

PERBANDINGAN KINERJA KODE HAMMING PADA CHANNEL AWGN

Gede Sukadarmika

Staf Pengajar Teknik Elektro, Fakultas Teknik, Universitas Udayana
Kampus Bukit Jimbaran Bali, 80361
Email : sukadarmika@unud.ac.id

Intisari

Noise merupakan suatu hal yang tidak bisa dihindari dalam sistem komunikasi. Desain sistem telekomunikasi dilakukan untuk meminimalkan pengaruh dari noise tersebut terhadap informasi yang ditransmisikan. Salah satu upaya yang dilakukan adalah dengan menerapkan *error control coding*. Kode Hamming merupakan contoh dari teknik error control coding yang ada. Kinerja kode Hamming dibedakan atas jumlah bit parity yang dimiliki. Pada tulisan ini dilakukan simulasi yang disusun berdasarkan beberapa model yaitu : model simulasi tanpa pengkodean, model simulasi dengan kode Hamming (7,4), (15,11), (31,26), (63,57), dan (127,120). Untuk modulasi di gunakan modulasi BPSK (*Binary Phase Shift Keying*), sedangkan untuk kanal yang di gunakan adalah kanal AWGN (*Additive White Gaussian Noise*). Data diperoleh dari hasil simulasi dengan parameter BER (*Bit Error Rate*) berdasarkan variasi SNR (*Signal to Noise Ratio*). Perhitungan dilakukan dengan membandingkan nilai BER pada keadaan nilai SNR yang tetap dan sebaliknya dengan membandingkan nilai SNR pada keadaan nilai BER yang tetap. Dari hasil simulasi diperoleh bahwa kode Hamming memberikan peningkatan kinerja dari sistem. Sedangkan, kinerja kode Hamming meningkat sesuai dengan peningkatan jumlah bit *parity*. BER paling kecil pada keadaan SNR tetap diperoleh pada saat menggunakan kode Hamming (127,120) atau pada kode Hamming dengan bit *parity* = 7. Demikian juga Kode Hamming (127,120) membutuhkan nilai SNR yang paling rendah yaitu 6,75 dB untuk memperoleh nilai BER = 10^{-6} .

Kata Kunci : *Noise, error control coding, Kode Hamming*

1. PENDAHULUAN

Keunggulan dari teknologi telekomunikasi digital dibandingkan dengan sistem analog merupakan daya tarik tersendiri dalam perkembangan teknik telekomunikasi belakangan ini, yang semakin menuntut akurasi dan kehandalan sistem telekomunikasi. Salah satu keunggulannya adalah dalam hal kemampuan mengatasi noise yang terjadi pada kanal telekomunikasi.

AWGN merupakan kanal yang sering menjadi model dasar untuk kanal pada sistem komunikasi digital dan juga digunakan sebagai standar untuk model kanal komunikasi yang lain.

Noise merupakan sinyal yang tidak diharapkan dalam sistem telekomunikasi karena bersifat mengganggu serta kehadirannya tidak bisa ditentukan (acak). Namun, noise selalu ada dalam setiap sistem telekomunikasi, khususnya pada sistem transmisi. Hal ini dapat menyebabkan kesalahan pada informasi yang diterima.

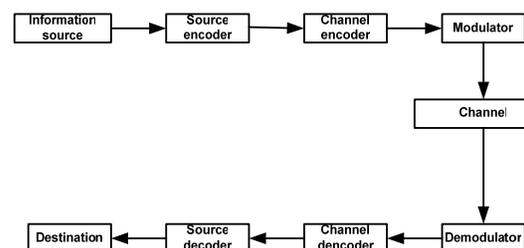
Pengaruh noise tidak sepenuhnya dapat dihilangkan, melainkan hanya dapat dikurangi. *error control coding* merupakan teknik yang digunakan untuk itu. Salah satu teknik pengkodean tersebut adalah teknik FEC (*Forward Error Correction*) yang merupakan teknik pengkodean dengan menambahkan bit-bit redundansi (*back up*) pada data atau sinyal informasi yang ditransmisikan pada kanal telekomunikasi.

