayleigh	64QAM	50%	Ne
test98	Nastavení 4	PedA	64QAM	50%	Ne
test99	Nastavení 4	PedB	64QAM	50%	Ne
test100	Nastavení 4	flat Rayleigh	QPSK	25%	Ne
test101	Nastavení 4	PedA	QPSK	25%	Ne
test102	Nastavení 4	PedB	QPSK	25%	Ne
test103	Nastavení 4	flat Rayleigh	16QAM	25%	Ne
test104	Nastavení 4	PedA	16QAM	25%	Ne
test105	Nastavení 4	PedB	16QAM	25%	Ne
test106	Nastavení 4	flat Rayleigh	64QAM	25%	Ne
test107	Nastavení 4	PedA	64QAM	25%	Ne
test108	Nastavení 4	PedB	64QAM	25%	Ne

Tab. A.14: Tabulka nastavení simulací, společné nastavení 4, část 2

Název	Nastavení	Typ prostředí	Modulace	vliv kanálu	ekvalizace
test109	Nastavení 4	flat Rayleigh	QPSK	75%	Ano
test110	Nastavení 4	PedA	QPSK	75%	Ano
test111	Nastavení 4	PedB	QPSK	75%	Ano
test112	Nastavení 4	flat Rayleigh	16QAM	75%	Ano
test113	Nastavení 4	PedA	16QAM	75%	Ano
test114	Nastavení 4	PedB	16QAM	75%	Ano
test115	Nastavení 4	flat Rayleigh	64QAM	75%	Ano
test116	Nastavení 4	PedA	64QAM	75%	Ano
test117	Nastavení 4	PedB	64QAM	75%	Ano
test118	Nastavení 4	flat Rayleigh	QPSK	50%	Ano
test119	Nastavení 4	PedA	QPSK	50%	Ano
test120	Nastavení 4	PedB	QPSK	50%	Ano
test121	Nastavení 4	flat Rayleigh	16QAM	50%	Ano
test122	Nastavení 4	PedA	16QAM	50%	Ano
test123	Nastavení 4	PedB	16QAM	50%	Ano
test124	Nastavení 4	flat Rayleigh	64QAM	50%	Ano
test125	Nastavení 4	PedA	64QAM	50%	Ano
test126	Nastavení 4	PedB	64QAM	50%	Ano
test127	Nastavení 4	flat Rayleigh	QPSK	25%	Ano
test128	Nastavení 4	PedA	QPSK	25%	Ano
test129	Nastavení 4	PedB	QPSK	25%	Ano
test130	Nastavení 4	flat Rayleigh	16QAM	25%	Ano
test131	Nastavení 4	PedA	16QAM	25%	Ano
test132	Nastavení 4	PedB	16QAM	25%	Ano
test133	Nastavení 4	flat Rayleigh	64QAM	25%	Ano
test134	Nastavení 4	PedA	64QAM	25%	Ano
test135	Nastavení 4	PedB	64QAM	25%	Ano