



UNIVERSITAS INDONESIA

**ANALISA DAN PERBANDINGAN TEKNIK *CLIPPING*
UNTUK REDUKSI PAPR DAN KODE KOREKSI ERROR
UNTUK PERFORMANSI BER PADA OFDM**

TESIS

YUDHI TRIPRASETYO

0906495715

**FAKULTAS TEKNIK
PROGRAM PASCASARJANA
DEPOK
DESEMBER 2011**



UNIVERSITAS INDONESIA

**ANALISA DAN PERBANDINGAN TEKNIK *CLIPPING*
UNTUK REDUKSI PAPR DAN KODE KOREKSI ERROR
UNTUK PERFORMANSI BER PADA OFDM**

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister Teknik

YUDHI TRIPRASETYO

0906495715

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
KEKHUSUSAN TEKNIK TELEKOMUNIKASI
DEPOK
DESEMBER 2011**

HALAMAN PERNYATAAN ORISINALITAS

**Tesis ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama : Yudhi Triprasetyo
NPM : 0906495715
Tanda Tangan : 
Tanggal : 16 Desember 2011

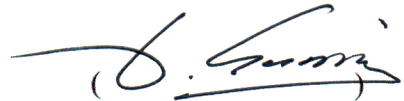
HALAMAN PENGESAHAN

Tesis ini diajukan oleh :
Nama : Yudhi Triprasetyo
NPM : 0906495715
Program Studi : Teknik Elektro
Judul Tesis : ANALISA DAN PERBANDINGAN TEKNIK
CLIPPING UNTUK REDUKSI PAPR DAN KODE
KOREKSI ERROR UNTUK PERFORMANSI BER
PADA OFDM

Telah berhasil dipertahankan dihadapan Dewan penguji dan sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Teknik pada Program Pascasarjana Departemen Teknik Elektro, Fakultas Teknik, Universitas Indonesia

DEWAN PENGUJI

Pembimbing : Prof.Dr.Ir. Dadang Gunawan M.Eng



Penguji : Dr.Ir. Arman D. Diponegoro



Penguji : Filbert Hilman Juwono S.T., M.T.



Penguji : Dr.Muhammad Suryanegara, ST., M.Sc



Ditetapkan di : Depok

Tanggal : 16 Desember 2011

KATA PENGANTAR/UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan kepada Tuhan YME, karena telah mengkaruniakan keadaan hingga saat ini dan seterusnya. Tesis ini dibuat untuk memenuhi salah satu syarat mencapai gelar Magister Teknik Program Studi Teknik Elektro, Kekhususan Teknik Telekomunikasi Fakultas Teknik Universitas Indonesia. Penulis mengucapkan terima kasih kepada:

1. Prof.Dr.Ir.Dadang Gunawan, M.Eng selaku dosen pembimbing yang telah bersedia meluangkan waktu untuk memberi pengarahan, diskusi dan bimbingan serta persetujuan sehingga tesis ini dapat selesai dengan baik
2. Filbert Juwono, ST.M.T, yang telah meluangkan waktunya untuk berdiskusi.
3. Orang tua yang selalu mendukung dan mendoakan.
4. Ryan Andhikasari yang selalu mendukung dan mendorong saya.
5. Teman-teman S2 Teknik Elektro UI yang menjadi teman diskusi yang baik.

Akhir kata, semoga tesis ini menjadi membawa manfaat bagi pengembangan ilmu.

Depok, 16 Desember 2011

Yudhi Triprasetyo

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan dibawah ini:

Nama : Yudhi Triprasetyo
NPM : 0906495715
Program studi : Teknik Telekomunikasi
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis karya : Tesis

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty Free Right*)** atas karya ilmiah saya yang berjudul:

ANALISA DAN PERBANDINGAN TEKNIK *CLIPPING* UNTUK REDUKSI
PAPR DAN KODE KOREKSI ERROR UNTUK PERFORMANSI BER PADA
OFDM

Beserta perangkat yang ada (juka diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalih media/format-kan mengelola dalam bentuk pangkalan data (database), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di: Depok

Pada tanggal : 16 Desember 2011

Yang menyatakan



(Yudhi Triprasetyo)

ABSTRAK

Nama : Yudhi Triprasetyo
Program Studi : Teknik Elektro
Judul : ANALISA DAN PERBANDINGAN TEKNIK *CLIPPING*
UNTUK REDUKSI PAPR DAN KODE KOREKSI ERROR
UNTUK PERFORMANSI BER PADA OFDM

OFDM merupakan teknik *multicarrier* yang kini menjadi pilihan teknologi telekomunikasi untuk transmisi *data rate* yang tinggi. Kelemahan OFDM adalah PAPR yang tinggi dan sensitif terhadap error frekuensi *carrier*. PAPR yang tinggi disebabkan superposisi dari IFFT dan menyebabkan kompleksitas pada *power amplifier*. Untuk menangani permasalahan tersebut, terdapat beberapa teknik reduksi PAPR. Teknik reduksi PAPR yang sederhana adalah *clipping*, namun memberi dampak distorsi *in-band* dan *out-of-band* yang dapat mempengaruhi kinerja sistem. Untuk menangani penurunan kinerja BER karena teknik *clipping*, dapat digunakan kode koreksi kanal (ECC). Terdapat beberapa jenis teknik *clipping*, diantara lain yaitu *deep clipping*, *classical clipping* dan *smooth clipping*. kode ECC yang saat ini mendekati kinerja Shannon limit adalah kode Turbo dan kode LDPC. Tesis ini merancang kombinasi variasi *clipping* dengan teknik kode koreksi Turbo dan LDPC untuk mendapatkan sistem terbaik dalam memperoleh kinerja PAPR dan BER yang baik. Dari hasil perancangan dan hasil simulasi tersebut, didapatkan teknik *classical clipping* dengan kode Turbo untuk reduksi PAPR yang baik tanpa memberi penurunan BER yang besar dibandingkan teknik *clipping* lainnya untuk sistem OFDM.

Kata kunci: *OFDM, Clipping, Kode LDPC, Kode Turbo*

ABSTRACT

Name : Yudhi Triprasetyo
Study Program : Telecommunication Engineering
Title : Analysis and Comparison Clipping Techniques for PAPR Reduction and Error Correction Code for BER Performance in OFDM System.

OFDM is a multicarrier technique that becomes the main choice of telecommunication technology for high data rate transmission. The disadvantages of OFDM are high PAPR and sensitivity to carrier frequency error. High PAPR that caused by the superposition of IFFT gives inefficiency in power amplifier. To address these problems, there are several PAPR reduction techniques. The simplest PAPR reduction technique is clipping, but the impact of in-band and out-of-band distortions affect the system's performance. Error Correction Channel (ECC) can be used to handle the BER degradation due to clipping technique. There are several types of clipping techniques such as deep clipping, classical clipping and smooth clipping. ECC codes which are close to Shannon limit are Turbo codes and LDPC codes. This thesis studies the effect of ECC codes to various clipping techniques to get better system's performance (better BER). From the simulation results, it is found that classical clipping technique with Turbo code yield better PAPR reduction without a large decrease in BER compared to other clipping techniques.

Keywords : *OFDM, Clipping, LDPC Code, Turbo Code*

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
UCAPAN TERIMA KASIH	iv
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI.....	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
DAFTAR SINGKATAN	xii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Permasalahan	6
1.3. Tujuan Penelitian	6
1.4. Batasan Masalah	6
1.5. Sistematika Penulisan	7
BAB II SISTEM OFDM.....	8
2.1. Prinsip Dasar OFDM	8
2.2. Modulasi dan Demodulasi OFDM	11
2.3. <i>Guard Interval</i>	13
2.4. <i>Guard Band</i>	14
2.5. PAPR.....	15
2.6. <i>Clipping</i>	16
2.7. Kode Koreksi Error (<i>Error Correcting Coding</i>).....	17
2.8 LDPC	18
2.9 Turbo	25
BAB III PEMODELAN SISTEM DAN PERANCANGAN SIMULASI.....	31
3.1. Pemodelan Sistem OFDM menggunakan teknik <i>Clipping</i>	31

3.2. Pemodelan Sistem LDPC-OFDM Menggunakan Teknik <i>Clipping</i>	32
3.3. Pemodelan Sistem Turbo-OFDM Menggunakan Teknik <i>Clipping</i>	33
3.4. Fungsi <i>Clipping</i>	35
3.5. Indikator Kinerja	36
3.5.1. Kinerja PAPR.....	36
3.5.2. Kinerja BER.....	37
3.5.3. Kinerja PSD	37

BAB IV HASIL DAN PEMBAHASAN..... 38

4.1. Analisa OFDM Menggunakan <i>Clipping</i>	39
4.2. Analisa Kinerja LDPC-OFDM dan Turbo-OFDM	42
4.3. Analisa Kinerja LDPC-OFDM Menggunakan <i>Clipping</i>	45
4.4. Analisa Kinerja Turbo-OFDM Menggunakan <i>Clipping</i>	50

BAB V KESIMPULAN..... 56

DAFTAR PUSTAKA

DAFTAR GAMBAR

Gambar 1.1 OFDM dalam jaringan nirkabel pita lebar	2
Gambar 2.1 Ilustrasi Fenomena <i>Intersymbol Interference</i>	8
Gambar 2.2 Perbandingan Skema Transmisi <i>Single Carrier</i> dan <i>Multicarrier</i>	9
Gambar 2.3 Ilustrasi Skema Modulasi <i>Multicarrier</i>	9
Gambar 2.4 Tanggap Frekuensi Sistem Transmisi <i>Multiple Carrier</i>	9
Gambar 2.5 Konsep Sinyal OFDM	10
Gambar 2.6 Sistem OFDM Dengan Bank Osilator	10
Gambar 2.7 Sistem OFDM Menggunakan IFFT/FFT	11
Gambar 2.8 Modulasi dan Demodulasi OFDM	12
Gambar 2.9 Prinsip Kerja <i>Cyclic Prefix</i>	13
Gambar 2.10 Ilustrasi Efek penggunaan <i>Cyclic prefix</i>	14
Gambar 2.11 Tanggap Impuls Kanal Frekuensi Selektif	14
Gambar 2.12 Ilustrasi Implementasi <i>Virtual Carrier</i> pada OFDM	15
Gambar 2.13 Perbandingan Fungsi Alih <i>Clipping</i>	17
Gambar 2.14 Ilustrasi Aplikasi Kode Koreksi Error Dalam Sistem Komunikasi	17
Gambar 2.15 Tanner <i>Graph</i>	19
Gambar 2.16 Diagram Alir Algoritma Bit Flipping	21
Gambar 2.17 Diagram Alir Algoritma Sum Product	24
Gambar 2.18 Blok Diagram Kode Turbo	25
Gambar 2.19 Skema Recursive Systematic Convolutional	27
Gambar 2.20 Penggunaan Puncture untuk menghasilkan <i>Code rate</i> 1/2	27
Gambar 2.21 Skema Dekoding Kode Turbo	28
Gambar 2.22 Ilustrasi Penggunaan LLR untuk Dekoding Turbo	29
Gambar 3.1 Sistem OFDM Menggunakan Teknik <i>Clipping</i>	31
Gambar 3.2 Sistem LDPC-OFDM Menggunakan Teknik <i>Clipping</i>	32
Gambar 3.3 Enkoding dan Dekoding LDPC	33
Gambar 3.4 Sistem Turbo-OFDM Menggunakan Teknik <i>Clipping</i>	34
Gambar 3.3 Enkoding dan Dekoding Turbo	33
Gambar 4.1 Simulasi OFDM Menggunakan Modulasi QPSK	38
Gambar 4.2 Power Spectral Density pada OFDM terhadap variasi <i>Clipping</i>	39
Gambar 4.3 Kinerja PAPR Sistem OFDM Menggunakan variasi <i>Clipping</i>	40

Gambar 4.4 BER Sistem OFDM Menggunakan Variasi <i>Clipping</i>	41
Gambar 4.5 Perbandingan Iterasi Dekoding LDPC-OFDM.....	43
Gambar 4.6 Perbandingan Iterasi Dekoding Turbo-OFDM.....	43
Gambar 4.7 Perbandingan BER OFDM Konvensional, Turbo-OFDM dan LDPC-OFDM	44
Gambar 4.8 PSD sistem LDPC-OFDM menggunakan <i>Clipping</i>	45
Gambar 4.9 Perbandingan PSD LDPC-OFDM dengan OFDM menggunakan variasi <i>Clipping</i>	46
Gambar 4.10 CCDF PAPR LDPC-OFDM Menggunakan Variasi <i>Clipping</i>	46
Gambar 4.11 Perbandingan PAPR LDPC-OFDM Menggunakan variasi <i>Clipping</i> terhadap OFDM Konvensional	47
Gambar 4.12 BER LDPC-OFDM menggunakan Variasi <i>Clipping</i>	48
Gambar 4.13 Perbandingan Variasi <i>Clipping</i> OFDM Konvensional dengan LDPC OFDM	49
Gambar 4.14 PSD Sistem Turbo OFDM Menggunakan <i>Clipping</i>	50
Gambar 4.15 Perbandingan PSD Turbo-OFDM dengan OFDM menggunakan variasi <i>Clipping</i>	51
Gambar 4.16 CCDF PAPR Turbo-OFDM Menggunakan Variasi <i>Clipping</i>	51
Gambar 4.17 Perbandingan PAPR Turbo-OFDM Menggunakan variasi <i>Clipping</i> terhadap OFDM Konvensional	52
Gambar 4.18 BER Turbo-OFDM menggunakan Variasi <i>Clipping</i>	53
Gambar 4.19 Perbandingan Variasi <i>Clipping</i> OFDM Konvensional dengan Turbo-OFDM	54
Gambar 4.20 Perbandingan Kinerja BER LDPC-OFDM dan Turbo-OFDM Menggunakan Variasi <i>Clipping</i>	55

DAFTAR TABEL

Tabel 2.1 Contoh Kelas Enkoder <i>Convolutional</i>	27
---	----

DAFTAR SINGKATAN



ACI	: Adjacent Channel Interference
ADC	: Analog to Digital Converter
APP	: A Posteriori Probabilities
BC	: Block Code
BER	: Bit Error Rate
BWA	: Broadband Wireless Access
CC	: convolutional code
CCDF	: Complementary Cumulative Distribution Function
CP	: Cyclic Prefix
DAC	: Digital to Analog Converter
DFT	: Discrete Fourier Transform
DMT	: Discrete Multitone
DVB-T	: Digital Video Broadcasting - Terrestrial
ECC	: Error Correction Coding
FDM	: Frequency Division Multiplexing
FFT	: Fast Fourier Transform
IDFT	: Inverse Discrete Fourier Transform
IFFT	: Inverse Fast Fourier Transform
ISI	: Inter Symbol Interference
LDPC	: Low Density Parity Check
LLR	: Log likelihood Ratio
LTE	: Long Term Evolution
MIMO	: Multiple Input Multiple Output
OFDM	: Orthogonal Frequency Division Multiplexing
OOB	: out of band
PAPR	: Peak to Average Power Ratio
PCCC	: Parallel Concatenated Convolutional Codes
QPSK	: Quadrature Phase Shift Keying
QAM	: Quadrature Amplitude Modulation
RC	: Raised Cosine
RSC	: Recursive Systematic Convolutional
SNR	: Signal to Noise Ratio
SQNR	: Signal to Quantization noise ratio

BAB 1

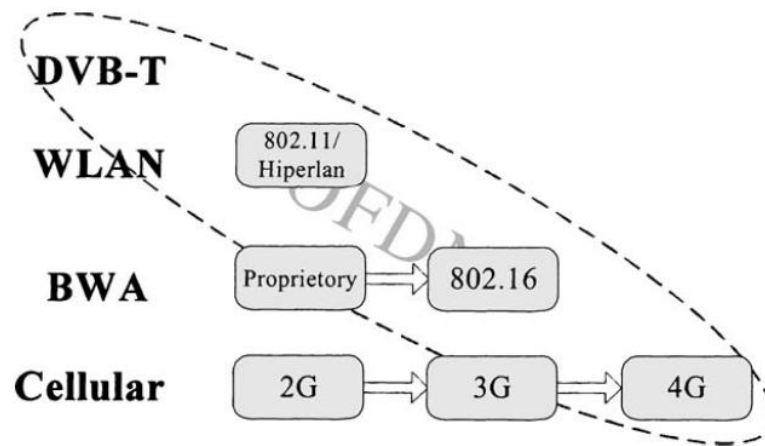
PENDAHULUAN

1.1 LATAR BELAKANG

Tuntutan layanan komunikasi nirkabel semakin tinggi dan diharapkan layanan komunikasi nirkabel meningkat tiap tahunnya [1]. Tuntutan ini memberi motivasi pada banyaknya penelitian dan pengembangan untuk mendapatkan teknologi 4G pada sistem nirkabel pita lebar yang dapat memberi peningkatan *data rate* dan kualitas layanan dari teknologi 3G. Teknologi nirkabel yang baru akan mendukung aplikasi multimedia dengan kebutuhan spesifikasi yang berbeda dalam bentuk kehandalan, *bitrate* dan *latency*.

Tantangan untuk menghasilkan teknologi 4G adalah konflik antar tuntutan kebutuhan *data rate* yang lebih tinggi dan keterbatasan spektrum radio, yang memberi kesimpulan akan kebutuhan penggunaan spektrum dengan efisiensi spektral yang tinggi. Untuk meningkatkan efisiensi spektral adalah penggunaan modulasi yang fleksibel dan skema coding, dimana sumber daya sistem secara adaptif dialokasikan ke pengguna berdasarkan *data rate* yang diminta dan kualitas kanal. Teknik *multicarrier* menjadi kandidat potensial untuk sistem nirkabel pita lebar karena keunggulannya dalam ketahanan terhadap distorsi kanal dan interferensi *narrowband*, efisiensi spektral yang tinggi, fleksibilitas yang tinggi dan mendukung skema modulasi adaptif, serta dapat dikombinasikan dengan teknologi MIMO (*Multiple Input Multiple Output*).

Transmisi *multicarrier*, seperti *Orthogonal Frequency Division Multiplexing* (OFDM) atau *Discrete Multitone* (DMT), adalah teknik yang mulai populer dalam aplikasi nirkabel dan kabel. Teknik ini mulai diminati karena perkembangan terkini dalam teknologi pemrosesan sinyal digital. OFDM adalah skema modulasi untuk transmisi data berkecepatan tinggi dalam lingkup tundaan dispersif [2]. Gambar 1.1 menunjukkan penggunaan sistem OFDM pada beragam sistem telekomunikasi seperti DVB-T, 802.16 dan selular seperti *Long Term Evolution* (LTE).



Gambar 1.1 OFDM Dalam Jaringan Nirkabel Pita Lebar [3]

Untuk aplikasi nirkabel, sistem OFDM memberi kekebalan pada *multipath fading* dan *impulse noise*, dan mengurangi penggunaan ekualiser, serta implementasi perangkat keras yang efisien dapat direalisasikan menggunakan teknik (*Fast Fourier Transform*) FFT. Teknologi *multicarrier* dalam bentuk OFDM secara umum dikenal sebagai salah satu skema akses yang menjanjikan untuk jaringan nirkabel.

Ide dari OFDM adalah membagi data berkecepatan tinggi menjadi sejumlah data dengan kecepatan rendah. Sejumlah data tersebut ditransmisikan secara paralel melalui sub-kanal yang ortogonal dengan spektra yang saling tumpang tindih. Dibandingkan dengan transmisi *single carrier*, OFDM memberi sistem dengan ketahanan terhadap interferensi *narrow band* dan distorsi kanal. Serta memberi fleksibilitas tinggi karena parameter modulasi seperti ukuran konstelasi dan *coding rate* secara independen dapat dipilih tiap masing-masing sub-kanal. OFDM juga dapat dikombinasikan dengan teknik multiple akses konvensional untuk operasi skenario *multi-user* [1].

OFDM sebagai salah satu aplikasi skema transmisi data paralel, yang mengurangi pengaruh dari *multipath fading* dan menghilangkan penggunaan ekualiser kompleks. Dalam sistem *multicarrier*, *fading* atau interferensi akan mempengaruhi sedikit dari *subcarrier* dan *Error Correction Coding* (ECC) dapat digunakan untuk memperbaiki error dari *subcarrier* tersebut.

Kelebihan skema transmisi OFDM adalah:

- OFDM adalah cara efektif untuk menangani *frequency selective fading*, penyebaran tundaan (*delay*), kompleksitas implementasi lebih rendah dibandingkan sistem *single carrier* yang menggunakan ekualiser.
- Pada kanal yang relatif bervariasi terhadap waktu, dimungkinkan untuk meningkatkan kapasitas secara signifikan dengan mengubah *data rate* per *subcarrier* berdasarkan SNR (*signal to noise ratio*) dari *subcarrier* tersebut.
- OFDM tahan terhadap interferensi *narrowband* karena efek interferensi hanya beberapa persen dari *subcarrier*.
- OFDM membuat jaringan frekuensi tunggal memungkinkan, yang menjadi menarik untuk aplikasi *broadcasting*.
- OFDM tahan terhadap ISI (*Intersymbol Interference*), dapat meningkatkan modulasi dan coding untuk tiap *subcarrier* dan ekualiser yang sederhana.
- Modulasi dengan kompleksitas rendah.
- OFDM mendapatkan efisiensi spektral yang lebih baik dibandingkan teknik *Frequency Division Multiplexing* (FDM) konvensional.
- Implementasi digital sederhana dengan menggunakan (*Discrete Fourier Transform*) DFT/Inverse-DFT.

Kelemahan OFDM adalah:

- OFDM lebih sensitif terhadap frekuensi *offset* dan gangguan fase. Sehingga membutuhkan spesifikasi ketat untuk osilator lokal.
- OFDM memiliki (*Peak to Average Power Ratio*) PAPR yang tinggi, memberi karakter gelombang tertransmisi yang cenderung mengurangi efisiensi daya dari amplifier RF.
- OFDM membutuhkan *power amplifier* yang bersifat linear karena PAPR yang tinggi memberi karakteristik sinyal yang ditransmisikan.
- Terdapat rugi-rugi efisiensi transmisi karena penggunaan *cyclic prefix*.

OFDM digunakan untuk aplikasi sebagai berikut:

Kabel:

- ADSL dan VDSL akses *broadband*
- DVB-C2 , versi peningkatan dari DVB-C TV kabel digital
- *Power Line Communication* (PLC)
- ITU-T G.hn, standar yang menyediakan jaringan berkecepatan tinggi menggunakan *home wiring* (*powerlines*, *phone lines*, dan kabel *coaxial*)
- Dan sebagainya.

Nirkabel:

- *Wireless LAN (WLAN) interface radio* IEEE 802.11a, g, n dan HIPERLAN/2
- Sistem radio digital DAB/EUREKA 147, DAB+, *Digital Radio Mondiale*, HD Radio, T-DMB dan ISDB-TSB.
- Sistem TV *Digital Terrestrial* DVB-T dan ISDB-T
- Sistem TV *mobile Terrestrial* DVB-H , TDMB, ISDB-T dan MediaFLO.
- *Wireless Personal Area Network (PAN) Ultra wideband* (UWB) IEEE 802.15.3a

Teknologi OFDM juga digunakan untuk jaringan selular seperti:

- Mode Mobilitas *wireless MAN/Broadband Wireless Access* (BWA) standar IEEE 802.16e (Mobile-WiMAX)
- *Mobile Broadband Wireless Access* (MBWA) standar IEEE 802.20
- 3GPP *Long Term Evolution Downlink* akses.

