

Perbandingan Nilai *Bit Error Rate* Penyandi Turbo dengan Sandi Komponen Konvolusional, *Block* dan Gabungan

Eva Yovita Dwi Utami¹, Ruth Johana Angelina², Andreas Ardian Febrianto³

Program Studi Teknik Elektro,
Fakultas Teknik Elektronika dan Komputer,
Universitas Kristen Satya Wacana, Salatiga
¹eva.utami@uksw.edu, ²ruth.johana@gmail.com, ³andreas.febrianto@uksw.edu

Ringkasan

Pada sistem komunikasi digital, dibutuhkan sistem penyandian kanal yang berfungsi untuk meminimalkan efek kerusakan atau hilangnya informasi yang diakibatkan oleh derau di dalam kanal. Sistem penyandian kanal yang baik memiliki nilai *Bit Error Rate* (*BER*) yang kecil. Penyandi Turbo merupakan salah satu teknik penyandian kanal pengoreksi galat dengan penggunaan dua sandi komponen yang dihubungkan secara paralel baik pada penyandi maupun pengawasandi. Pada penelitian ini, dilaporkan perbandingan nilai *BER* Penyandi Turbo dengan sandi komponen Konvolusional, *Block* dan Gabungan berdasarkan jumlah iterasi yang dilakukan, ukuran *interleaver*, efek *puncturing* serta kanal yang dilalui oleh isyarat dari pengirim menuju penerima. Dari hasil simulasi, secara umum Turbo Konvolusional memiliki kinerja yang paling baik dibandingkan dengan Turbo *Block* dan Turbo Gabungan pada setiap nilai iterasi, pada data tersandi dengan dan tanpa efek *puncturing* maupun pada setiap kanal yang dilewati. Kinerja *BER* semua jenis penyandi mengalami perbaikan ketika proses iterasi ditingkatkan. Sementara itu efek *puncturing* akan menurunkan kinerja Penyandi Turbo. Pada kanal AWGN, Turbo Konvolusional dapat mencapai nilai *BER* sebesar 0 saat $E_b/N_0 = 2$ dB sedangkan Turbo Gabungan dan Turbo *Block* hanya mencapai 0,0087 dan 0,0155. Pada kanal berderau AWGN yang ditambah *multipath Rayleigh fading*, Turbo Konvolusional pada $E_b/N_0 = 2$ dB tetap memiliki kinerja yang paling baik dengan nilai *BER* sebesar 0,0114 sementara Turbo Gabungan dan Turbo *Block* dengan nilai *BER* 0,0458 dan 0,0163.

Kata kunci: penyandian, Turbo, RSC, konvolusional

1. Pendahuluan

Tuntutan kebutuhan akan laju data dan *bandwidth* yang tinggi, serta keandalan transmisi data pada sistem komunikasi nirkabel meningkat dengan sangat cepat. Di sisi lain kanal radio bersifat acak dan sulit diprediksi yang menyebabkan derau dan interferensi tidak dapat dihindari. Konsekuensinya, untuk memastikan bahwa data yang diterima memiliki kemungkinan kesalahan sekecil mungkin, dipersyaratkan teknik-teknik *channel encoding* atau penyandian kanal menggunakan *code* atau sandi tertentu untuk mengoreksi kesalahan yang semakin baik.

Claude Shannon telah menunjukkan bahwa sandi-sandi yang acak dapat dibangkitkan sebelum proses transmisi untuk menjamin keandalan komunikasi pada kecepatan data yang mendekati kapasitas kanal [1]. Teori penyandian kanal telah

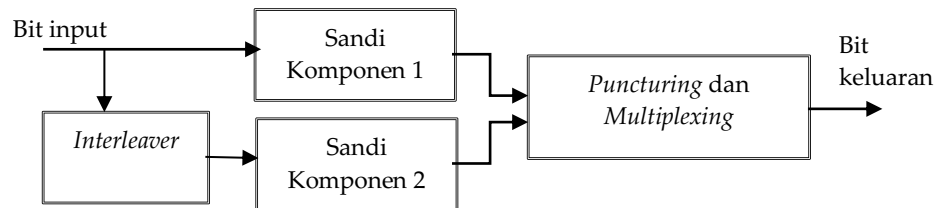
menghasilkan sejumlah sandi yang bertujuan meningkatkan transmisi data yang bebas kesalahan. Teknik penyandian Turbo yang diperkenalkan oleh Berrou *et al* pada tahun 1993 merupakan penyandi berbasis deteksi *soft decision* secara iteratif dari suatu sandi paralel dan dikenal menghasilkan bati penyandian yang tinggi [2]. Penyandi Turbo selanjutnya banyak diadopsi sebagai proteksi terhadap galat pada sistem komunikasi bergerak seluler, komunikasi serat optik, komunikasi *Digital Video Services (DVB)* melalui satelit, *wireless personal area network (WPAN)* dan pada teknik *MC-CDMA* [2],[3],[4].

Penyandi Turbo memanfaatkan dua sandi komponen (*component code*) yang dihubungkan secara paralel dan mengolah data informasi yang sama dalam urutan yang berbeda. Jenis penyandi yang biasa digunakan sebagai proteksi galat menurut prosesnya dapat dibagi menjadi sandi Konvolusional dan sandi *Block* [5]. Pada Penyandi Turbo kedua jenis sandi ini dapat digunakan sebagai sandi komponen. Penyandi Turbo Konvolusional memiliki kelebihan sangat fleksibel dan mudah beradaptasi pada rentang ukuran data dan laju penyandian yang sangat besar [3]. Sementara itu penyandi *Block* pada Turbo dapat memberikan keuntungan dalam hal pengurangan kompleksitas untuk aplikasi dengan laju data dan efisiensi spektral yang tinggi [6].