Bagian dari teknik pengkodean FEC adalah kode blok Hamming yang memiliki kemampuan mengoreksi *single error* dengan algoritma yang sederhana. Kode ini memiliki jenis kode yang berbeda-beda sesuai dengan *parity bit* yang dimiliki, dan memiliki perhitungan matrik yang berbeda-beda yang tentu saja akan berpengaruh pada penerimaan sinyal dan performansinya.

2. TINJAUAN PUSTAKA

2.1 Sistem Komunikasi Digital

Sistem komunikasi bertujuan untuk mengirimkan informasi dari sumber ke penerima.



Gambar 1 Bagian Sistem Komunikasi Digital

Ada tiga komponen utama pada suatu sistem komunikasi, yaitu : pengirim, media transmisi, dan penerima. Pada bagian pengirim terdapat proses *source encoding*, *channel encoding* dan modulasi,

sedangkan pada bagian penerima terdapat proses demodulasi, *channel decoding* dan *source decoding*. Bagan sebuah sistem komunikasi dapat dilihat pada gambar 1.

2.2 Teknik Modulasi Digital

Modulasi merupakan teknik penumpangan sinyal informasi pada suatu sinyal *carrier*. Ada tiga teknik *Digital Analog Modulation (DAM)* yang umum digunakan, yaitu :

1. *Amplitude Shift Keying (ASK)*

ASK merupakan suatu teknik modulasi dengan cara mengalikan sinyal *carrier* dengan sinyal informasi, secara matematis dapat dituliskan :

$$S(t) = A \cos(2\pi f_c t + \phi) \dots\dots\dots(1)$$

2. *Frequency Shift Keying (FSK)*

FSK menggunakan frekuensi berbeda untuk bit 1 dan 0. Hasil modulasi didapatkan dengan menjumlahkan 2 amplitudo dari sinyal yang telah dimodulasi dari frekuensi *carrier* yang berbeda, secara matematis hal tersebut dapat ditunjukkan pada persamaan berikut :

$$S(t) = A_1 \cos(2\pi f_{c1} t + \phi) + A_2 \cos(2\pi f_{c2} t + \phi) \dots\dots\dots(2)$$

3. *Phase Shift Keying (PSK)*

PSK merupakan teknik modulasi digital dengan cara mengubah phase dari sinyal pembawa, teknik modulasi tersebut secara matematis dapat dituliskan sebagai berikut :

$$S(t) = \cos(2\pi f_c t + \phi(t)) \dots\dots\dots(3)$$

Untuk selanjutnya hanya akan diterangkan mengenai teknik modulasi *Phase Shift Keying (PSK)* saja.

2.3 BPSK (Binary Phase Shift Keying)

Dalam BPSK, dua output fase membawa informasi *binary*. Satu fase output mewakili logic 1 dan yang lainnya logic 0. Sesuai dengan perubahan keadaan sinyal input digital, fase pada output *carrier* bergeser diantara dua sudut yang keduanya terpisah 180°. Nama lain untuk BPSK adalah *Phase Reversal Keying (PRK)* dan *biphase modulation*. BPSK adalah suatu bentuk *suppressed carrier* memodulasi suatu sinyal *continuous wave (CW)* (Bateman, 1998).

Pada modulator BPSK, sinyal output didapatkan dari perkalian sinyal *carrier* dengan data biner. Jika bit 1 direpresentasikan oleh tegangan +1 V dan bit -1V untuk bit 0, kemudian sinyal *carrier* dikalikan dengan +1 atau -1, maka sinyal output adalah +sin $\omega_c t$ atau -sin $\omega_c t$.

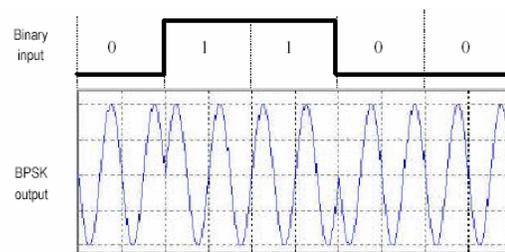
Parameter – parameter dari sinyal (amplitude, frekuensi, dan phase) mengalami perubahan setiap detik, hal ini disebut *signaling rate* yang dinyatakan dalam baud, 1 baud sama dengan 1 perubahan setiap detiknya. Dengan menggunakan teknik modulasi digital, seperti ASK, FSK dan BPSK, maka besarnya *signaling rate* sama dengan nilai bit rate, sedangkan

dengan menggunakan QPSK dan M-ary PSK, maka nilai bit rate dapat melebihi nilai baud rate.