Salah satu kelemahan dari transmisi *multicarrier* adalah PAPR yang tinggi dari sinyal transmisi [4]. PAPR terjadi karena sinyal OFDM merupakan superposisi dari aliran data kecepatan rendah yang dimodulasi pada frekuensi yang berbeda. Jika daya puncak transmisi dibatasi oleh regulasi atau pembatasan aplikasi, efeknya adalah mengurangi daya rata-rata yang diizinkan dalam

transmisi *multicarrier* relatif dibawah konstanta teknik modulasi daya. Hal tersebut dapat mengurangi jangkauan dari transmisi *multicarrier*, bahkan mencegah pembesaran spektral dari sinyal *multicarrier* dalam bentuk intermodulasi antar *subcarrier* dan radiasi OOB (*out-of band*). Penguat daya transmisi harus beroperasi dalam area linear dimana konversi daya tidak efisien. Dan menyebabkan konsumsi daya pada aplikasi *mobile*. Dalam banyak aplikasi *low-cost*, PAPR yang tinggi dapat mengurangi keuntungan potensial dari sistem transmisi *multicarrier*.

Sejumlah pendekatan telah diajukan untuk menangani masalah PAPR, teknik tersebut adalah *clipping*, *coding*, *tone reservation*, *tone injection*, *active constellation extention*, dan teknik representasi sinyal seperti *partial transmit sequence* (PTS), *selected mapping* (SLM) dan *interleaving*. Teknik tersebut memperoleh reduksi PAPR dengan dampak peningkatan daya sinyal transmisi, peningkatan BER, *data-rate loss*, peningkatan kompleksitas komputasi dan sebagainya.

Terdapat beberapa penelitian yang telah dilakukan menggunakan teknik *clipping*. Penelitian Langlais et al [5] melakukan pendekatan kode *Error Correction Coding* (ECC) untuk reduksi PAPR pada sistem OFDM dengan mengkombinasikan Kode *Low Density Parity Check* (LDPC) dengan teknik *Clipping* untuk reduksi PAPR. Penggunaan kode LDPC juga diterapkan pada penelitian [6] dalam perbandingan kinerja *Bit Error Rate* (BER) pada sistem LDPC-OFDM dengan Turbo-OFDM. Penelitian Palicot et al [7] menunjukkan kinerja dari beberapa teknik *Clipping* seperti *Classical Clipping*, *Heavyside Clipping*, *Deep Clipping* dan *Smooth Clipping* untuk sistem OFDM. Variasi teknik *Clipping* yang dibahas pada penelitian tersebut menghasilkan perbedaan kinerja pada reduksi PAPR dan BER. Saat ini, belum terdapat penelitian yang melakukan penggunaan teknik *Deep Clipping*, dan *Smooth Clipping* pada kode LDPC dan kode Turbo pada OFDM.

Pada tulisan ini, akan dilakukan kombinasi OFDM dengan menggunakan teknik ECC yaitu *Low Density Parity Check* (LDPC) dan Turbo dengan kombinasi teknik variasi *Clipping* untuk mendapatkan kinerja terbaik pada sistem

OFDM. Kombinasi kode ECC dengan teknik *clipping* diharapkan memberi kontribusi dalam mengetahui kapabilitas ECC untuk koreksi error yang disebabkan variasi teknik *clipping* dan mendapatkan teknik terbaik dari kombinasi tersebut.

1.2 PERMASALAHAN

Salah satu kelemahan dari sistem OFDM adalah PAPR yang tinggi yang disebabkan rentang dinamik sinyal yang lebar. PAPR terjadi karena sinyal OFDM merupakan superposisi dari aliran data kecepatan rendah yang dimodulasi pada frekuensi yang berbeda. Beberapa teknik reduksi PAPR yang ada memiliki keunggulan disertai dengan efek samping dari teknik reduksi seperti peningkatan daya transmisi atau degradasi BER (*bit error rate*). Reduksi PAPR menggunakan teknik *clipping* merupakan teknik paling sederhana dibandingkan teknik reduksi lainnya. Namun kelemahan dari *clipping* mengurangi kinerja BER karena menyebabkan radiasi *in-band* dan *out-of-band* (OOB). Kelemahan dari teknik distorsi sinyal (*clipping*) ini dapat dikurangi menggunakan coding ECC. Error yang disebabkan oleh simbol yang mengalami degradasi yang cukup besar dapat dikoreksi menggunakan simbol sekelilingnya.

1.3 TUJUAN PENELITIAN

Tujuan dari tulisan ini adalah untuk merancang dan menganalisa teknik reduksi PAPR menggunakan variasi *clipping* dan kombinasi variasi teknik kode koreksi error LDPC dan Turbo.

1.4 BATASAN MASALAH

Untuk mempermudah dan menyederhanakan analisa, unjuk kerja dilakukan menggunakan simulasi berbasis komputasi yang dikembangkan dari referensi [8] untuk LDPC, [9] untuk Turbo dan [10] untuk OFDM. Unjuk kerja dilakukan tanpa mengesampingkan prinsip-prinsip dasar dari sistem reduksi PAPR pada sistem OFDM dan ECC dan diberikan beberapa batasan masalah,

yaitu parameter indikator kinerja yang digunakan adalah *Complementary Cumulative Distribution Function* (CCDF) untuk analisa PAPR, kinerja BER dan *Power Spectral Density* (PSD).

1.5 SISTEMATIKA PENULISAN

Pembahasan yang dilakukan pada penelitian ini meliputi empat bab, yaitu:

Bab 1 Pendahuluan

Bagian ini terdiri dari latar belakang masalah, tujuan penelitian, batasan masalah dan sistematika penulisan.

Bab 2 Sistem OFDM dan PAPR

Bagian ini berisi tentang Sistem OFDM, PAPR, ECC, LDPC dan Turbo serta Teknik *Clipping*.

Bab 3 Pemodelan Sistem dan Perancangan Simulasi.

Bab ini berisi Pemodelan sistem yang diusulkan dan metode simulasi.

Bab 4 Hasil Simulasi dan Analisis

Bab ini berisikan hasil simulasi dan pembahasan dari kinerja reduksi PAPR untuk OFDM yang diusulkan pada tesis ini.

Bab 5 Penutup

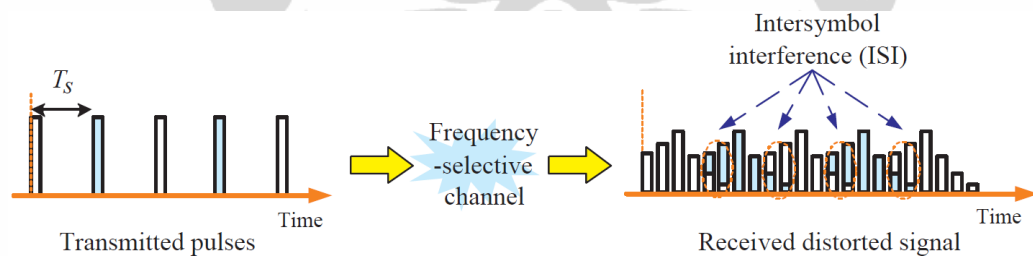
Bagian ini berisikan penutup dari tesis.

BAB 2

SISTEM OFDM

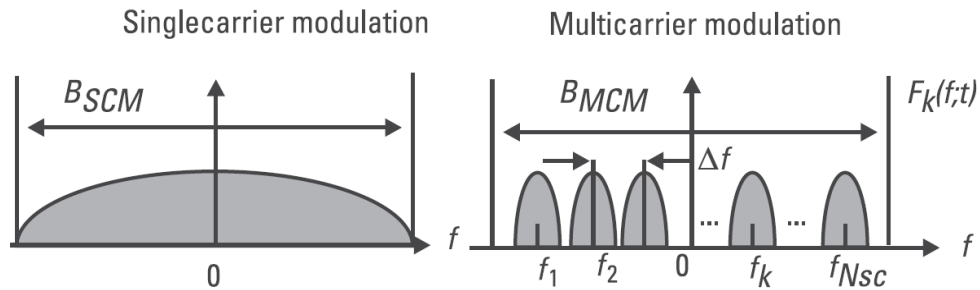
2.1 Prinsip Dasar OFDM

Ketika sinyal transmisi dengan *bandwidth* yang lebih besar dari koherensi kanal, komponen spektral dari sinyal akan mengalami atenuasi yang berbeda saat melakukan propagasi ke penerima. Kanal tersebut disebut selektif-frekuensi dan sinyal yang diterima secara linear terdistorsi dibandingkan sinyal transmisi. Efek dari fading selektif frekuensi adalah interferensi antar simbol (*Intersymbol Interference* (ISI)) pada sinyal yang diterima yang diilustrasikan pada Gambar 2.1. Cara mengkompensasi ISI adalah merancang filter linear yang disebut ekualiser kanal. Namun ekualiser kanal yang efisien membutuhkan filter yang kompleks sehingga meningkatkan kompleksitas pada perangkat terminal penerima. Solusi lain untuk menangani ISI adalah membuat durasi simbol lebih banyak dibandingkan tundaan penyebaran kanal maksimum (*maximum channel delay spread*). Namun metode pendekatannya akan mengurangi *throughput*.

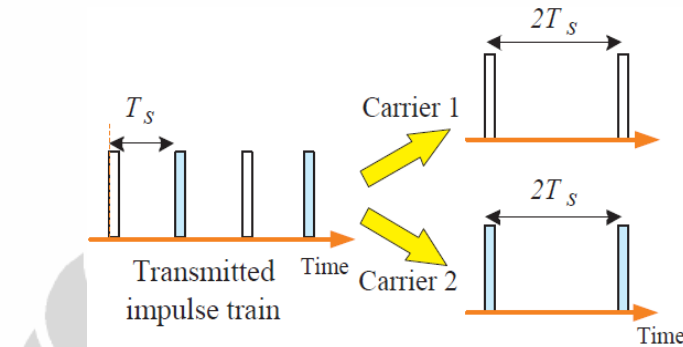


Gambar 2.1 Ilustrasi Fenomena *Intersymbol Interference* [1]

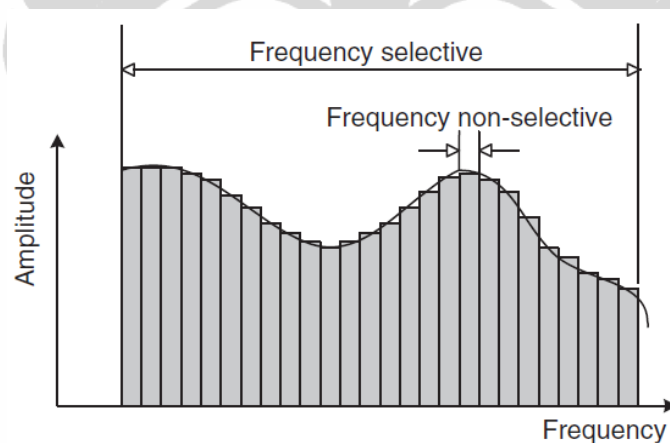
Untuk menangani selektifitas frekuensi dalam kanal pita lebar yang dialami oleh transmisi *single carrier* untuk transmisi data berkecepatan tinggi, *multicarrier* dapat digunakan untuk transmisi tersebut [11]. Gambar 2.2 menunjukkan perbandingan skema transmisi *single carrier* dan *multicarrier*, dimana B_{SCM} adalah *bandwidth* dari *single carrier* dan B_{MCM} *bandwidth* dari *multicarrier*, N_{SC} adalah total *subcarrier*. Skema *multicarrier* memberi ketahanan dan efektif dalam kanal nirkabel yaitu menangani *fading* frekuensi selektif, dan tidak memerlukan ekualisasi pada tiap *subcarrier*.



Gambar 2.2 Perbandingan Skema Transmisi *Single Carrier* dan *Multicarrier* [12]



Gambar 2.3. Ilustrasi Skema Modulasi *Multicarrier* [1]

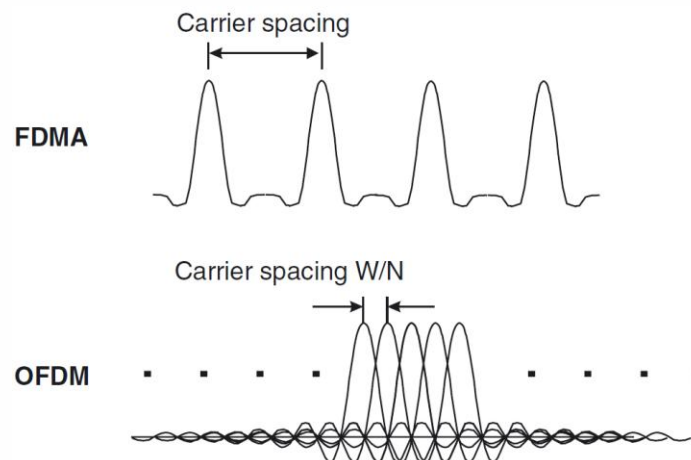


Gambar 2.4 Tanggap Frekuensi Sistem Transmisi Multiple Carrier [11].

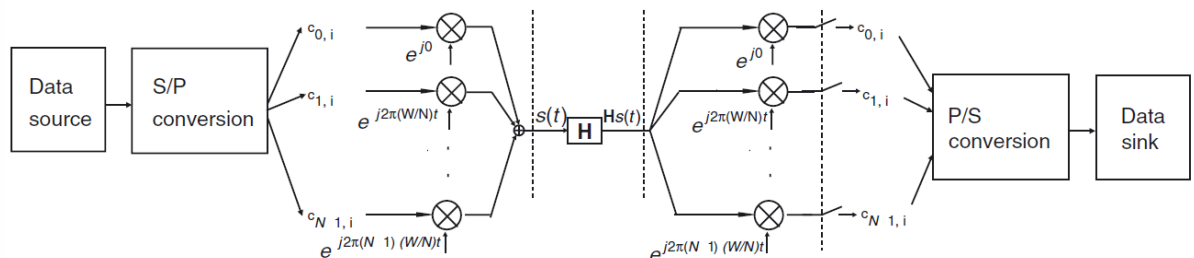
Gambar 2.3 menunjukkan contoh sederhana skema *multicarrier* dimana durasi simbol menjadi dua kali lipat pada *subcarrier* dengan membagi data menjadi data paralel. Gambar 2.4 menunjukkan kanal pita lebar frekuensi-selektif dapat diestimasi dengan kanal pita sempit flat multi frekuensi. Frekuensi non-selektif dari kanal pita sempit dapat mengurangi kompleksitas ekualiser untuk tiap subkanal.

Skema *multicarrier* konvensional tidak efisien dalam penggunaan spektrum. Perbandingan konsep skema OFDM yang saling tumpang tindih dengan

multicarrier konvensional pada Gambar 2.5 menunjukkan OFDM dapat memberi efisiensi spektral yang lebih. Spasi antar *subcarrier* (W/N) dinyatakan dalam $\Delta f = 1/T$ dimana T adalah waktu simbol.

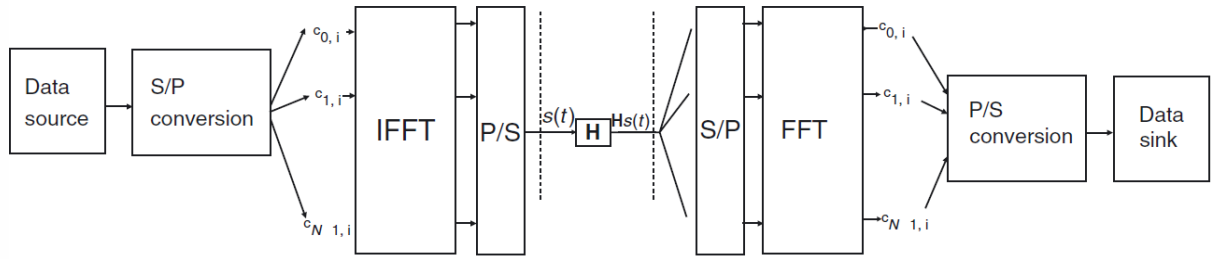


Gambar 2.5 Konsep Sinyal OFDM [2]: (a) Teknik *Multicarrier* Konvensional dan (b) OFDM.



Gambar 2.6 Sistem OFDM Dengan Bank Osilator [2]

OFDM dapat diinterpretasikan dalam dua cara, yang pertama adalah analog dimana data dijadikan data paralel N dimana masing-masing data melalui osilator lokal yang akan melakukan osilasi pada frekuensi *subcarrier*. Bank osilator tersebut membutuhkan perangkat lokal osilator yang banyak sehingga akan sulit diimplementasi secara aktual. Sistem OFDM analog diilustrasikan pada Gambar 2.6. Cara kedua adalah digital, dimana data paralel melalui transformasi IFFT/FFT atau DFT/IDFT. Cara ini dapat menghilangkan penggunaan bank osilator untuk implementasi OFDM. Sistem diilustrasikan pada Gambar 2.7.



Gambar 2.7. Sistem OFDM Menggunakan IFFT/FFT [2].

Skema transmisi OFDM adalah tipe lain dari sistem *multicarrier* dengan spektra yang dirancang saling tumpang tindih untuk mendapatkan efisiensi *bandwidth*. Sinyal *subcarrier* yang ortogonal agar saling tumpang tindih dihasilkan dengan mengaplikasikan sinyal *single carrier* dengan kriteria Nyquist pada persamaan 2.1 kedalam kriteria sinyal *multicarrier*.

$$\sum_{i=-\infty}^{\infty} G\left(f - \frac{i}{T}\right) = T \quad \dots(2.1)$$

Dalam aplikasinya, DFT dan IDFT berguna untuk implementasi sinyal ortogonal. DFT dan IDFT dapat diimplementasi secara efisien menggunakan *Fast Fourier transform* (FFT) dan *Inverse FFT* (IFFT).

2.2 Modulasi dan Demodulasi OFDM

Transmitter OFDM memetakan bit informasi menjadi sekuen simbol PSK (*Phase Shift Keying*) atau QAM (*Quadrature Amplitude Modulation*) yang akan dikonversi secara subsekuensial ke aliran N paralel [11]. Tiap simbol N dari konversi S/P (*serial to parallel*) dibawa ke *subcarrier* yang berbeda. $X_l[k]$ adalah simbol transmit l pada *subcarrier* k , untuk $l = 0, 1, 2, \dots, \infty$, $k = 0, 1, 2, \dots, N - 1$. Karena konversi S/P, durasi waktu transmisi untuk simbol N diperpanjang hingga NT_s , yang membentuk simbol OFDM tunggal dengan panjang T_{sym} ($T_{sym} = NT_s$). Sinyal OFDM untuk *subcarrier* k ditunjukkan pada persamaan 2.2.

$$\psi_{l,k}(t) = \begin{cases} e^{j2\pi f_k(t-lT_{sym})}, & 0 < t \leq T_{sym} \\ 0, & \text{elsewhere} \end{cases} \quad \dots(2.2)$$

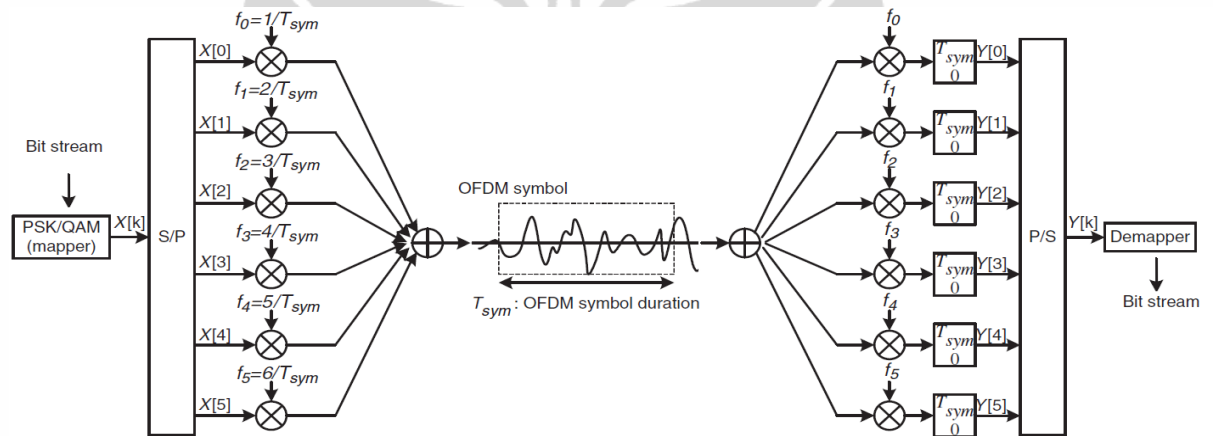
Maka sinyal OFDM pada ranah waktu diskrit adalah:

$$x_l[n] = \sum_{k=0}^{N-1} X_l[k] e^{j \frac{2\pi k n}{N}}, \text{ untuk } n = 0, 1, \dots, N-1 \quad \dots(2.3)$$

persamaan 2.3 diatas dapat disebut N -point IDFT dari PSK atau QAM simbol data $\{X_l[k]\}_{k=0}^{N-1}$ dan dapat dikomputasi secara efisien dengan IFFT.

Simbol OFDM baseband yang diterima $y_l(t) = \sum_{k=0}^{N-1} X_l[k] e^{j 2\pi f_k (t - lT_{sym})}$, $lT_{sym} < t \leq lT_{sym} + nT_s$, dimana simbol tertransmisi $X_l[k]$ dapat direkonstruksi oleh ortogonalitas antar *subcarrier* pada persamaan 2.4.

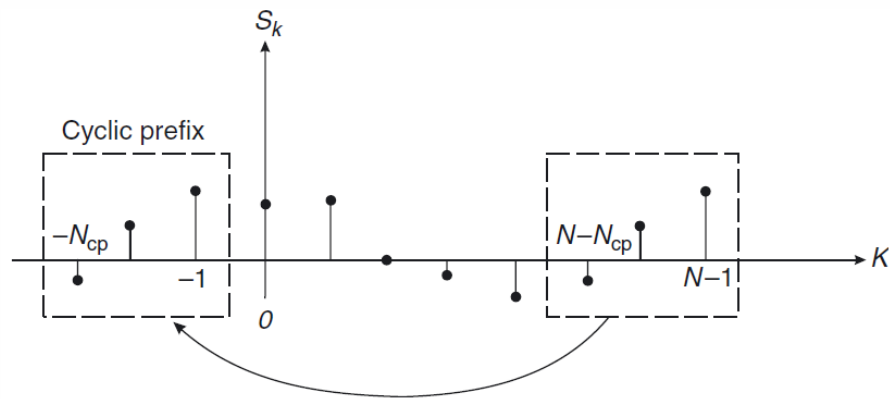
$$Y_l[k] = \frac{1}{N} \sum_{n=0}^{N-1} \sum_{i=0}^{N-1} X_l[i] e^{j 2\pi (i-k)n/N} = X_l[k] \quad \dots(2.4)$$



Gambar 2.8 Modulasi dan Demodulasi OFDM [11]

Persamaan 2.4 adalah DFT N -point dari fungsi $y(n)$ dan dapat dikomputasi secara efisien menggunakan FFT. Gambar 2.8 mengilustrasikan proses modulasi dan demodulasi OFDM. Simbol ranah frekuensi $X[k]$ memodulasi *subcarrier* dengan frekuensi dari $f_k = k/T_{sym}$, dan kemudian melalui proses demodulasi menggunakan ortogonalitas antar *subcarrier* pada *receiver*. Panjang simbol bertambah pada $T_{sym} = NT_s$ dengan mentransmisikan simbol N secara paralel.