Penelitian mengenai Penyandi Turbo telah dilakukan sebelumnya. Pada penelitian [7] dilaporkan kinerja Penyandi Turbo menggunakan sandi komponen *RSC* dengan parameter yang diubah adalah laju sandi, teknik pengawasandi, iterasi, panjang sandi dan kanal yang dilewati. Pada [8] dilakukan modifikasi pada Turbo *RSC* dengan penggunaan *random interleaver* dan *deinterlacer* dan dilakukan simulasi variasi laju sandi dan iterasi. Penelitian untuk meningkatkan kinerja Turbo *Block* dilakukan pada [9] dengan cara peningkatan algoritma pengawasandi berbasis *Particle Swarm Optimization* dan *Support Vector Machine*. Dalam makalah ini akan dibahas hasil penelitian tentang penyandi Turbo dengan melakukan simulasi perbandingan nilai *bit error rate (BER)* antara Penyandi Turbo yang menggunakan penyandi komponen konvolusional dan sandi *Block*. Selain itu dilakukan simulasi dengan menggabungkan kedua sandi komponen dalam satu Penyandi Turbo menggunakan Matlab. Sandi komponen yang digunakan untuk Turbo Konvolusional adalah sandi *Recursive Systematic Convolutional (RSC)*, sedangkan pada Turbo *Block* menggunakan sandi *BCH*. Pada Turbo Gabungan digunakan kedua sandi komponen tersebut. Untuk mengetahui kinerja galat ketiga jenis penyandian tersebut, nilai *BER* akan dibandingkan pada beberapa nilai iterasi, pada ukuran *block interleaver* yang berbeda, pada saat dilakukan *puncturing* dan tidak dilakukan *puncturing* serta pada kanal *AWGN* dan *Rayleigh*.

2. Metode Penelitian

Penyandi *Turbo* memanfaatkan dua sandi komponen yang dihubungkan secara paralel. Bit-bit data akan disandikan sesuai dengan teknik sandi komponen yang digunakan. Pada Gambar 1 dapat dilihat bahwa data yang masuk ke penyandi komponen kedua dilewatkan dahulu pada *interleaver*. *Interleaver* pada penyandi *Turbo* berfungsi untuk mengubah urutan data bit sehingga bit data yang dipancarkan oleh kedua komponen sandi tersebut menghasilkan *parity bit* yang saling tak gayut atau berbeda sama sekali [1],[10]. Dua komponen sandi dapat dihubungkan secara opsional dengan blok *puncturing*. Blok ini digunakan untuk mendapatkan *code rate* (laju penyandian) yang lebih tinggi. Pengawasandi *Turbo* menggunakan dua komponen *Maximum A-Posteriori Algoritma (MAP)* yang dihubungkan secara paralel dengan *interleaver* dan *deinterleaver*.



Gambar 1. Bagan kotak Penyandi Turbo dasar [10]

2.1. Penyandi Turbo Konvolusional

Sandi komponen yang digunakan pada penyandi Turbo Konvolusional adalah *Recursive Systematic Convolutional (RSC)*. Sandi ini disebut *Recursive* karena terdapat keluaran yang diumpanbalikkan dan disebut *sistematik* karena terdapat bit masukan yang langsung menjadi keluaran. Sandi *RSC* dibangkitkan dengan suatu generator polynomial dengan *delay* dan laju penyandian tertentu. Laju penyandian merupakan perbandingan jumlah bit masukan dengan jumlah bit keluaran.

Secara lengkap Penyandi Turbo Konvolusional dimodelkan dalam sistem komunikasi seperti Gambar 2. Dalam simulasi data bit acak dibangkitkan dan akan menjadi masukan bagi penyandi Turbo. Data disandikan oleh sandi komponen *RSC* yang mulanya memiliki laju penyandian $\frac{1}{2}$ dihubungkan secara paralel dengan sebuah sandi *RSC* yang sama dan membentuk penyandi *Turbo Konvolusional* dengan laju penyandian menjadi $\frac{1}{3}$. Keluaran pertama adalah bit sistematik yang hanya berasal dari sandi *RSC* 1. Hal ini dikarenakan bit data masukan dari sandi *RSC* 1 dan 2 sama walaupun berbeda urutan. Keluaran kedua adalah *parity bit* yang dihasilkan sandi *RSC* 1 dan keluaran ketiga adalah *parity bit* yang dihasilkan sandi *RSC* 2. Penyandi *RSC* yang digunakan memiliki laju penyandian $=1/2$ dengan *generator polynomial* $[1\ 5/7]$, blok *delay* = 2 dengan cara kerjanya dapat dilihat pada referensi [11].