2.3 Guard Interval

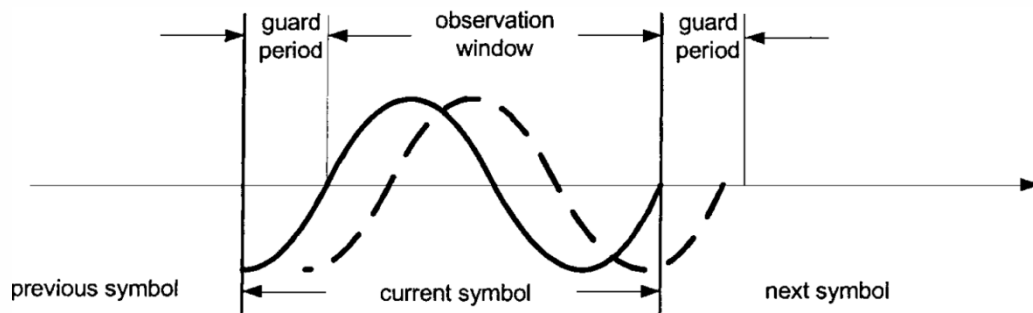


Gambar 2.9 Prinsip Kerja *Cyclic Prefix*[2]

Guard Interval untuk menghindari interferensi antar simbol OFDM yang berdekatan karena kurangnya interval waktu yang memisahkan kedua simbol tersebut [12]. Sinyal OFDM dapat didemodulasi tanpa interferensi antar *subcarrier* untuk kondisi tidak teratenuasi. Namun, pada kanal dispersif-waktu, ortogonalitas sinyal OFDM sebagian akan hilang. Hal ini disebabkan rugi-rugi ortogonalitas karena interval korelasi demodulator untuk satu jalur akan tumpang tindih dengan jalur lainnya.

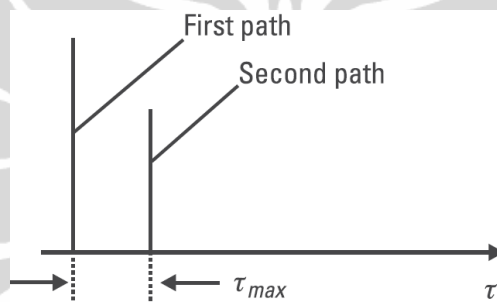
Salah satu prinsip dari sistem OFDM adalah penggunaan *Cyclic Prefix* sebagai interval antar sinyal *subcarrier* dari OFDM. *Cyclic prefix* digunakan untuk menambah simbol OFDM dengan menduplikat sampel terakhir dari simbol OFDM dan diletakkan dari awal simbol OFDM seperti ditunjukkan pada Gambar 2.9. Panjang dari sampel *cyclic prefix* harus lebih panjang atau sama dengan tundaan maksimum dari kanal *multipath*. *Cyclic prefix* memberi keunggulan pada kanal tundaan dispersif.

Ketika data ditransmisi melalui kanal tundaan dispersif, sinyal yang datang adalah konvolusi linear dari sinyal tertransmisi dengan tanggap impuls kanal. *Cyclic prefix* mengkonversi konvolusi linear menjadi konvolusi siklus pada waktu $-T_{cp} < t < -T_{cp} + \tau_{max}$, dimana τ_{max} adalah maksimum *excess delay* dari kanal. Sinyal diterima yang mengalami ISI sebagai bagian terakhir dari simbol awal yang menginterferensi simbol tersebut. ISI dapat dihilangkan dengan mengabaikan sinyal yang mengalami interferensi.



Gambar 2.10 Ilustrasi Efek Penggunaan *Cyclic Prefix* [3]

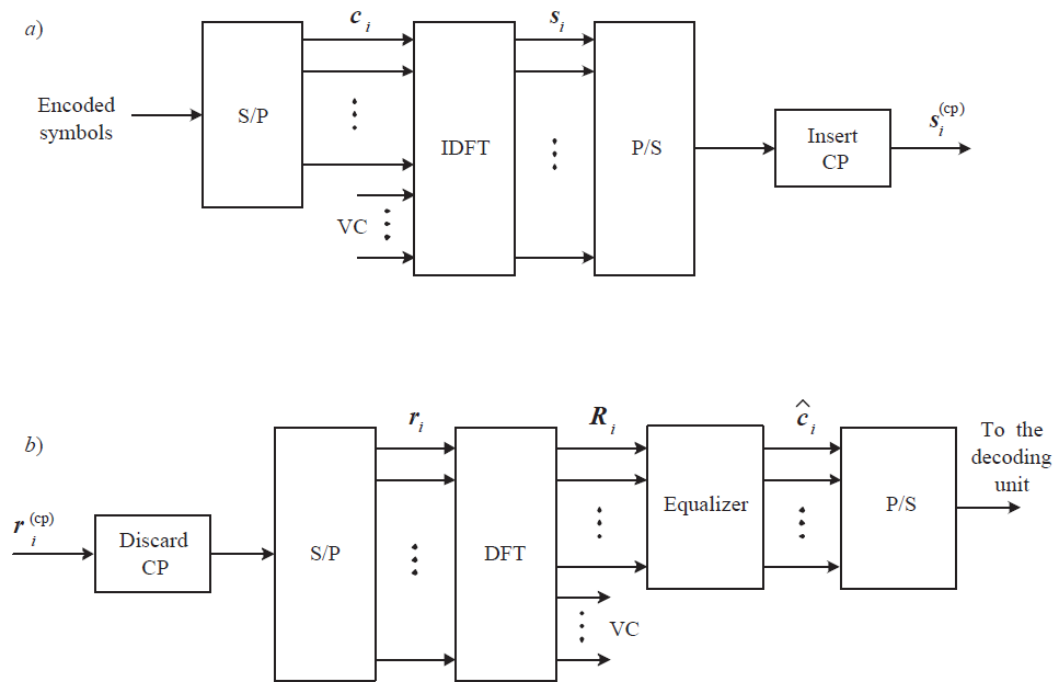
Gambar 2.10 Menunjukkan penggunaan komponen *multipath* dengan tundaan kurang atau sama dengan τ_{max} akan menjaga gelombang eksponensial kompleks. τ_{max} adalah waktu tundaan antar *path* seperti ditunjukkan Gambar 2.11.



Gambar 2.11 Tanggap Impuls Kanal Frekuensi Selektif [13].

2.4 Guard Band

Tiap komponen simbol OFDM dengan durasi efektif T_{sub} dapat dianggap sebagai sinyal *single-tone* dikalikan dengan *window rectangular* dengan durasi T_{sub} . Dimana spektrum tersebut adalah fungsi *sinc* dengan *bandwidth* $2/T_{sub}$. Maka spektrum daya sinyal OFDM adalah jumlah dari fungsi *sinc* pergeseran frekuensi, yang memberi radiasi *out-of-band* seperti *Adjacent Channel Interference* (ACI). Maka dibutuhkan *guard band* untuk mengurangi efek dari ACI dalam OFDM. Untuk mengurangi daya *out-of-band* dari simbol OFDM dapat digunakan fungsi *shaping* ranah waktu seperti *raised-cosine* (RC) *windowing*. Alternatif lainnya untuk mengurangi ACI adalah menggunakan *virtual carrier* (VC). Namun VC mengurangi efisiensi *bandwidth* karena adanya *subcarrier* yang tidak digunakan. Penggunaan *virtual carrier* pada OFDM ditunjukkan pada Gambar 2.12.



Gambar 2.12 Ilustrasi Implementasi *Virtual Carrier* pada OFDM [1]

2.5 Peak to Average Power Ratio (PAPR)

Sinyal transmisi pada sistem OFDM dapat memiliki nilai puncak tinggi pada ranah waktu karena komponen *subcarrier* ditambah melalui operasi IFFT [11]. Maka sistem OFDM memiliki PAPR yang tinggi dibandingkan sistem *single carrier*. PAPR yang tinggi adalah aspek yang merusak pada sistem OFDM, dapat menurunkan SQNR (*signal to Quantization Noise Ratio*) dari ADC (*Analog to Digital Converter*) dan DAC (*Digital to Analog Converter*) dan mengurangi efisiensi daya *amplifier* pada transmitter.

PAPR adalah rasio antar daya maksimum dan daya rerata dari sinyal kompleks $s(t)$ yaitu:

$$PAPR\{\tilde{s}(t)\} = \frac{\max |Re(\tilde{s}(t)e^{j2\pi f_c t})|^2}{E\{|Re(\tilde{s}(t)e^{j2\pi f_c t})|^2\}} = \frac{\max |s(t)|^2}{E\{|s(t)|^2\}} \quad \dots(2.5)$$

Pada sistem PSK/OFDM dengan N *subcarrier*, daya maksimum muncul ketika seluruh komponen N SC ditambah dengan fase identik. Dengan asumsi *rata – rata daya* = 1, menghasilkan $PAPR = N$, sehingga daya maksimum sama dengan $N \times \text{daya rata – rata}$. Dan kemungkinan kemunculan daya

maksimum sinyal menurun seiring peningkatan N .

2.6 Clipping

Cara paling sederhana untuk mengurangi PAPR adalah memotong sinyal amplitudo puncak menjadi terbatas ke level maksimum yang dibutuhkan. Walau menjadi solusi tersederhana, *clipping* menyebabkan terjadinya distorsi *in-band*, yang menyebabkan penurunan BER, kedua distorsi non-linear pada sinyal OFDM secara signifikan meningkatkan level emisi OOB. Teknik *Clipping* yang digunakan pada tesis ini adalah *Classical Clipping* pada persamaan 2.6, *Deep Clipping* dengan persamaan yang ditunjukkan pada persamaan 2.7 dan *Smooth Clipping* dengan persamaan 2.8.

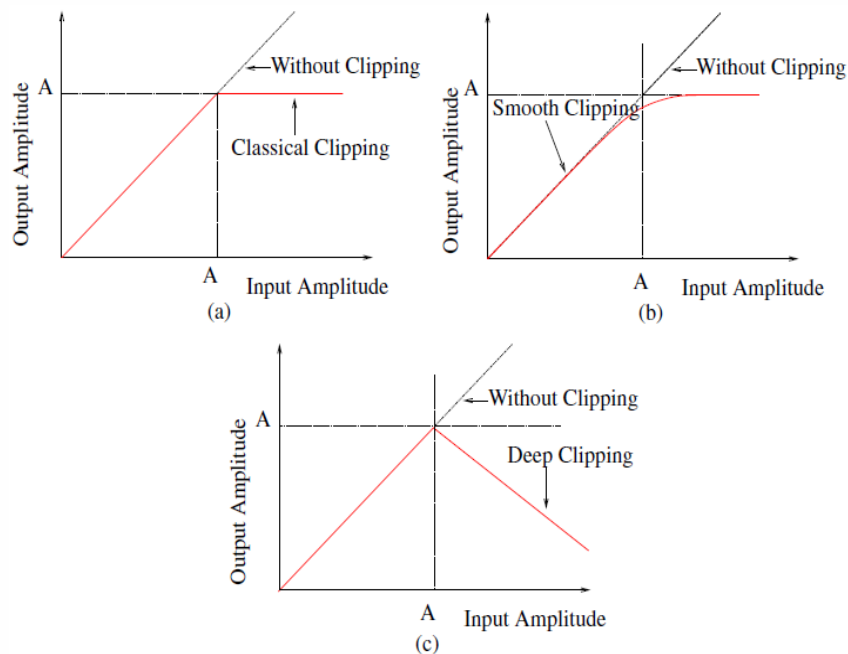
$$f(r) = \begin{cases} r, & r \leq A \\ A, & r > A \end{cases} \quad \dots(2.6)$$

$$f(r) = \begin{cases} r, & r \leq A \\ A - p(r - A), & A < r \leq \frac{1+p}{p}A \\ 0, & r > \frac{1+p}{p}A \end{cases} \quad \dots(2.7)$$

$$f(r) = \begin{cases} r - \frac{1}{b}r^3, & r \leq \frac{3}{2}A \\ A, & r > \frac{3}{2}A \end{cases} \quad \dots(2.8)$$

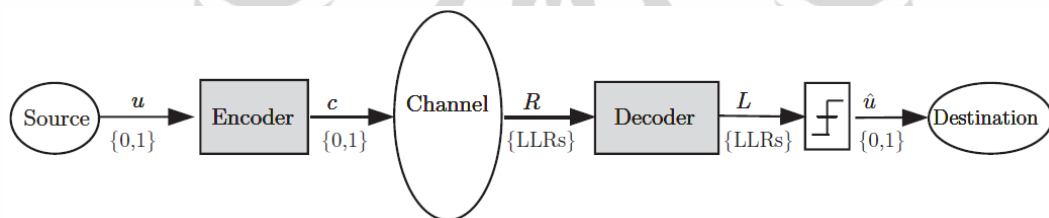
Dimana A adalah level *clipping* yang ditentukan, p adalah faktor kedalaman (*depth factor*) *clipping* dan $b = \frac{27}{4}A^2$.

Karakteristik perbandingan teknik *clipping* untuk *Deep Clipping* (DC), *Classical Clipping* (CC), dan *Smooth Clipping* (SC) ditunjukkan pada Gambar 2.13 untuk perbandingan fungsi alih *Clipping*.



Gambar 2.13. Perbandingan Fungsi Alih Clipping. [7] (a). *Classical Clipping*, (b). *Deep Clipping* dan (c) *Smooth Clipping*.

2.7 Kode Koreksi Error (*Error Correction Coding*)



Gambar 2.14 Ilustrasi Aplikasi Kode Koreksi Error Dalam Sistem Komunikasi [14].

Dalam sistem komunikasi digital, pilihan yang dapat diterapkan untuk mengubah kualitas data transmisi adalah menggunakan *Error Correction Coding* (ECC). ECC adalah sarana dimana error yang terjadi dalam data digital sebagai hasil dari transmisi dari kanal komunikasi dapat dikoreksi berdasarkan data yang diterima [8]. Gambar 2.14 Menunjukkan blok diagram penggunaan ECC dalam sistem komunikasi dimana bit info u di-encode menjadi c dan pada transmitter didekode menjadi bit info yang diterima. Penggunaan ECC membutuhkan sumber daya transmisi (*bandwidth*) karena penggunaan data redundansi. Jenis ECC diklasifikasikan menjadi dua berdasarkan cara redundansi ditambahkan, yaitu *Block Code* (BC) dan *Convolutional Code* (CC) [15]. BC melakukan encode dan

dekode berbasis blok per blok dengan blok data yang independen dan operasi *memoryless* serta dapat diimplementasikan menggunakan kombinasi logik. BC mengumpulkan data menjadi blok dan dari nilai bit dalam blok tersebut dikomputasi menjadi *codeword* yang lebih panjang dan ditransmisikan. Semakin kecil *code rate* (rasio jumlah bit dari data dan *codeword*), semakin kecil tingkat redundansinya, semakin tinggi probabilitas error dapat dikoreksi.

CC bekerja sebagai *stream* data kontinu, dan proses encode dan dekode bergantung juga pada data sebelumnya. CC menggunakan memori dan harus diimplementasikan menggunakan logik sekuensial. *Codeword* terdiri dari bit info dan redundansi, jika redundansi di tanamkan secara implisif (*embedded*) disebut non-sistematik. Jika redundansi ditambahkan disebut sistematik.

Suatu kode dinyatakan linear jika dua *codeword* dalam kode dapat ditambahkan dalam modul 2 aritmatik untuk menghasilkan *codeword* ketiga dalam kode. Kode linear biner memenuhi syarat, jika $x_1, x_2 \in \mathcal{C}$ maka $x_1 + x_2 \in \mathcal{C}$. *Codeword* untuk $x \in \mathcal{C}$ adalah $x = u_1 v_1 + \dots + u_k v_k$ [16].

Kode *Low Density Parity Check* (LDPC) dan kode *Turbo* adalah teknik ECC yang termasuk dalam jenis iteratif ECC karena dalam proses decoding menggunakan iterasi. LDPC digunakan sebagai standar DVB-S2 yang mengalahkan 6 Turbo Code untuk transmisi satelit televisi digital. LDPC juga digunakan sebagai standar ITU-T G.hn mengalahkan skema *Convolutional Turbo Code*, serta digunakan pada 10Gbase-T Ethernet dan standar Wi-Fi 802.11 sebagai bagian opsional dari 802.11n dan digunakan pada WiMAX [17].

2.8 LDPC

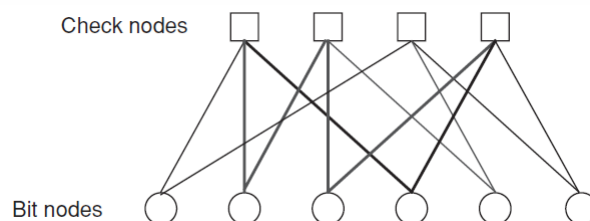
Kode LDPC adalah kode blok linear dengan algoritma decoding iteratif yang diajukan oleh Gallager tahun 1962 dan hampir dilupakan hampir selama 30 tahun. Kini LDPC dikenal sebagai kode ECC yang baik yang mendekati performa Shannon limit.

LDPC dijelaskan sebagai kode LDPC dengan matriks *parity check* $\mathbf{H}$ $M \times N$ sebagai (N, K) LDPC. Dimana $K = N - M$ dan *code rate* $R = K / N$. Jika $\mathbf{H}$ tidak memiliki *full rank* ($K > N - M$) sehingga kinerja koreksi error dari kode LDPC menjadi buruk. Dalam merancang matriks *parity-check* $\mathbf{H}$, diperlukan

seluruh baris dari matriks independen secara linear. LDPC dapat di-dekode-kan menggunakan algoritma probabilitas propagasi yang diketahui Algoritma *Sum-Product* (SPA) atau algoritma *Belief-Propagation*. Kode LDPC memiliki kinerja blok error yang lebih baik dibandingkan kode Turbo [18], karena jarak minimum dari kode LDPC meningkat proporsional ke panjang kode dengan probabilitas tinggi. Sifat yang diinginkan untuk transmisi dengan *bit rate* yang tinggi dengan probabilitas error *frame* yang rendah. Pada sisi negatifnya, kode LDPC memiliki proses enkoding yang lebih kompleks dibandingkan Turbo serta proses dekoding dapat memerlukan iterasi yang lebih banyak dibandingkan Turbo yang memberi dampak pada *latency*.

Kode LDPC diklasifikasikan menjadi dua jenis, yaitu kode LDPC regular dan kode LDPC iregular. Kode regular LDPC memiliki jumlah ‘1’ untuk kolom dan baris yang tetap sedangkan kode LDPC iregular tidak seragam pada matriks *parity check*. Aspek penting dari perancangan LDPC adalah tidak melalui pembangkit matriks $\mathbf{G}$ namun melalui matriks *parity check* $\mathbf{H}$. Hal ini memberi kemudahan untuk proses dekoding karena struktur yang telah didefinisikan dengan ukuran blok dan dimensi matriks yang besar. Jumlah “1” pada *parity check* dirancang agar sedikit sehingga kode ini disebut “*low density*”. Kode ini didefinisikan sebagai kode menggunakan matriks *sparse parity check* dengan jumlah ‘1’ per kolom dan sejumlah ‘1’ per baris, dimana keduanya sangat kecil dibandingkan dengan panjang blok. *Sparse* didefinisikan sebagai banyak komponen ‘0’ dan sedikit ‘1’.

LDPC dapat direpresentasikan oleh *Tanner/Factor Graph* yang terdiri atas dua tipe node. Yaitu bit/variabel node dan cek/*constraint node*.



Gambar 2.15 Tanner Graph [14]

Gambar 2.15 menunjukkan contoh Tanner Graph, tiap bit node

berhubungan ke kolom dari matriks *parity check* yang juga korespon dengan bit *codeword*. Tiap cek node korespon ke baris matriks *parity check* yang mewakili persamaan *parity check*.

Dalam lingkungan *multipath*, beberapa *subcarrier* dari OFDM dapat hilang karena *fading*. Maka untuk kasus ini, diharapkan banyak perbaikan error pada beberapa *subcarrier* dan error dua dimensi error dalam ranah waktu dan frekuensi yang muncul. Maka kode LDPC diaplikasikan agar dapat mengkompensasi error dua dimensi pada sistem OFDM.

Matriks *parity check* dapat dirancang menggunakan beberapa aturan sederhana, pertama bagi matriks secara horisontal menjadi sub-matriks p dengan jumlah yang sama, kemudian masukkan satu “1” tiap kolom pada sub-matriks. dengan submatriks pertama sebagai matriks identitas. Kemudian submatriks berikutnya menjadi perulangan kolom acak dari submatriks pertama.