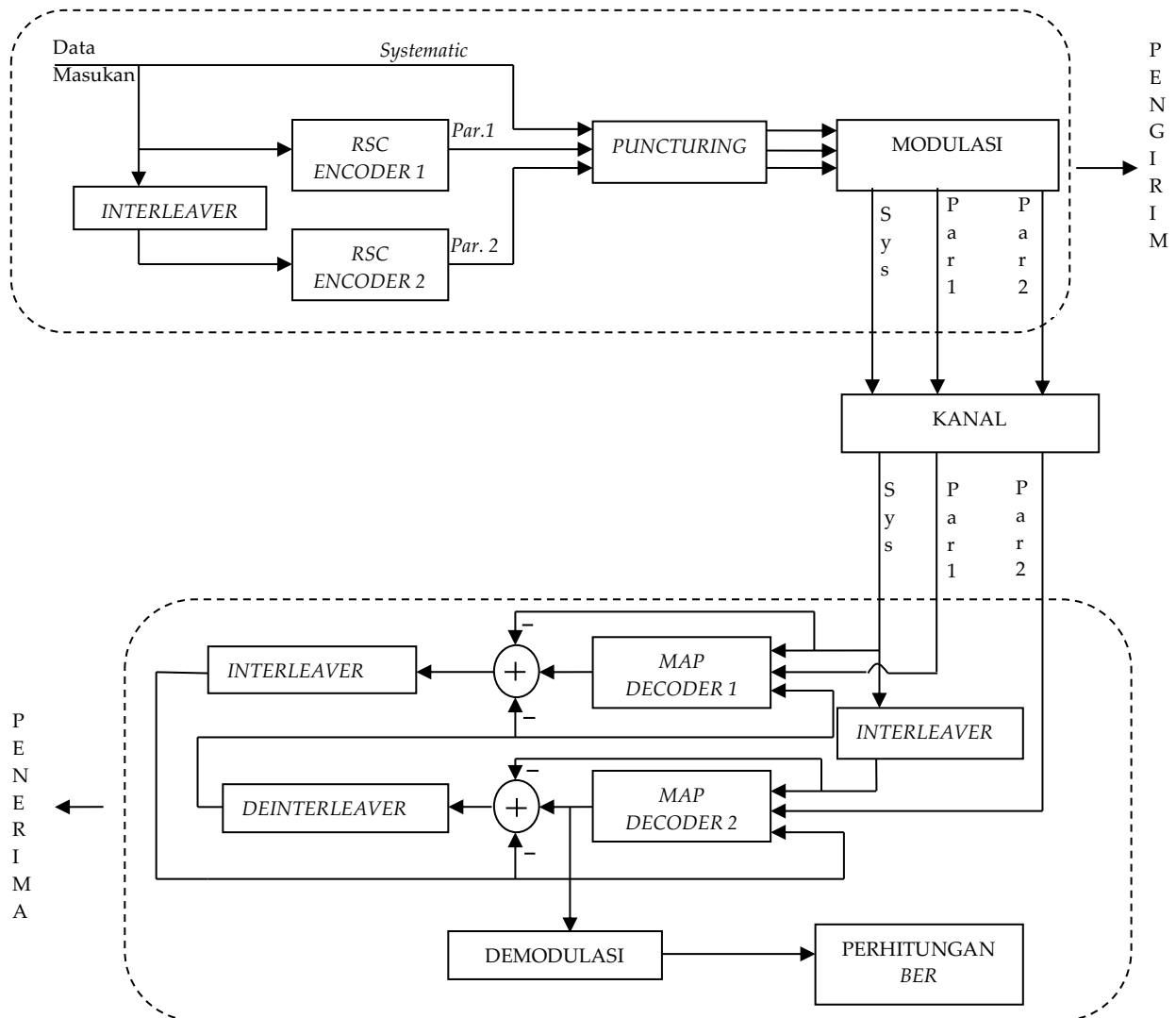
Interleaver berfungsi untuk mengubah urutan data dengan aturan tertentu. Data masukan mengisi matriks *interleaver* dengan jumlah baris dan kolom tertentu. Data masuk diurutkan baris demi baris dan keluarannya diurutkan kolom demi kolom. Hal ini dilakukan agar sandi komponen pertama dan kedua menghasilkan sandi yang berbeda, cara yang dimanfaatkan Penyandi Turbo untuk mencapai kinerja mendekati batas teori kapasitas kanal [8].

Puncturing merupakan proses yang dilakukan untuk meningkatkan laju penyandian. Pada data yang di-*puncture* terdapat *parity bit* yang tidak ditransmisikan. Pola *puncturing* ditentukan oleh matriks *puncturing* dan laju penyandian yang diinginkan. Matriks ini akan membuat *parity bit* dari setiap sandi komponen *RSC* muncul bergantian.

Data yang telah disandikan Turbo dikirimkan melalui kanal dengan terlebih dahulu dimodulasi dengan *Binary Phase Shift Keying* dengan frekuensi pembawa f_c . Isyarat hasil modulasi melewati kanal *Rayleigh*. Parameter kanal *Rayleigh* pada simulasi ini digunakan frekuensi pembawa 2140 MHz dengan kecepatan kendaraan adalah 40 km/jam, sehingga didapatkan frekuensi Doppler sebesar 79,26Hz. Selain itu isyarat yang melalui kanal akan ditambahkan dengan derau *AWGN*.

Pada bagian penerima isyarat yang sudah didemodulasi akan diawasandikan. Pengawasandi menggunakan *MAP* untuk menghitung nilai *LLR (Log Likelihood Ratio)* sebuah bit. *MAP* memanfaatkan diagram *Trellis* yang dihasilkan oleh penyandi *RSC* dan menghitung kemungkinan jalur yang dilewati data masukan. *MAP* membaginya menjadi

tiga bagian yaitu penghitungan *branch matrix*, penghitungan rekursi maju dan penghitungan rekursi mundur secara berurutan. Setelah ketiga nilai tersebut diketahui, kemudian bisa diperoleh nilai ekstrinsik yang menjadi informasi *a-priori* pada sebuah pengawasandi [10].

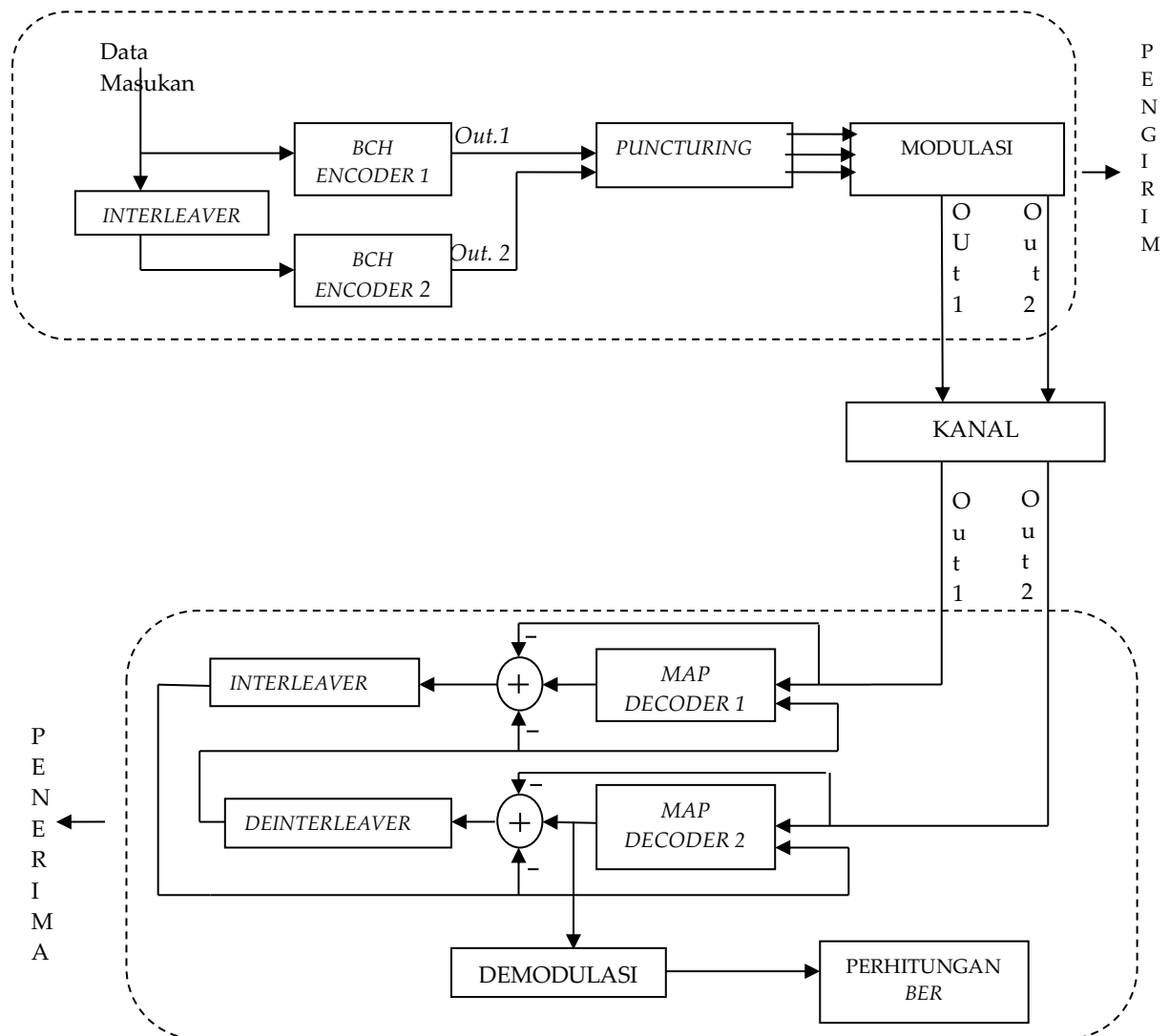


Gambar 2. Diagram sistem komunikasi dengan Penyandi Turbo Konvolusional

Terdapat dua pengawasandi Turbo yang dihubungkan secara paralel dengan *interleaver* maupun *deinterleaver*. Pada pengawasandi 1, masukan dan *parity bit* keluaran penyandi RSC 1 serta hasil *soft decision* yang dihasilkan oleh pengawasandi 2 dimanfaatkan oleh MAP untuk mendapatkan hasil *soft decision* suatu bit. Semuanya ini dilakukan dengan menghitung kemungkinan suatu jalur dalam diagram *Trellis* dilewati oleh data masukan. Hasil *soft decision* pengawasandi 1 terlebih dahulu diubah urutannya dalam *interleaver* sebelum menjadi masukan pengawasandi 2 agar didapati urutan data yang sama seperti data yang masuk pada pengawasandi 2. Penukaran informasi ini akan berlangsung sampai batas iterasi yang ditentukan. Proses ini juga berlaku untuk pengawasandi 2.

2.2. Penyandi Turbo Block

Sandi komponen yang digunakan pada penyandi *Turbo Block* adalah sandi *Bose Chaudhuri Hocqueghem* atau *BCH (n,k)*. Sandi *BCH* yang digunakan adalah *BCH (7,4)* yang memiliki laju penyandian sebesar $4/7$ dengan *generator polynomial* $1+x+x^3$. Ketika dihubungkan secara paralel dengan sandi *BCH (7,4)* lain untuk membentuk penyandi *Turbo Block*, maka laju penyandian dari *Turbo Block* adalah $0,4$ [10]. Pada sandi *BCH (7,4)* untuk setiap 4 bit yang disandikan akan menghasilkan 3 *parity bit* atau terdapat total keluaran 7 bit. Proses yang disimulasikan secara lengkap untuk *Turbo BCH* diperlihatkan pada Gambar 3.



Gambar 3. Diagram sistem komunikasi dengan Penyandi Turbo BCH

Pada bagian pengawasandi terdapat dua pengawasandi Turbo yang dihubungkan secara paralel dengan *interleaver* maupun *deinterleaver*. Prinsip kerjanya sama dengan pengawasandi Turbo Konvolusional, tetapi data yang masuk terlebih dahulu dibagi menjadi beberapa *frame*. Tiap *frame* sandi *BCH (7,4)* terdiri dari 4 bit dengan 3 *parity bit* yang dihasilkan. Perhitungan dalam diagram *Trellis* dilakukan per *frame*.