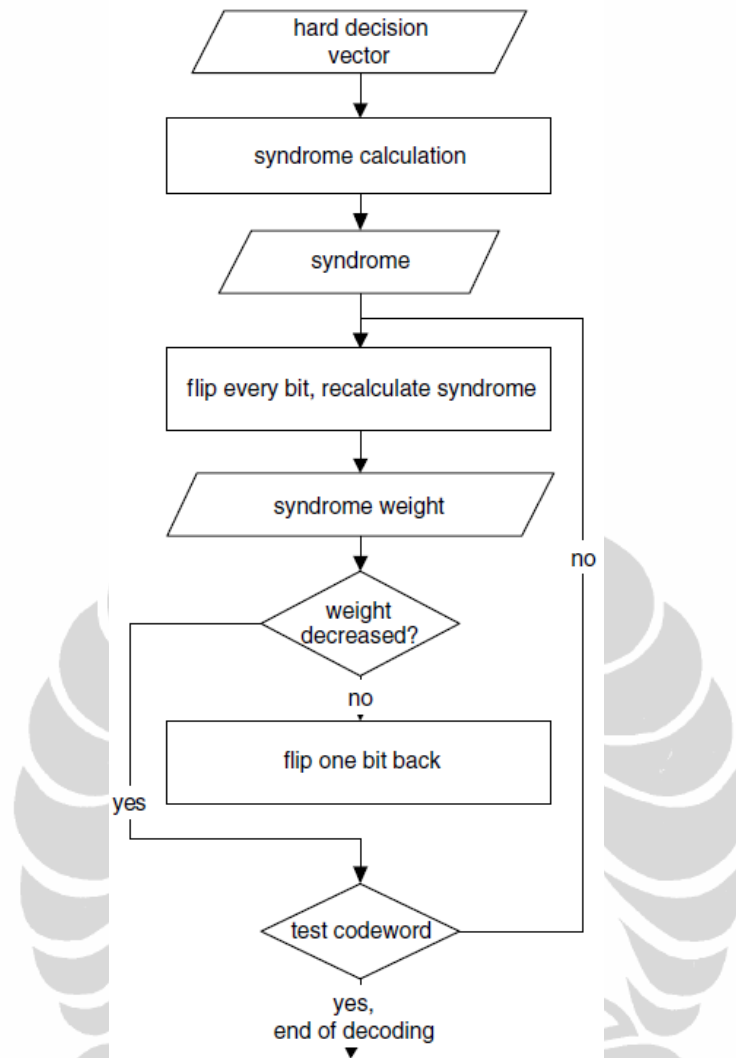
Karena kode LDPC ditentukan melalui matriks *parity check*, proses enkoding lebih kompleks dibandingkan kode blok lainnya [2]. Untuk kode blok pada umumnya proses enkoding adalah perkalian dari vektor *codeword* dengan matriks pembangkit. Namun untuk kode LDPC, matriks pembangkit belum diketahui. Untuk mendapatkan matriks pembangkit G , didapatkan dengan persamaan 2.9. dimana parameter A didapatkan dari persamaan 2.10.

$$G = [I_K \ A^T] \quad \dots(2.9)$$

$$H = [A \ I_{N-K}] \quad \dots(2.10)$$

Dimana I adalah matriks identitas dan A adalah *parity check* matriks. Operasi enkoding cukup sederhana karena dilakukan pada ranah diskrit dalam bit yang dilakukan menggunakan persamaan 2.11.

$$c = u G \quad \dots(2.11)$$



Gambar 2.16 . Diagram Alir Algoritma *Bit Flipping* [19]

Message Passing adalah kelas algoritma yang digunakan untuk decode LDPC yang merupakan algoritma dekoding iteratif dimana data diolah melalui *bit node* dan *check node* secara iteratif hingga mencapai iterasi maksimum atau mendapatkan hasil dekoding.

Skema dekoding untuk kode LDPC yang diperkenalkan oleh Gallager adalah algoritma *Bit Flipping* untuk *hard decision* dan algoritma *belief propagation* untuk *soft decision* [19]. Algoritma *belief propagation* dapat digunakan dengan probabilitas dan *log likelihood ratio* dan memberi hasil yang baik pada kanal AWGN. Prinsip dari *Bit Flipping* adalah untuk beberapa simbol c_i yang diterima, simbol lainnya tidak dapat lebih dari satu simbol c_i karena matriks *parity check* yang *sparse* dan tidak adanya siklus pendek dalam grafik Tanner atau dapat

dikatakan *parity check* ortogonal pada simbol c_i . Langkah dekoding untuk *Bit Flipping* adalah sebagai berikut:

1. Kalkulasi sindrom ($c \cdot H^T$) dari sinyal yang diterima, atau dari hasil iterasi, jika sindrom adalah nol atau iterasi maksimal sudah tercapai maka prosedur iterasi dekoding selesai.
2. Kalkulasi data yang tidak memenuhi *parity check* untuk tiap simbol c_i .
3. ‘Balik (*flip*)’ simbol yang memiliki data terbanyak yang tidak memenuhi langkah no.2.
4. Kembali kelangkah pertama.

Diagram alir algoritma *Bit Flipping* ditunjukkan pada Gambar 2.16. Algoritma *Bit Flipping* memiliki kompleksitas yang rendah karena untuk tiap bit yang telah diperbarui dapat digunakan.

Algoritma *Sum Product* adalah algoritma dekoding *message passing* untuk *soft decision*. Skema dekoding *Sum Product* mirip dengan algoritma *Bit Flipping* namun data yang digunakan adalah probabilitas atau *Log-Likelihood Ratio* (LLR). *Soft decision* sebagai input adalah *a posterior probabilities* (APP):

$$p_n = P(c_n = x | r_n) \quad \dots(2.12)$$

Untuk inisiasi awal, parameter $q_{mn}(x) = p_n(x)$ untuk semua (m,n) dengan $A(m,n) = 1$. Pada langkah horisontal untuk tiap (m,n) dengan $A(m,n) = 1$ dilakukan komputasi $\delta_{q_{ml}} = q_{ml}(0) - q_{ml}(1)$. Kemudian komputasi:

$$\delta_{r_{mn}} = \prod_{\{n' \in \mathcal{N}_{m,n}\}} \delta_{q_{mn'}} \quad \dots(2.13)$$

Komputasi $r_{mn}(1) = \frac{1-\delta_{r_{mn}}}{2}$ dan $r_{mn}(0) = \frac{1+\delta_{r_{mn}}}{2}$. Pada langkah vertikal, dilakukan komputasi

$$q_{mn}(0) = \alpha_{mn} p_n(0) \prod_{\{m' \in \mathcal{M}_{m,n}\}} r_{m'n}(0) \quad \dots(2.14)$$

$$q_{mn}(1) = \alpha_{mn} p_n(1) \prod_{\{m' \in \mathcal{M}_{m,n}\}} r_{m'n}(1)$$

Dimana α_{mn} dipilih agar $q_{mn}(0) + q_{mn}(1) = 1$. Serta komputasi:

$$q_n(0) = \alpha_n p_n(0) \prod_{\{m \in \mathcal{M}_n\}} r_{m'n}(0)$$

$$q_n(1) = \alpha_n p_n(1) \prod_{\{m \in \mathcal{M}_n\}} r_{m'n}(1)$$

...(2.15)

Dimana α_n dipilih agar $q_n(0) + q_n(1) = 1$. Kemudian dilakukan keputusan tentatif dimana $\hat{c}_n = 1$ jika $q_n(1) > 0.5$ jika tidak maka $\hat{c}_n = 0$. Jika $c.H^T = 0$ maka iterasi berhenti, jika tidak iterasi dilakukan kembali pada tahap horisontal hingga batas maksimum iterasi.

Proses algoritma Sum Product menggunakan LLR dijelaskan sebagai berikut:

1. Vektor *Soft Decision* :Inisiasi LLR $L(P_{ij}) = L(c_i)$
2. Dari *check node* ke *bit node* : untuk tiap *check node* i dengan *edge* ke *bit node* j : perbarui

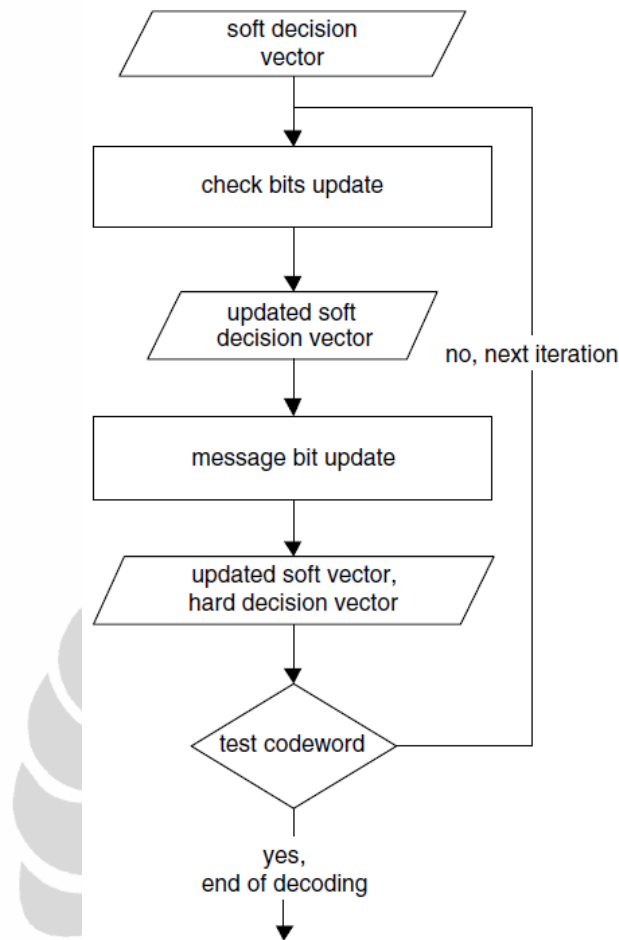
$$L(Q_{ij}) = \prod_{j'} \alpha_{ij'} \phi \left[\sum_{j'} \phi(\beta_{ij'}) \right]$$

...(2.16)

Untuk $(j' = 1, 2, \dots, n)$ dan $j' \neq j$. $\alpha_{ij'} \triangleq \text{sign}(L(P_{ij}))$ dan $\beta_{ij'} \triangleq \text{sign}(L(P_{ij}))$.

$$\phi(x) = -\ln \left[\tanh \left(\frac{x}{2} \right) \right] = \ln \left[\frac{(e^x + 1)}{(e^x - 1)} \right]$$

...(2.17)



Gambar 2.17 Diagram Alir Algoritma Sum Product [19].

3. *Message bit*: Dari *bit node* ke *check node*: untuk tiap *bit node j* dengan *edge* ke *check node i*:
Perbarui:

$$L(P_{ij}) = L(c_j) + \sum_{i'} L(Q_{ij}) \quad \dots(2.18)$$

untuk $i' = 1, 2, \dots, m$ dan $i' \neq i$.

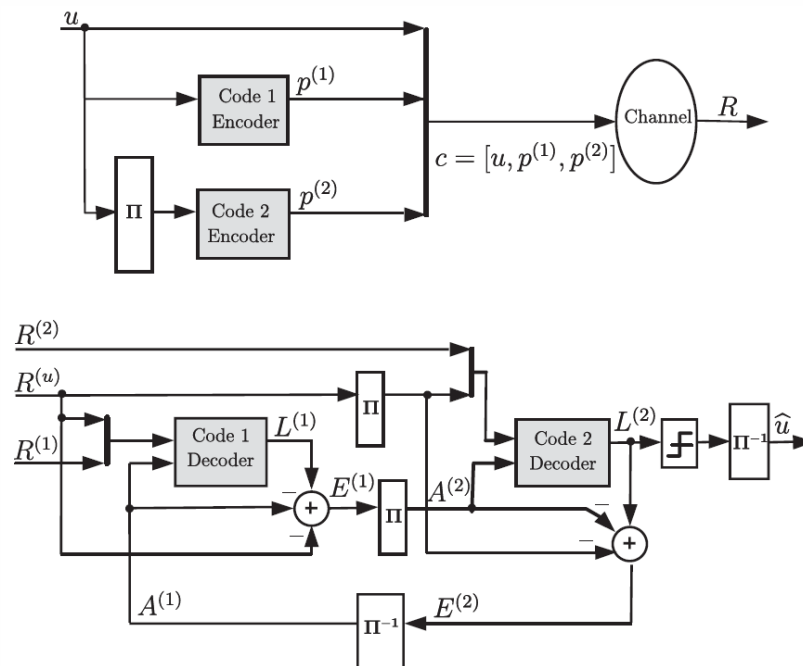
4. Nilai $L(P_{ij})$ digunakan untuk keputusan tentatif sebagai hasil dekoding (vektor *hard decision*) yang ditunjukkan persamaan 2.21.

$$c_j = \begin{cases} 1, & \text{jika } L(P_j) < 0 \\ 0, & \text{selainnya} \end{cases} \quad \dots(2.19)$$

5. Jika $c \cdot H^T = 0$ atau iterasi mencapai iterasi maksimum yang telah ditentukan maka proses dekoding selesai, jika tidak kembali ke langkah 2.

Diagram alir skema dekoding Sum Product ditunjukkan pada Gambar 2.17.

2.9 Kode Turbo



Gambar 2.18 Blok Diagram Kode Turbo [14].

Kode Turbo atau disebut sebagai *Parallel Concatenated Convolutional Code* (PCCC) adalah kode yang diperkenalkan oleh Berrou, Glavieux dan Thitimajshima. Kode Turbo adalah penggabungan dua CC dimana pada penerima saling berbagi informasi antar dekoder. Enkoder menggunakan dua enkoder konvolusional dan pada dekoder menggunakan dua dekoder BCJR [14]. Penggabungan dua enkoder (*Concatenate*) dilakukan agar mendapatkan prinsip dari dekoder iteratif [19]. Enkoder Turbo menggunakan dua atau lebih BC sistematis yang berbagi bit info menggunakan *interleaver*. Dalam realisasinya, kode *Recursive Systematic Convolutional* (RSC) digunakan untuk enkoder Turbo [8]. RSC untuk input sekuen *low weight* dapat membangkitkan output sekuen *infinite weight* dibandingkan enkoder CC konvolusional. Kombinasi RSC dan *interleaver* dapat memberi peningkatan struktur jarak dari kode Turbo karena tidak menghasilkan pasangan sekuen kode *low weight*.

Dalam algoritma dekoding iteratif, dekoder dari tiap enkoder bergantian

mengolah data yang diterima, tiap dekoder menghasilkan estimasi probabilitas dari simbol yang ditransmisikan. Dekoder saling mengirimkan probabilitas antar dekoder hingga iterasi berakhir. Karena output dari satu dekoder adalah input dari dekoder lainnya, maka algoritma dekoding ini disebut dekoder Turbo. Blok diagram dari sistem kode Turbo ditunjukkan pada Gambar 2.18. Kinerja kode Turbo ditentukan oleh penggunaan *interleaver* dan algoritma iteratif dengan kompleksitas rendah.

Kode RSC sebagai realisasi alternatif dari non sistematis rate $1/n$ CC. CC konvensional dikonstruksi dalam cara *feed forward*, yaitu enkoder tanpa *feedback*. RSC melibatkan *feedback* pada proses encoding secara ekuivalen. Ekuivalen berarti kode rekursif memiliki sifat jarak yang sama seperti *non-recursive counterpart*. Untuk rekonstruksi kode RSC, transformasi pembangkit kode non sistematis *feed-forward* dibutuhkan untuk menghasilkan CC konvensional menjadi generator sistematis *feedback* atau output dari enkoder diumpan balik ke state enkoder [14]. Matriks pembangkit G untuk RSC adalah:

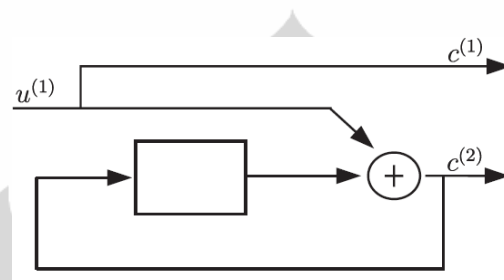
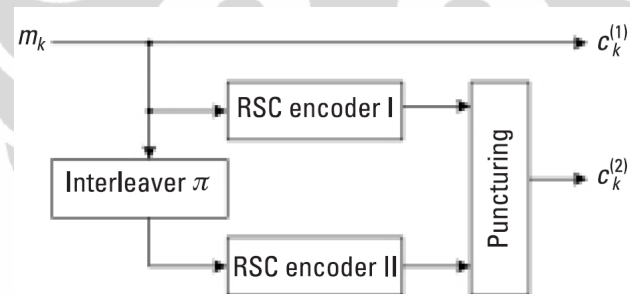
$$G(D) = \begin{bmatrix} 1 & \frac{g^{(2)}(D)}{g^{(1)}(D)} \end{bmatrix} \quad \dots(2.20)$$

Dimana matriks pembangkit disebut matriks transfer function dimana $g^{(1)}(D)$ dan $g^{(2)}(D)$ adalah transfer function polinomial. Enkoder yang memiliki fungsi rasio dalam matriks *transfer function* disebut *feedback* atau enkoder IIR [8]. Tabel 2.1 menunjukkan perbandingan fungsi $G(D)$ dengan variasi skema enkoder CC.

Flushing bit ditambahkan pada akhir encoding untuk memaksa enkoder RSC pertama kedalam keadaan nol atau terminasi kode. RSC kedua tidak atau perlu dalam keadaan nol. Dua enkoder RSC sering disebut sebagai komponen enkoder. Gambar 2.19 menunjukkan enkoder biner RSC dengan *code rate* $\frac{1}{2}$ dimana kode yang dihasilkan adalah sistematis karena bit input $u^{(1)}$ dipetakan langsung ke bit output $c^{(1)}$ dan disebut rekursif karena bit output $c^{(2)}$ dikirimkan kembali ke *shift register*.

Tabel 2.1. Contoh Kelas Enkoder Convolutional [20].

Kelas Enkoder CC	Contoh $G(D)$
non-rekursif non-sistematik	$[1 + D + D^2 \quad 1 + D^2]$
rekursif-sistematik	$\begin{bmatrix} 1 & \frac{1 + D^2}{1 + D + D^2} \end{bmatrix}$
rekursif non-sistematik	$\begin{bmatrix} \frac{1 + D^2}{1 + D + D^2} & 0 & 1 + D \end{bmatrix}$
non-rekursif sistematik	$[1 \quad 1 + D + D^2]$

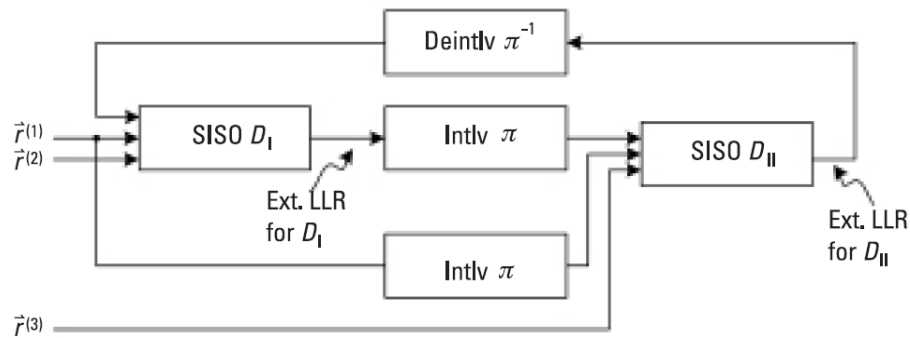
Gambar 2.19 Skema *Recursive Systematic Convolutional* [14].Gambar 2.20 Penggunaan *Puncture* Untuk Menghasilkan *Code Rate* $\frac{1}{2}$. [15].

Kode dapat di *puncture* yang berguna untuk menghapus bit yang terpilih agar mendapatkan *coding rate* yang lebih tinggi, *puncturing* hanya dapat beroperasi pada sekuen paritas. Bit sistematik tidak dilakukan proses *puncture*. Gambar 2.20 menghasilkan kenaikan *rate* menjadi $\frac{1}{2}$ dengan menggunakan *puncture*. *Interleaver* memegang peranan penting, yaitu melakukan permutasi data sekuen yang dikirim ke enkoder kedua dan juga membangkitkan paritas (*codeword*) kedua yang independen dari paritas yang dibangkitkan enkoder pertama.

Sebelum proses dekoding, sekuen r dilakukan *demultiplexing* menjadi 3 subsekuen dari $\tilde{c}^{(1)}$, $\tilde{c}^{(2)}$ dan $\tilde{c}^{(3)}$ menjadi $\tilde{r}^{(1)}$, $\tilde{r}^{(2)}$ dan $\tilde{r}^{(3)}$. Sistem kerja dari

Turbo dekoder dijelaskan sebagai berikut dengan struktur pada Gambar 2.21.

1. $\vec{r}^{(1)}, \vec{r}^{(2)}$ menjadi input D_I .
2. D_I melakukan dekoding dan menghasilkan informasi ekstrinsik LLR (a posteriori ke D_I)
3. Informasi ekstrinsik LLR dilakukan *interleaving* dan dikirim ke D_{II} .
4. D_{II} menggunakan input dari D_I sebagai *priori* untuk dekoding dan menghasilkan informasi ekstrinsik yang lebih akurat untuk D_I .
5. Informasi ekstrinsik dari D_{II} di *deinterleaving* dan dikirim kembali ke D_I sebagai iterasi baru dekoding.
6. Iterasi dilakukan hingga mencapai iterasi maksimum.



Gambar 2.21 Skema Dekoding Kode Turbo [15].

Dekoder menerima *soft* LLR dan membangkitkan *soft* LLR sehingga disebut *soft input soft output* dekoder. LLR dapat dipisah menjadi jumlah 3 bagian, kuantitas yang terkait dengan kondisi kanal χ_k dari persamaan 2.21, *a priori* probabilitas L_k^{apri} , dan *a posteriori* probabilitas $L_k^{a pos}$ yang ditunjukkan persamaan 2.22.

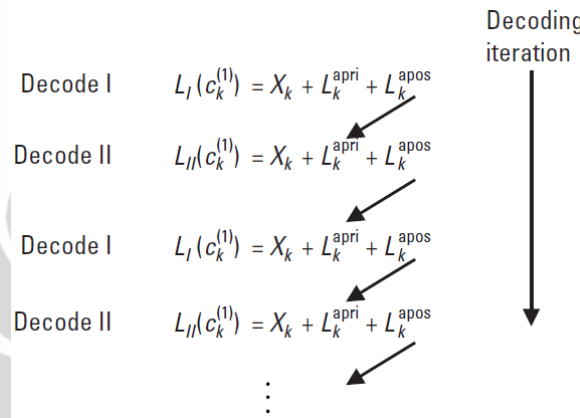
$$\chi_k = L_c r_k^{(1)} \quad \dots(2.21)$$

$$L(c_k^{(1)}) = \chi_k + L_k^{apri} + L_k^{a pos} \quad \dots(2.22)$$

L_k^{apri} adalah LLR yang sangat ekstrinsik oleh komponen dekoder. $L_k^{a pos}$ adalah ekstrinsik LLR yang komponen dekoder bangkitkan untuk digunakan pada dekoder berikutnya. Penggunaan LLR dalam proses dekoding diilustrasikan pada

Gambar 2.22.

Skema dekoding BCJR harus menggunakan multiplikasi yang banyak. Untuk mengurangi beban ini, Max Log MAP dan Log MAP sebagai algoritma simplifikasi dari BCJR. Kedua algoritma ini menggantikan penambahan menjadi multiplikasi. Alternatif dari dekoding MAP adalah Soft Output Viterbi Algorithm (SOVA). SOVA adalah versi pembaruan dari algoritma Viterbi konvensional. SOVA menggunakan input *a priori* dan output tambahan *soft* reliabilitas dari bit terdekode.



Gambar 2.22 Ilustrasi Penggunaan LLR Untuk Dekoding Turbo [15].