2.3 Penyandi Turbo Gabungan

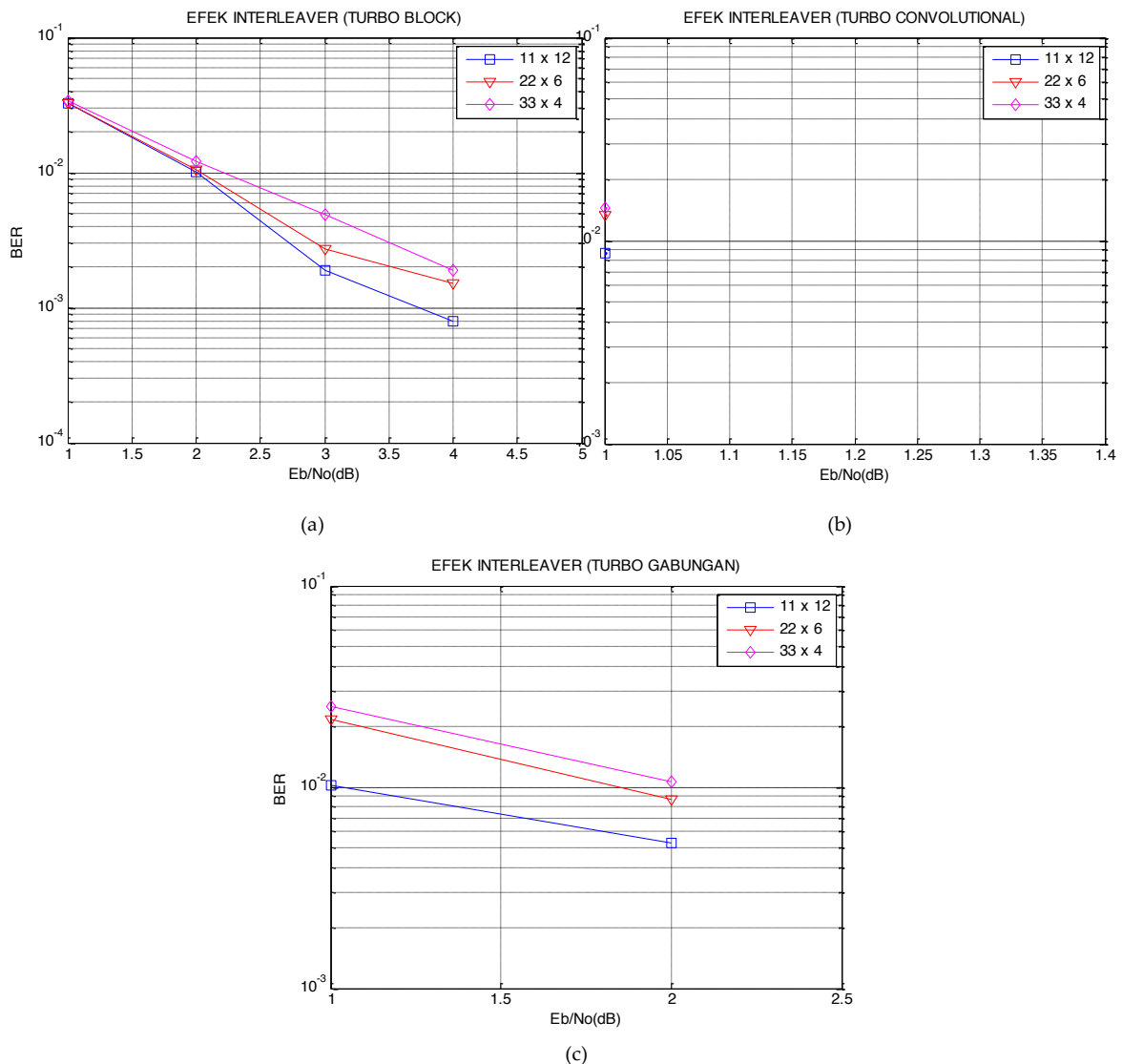
Penyandi Turbo Gabungan menggunakan sandi RSC dengan blok *delay* 2 untuk sandi komponen 1 dan sandi BCH (7,4) sebagai sandi komponen 2. Demikian juga pada penerima terdapat satu pengawasandi Turbo Konvolusional dan satu pengawasandi Turbo Block yang dihubungkan secara paralel dengan *interleaver* ataupun *deinterleaver*. Prinsip yang digunakan sama, kedua pengawasandi saling bertukar nilai *soft decision* hingga iterasi yang diinginkan. Saat informasi dari pengawasandi Turbo Konvolusional menjadi masukan pengawasandi Turbo Block, informasi terlebih dahulu dibagi menjadi beberapa *frame* bergantung pada jenis sandi BCH yang digunakan pada Turbo Block.

3. Hasil dan Pembahasan

3.1 Hasil Simulasi Berdasarkan *Interleaver* yang Digunakan

Interleaver berfungsi untuk mengubah urutan data dengan aturan tertentu. Dalam simulasi ini diterapkan *Block interleaving* dengan tiga ukuran baris dan kolom matriks yang berbeda, yaitu 11×12, 22×6, 33×44. Hasil simulasi menunjukkan semakin besar kolom matriks *interleaver* maka nilai BER dapat ditekan sehingga kinerja sistem semakin baik. Sebaliknya, jika kolom matriks *interleaver* semakin kecil maka nilai BER akan meningkat. Hal ini ditunjukkan oleh matriks *interleaver* 33 × 4, memiliki ukuran kolom terkecil di antara ketiga *interleaver* yaitu 4. Kinerja menurun, dikarenakan bit yang berurutan ketika dilakukan *interleaving* terlalu dekat jarak pemisahannya maka jika terjadi *burst error* pada penerima galat masih terkumpul (tidak tersebar). Hal ini menyebabkan deteksi kesalahan menjadi lebih sulit.

Hal ini terjadi baik pada Turbo Block, Turbo Konvolusional maupun Turbo Gabungan. Matriks *interleaver* dengan ukuran 33 × 4 menyebabkan kinerja sistem Turbo Konvolusional maupun Turbo Gabungan menurun sedangkan matriks *interleaver* dengan ukuran 11 × 12 memiliki kinerja paling baik. Gambar 4 menunjukkan kinerja sistem penyandi Turbo berdasarkan *interleaver* yang digunakan. Pada simulasi Turbo Block yang dilakukan, ketika diberikan $E_b/N_0 > 4$ dB, nilai BER sudah mendekati nilai nol sehingga tidak muncul pada grafik. Demikian pula BER pada simulasi Turbo Konvolusional mencapai nol pada E_b/N_0 di atas 1 dB, dan pada Turbo Gabungan mencapai nol saat E_b/N_0 di atas 2 dB



Gambar 4. Grafik Kinerja Penyandi Turbo Berdasarkan *Interleaver* yang Digunakan: (a). Turbo BCH, (b) Turbo Konvolusional, (c). Turbo Gabungan