Langkah algoritma dekoder BCJR MAP dijelaskan sebagai berikut:

1. Inisiasi nilai α'_0 dan β'_N
2. Untuk $k = 0, 1, \dots, N - 2$:

$$\alpha_k(S) = \sum_{S'} \gamma_k(S', S) \alpha_{k-1}(S') \quad \dots(2.23)$$

3. Untuk $k = N - 1, N - 2, \dots, 1$:

$$\beta_{k-1}(S') = \sum_S \gamma_k(S', S) \beta_k(S) \quad \dots(2.24)$$

4. Kemudian komputasi probabilitas *posterior* untuk s_k :

$$P(s_{k-1} = S', s_k = S, \vec{r}) = \sum \alpha_{k-1} \gamma_k(S', S) \beta_k(S) \quad \dots(2.25)$$

persamaan 2.26 adalah algoritma Log-MAP dilakukan dengan aplikasi logaritmik natural pada probabilitas α , β dan γ .

$$\begin{aligned} A_k(S) &\triangleq \ln[\alpha_k(S)] \\ B_k(S) &\triangleq \ln[\beta_k(S)] \\ \Gamma_k(S', S) &\triangleq \ln[\gamma_k(S', S)] \end{aligned} \quad \dots(2.26)$$

Persamaan algoritma Max-Log-MAP menggunakan aproksimasi pada persamaan 2.27 Untuk menghindari penjumlahan eksponensial dalam penghitungan A_k , B_k dan $L(c_k^{(1)})$.

$$\ln\left(\sum_i e^{x_i}\right) \approx \max(x_i) \quad \dots(2.27)$$

Sehingga didapatkan A_k dan B_k pada persamaan 2.30.

$$\begin{aligned} A_k(S) &\approx \max_S [\Gamma_k(S', S) + A_{k-1}(S)] \\ B_k(S) &\approx \max_S [\Gamma_k(S', S) + B_k(S)] \end{aligned} \quad \dots(2.28)$$

Maka persamaan LLR untuk metode BCJR MAP, Log MAP dan Max-Log-MAP masing-masing berturut-turut ditunjukkan oleh persamaan 2.29, 2.30 dan 2.31.

$$L(c_k^{(1)}) = \ln \frac{\sum_{S^+} \alpha_{k-1}(S') \gamma_k(S', S) \beta_k(S)}{\sum_{S^-} \alpha_{k-1}(S') \gamma_k(S', S) \beta_k(S)} \quad \dots(2.29)$$

$$L(c_k^{(1)}) = \ln \frac{\sum_{S^+} \exp[A_{k-1}(S') + B_k(S) + \Gamma_k(S', S)]}{\sum_{S^-} \exp[A_{k-1}(S') + B_k(S) + \Gamma_k(S', S)]} \quad \dots(2.30)$$

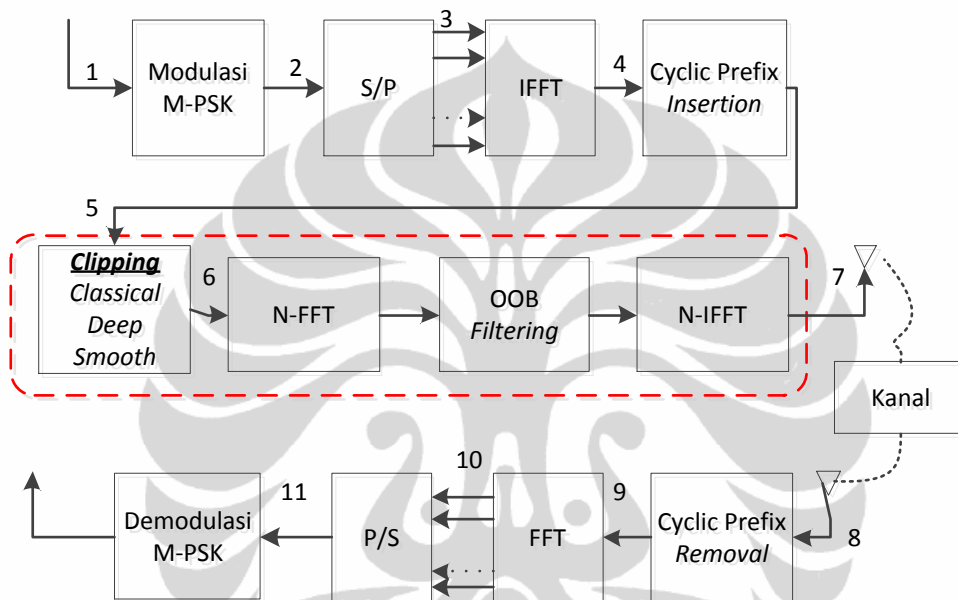
$$\begin{aligned} L(c_k^{(1)}) &\approx \max_{S^+} [A_{k-1}(S') + B_k(S) + \Gamma_k(S', S)] \\ &\quad - \max_{S^-} [A_{k-1}(S') + B_k(S) + \Gamma_k(S', S)] \end{aligned} \quad \dots(2.31)$$

BAB 3

PEMODELAN SISTEM DAN PERANCANGAN SIMULASI

3.1 Pemodelan Sistem OFDM Menggunakan Teknik *Clipping*

Blok diagram sistem ditunjukkan pada Gambar 3.1 yang menunjukkan skema transmisi OFDM menggunakan *clipping*.

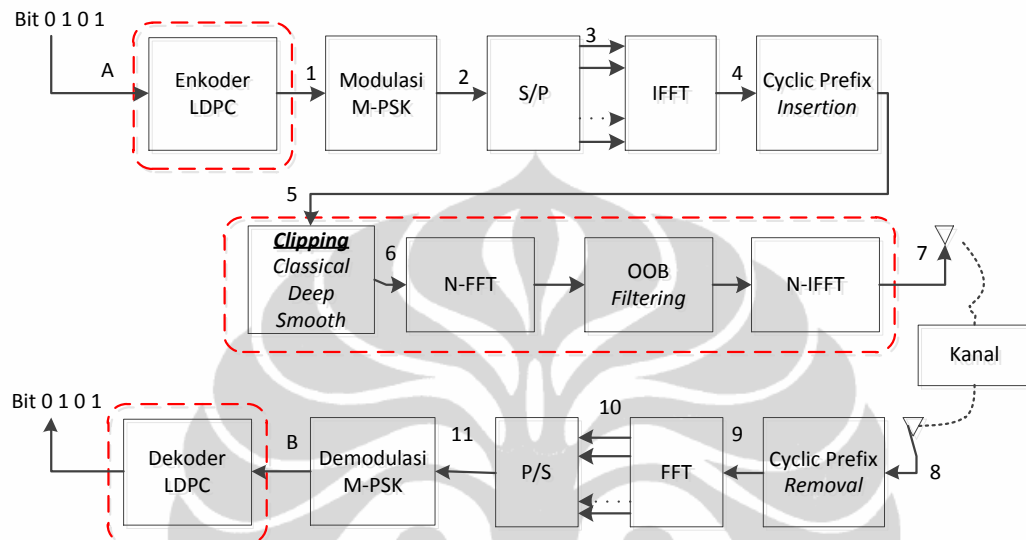


Gambar 3.1 Sistem OFDM Menggunakan *Clipping*.

Sistem pada Gambar 3.1 dijelaskan sebagai berikut. Langkah 1, bit informasi dipetakan menggunakan modulator QPSK menjadi simbol kompleks. Sekuen simbol dijadikan paralel pada langkah 2, kemudian dilakukan transformasi menggunakan IFFT pada langkah 3 yang menghasilkan simbol OFDM. Sinyal OFDM kemudian melalui proses penyisipan *Cyclic prefix* pada langkah 4. Sinyal OFDM dilakukan proses *Clipping* pada langkah 5 untuk reduksi PAPR dan dilakukan *filtering* untuk menurunkan OOB pada langkah 6. Teknik *clipping* yang digunakan adalah *Classical clipping*, *Deep clipping* dan *Smooth clipping*. Sinyal pada ranah waktu kemudian ditransmisikan melalui kanal pada langkah 7. Pada blok penerima, sinyal OFDM dihilangkan *cyclic prefix*-nya pada langkah 8 dan ditransformasi menggunakan FFT pada langkah 9. Sinyal OFDM pada ranah frekuensi yang telah dijadikan serial pada langkah 10 didemodulasi pada langkah

11. Skema pemodelan ini sebagai simulasi pembanding dengan simulasi OFDM menggunakan ECC dan variasi *clipping*.

3.2 Pemodelan Sistem LDPC-OFDM Menggunakan Teknik *Clipping*

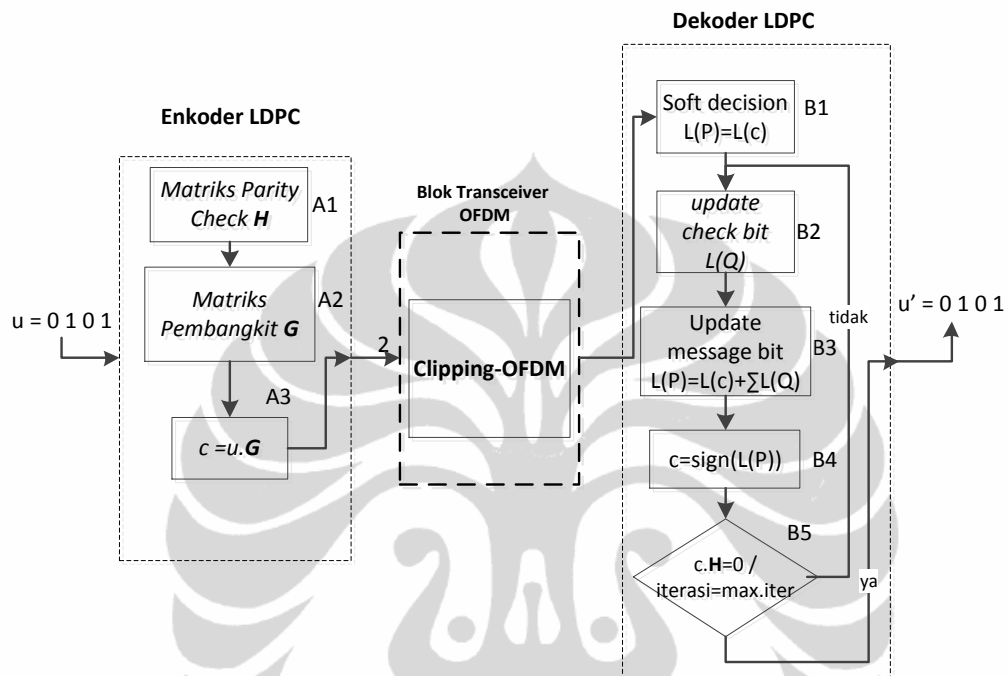


Gambar 3.2 Sistem LDPC-OFDM menggunakan *clipping*

Gambar 3.2 menjelaskan penambahan blok enkoder LDPC dan blok dekoder LDPC sebagai sistem OFDM yang menggunakan kode LDPC (LDPC-OFDM) dengan teknik *clipping*. Langkah A dan B sebagai tambahan dari sistem OFDM menggunakan *clipping* yang ditunjukkan Gambar 3.1. Langkah A adalah proses enkoding yang menghasilkan *codeword* sebagai input pada langkah 1. Langkah B adalah proses dekoding LDPC yang melakukan perbaikan error. Gambar 3.3 menunjukkan proses enkoding dan dekoding LDPC-OFDM. Proses enkoding LDPC dilakukan dengan pembangkitan matriks *parity check* $\mathbf{H}$ pada langkah A1 dengan konfigurasi yang telah ditentukan. Kemudian pada langkah A2 matriks $\mathbf{H}$ menjadi referensi matriks generator $\mathbf{G}$ untuk enkoder pada langkah A3 yang menggunakan input dari bit biner informasi yang akan dikodekan. Output dari enkoder LDPC adalah *codeword* dengan *code rate* yang telah ditentukan yang akan dilanjutkan ke proses modulasi. Penggunaan *code rate* adalah $1/2$ mengacu pada referensi [6].

Metode yang digunakan untuk membangkitkan matriks *parity check* $\mathbf{H}$

non sistematis mengacu dari referensi [8] yang disebut konstruksi Mackay 1A yang merupakan skema regular LDPC, matriks ini dinyatakan sebagai matriks $\mathbf{A}$ yang menjadi referensi dekoding. Matriks $\mathbf{H}$ didapatkan dengan melakukan eliminasi Gaussian dengan *column pivoting* seperlunya dengan aritmatika biner.



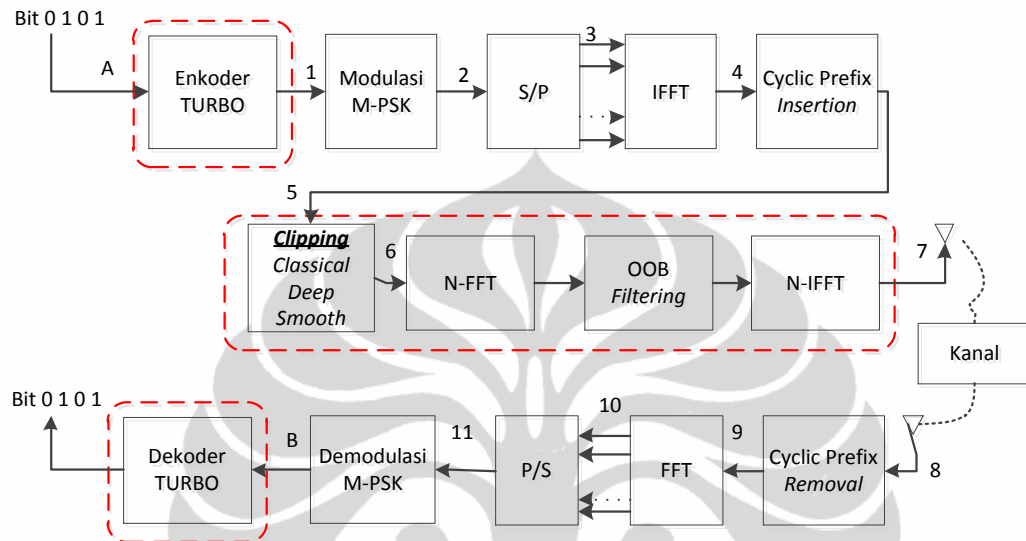
Gambar 3.3 Enkoding dan Dekoding LDPC

Proses dekoding LDPC dilakukan setelah proses FFT pada penerima. Langkah B1 adalah inisiasi nilai $L(P)$ dari kalkulasi *soft decision*. Langkah B2 menghitung $L(Q)$ sesuai dengan persamaan 2.16 dan B3 menghitung $L(P)$ dari persamaan 2.18. *Codeword* didapatkan pada langkah B4 dan keputusan tentatif pada langkah B5 untuk menghentikan iterasi jika $c \cdot \mathbf{H}^T = 0$ atau mencapai batas maksimum iterasi yang telah ditentukan, jika tidak kembali kelangkah B2 hingga syarat B5 terpenuhi. Skema dekoding yang digunakan pada simulasi ini adalah algoritma Log-Sum Product. Simulasi LDPC dikembangkan dari referensi [8].

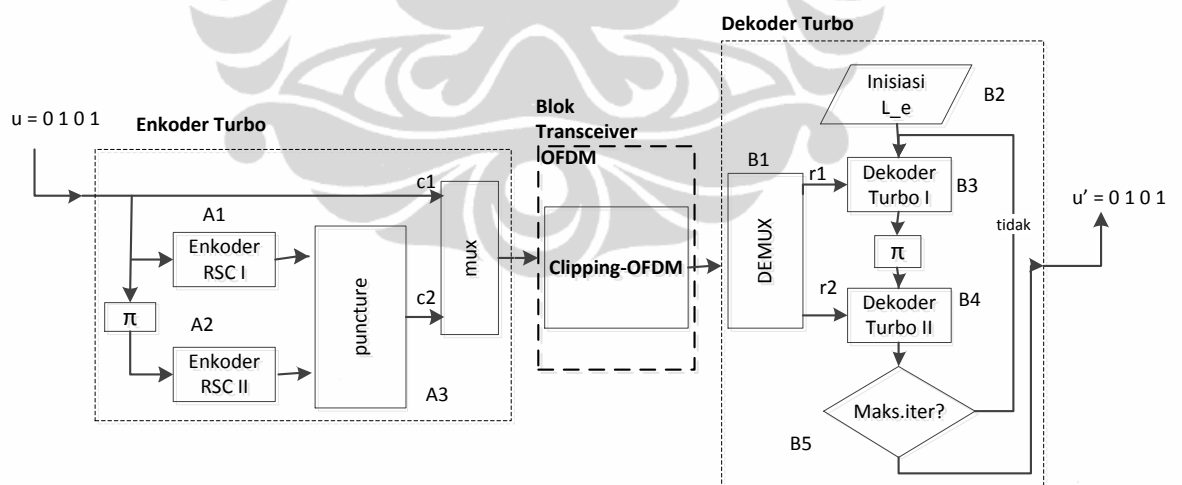
3.3 Pemodelan Sistem LDPC-OFDM Menggunakan Teknik *Clipping*

Gambar 3.4 menunjukkan sistem OFDM menggunakan kode Turbo (Turbo-OFDM) dengan teknik *clipping*. Langkah A adalah blok enkoder Turbo dan langkah B adalah dekoder Turbo yang ditambahkan seperti ditunjukkan pada

Gambar 3.4. Proses encoding Turbo menggunakan dua RSC encoder dan *random interleaver*. Skema encoding menggunakan matriks $G = [1\ 1\ 1; 1\ 0\ 1]$ yang mengacu dari referensi [20]. Simulasi kode Turbo dikembangkan dari referensi [9] untuk membangkitkan *code rate* $\frac{1}{2}$.



Gambar 3.4. Sistem Turbo-OFDM Menggunakan Clipping.



Gambar 3.5. Encoding dan Dekoding Turbo

Skema enkoder dan dekoder Turbo ditunjukkan pada Gambar 3.5. Pada langkah A1 bit informasi menjadi input enkoder RSC I dan langkah A2 bit informasi yang dilakukan *interleave* dan menjadi input enkoder RSC II, hasil encoding A1 dan

A2 dilakukan *puncture* pada A3 menjadi c_2 yang hasilnya digabungkan dengan c_1 . *Codeword* c_1 adalah bit informasi. *Codeword* digunakan sebagai input pada blok *transceiver* OFDM seperti yang sudah dijelaskan pada subbab 3.1. Pada dekoder Turbo, B1 membagi input dari blok *Clipping*-OFDM yang masing-masing untuk dekoder I(B3) dan dekoder II (B4). Langkah B2 untuk inisiasi informasi ekstrinsik LLR yang menjadi input B3. Output dari B3 adalah informasi ekstrinsik LLR yang setelah di *interleave* menjadi input B4. Informasi ekstrinsik LLR dari B4 diubah menjadi bit hasil dekode pada B5 dan menjadi input iterasi ke B3, proses dekoding selesai ketika telah mencapai maksimum iterasi yang telah ditentukan. Skema dekoder I dan dekoder II menggunakan Max-Log-MAP yang telah dijelaskan pada bab sebelumnya.

3.4 Fungsi *Clipping*

Fungsi dari *Classical Clipping*, *Deep clipping* dan *Smooth Clipping* adalah memotong sinyal sesuai dengan level *clipping* yang ditentukan sebagai ambang batas amplitudo dari sinyal. Proses *clipping* dijelaskan sebagai berikut:

1. *Clipping* menggunakan input sinyal OFDM setelah proses penyisipan *cyclic prefix* pada transmitter.
2. Nilai σ RMS sebagai normalisasi rasio *clipping* dikalkulasi menggunakan input sinyal OFDM.
3. Rasio *clipping* CR yang digunakan sebagai ambang batas proses *clipping* dikalkulasi dengan persamaan 3.3, dengan A adalah level *clipping* yang telah ditentukan.

$$CR = \frac{A}{\sigma} \quad \dots(3.3)$$

4. *Classical Clipping* menggunakan 2 syarat yaitu. Amplitudo dari sinyal OFDM akan diloloskan ketika amplitudo dari sinyal lebih kecil atau sama dengan CR . Ketika sinyal r pada kondisi lebih besar dari ambang batas CR , maka sinyal mengalami proses *clipping* dengan pemotongan amplitudo dengan level CR .
5. *Deep Clipping* menggunakan 3 syarat yaitu: Amplitudo dari sinyal OFDM akan diloloskan ketika amplitudo dari sinyal lebih kecil atau sama dengan

CR . Ketika sinyal r pada kondisi $CR < r \leq \frac{1+p}{p}CR$, maka sinyal mengalami proses *clipping* dengan persamaan $A - p * (r - A)$. Ketika $r > \frac{1+p}{p}CR$, maka amplitudo sinyal mengalami proses *clipping* dengan nilai amplitude '0'.

6. *Smooth Clipping* menggunakan 2 syarat yaitu: Ketika sinyal r pada kondisi $r \leq \frac{3}{2}A$, maka proses *clipping* yang dilakukan adalah $r - \frac{1}{b}r^3$, jika $r > \frac{3}{2}A$, maka dilakukan pemotongan amplitudo dengan batas level *clipping* yang telah ditentukan.

Teknik *clipping* yang diaplikasikan pada sistem menggunakan teknik *Clipping* dan *Filtering*. *Filtering* dilakukan setelah proses *clipping* untuk mengurangi efek radiasi *out-of band* yang disebabkan oleh *clipping*. Komponen *out-of band* dari sinyal *clipping* diubah menjadi nol pada ranah frekuensi [21]. Langkah-langkah *filtering* dijelaskan sebagai berikut [22]:

1. Setelah proses *clipping*, sinyal diproses FFT.
2. Komponen-komponen *in-band* dari sinyal yang telah dilakukan *clipping* $c_0 \dots c_{\frac{N}{2}-1}, c_{\frac{N}{2}L-\frac{N}{2}+1} \dots c_{NL-1}$ dilewatkan ke IFFT kedua dengan faktor *oversampling* L_2 .
3. Komponen-komponen *out-of-band* $c_{\frac{N}{2}+1} \dots c_{NL-\frac{N}{2}}$ diubah menjadi nol.
4. Kemudian sinyal ditransformasi IFFT.

3.5. INDIKATOR KINERJA

Sebagai pembatasan masalah dan analisa. Hasil simulasi yang akan dianalisa adalah hasil reduksi PAPR, kinerja BER, dan *Power Spectral Density* (PSD). Hasil simulasi akan dibandingkan dengan penggunaan teknik *Clipping* pada referensi acuan.

3.5.1 Kinerja PAPR

PAPR dapat dikalkulasi dengan persamaan 2.6 pada Bab 2. Umumnya,

PAPR dideskripsikan secara statistik dengan menggunakan *Complementary Cumulative Distribution Function* (CCDF) dimana $CCDF = 1 - CDF$. CDF merupakan kumulatif dari nilai PAPR. Setelah didapatkan nilai kumulatif CDF maka didapatkan nilai CCDF dimana hasilnya dibagi dengan jumlah simbol yang dikirimkan untuk mendapatkan probabilitas munculnya nilai PAPR lebih dari nilai PAPR tertentu. Hasil PAPR digambarkan dengan grafik CCDF terhadap probabilitas. CCDF digunakan untuk mengetahui probabilitas level PAPR yang melebihi ambang batas tertentu. PAPR dikalkulasi setelah proses *clipping* dan *filtering*.

3.5.2 Kinerja Probabilitas Error

BER dalam simulasi dapat dikalkulasi dengan cara membandingkan urutan bit informasi yang dikirim dan diterima, kesalahan dari bit informasi dibagi dengan jumlah bit informasi yang dibangkitkan atau bit yang dikirim.