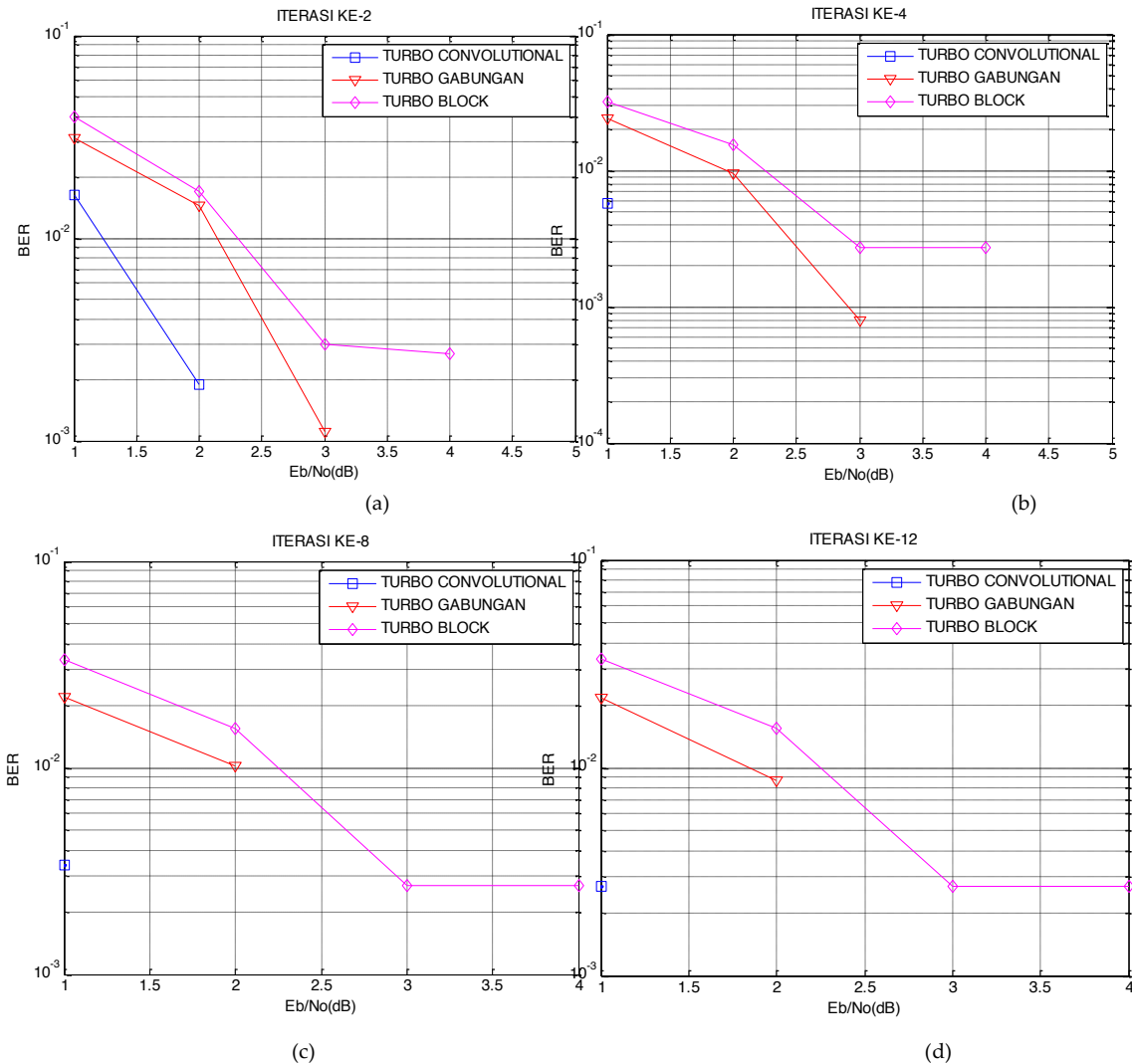
3.2 Hasil Simulasi Berdasarkan Iterasi yang Dilakukan

Ketiga jenis penyandi Turbo disimulasikan pada iterasi kedua, keempat, kedelapan dan kedua belas. Nilai E_b/N_0 dinaikkan untuk melihat penurunan BER. Gambar 5 menunjukkan terjadi penurunan nilai BER yang lebih cepat pada saat iterasi ditingkatkan. Hal ini menunjukkan untuk mencapai nilai BER tertentu iterasi yang meningkat akan menurunkan nilai E_b/N_0 yang dipersyaratkan. Meskipun demikian pada peningkatan iterasi kedelapan dan kedua belas penurunan BER menjadi kurang signifikan karena hanya terjadi sedikit penurunan nilai BER bahkan untuk Turbo Block tidak terjadi penurunan nilai BER.

Membandingkan kinerja dari ketiga sistem dapat dilihat sistem yang paling baik adalah Turbo Konvolusional diikuti oleh Turbo Gabungan dan Turbo Block. Dari simulasi tersebut, diketahui bahwa sistem Turbo bekerja maksimal saat nilai E_b/N_0 semakin besar. Turbo Gabungan memiliki nilai BER yang lebih kecil dari Turbo Block. Hal ini disebabkan oleh sandi komponen yang digunakan oleh Turbo Gabungan terdiri dari satu sandi BCH

dan satu sandi *RSC*. Turbo Gabungan mendapat keuntungan dari kinerja *RSC* yang baik sehingga bisa mengungguli kinerja Turbo *Block* yang hanya terdiri dari dua sandi komponen *BCH*.

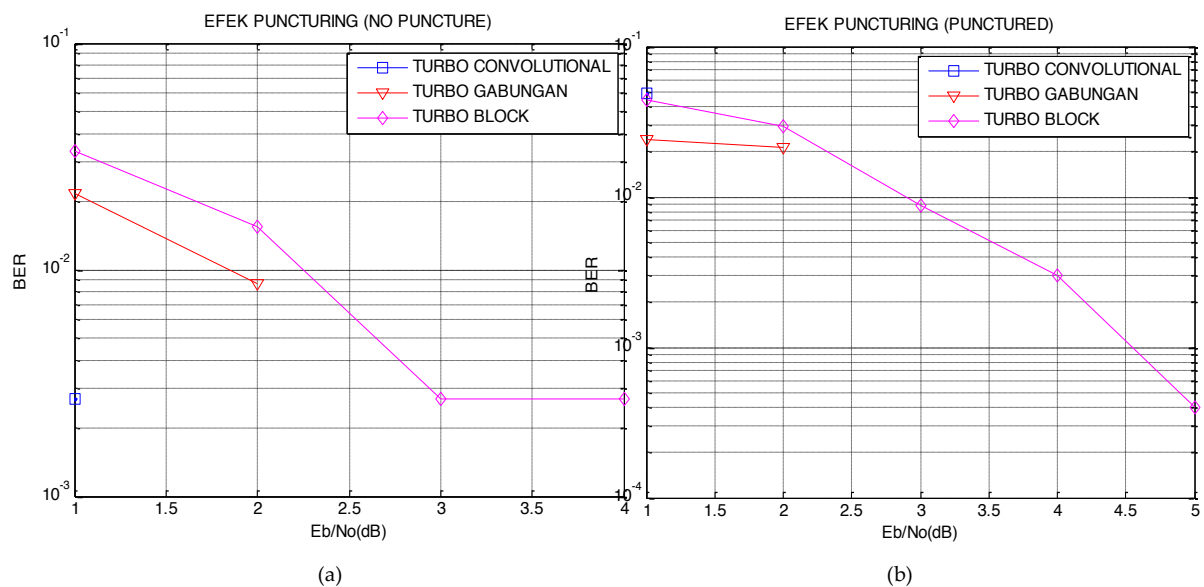
Sebagai contoh pada Gambar 5(c) saat iterasi ke-8, ketika E_b/N_0 bernilai 2 dB Turbo *Block* memiliki nilai *BER* yang lebih besar dibandingkan sistem yang lain yaitu 0,0155 sedangkan Turbo Gabungan bernilai 0,0102 dan nilai *BER* Turbo Konvolusional bernilai mendekati 0 atau bebas dari galat. Pada iterasi ke-12 seperti pada Gambar 5(d), hanya terjadi penurunan yang sangat sedikit untuk Turbo Konvolusional yaitu sebesar 0,0007, 0,0004 untuk Turbo Gabungan saat E_b/N_0 bernilai 1 dB dan untuk Turbo *Block* tidak terjadi penurunan nilai *BER*. Dari data hasil simulasi diperoleh bahwa pada iterasi kedelapan dan keduabelas, *BER* Turbo Konvolusional mendekati nol ketika diberikan $E_b/N_0 > 1$ dB, *BER* Turbo Gabungan mendekati nol ketika diberikan $E_b/N_0 > 2$ dB, dan *BER* Turbo *Block* mendekati nol ketika diberikan $E_b/N_0 > 4$ dB, sehingga pada nilai-nilai yang lebih besar daripada E_b/N_0 tersebut, nilai *BER* tidak muncul pada grafik.



Gambar 5. Grafik Kinerja Penyandi Turbo Berdasarkan Iterasi (a). Iterasi Kedua, (b) Iterasi Keempat, (c).Iterasi Kedelapan, (d) Iterasi Keduabelas

4.2 Hasil Simulasi Berdasarkan Efek *Puncturing*

Puncturing adalah proses yang dilakukan untuk meningkatkan laju penyandian. Dalam hal ini, *puncturing* dilakukan dengan tidak semua *parity bit* pada proses penyandian dikirimkan. Hal ini dilakukan agar sistem dapat menghemat penggunaan *bandwidth* pada sistem komunikasi yang menuntut kecepatan tinggi dengan *bandwidth* terbatas yang ada [11]. Untuk pola *puncturing* disesuaikan dengan matriks *puncturing*. Dengan *parity bit* yang semakin sedikit maka kinerja sistem akan menurun karena bit yang berfungsi sebagai proteksi semakin sedikit sehingga kemungkinan terjadi galat semakin besar. Hal ini dapat dilihat pada Gambar 6(a) untuk sistem tanpa *puncturing* dan Gambar 6(b) untuk sistem dengan *puncturing*.

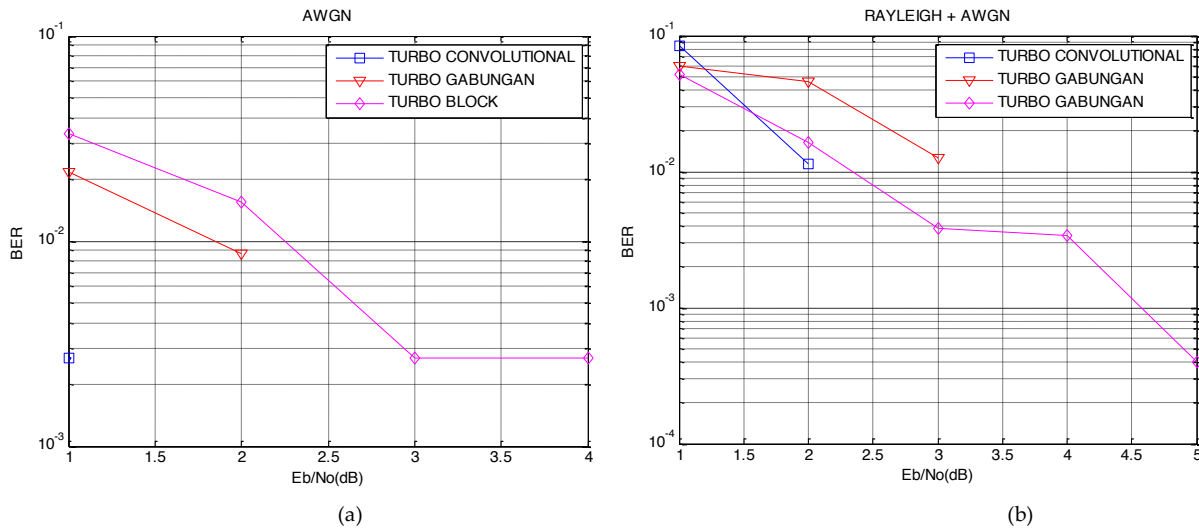


Gambar 6. Grafik Kinerja Penyandi Turbo Berdasarkan Efek *Puncturing* (a). Tanpa *Puncturing*, (b) Dengan *Puncturing*