3.5.3 Kinerja Power Spectral Density

PSD digunakan untuk melihat tingkat OOB yang diakibatkan dari teknik *clipping*. PSD dikalkulasi setelah proses *clipping* dan *filtering*. Grafik PSD dilakukan berdasarkan periodogram Welch yang dijelaskan sebagai berikut:

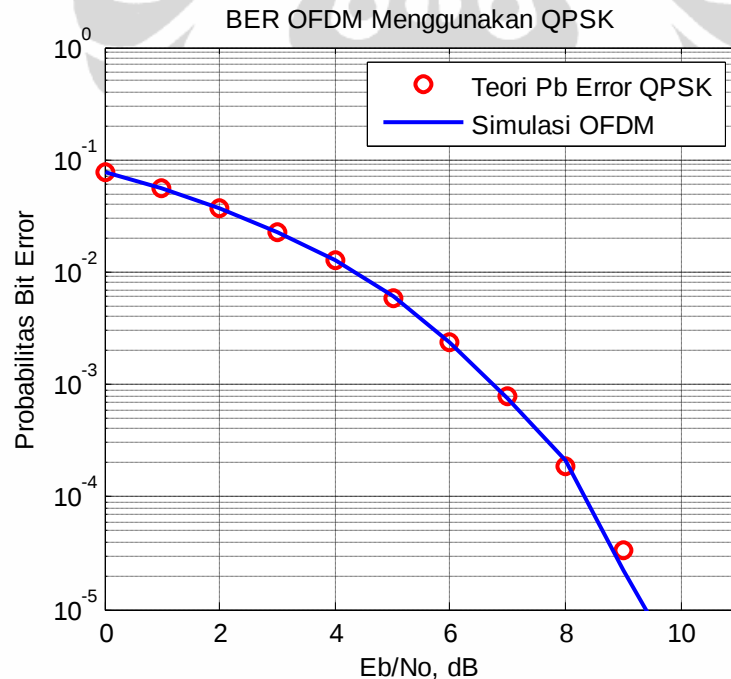
1. Vektor sinyal *input* x dibagi menjadi k segmen yang saling overlap berdasarkan nilai *window* dan *noverlap* (atau nilai yang telah ditetapkan) dengan $k = \frac{m-o}{l-o}$. Dengan m adalah panjang vektor x , o adalah banyaknya sampel yang *overlap* (*noverlap*) dan l adalah panjang setiap segmen (panjang *window*)
2. Hamming *window* diterapkan pada setiap segmen x
3. Setelah itu dilakukan proses IFFT pada data yang telah di *window*.
4. Periodogram dari setiap segmen yang telah di *window* dihitung.
5. Kemudian dirata-ratakan untuk mendapatkan spektrum estimasi $S(e^{j\omega})$
6. Estimasi spektrum tersebut diskalakan untuk menghitung $S(e^{j\omega})/F$, dengan F adalah frekuensi *sampling* fs .

BAB 4

HASIL DAN PEMBAHASAN

Simulasi yang dilakukan menggunakan parameter sebagai berikut :

- Modulasi QPSK (*Quadrature Phase Shift Keying*)
- Kanal AWGN (*Additive White Gaussian Noise*)
- Jumlah bit info 640.000
- Jumlah titik FFT / IFFT = 256
- Oversampling 4 x
- Panjang Cyclic Prefix = 25 % [23]
- Rasio *clipping* = 1.4 [24]
- Depth Factor = 0.6 [25]
- *Code rate* $\frac{1}{2}$ Regular LDPC (64,128)
- Kode Turbo menggunakan panjang *interleaver* 64 dengan *code rate* $\frac{1}{2}$.
- Dekoder LDPC menggunakan Algoritma Sum Product sedangkan Dekoder Turbo menggunakan algoritma BCJR Max Log MAP.

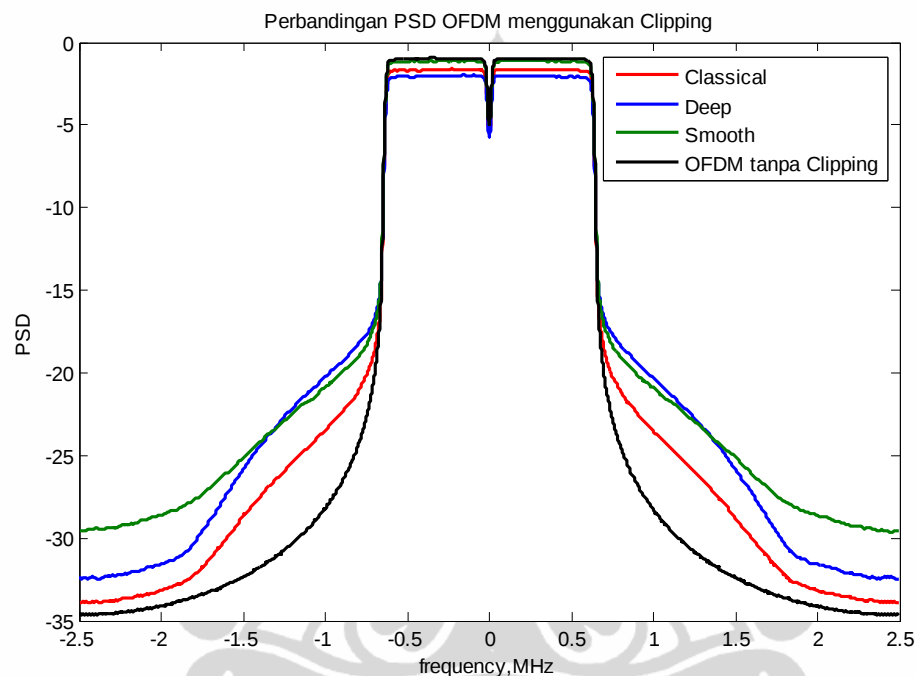


Gambar 4.1 Simulasi OFDM Menggunakan Modulasi QPSK.

Untuk validasi simulasi OFDM dilakukan dengan membandingkan hasil simulasi OFDM menggunakan modulasi QPSK dengan AWGN dengan teori

probabilitas error untuk QPSK menggunakan AWGN, acuan validasi ini digunakan karena tidak ada perbaikan atau rugi-rugi kinerja dengan menggunakan modulasi OFDM terhadap AWGN. Gambar 4.1 menunjukkan perbandingan hasil simulasi terhadap modulasi QPSK. Dari hasil simulasi menunjukkan BER dari simulasi sama dengan teori QPSK.

4.1 Analisa OFDM Menggunakan *Clipping*

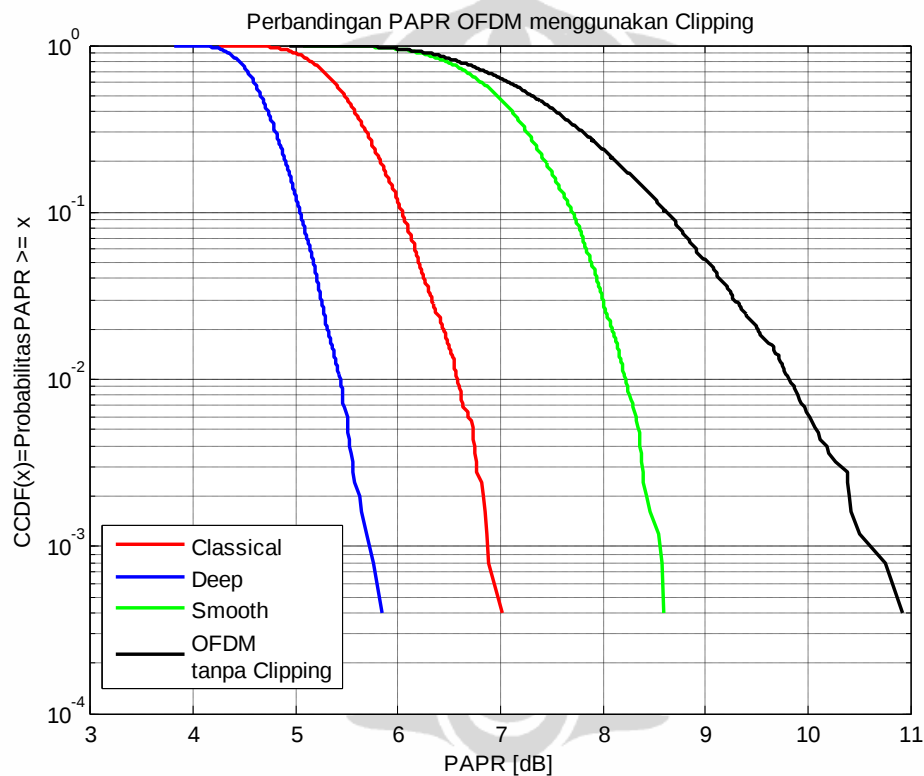


Gambar 4.2 Power Spectral Density Pada OFDM Terhadap Variasi *Clipping*

Observasi efek *clipping* pada radiasi riak (*ripple*) dari *in-band* dan radiasi OOB dapat dilakukan menggunakan kinerja PSD. Efek dari variasi *Clipping* ditunjukkan dalam kinerja PSD pada Gambar 4.2. Penentuan kinerja algoritma *Deep Clipping* untuk faktor kedalaman (*Depth Factor*) diacu dari referensi dimana nilai $p = 0.6$. Gambar 4.2 menunjukkan perbandingan PSD dari teknik *clipping* untuk OFDM konvensional (OFDM yang tidak menggunakan koreksi error).

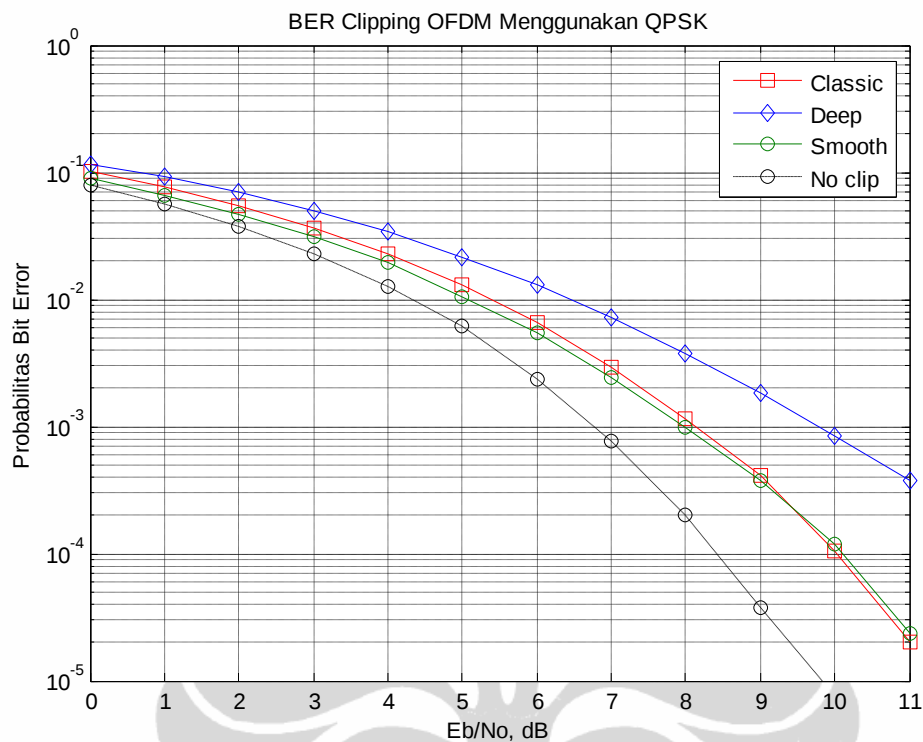
Dari hasil simulasi dapat diketahui efek dari *Classical clipping* memberi radiasi atau peningkatan *out-of band* sekitar 0.74 dB, dan *Deep clipping* serta *Smooth clipping* juga memberi peningkatan OOB sebesar 2.2 dB dan 5.07 dB.

Dapat disimpulkan *Smooth clipping* memberi peningkatan radiasi OOB paling tinggi dan *Classical clipping* memberi peningkatan OOB paling rendah. Sehingga hasil PSD juga membuktikan secara teori dampak dari teknik reduksi distorsi sinyal dengan teknik *clipping* mempengaruhi OOB. OOB yang tinggi berarti dapat menyebabkan interferensi antar kanal (ICI). Pengaruh *clipping* pada peningkatan radiasi OOB dapat dikurangi dengan menggunakan *filtering*. Karena *filtering* dapat menurunkan OOB dengan cara menghilangkan frekuensi tinggi.



Gambar 4.3. Kinerja PAPR Sistem OFDM Menggunakan Variasi *Clipping*.

Kinerja PAPR ditunjukkan pada Gambar 4.3 untuk sistem OFDM menggunakan variasi *clipping*. Reduksi PAPR paling rendah dihasilkan menggunakan teknik *Deep Clipping* dengan 4,4 dB pada probabilitas 10^{-2} terhadap PAPR OFDM. *Classical clipping* menghasilkan penurunan PAPR 3,3 dB dan *Smooth clipping* pada 1,6 dB. Dari segi kinerja PAPR, *Deep clipping* memberi penurunan BER yang paling baik diantara teknik *clipping* lainnya.



Gambar 4.4. BER Sistem OFDM Menggunakan Variasi *Clipping*.

Kinerja BER sistem OFDM menggunakan variasi *clipping* ditunjukkan Gambar 4.4. Pada probabilitas error 10^{-3} , *Smooth clipping*, *classical clipping* dan *deep clipping* berturut-turut memberi penurunan kinerja BER hingga 1,1 dB, 1,3 dB dan 3 dB. Dari hasil simulasi dapat disimpulkan penurunan kinerja BER terbesar adalah menggunakan *Deep clipping*, sedangkan *classical* dan *smooth clipping* memberi kinerja BER yang hampir sama.

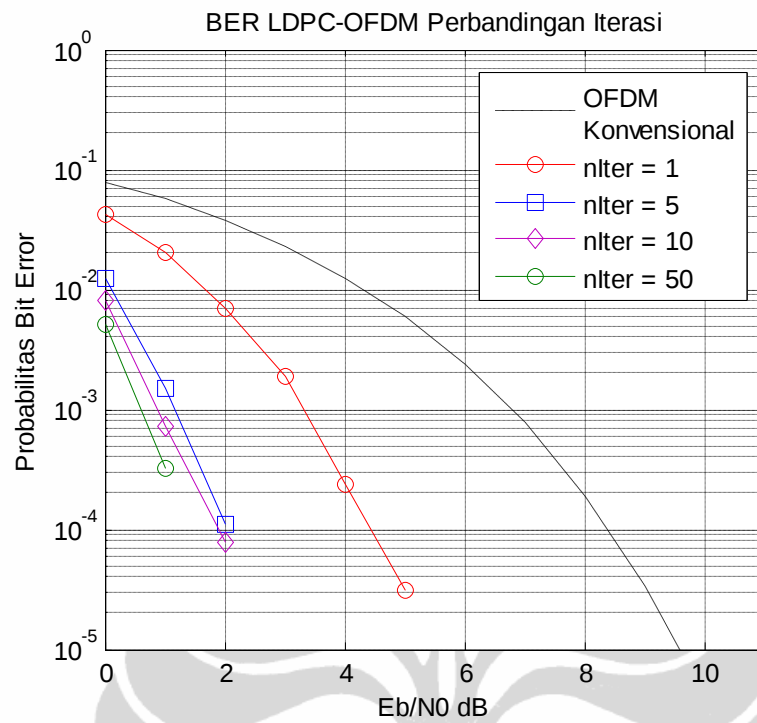
Penjelasan hasil kinerja dari variasi teknik *clipping* dijelaskan sebagai berikut. Mengacu dari fungsi alih variasi teknik *clipping* yang ditunjukkan Gambar 2.13. *Smooth clipping* dan *deep clipping* memberi banyak pemotongan amplitudo sinyal dibandingkan *classical clipping*, sehingga memberi radiasi OOB yang lebih tinggi dibandingkan dengan *classical*. Level radiasi OOB dari *smooth clipping* tinggi karena dari algoritmanya, seluruh sinyal mengalami pemotongan sesuai dengan ketentuannya dan hal ini memberi radiasi OOB yang lebih tinggi.

Walaupun terjadi pemotongan sinyal dengan level *clipping* yang sama, nilai rata-rata dari sinyal berubah sesuai dengan algoritma masing-masing *clipping*. Sehingga dapat dikatakan *Deep clipping* meningkatkan nilai rata-rata

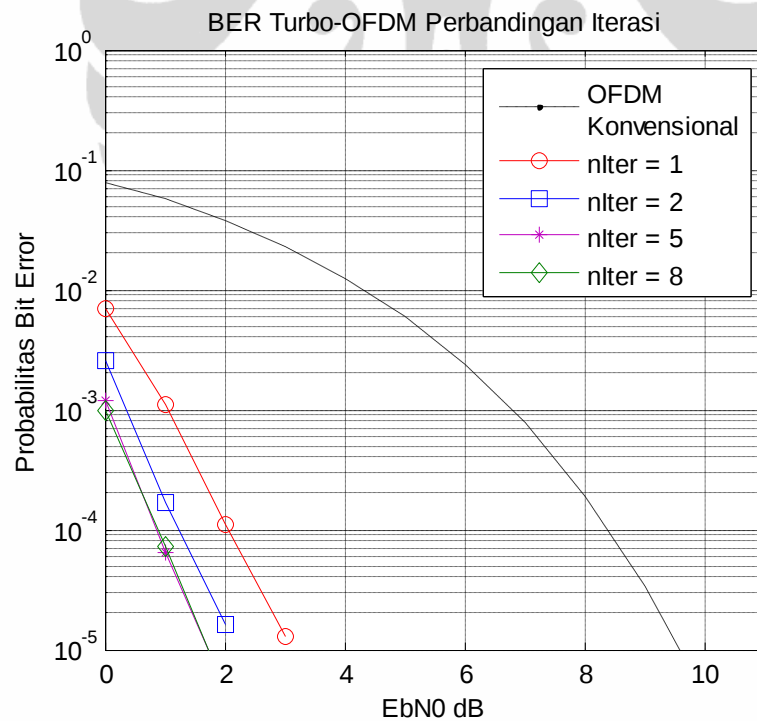
dari daya sinyal sehingga menghasilkan reduksi PAPR yang paling rendah. Hal ini juga menghasilkan distorsi *in-band* yang mempengaruhi penurunan kinerja BER. *Smooth clipping* melakukan pemotongan pada seluruh sinyal sehingga mempengaruhi nilai rata-rata dari kalkulasi PAPR dan walaupun terjadi pemotongan daya puncak, nilai rata-rata untuk kalkulasi PAPR juga menurun sehingga hasil kalkulasi PAPR menghasilkan reduksi PAPR yang paling sedikit dibandingkan teknik *clipping* lainnya. *Classical clipping* menghasilkan kinerja yang paling baik dibandingkan teknik *clipping* lainnya mengacu dari hasil kinerja PAPR dan BER. *Deep clipping* memberi hasil reduksi PAPR yang paling baik namun memberi dampak hasil BER yang paling buruk.

4.2 Analisa Kinerja LDPC-OFDM dan Turbo-OFDM

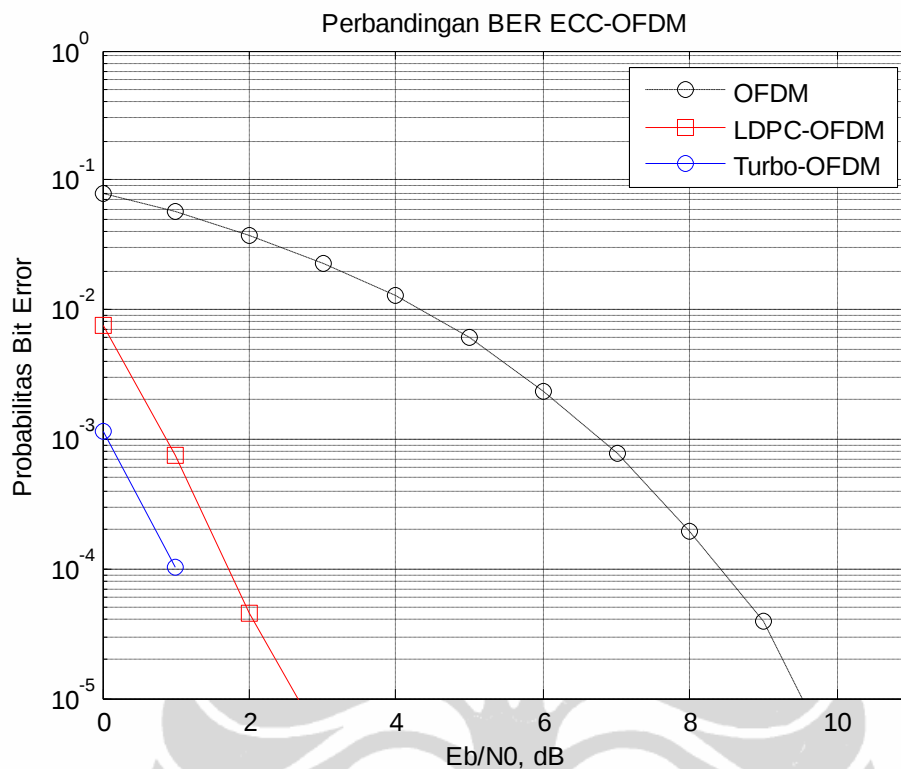
Perbandingan iterasi dekoding untuk Turbo-OFDM dan LDPC OFDM dilakukan untuk mendapatkan nilai iterasi optimal untuk digunakan dalam simulasi. Gambar 4.5 dan Gambar 4.6 berturut-turut menunjukkan perbedaan kinerja BER sesuai dengan banyak iterasi yang digunakan pada LDPC dan Turbo. Banyak iterasi optimal untuk LDPC adalah 10 iterasi sedangkan Turbo adalah 5 iterasi. Dapat disimpulkan dari perbandingan iterasi dari LDPC dan Turbo, dimana Turbo menghasilkan hasil koreksi error yang lebih baik dengan iterasi yang lebih sedikit dibandingkan kode LDPC. Hal ini memberi pengaruh terhadap *latency* dalam proses dekoding.