4.3 Hasil Simulasi Berdasarkan Kanal yang Dilewati

Kanal pada simulasi ini dibagi menjadi dua model yaitu kanal dengan derau AWGN saja dan kanal berderau AWGN yang ditambah *multipath Rayleigh fading*. Gambar 7(a) menunjukkan kinerja sistem Turbo pada kanal dengan derau AWGN dan Gambar 7(b) menunjukkan kinerja sistem Turbo pada kanal berderau AWGN yang ditambah *multipath Rayleigh fading*. Dapat dilihat pada Gambar 7(b), kinerja ketiga sistem Turbo mengalami penurunan dibandingkan ketika kanal yang dilewati hanya mengalami derau AWGN. Data hasil simulasi dan Gambar 7(a) menunjukkan nilai BER akan mencapai nol pada saat diberikan $E_b/N_0 > 1$ dB untuk Turbo Konvolusional, $E_b/N_0 > 2$ dB untuk Turbo Gabungan dan $E_b/N_0 > 4$ dB untuk Turbo Block pada lingkungan derau AWGN. Hasil ini menjadi menurun pada Gambar 7(b) untuk kanal *Rayleigh*, yaitu bahwa dari data simulasi diperoleh nilai BER akan mencapai nol pada saat diberikan $E_b/N_0 > 2$ dB untuk Turbo Konvolusional, $E_b/N_0 > 3$ dB untuk Turbo Gabungan dan untuk Turbo Block ketika $E_b/N_0 = 5$ dB masih mempunyai nilai BER sebesar 0,0004. Pada kanal radio sistem komunikasi nirkabel, adanya *multipath Rayleigh fading* berpengaruh pada penerimaan isyarat. Pengaruh *fading* berakibat pada penerimaan isyarat jalur jamak dengan waktu tunda bervariasi. Selain itu pergerakan penerima menyebabkan modulasi frekuensi acak karena efek Doppler. Kedua hal ini diperhitungkan dalam sistem penyandi. Adanya

fenomena jalur jamak ini menyebabkan isyarat terdistorsi akibat waktu tunda (*delay*) dari isyarat jalur jamak. Distorsi yang terjadi menimbulkan peningkatan galat dalam proses deteksi. Demikian juga modulasi frekuensi acak menyebabkan kanal menjadi *time-varying* dan isyarat kesulitan mengatasi perubahan waktu kanal yang lebih cepat, sehingga dapat juga meningkatkan galat. Pada kanal dengan derau AWGN maupun kanal *Rayleigh* dengan derau AWGN, Turbo Konvolusional memiliki kinerja paling baik yang ditunjukkan dengan nilai BER rendah dibandingkan kedua sistem yang lain.



Gambar 7. Grafik Pengaruh Kanal pada Penyandi Turbo (a).AWGN, (b) *Rayleigh*

7. Kesimpulan

Berdasarkan simulasi yang telah dilakukan, Turbo Konvolusional memperoleh kinerja BER terbaik dibandingkan Turbo Gabungan dan Turbo Block. Hal ini dapat ditunjukkan dari nilai BER yang paling rendah dalam simulasi berdasarkan jumlah iterasi yang dilakukan, simulasi tanpa maupun dengan proses *puncturing* serta simulasi pada kanal AWGN maupun kanal AWGN yang ditambah *Rayleigh*. Ukuran matriks *interleaver* dengan kolom yang semakin besar memiliki kinerja yang baik dalam hal ini matriks *interleaver* dengan ukuran 11×12 . Disisi lain, ukuran 33×4 menjadikan kinerja sistem menurun karena jarak antar bit yang terlalu dekat sehingga tidak dapat membuat *burst error* menjadi tersebar. Akibatnya, galat tidak dapat dideteksi dan nilai BER mengalami peningkatan. Dengan bertambahnya iterasi yang dilakukan, maka kinerja ketiga sistem akan semakin baik. Kinerja sistem yang paling baik adalah Turbo Konvolusional diikuti dengan Turbo Gabungan yang menggunakan sandi BCH dan sandi RSC yang bekerja baik dibandingkan dengan menggunakan sandi BCH saja seperti pada Turbo Block. Pada sistem dengan *puncturing*, simulasi menunjukkan peningkatan nilai BER saat sistem di-*puncture*, karena berkurangnya jumlah *parity bit* yang memproteksi data dari galat. Kanal yang dilewati mempengaruhi kinerja sistem Turbo. Kinerja ketiga sistem menurun saat kanal yang dilewati adalah kanal *Rayleigh* dengan derau AWGN. Hal ini disebabkan oleh fenomena *multipath* dan pergerakan penerima yang terjadi pada kanal *Rayleigh* sehingga menyebabkan kemungkinan galat yang terjadi semakin besar.

Daftar Pustaka

- [1] S. J. Johnson, *Iterative Error Correction: Turbo, Low-Density Parity-Check and Repeat-Accumulate Codes*, New York: Cambridge University Press, 2009.
- [2] S. E. Makhtari, M. Moussaoui, H. Samadi, "Improved Turbo Decoding Using Soft Combining Principle", *Wireless Personal Communications: An International Journal*, vol. 78, no. 2, September 2014, h. 1115-1127.
- [3] K. Perutka, "Turbo codes - Introduction and applications", *Annals of DAAAM and Proceedings of the International DAAAM Symposium*, 2010, h. 191-192.
- [4] A. R. Enayati, P. Azmi, Paeiz, Y. Taghinia, & A. Salahi, "A novel bandwidth efficient SOC-based turbo coding scheme mid reduced complexity MUD for SA-based MC-CDMA systems", *Telecommunication Systems - TELSIS*. vol. 50, 2010, h. 71-78
- [5] M. Borda, *Fundamentals in Information Theory and Coding*, Germany, Springer, 2011
- [6] R. M. Pyndiah, "Near-Optimum Decoding of Product Codes: Block Turbo Codes", *IEEE Transactions on Communications*, vol. 46, no. 8, Agustus 1998, h. 1003-1010.
- [7] S. Jasim, A. Abbas "Performance of Turbo Code with Different Parameters", *Journal of Babylon University/Engineering Sciences*, vol. 25, no. 5, 2017, h. 1684-1692
- [8] S. Kumar, H. Dalal, "Performance Comparison of Turbo and Modified Turbo Codes with Different Rate", *International Journal of Sciences, engineering and Technology Research (IJSETR)*, vol. 3, no. 5, Mei 2014, h. 1520-1523
- [9] V. Sudharsan , V. Karthik, J.S. Vaishnavi, S. Abirami, B. Yamuna, "Performance Enhanced Iterative Soft-Input-Soft-Output Decoding Algorithms for Block Turbo Codes", *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 8, no. 5, 2016, 105-110
- [10] L. Hanzo, T.H. Liew, B.L.Yeap, *Turbo Coding, Turbo Equalisation and Space-Time Coding*, England: John Wiley &sons Ltd, 2002
- [11] A. Imran, *Software Implementation and Performance of UMTS Turbo Code*, M.Sc. Thesis, Tampere University of Technology, 2012