Gambar 4.5. Perbandingan Iterasi Dekoding LDPC-OFDM



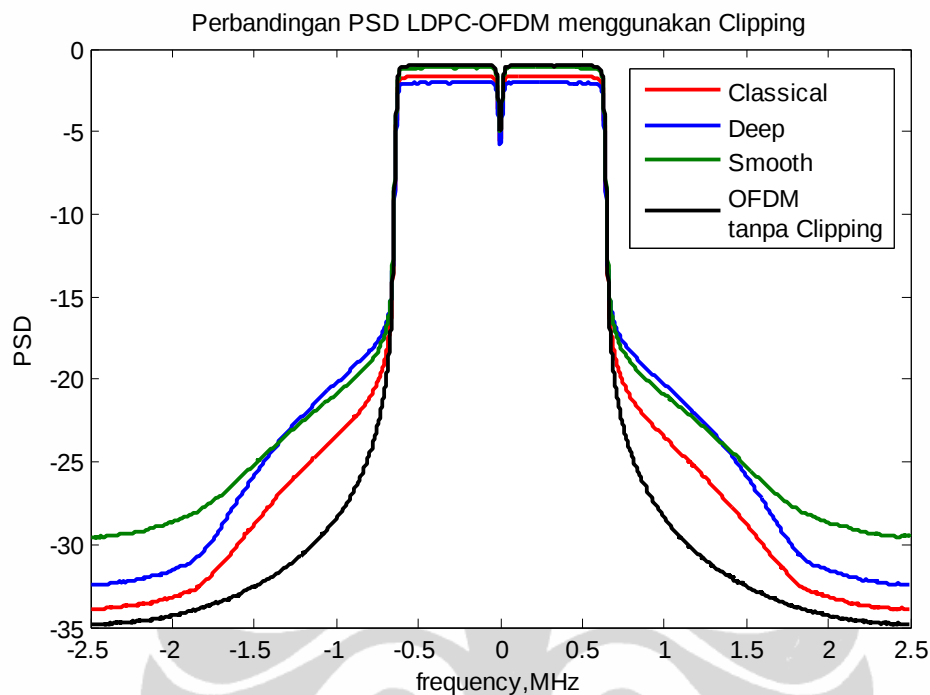
Gambar 4.6. Perbandingan Iterasi Dekoding Turbo-OFDM



Gambar. 4.7. Perbandingan BER OFDM Konvensional, Turbo-OFDM dan LDPC-OFDM

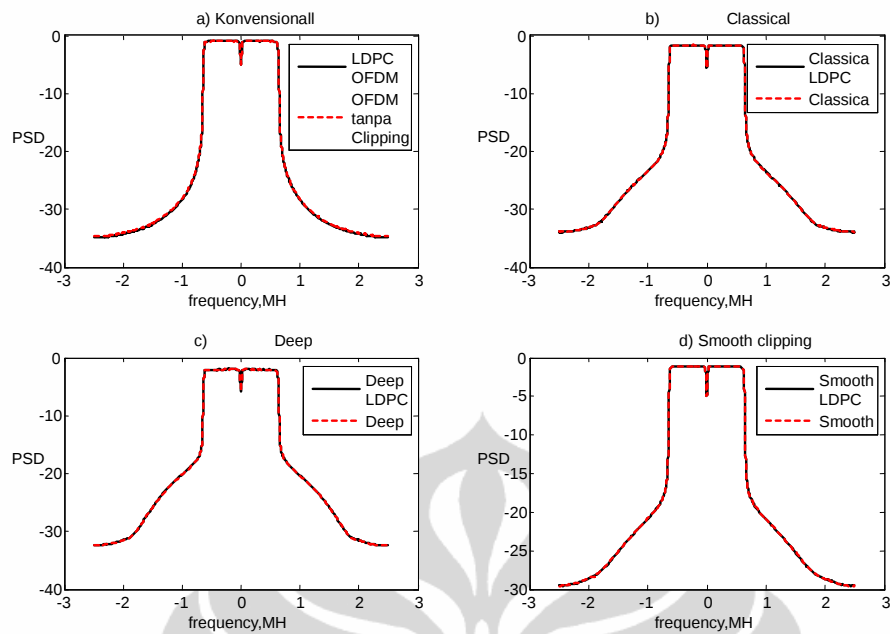
Kinerja BER untuk OFDM menggunakan ECC ditunjukkan pada Gambar 4.7. Turbo-OFDM dengan konfigurasi spesifik dan iterasi 5 kali memberi kinerja BER yang lebih baik dibandingkan LDPC-OFDM dengan konfigurasi spesifik dan iterasi 10 kali. Perbandingan kode LDPC dan Turbo dilakukan dengan parameter *code rate* $\frac{1}{2}$ dan dengan input bit info 64. Spesifikasi LDPC adalah menggunakan kode Regular LDPC dengan *weight* pada kolom matriks parity check $\mathbf{H}$ adalah 3 dan *weight* pada baris kolom matriks parity check adalah 6. Spesifikasi kode Turbo yang digunakan adalah penggunaan matriks polinomial $\mathbf{G}$ $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$ dan skema *interleaver random* dengan panjang *interleaver* adalah 64. Untuk spesifikasi yang digunakan untuk simulasi pada tesis ini, kode Turbo memberi hasil kinerja yang lebih baik dibandingkan dengan kode LDPC untuk OFDM. Turbo OFDM memberi peningkatan pada probabilitas error 10^{-3} mencapai sekitar 6.7 dB sedangkan LDPC-OFDM pada 5.8 dB dibandingkan dengan OFDM konvensional.

4.3 Analisa Kinerja LDPC-OFDM Menggunakan *Clipping*

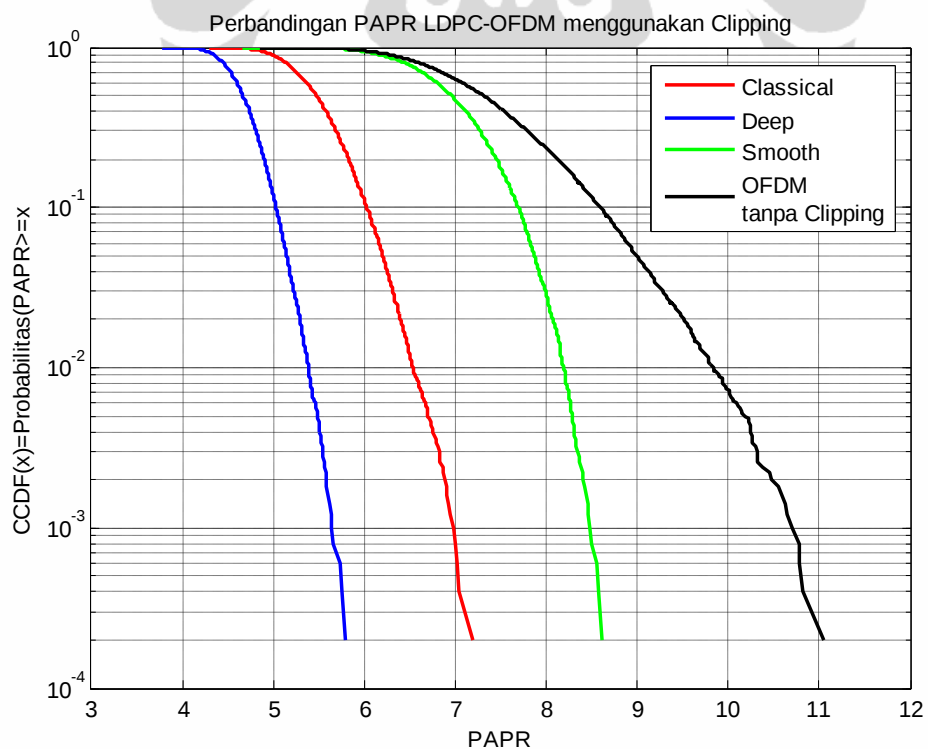


Gambar 4.8. PSD Sistem LDPC-OFDM Menggunakan Variasi *Clipping*.

PSD dari sistem LDPC-OFDM menggunakan *clipping* ditunjukkan pada Gambar 4.8. peningkatan radiasi OOB *Classical clipping*, *Deep clipping* dan *Smooth clipping* berturut-turut sekitar 0.74 dB, 2.2 dB dan 5.07 dB, yang merupakan hasil PSD yang sama dengan OFDM konvensional. Hasil kinerja PSD LDPC-OFDM dan OFDM yang menggunakan variasi *clipping* adalah sama, karena penggunaan kode-LDPC berada pada input bit informasi yang akan dikirimkan, dan tidak melakukan distorsi sinyal seperti teknik *clipping*. Gambar 4.9 menunjukkan tidak ada perubahan dari kinerja PSD penggunaan kode LDPC untuk semua variasi *clipping* pada sistem OFDM dibandingkan dengan OFDM konvensional dengan variasi *clipping*.

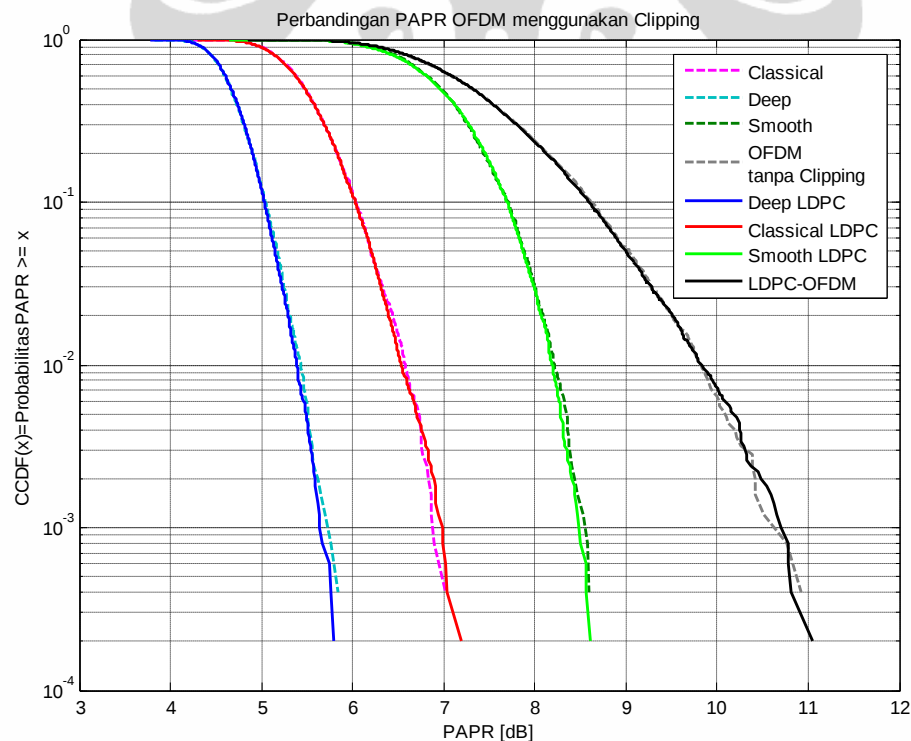


Gambar 4.9 Perbandingan PSD LDPC-OFDM dengan OFDM Menggunakan Variasi *Clipping*. a) Konvensional, b).Classical, c) Deep, d) Smooth.

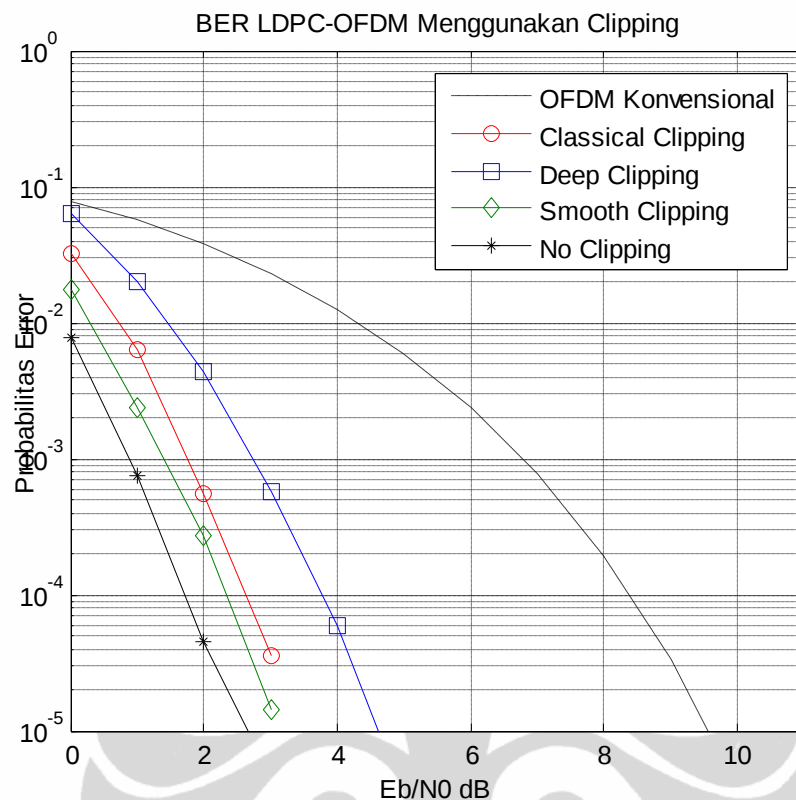


Gambar 4.10 CCDF PAPR LDPC-OFDM Menggunakan Variasi *Clipping*.

Hasil PAPR dalam CCDF ditunjukkan pada Gambar 4.10 Untuk LDPC-OFDM menggunakan variasi *clipping*. Kinerja reduksi PAPR dari *Classical*, *Deep* dan *Smooth clipping* pada LDPC-OFDM pada probabilitas 10^{-2} berturut-turut adalah 3,3 dB, 4,5 dB dan 1,6 dB terhadap LDPC-OFDM tanpa *clipping*. Kinerja PAPR LDPC-OFDM dengan OFDM konvensional menggunakan variasi *clipping* menghasilkan reduksi PAPR yang sama. Tidak terdapat perubahan kinerja PAPR menggunakan kode LDPC karena diasumsikan tidak terjadi skema encoding yang spesifik mengatur pencegahan superposisi koheren dari sinyal *multicarrier*, seperti pada teknik *Huffman coding* [4]. Asumsi dilakukan karena tidak ada spesifikasi atau informasi yang menjelaskan reduksi simbol pada kode LDPC dari referensi. Perbandingan kinerja PAPR LDPC-OFDM dengan OFDM konvensional menggunakan *clipping* ditunjukkan pada Gambar 4.11 Dimana penggunaan kode PAPR tidak memberi peningkatan atau penurunan dari PAPR OFDM konvensional.

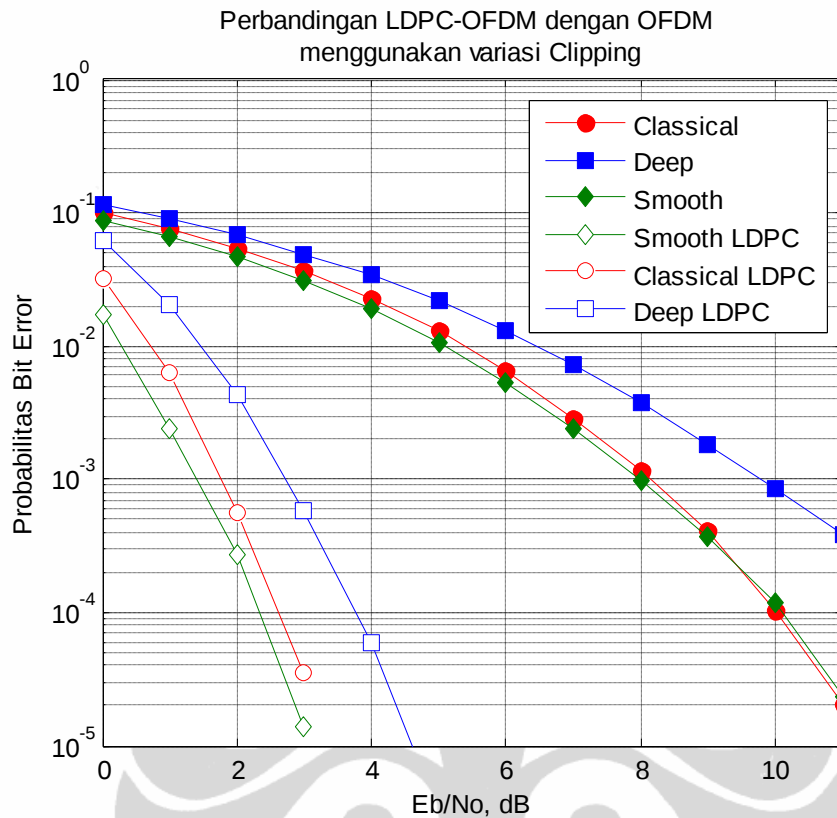


Gambar 4.11 Perbandingan PAPR LDPC-OFDM Menggunakan Variasi *Clipping* Terhadap OFDM Konvensional.



Gambar 4.12 BER LDPC-OFDM Menggunakan Variasi *Clipping*.

Kinerja BER untuk LDPC-OFDM menggunakan variasi *clipping* ditunjukkan pada Gambar 4.12. Pada probabilitas 10^{-3} , *Classical clipping* dengan kode LDPC memberi penurunan BER hampir 1 dB terhadap LDPC-OFDM dan peningkatan BER hingga 5 dB pada OFDM konvensional. *Deep Clipping* memberi penurunan hingga 1,8 dB terhadap LDPC-OFDM dan peningkatan BER 4 dB terhadap OFDM konvensional. *Smooth clipping* memberi penurunan hingga 0,5 dB pada LDPC-OFDM dan peningkatan BER hingga 5,4 dB. Peningkatan BER pada masing-masing teknik *clipping* terhadap OFDM konvensional karena kemampuan kode LDPC untuk mengoreksi error yang terjadi karena penggunaan *clipping*.

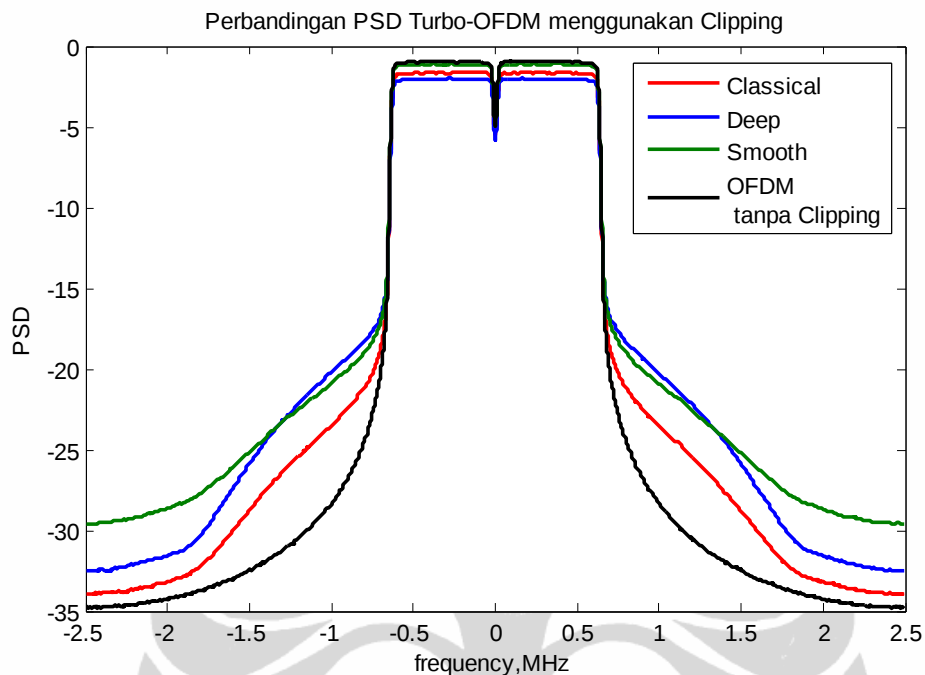


Gambar 4.13 Perbandingan Variasi *Clipping* OFDM Konvensional dengan LDPC-OFDM

Gambar 4.13 Menunjukkan perbandingan variasi *clipping* pada OFDM konvensional dengan LDPC-OFDM. Untuk probabilitas 10^{-3} dengan perbandingan OFDM konvensional, pada *classical clipping*, kode LDPC menghasilkan peningkatan kinerja BER hingga 6,4 dB. Pada *smooth clipping*, kode LDPC menghasilkan peningkatan kinerja BER hingga 6,5 dB, Pada *deep clipping*, kode LDPC memberi peningkatan kinerja BER hingga 7 dB.

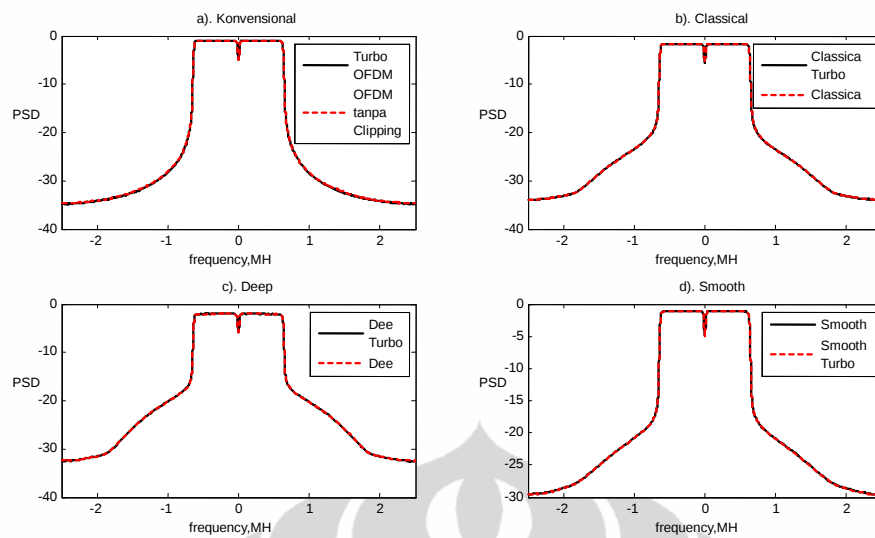
Dari hasil simulasi, kode LDPC dapat mengoreksi error yang dihasilkan seluruh teknik *clipping*. Walaupun terjadi distorsi *in-band* yang mempengaruhi kinerja BER dari *clipping*, kode LDPC mampu mengoreksi error karena *clipping*.

4.4 Analisa Kinerja Turbo-OFDM Menggunakan *Clipping*

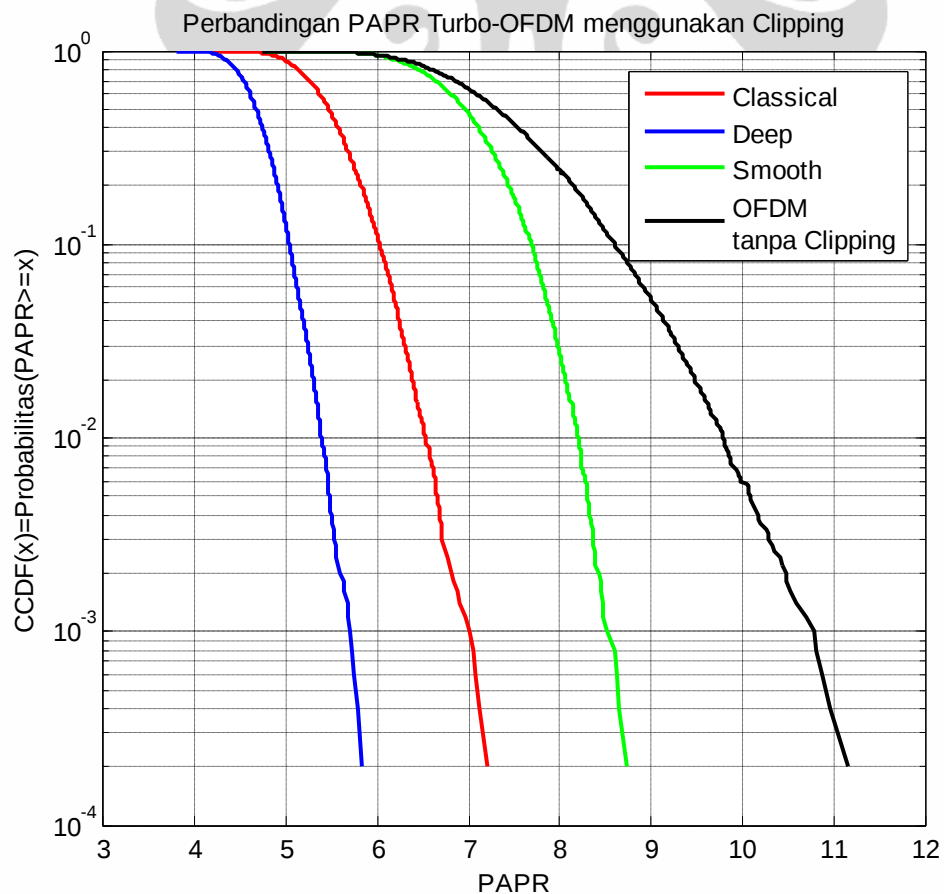


Gambar 4.14. PSD Sistem Turbo-OFDM Menggunakan *Clipping*.

PSD dari sistem Turbo-OFDM menggunakan *clipping* ditunjukkan pada Gambar 4.14. peningkatan radiasi OOB *Classical clipping*, *Deep clipping* dan *Smooth clipping* berturut-turut sekitar 0.8 dB, 2.2 dB dan 5.15 dB, yang merupakan hasil PSD yang sama dengan OFDM konvensional. Serupa dengan kinerja OFDM konvensional dan LDPC-OFDM, Turbo-OFDM juga tidak melakukan distorsi sinyal seperti teknik *clipping* dan proses encoding Turbo berada pada input bit untuk dikirimkan. Gambar 4.15 menunjukkan perbandingan masing-masing teknik *clipping* dalam kinerja PSD untuk OFDM konvensional dan Turbo-OFDM.

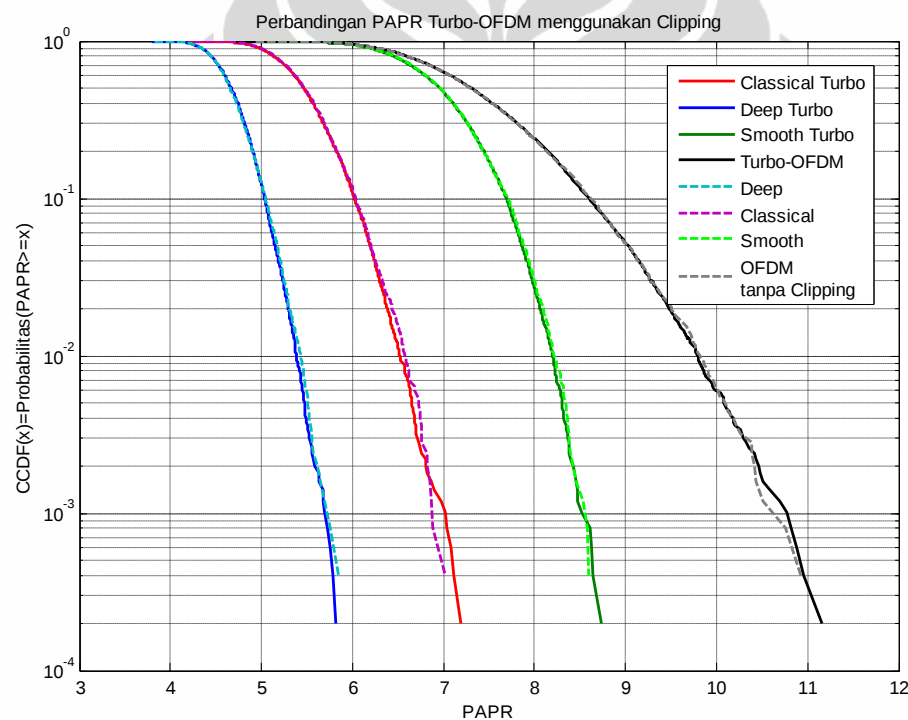


Gambar 4.15 Perbandingan PSD Turbo-OFDM dengan OFDM Menggunakan Variasi *Clipping*. a) Konvensional, b). *Classical*, c) *Deep*, d) *Smooth*.

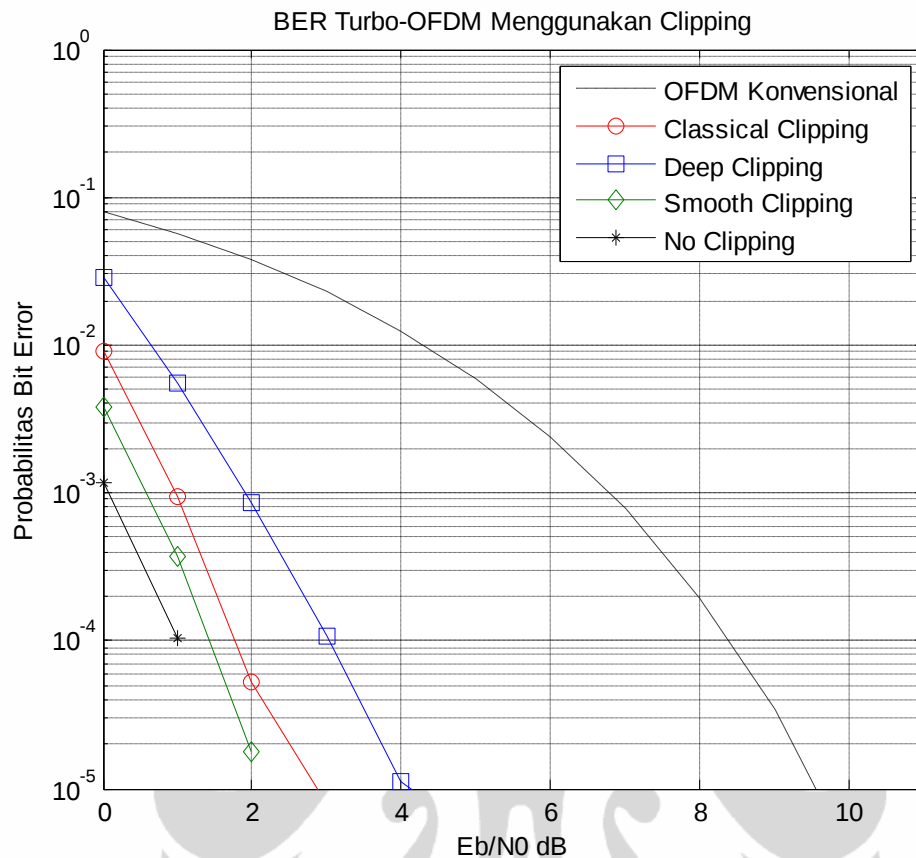


Gambar 4.16 CCDF PAPR Turbo-OFDM Menggunakan Variasi *Clipping*.

Hasil PAPR dalam CCDF ditunjukkan pada Gambar 4.16 Untuk Turbo-OFDM menggunakan variasi *clipping*. Kinerja reduksi PAPR dari *Classical*, *Deep* dan *Smooth clipping* pada Turbo-OFDM pada probabilitas 10^{-2} berturut-turut adalah 3,2 dB, 4,4 dB dan 1,6 dB terhadap Turbo-OFDM tanpa *clipping*. Hasil ini serupa dengan LDPC-OFDM dan OFDM konvensional, sehingga dapat disimpulkan untuk penggunaan kode Turbo untuk OFDM tidak memberi peningkatan atau penurunan kinerja dari PAPR pada variasi *clipping*. Perbandingan kinerja PAPR Turbo-OFDM dengan OFDM konvensional menggunakan *clipping* ditunjukkan pada Gambar 4.17

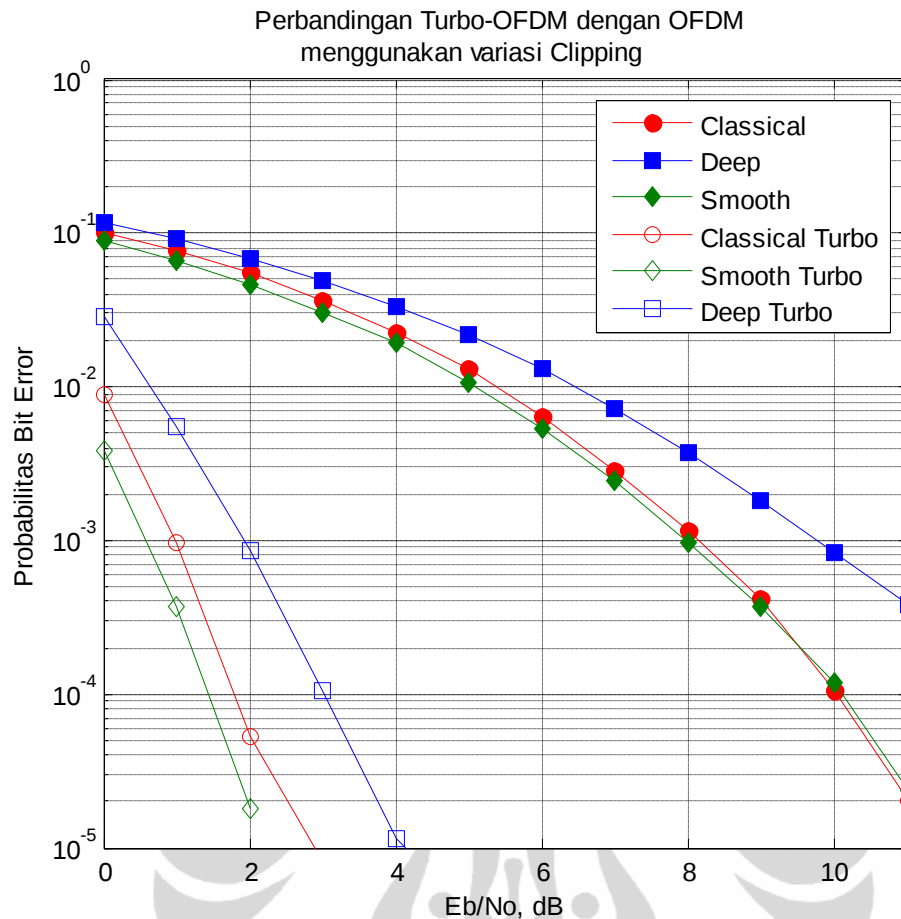


Gambar 4.17 Perbandingan PAPR Turbo-OFDM Menggunakan Variasi *Clipping* Terhadap OFDM Konvensional.



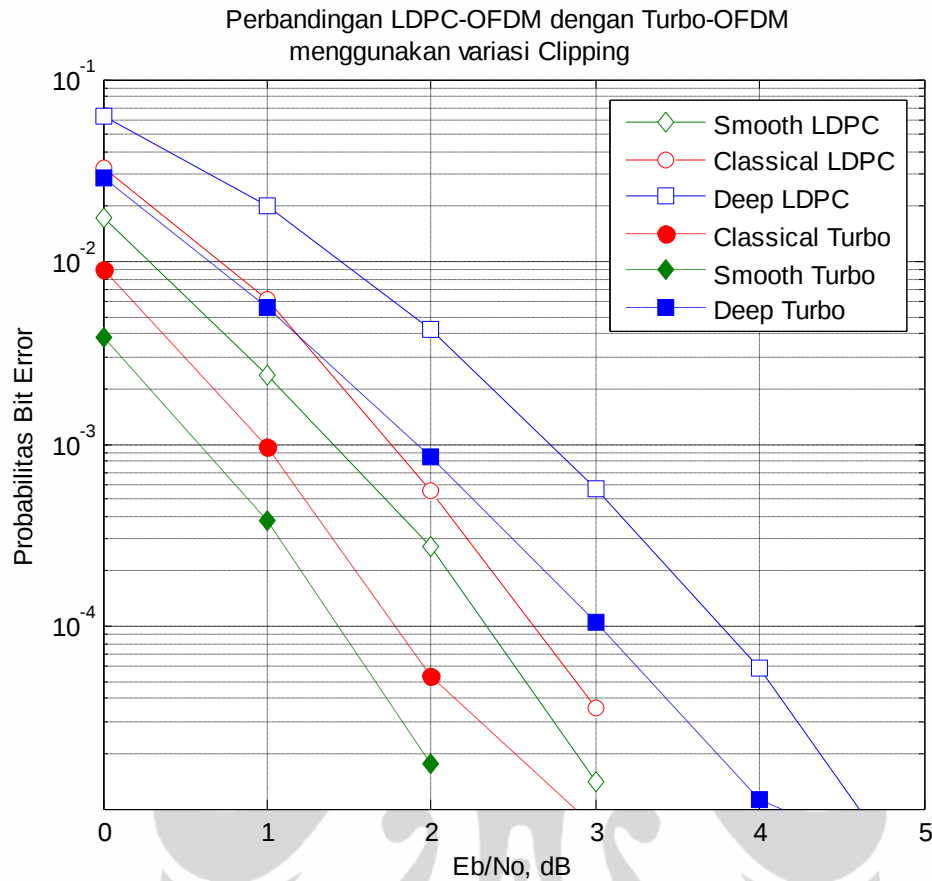
Gambar 4.18 BER Turbo-OFDM Menggunakan Variasi Clipping.

Kinerja BER untuk Turbo-OFDM menggunakan variasi *clipping* ditunjukkan pada Gambar 4.18. Pada probabilitas 10^{-3} , *Classical clipping* dengan kode Turbo memberi penurunan kinerja BER hampir 1 dB terhadap Turbo-OFDM dan peningkatan kinerja BER hingga 5,7 dB pada OFDM konvensional. *Deep Clipping* memberi penurunan kinerja hingga 1,8 dB terhadap Turbo-OFDM dan peningkatan BER 4,8 dB terhadap OFDM konvensional. *Smooth clipping* memberi penurunan kinerja hingga 0,5 dB pada Turbo -OFDM dan peningkatan kinerja BER hingga 6,2 dB. Sama dengan kode LDPC, kode Turbo mampu melakukan koreksi error terhadap efek dari distorsi sinyal dari *clipping*.



Gambar 4.19. Perbandingan Variasi *Clipping* OFDM Konvensional dengan Turbo-OFDM.

Gambar 4.19 Menunjukkan perbandingan variasi *clipping* pada OFDM konvensional dengan Turbo-OFDM. Untuk probabilitas 10^{-3} dengan perbandingan OFDM konvensional, pada *classical clipping*, kode Turbo menghasilkan peningkatan kinerja BER hingga 7,5 dB. Pada *smooth clipping*, kode Turbo menghasilkan peningkatan kinerja BER hingga 6,6 dB, Pada *deep clipping*, kode Turbo memberi peningkatan kinerja BER hingga 7,8 dB. Perbandingan penggunaan kode-Turbo pada OFDM menggunakan *clipping* dengan OFDM konvensional menghasilkan kesimpulan Turbo mampu meningkatkan kinerja BER terhadap OFDM konvensional menggunakan *clipping*. Dampak dari distorsi sinyal yang mempengaruhi *in-band* adalah penurunan kinerja BER. Kode Turbo dapat memperbaiki dan meningkatkan kinerja BER akibat dari *clipping*.



Gambar 4.20 Perbandingan Kinerja BER LDPC-OFDM dan Turbo-OFDM Menggunakan Variasi Clipping.

Gambar 4.20 Menunjukkan perbandingan kinerja BER LDPC-OFDM dan Turbo-OFDM menggunakan *clipping*. Pada probabilitas 10^{-3} , Turbo-OFDM menghasilkan peningkatan BER 0,7 dB untuk *Classical clipping*, 0,8 dB untuk *Deep clipping* dan *Smooth clipping* dibandingkan dengan LDPC-OFDM. Seperti hasil BER tanpa *clipping* untuk LDPC-OFDM dan Turbo-OFDM yang ditunjukkan Gambar 4.7, kode hasil koreksi error yang paling baik adalah menggunakan kode Turbo.

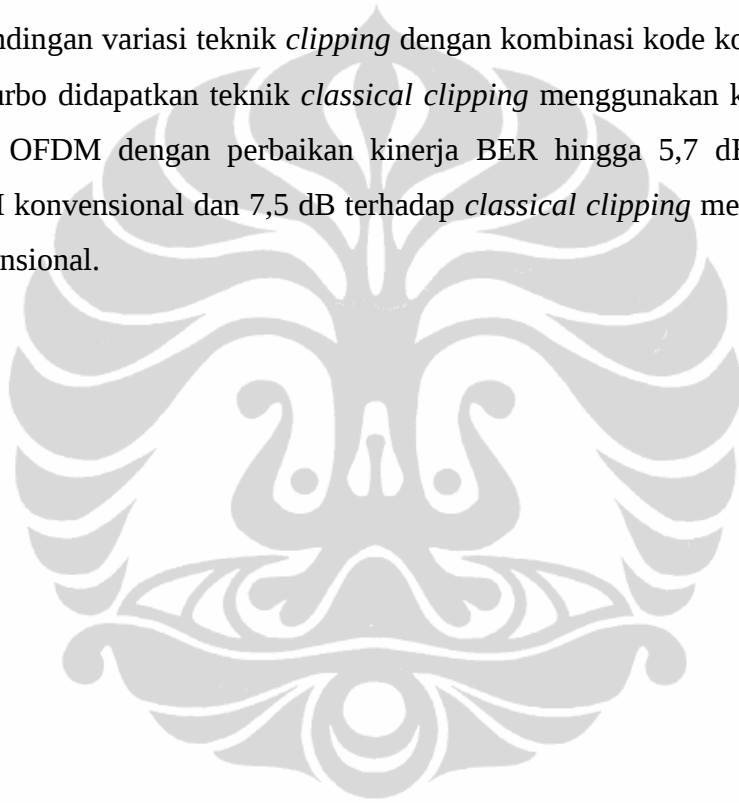
Secara keseluruhan hasil simulasi, hasil reduksi PAPR yang paling baik adalah menggunakan teknik *classical clipping*, karena untuk sistem OFDM dan kode-OFDM (Turbo dan LDPC), penurunan BER yang harus dikompensasi tidak terlalu besar dibandingkan *Deep clipping* dengan hasil reduksi PAPR yang lebih baik.

BAB 5

KESIMPULAN

Dari perbandingan variasi teknik *clipping* pada sistem OFDM yang disimulasikan, didapatkan teknik *classical clipping* sebagai teknik reduksi PAPR terbaik dengan tingkat radiasi OOB dan *in-band* yang rendah dan hasil reduksi PAPR hingga 3,3 dB dengan degradasi BER 1,3 dB terhadap sistem OFDM konvensional.

Perbandingan variasi teknik *clipping* dengan kombinasi kode koreksi error LDPC dan Turbo didapatkan teknik *classical clipping* menggunakan kode Turbo untuk sistem OFDM dengan perbaikan kinerja BER hingga 5,7 dB terhadap sistem OFDM konvensional dan 7,5 dB terhadap *classical clipping* menggunakan OFDM konvensional.



DAFTAR REFERENSI

- [1] Man-On Pun., Morelli, M., & Kuo., C-C.J (2007). *Multi-Carrier Techniques for Broadband Wireless Communications: A Signal Processing Perspective*. UK: Imperial College Press.
- [2] Molisch, Andreas F (2011). *Wireless Communications 2nd Edition*. UK: John Wiley & Sons.
- [3] Hui Liu & Guoqing Li. (2005). *OFDM-Based Broadband Wireless Networks : Design and Optimization*. Canada: John Wiley & Sons.
- [4] Juwono, Filbert Hilman.(2009). *Reduksi PAPR menggunakan Huffman Coding yang dikombinasikan dengan Clipping dan Filtering untuk Transmitter OFDM*. Indonesia: Teknik Elektro Universitas Indonesia.
- [5] Langlais. C., Haddad. S., Louet. Y., & Mazouz. N.(2011). Clipping noise mitigation with capacity approaching FEC codes for PAPR Reduction of OFDM signals. *IEEE Conference MC-SS 8th International Workshop*
- [6] Soriano. R.D., & Marciano. J.S.(2006). The Effect of Signal Distortion Techniques for PAPR Reduction on the BER Performance of LDPC and Turbo Coded OFDM System. *TENCON 2006 IEEE Region 10 Conference*.
- [7] Guel.D. & Palicot.J. (2009) Analysis and comparison of Clipping techniques for OFDM Peak-to-Average Power Ratio Reduction. *Digital Signal Processing 16th International Conference*.
- [8] Moon. Tod.K. (2005). *Error Correction Coding: Mathematical Methods and Algorithms*. Canada: John Wiley & Sons.
- [9] Wu Yu Fei.(2000). *Implementation of Paralel and Serial Concatenated Convolutional Codes*. Faculty of the Virginia Polytechnic Institute. Virginia.
- [10] Pillai,K.S.M.(2008). BER for BPSK in OFDM with Rayleigh multipath channel. August 26,2008. www.dspplog.com
- [11] Yong Soo Cho., Jeakwon Kim., Won Young Yang & Chung-Gu Kang.(2010). *MIMO-OFDM Wireless Communication with MATLAB*. Singapore: John Wiley & Sons.
- [12] Yang. Samuel C.(2010). *OFDM System Analysis and Design*. London: Artech

House.

- [13] Shinsuke Hara & Prasad. Ramjee.(2003). *Multi-Carrier Techniques for 4G Mobile Communications*. USA: Artech House.
- [14] Johnson. Sarah.J. (2010). *Iterative Error Correction*. USA: Cambridge University Press.
- [15] Yuan Jiang. (2010). *A practical Guide to Error Control Coding Using MATLAB*. London: Artech House.
- [16] Madhow. Upamanyu. (2008). *Fundamental of Digital Communication*. USA: Cambridge University Press.
- [17] www.wikipedia.org/wiki/Ldpc
- [18] Futaki.H. & Ohtsuki. T. (2002). Performance of Low Density Parity Check (LDPC) Coded-OFDM Systems. IEEE International Communication. ICC.
- [19] Kabatiansky.G., Krouk.E. & Semenov.S .(2005). *Error Correcting Coding and Security for Data Networks*. England: John Wiley & Sons.
- [20] Ryan. William.E. & Shu Lin.(2009). *Channel Codes: Classical and Modern*.USA: Cambridge University Press.
- [21] Amstrong, J. (2002). Peak-to-Average Power Reduction for OFDM by repeated Clipping and frequency domain Filtering. Electronic Letters IEEE.
- [22] Juwono.Filbert Hilman & Gunawan. Dadang. (2010). *Prinsip-Prinsip OFDM*. Indonesia: Penerbit ANDI.
- [23] Pradabpet, C., et. al.(2008). A New PAPR Reduction Technique for OFDM-WLAN in 802.11a Systems. Proceedings of the 2008 Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, pp. 179 – 184.
- [24] Li, Xiaodong dan Leonard J. Cimini, Jr. (1998). Effects of Clipping and Filtering on the Performance of OFDM. IEEE Communication Letters, vol. 2, No. 5, pp. 131 – 133, May 1998.
- [25] Kimura.S., Nakamura.T., Saito.M., & Okada.M.(2008). PAR Reduction for OFDM Signals based on Deep Clipping. ISCCP Malta.