Balanced modulator adalah suatu produk dari dua sinyal input. Frekuensi fundamental (*f*) pada sederetan bit sebanding dengan setengah bit rate (*f_b/2*). Secara matematis, fase output pada suatu modulator BPSK adalah :

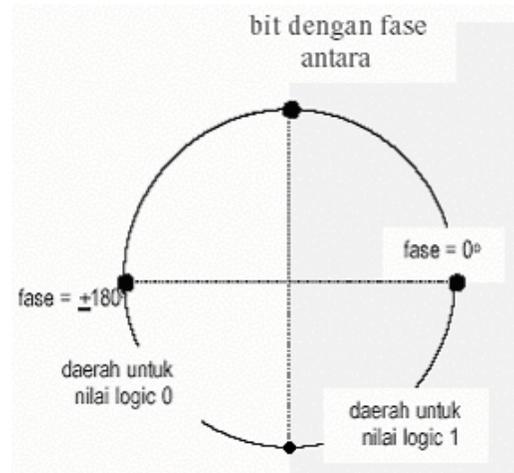
$$\begin{aligned} \text{Output} &= (\sin 2\pi f a t) \times (\sin 2\pi f c t) \\ &= \frac{1}{2} \cos(2\pi f c t \times 2\pi f a t) - \frac{1}{2} \cos(2\pi f c t \times 2\pi f a t) \dots\dots\dots(4) \end{aligned}$$

Sebagai konsekuensinya, *double -sided minimum Nyquist (F_n)* adalah 2(2πf_at), dan karena 2πf_at = F_b/2, maka: F_n=2(F_b/2) = F_b. Gambar (2) menunjukkan fase output sebagai fungsi dari data yang masuk pada sistem BPSK. Terlihat bahwa spektrum pada suatu modulator BPSK adalah senilai sinyal *double sided band suppressed carrier* dimana *upper side* dan *lower side* frekuensi terpisah dari frekuensi *carrier* oleh nilai yang sebanding dengan ½ bit rate.



Gambar 2. Fase Output Sebagai Fungsi Dari Data Yang Masuk Pada Sistem BPSK

Sebagai konsekuensinya, bandwidth minimum (*F_n*) yang diperlukan untuk meloloskan sinyal output BPSK pada kasus terburuk adalah senilai bit rate input.



Gambar 3. Daerah Keputusan Untuk BPSK

Perbandingan Kinerja BPSK dan Diferensial BPSK ditentukan oleh kemampuan receiver dalam menterjemahkan setiap bit informasi yang dikirimkan pemancar dengan menggunakan gelombang carrier. Probabilitas sebuah informasi diterjemahkan secara salah, yang dalam hal ini disebabkan penerimaan bit secara salah (PE), karena satu simbol informasi dalam sistem BPSK berisi satu bit.

Amplitudo sinyal yang diterima tidak dipermasalahkan pada sistem BPSK karena informasinya terkandung dalam fase gelombang carrier yang termulasi. Pada kondisi ideal, fase gelombang carrier termulasi bisa bernilai 0° untuk nilai bit '1' dan bernilai 180° untuk nilai bit '0'. Tapi dalam kondisi real, hal ini jarang terjadi. Dengan adanya gangguan yang menyebabkan pergeseran fase akan muncul suatu kondisi dimana fase gelombang termulasi yang pada awalnya bernilai 0° karena membawa bit informasi '1' akan bergeser sehingga fasenya menjadi 95°. Tentu saja oleh receiver akan diterjemahkan sebagai bit '0' dan ini akan menimbulkan satu kesalahan atau bit error. Ada juga suatu kondisi dimana pergeseran fase menyebabkan fase berada pada nilai 90° sehingga receiver sulit menterjemahkannya. Kemungkinan atau probabilitas terjadinya kesalahan (*probability of error*) dalam menterjemahkan informasi yang dalam hal ini satu simbol yang terdiri dari satu bit untuk sistem BPSK berbeda dengan sistem Diferensial BPSK. Untuk sistem BPSK nilai *probability of error* (*Pe*) dapat didekati dengan suatu fungsi matematis:

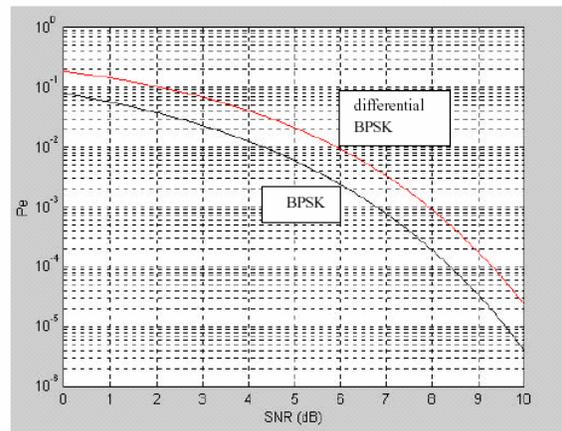
$$p(e) = 0.5 \operatorname{erfc} \left[\frac{\sqrt{2} \cdot snr}{\sqrt{2}} \right] \dots\dots\dots(5)$$

Sedangkan untuk sistem Diferensial BPSK dinyatakan sebagai berikut:

$$Pe = \frac{1}{2} \exp\left(-\frac{Eb}{No}\right) \dots\dots\dots(6)$$

dimana:
 Eb = Energi perbit rata-rata untuk BPSK maupun Diferensial BPSK.
 No = Nilai mean dan sekaligus varians dari noise Gaussian.

Kedua persamaan diatas berlaku untuk kondisi kanal yang memiliki noise yang terdistribusi secara Gaussian yang dapat diakibatkan oleh noise temperatur, noise yang ada pada medium kabel tembaga dan sebagainya. Dengan probability of error yang digambarkan sebagai fungsi Eb/No gambaran kinerja untuk sistem BPSK dan Diferensial PSK dapat kita lihat seperti Gambar (4). Terlihat bahwa kinerja sistem BPSK lebih baik, yang ditunjukkan dengan kemampuan untuk menghasilkan probability of error yang lebih kecil untuk SNR yang sama.



Gambar 4. Kinerja BPSK dan Differential BPSK, (*Pe*) Sebagai Fungsi (SNR)

Adapun contoh pengaplikasian dari modulasi BPSK (Binary Pashe Shift Keying) adalah pada CDMA 2000 dan CDMA 20001X.

2.4 Noise

Noise merupakan sinyal yang tidak diharapkan dalam sistem telekomunikasi karena bersifat mengganggu serta kehadirannya tidak bisa ditentukan (acak). Sehingga hanya dapat dirumuskan probabilitas ataupun kisaran nilai (*range*) dari *noise* tersebut.

Menurut Carlson (1986), *Probability Density Function (PDF)* dari sinyal acak atau variabel acak *x*, didefinisikan sebagai probabilitas dari variabel acak *x* yang mempunyai nilai antara *xj* dan *xj + ∂x*, sehingga dapat dituliskan dengan persamaan :

$$P(x_j) \partial x = P\{x_j \leq x \leq x_j + \partial x\} \dots\dots\dots(7)$$

Probabilitas dari variabel acak didapatkan dengan mengintegrasikan fungsi pdf dari sinyal tersebut dengan interval *x1* dan *x2*.

$$P\{a \leq x \leq b\} = \int_a^b p(x) \partial x \dots\dots\dots(8)$$

$P\{-\infty < x < \infty\}$ adalah *unity*, sehingga :

$$\int_{-\infty}^{\infty} p(x) \partial x = 1$$

Distribusi dari fungsi probabilitas *p(x)* didefinisikan sebagai probabilitas dari variabel acak, dimana *x* kurang dari *xj*.

$$P(x) = \{x < x_j\} = \int_{-\infty}^{x_0} p(x) dx \dots\dots\dots(9)$$

dengan menggunakan aturan pada proses pengintegrasian :

$$P\{x_1 < x < x_2\} = P(x_2) - P(x_1) \dots\dots\dots(10)$$

dengan menggunakan persamaan (8) dan (9), didapatkan :

$$P(\infty) = 1, P(-\infty) = 0 \dots\dots\dots(11)$$

Pada sistem komunikasi, SNR merupakan faktor yang penting dalam menentukan kualitas kanal komunikasi, besarnya didapatkan seperti berikut :

$$SNR = 10 \log_{10} (S/\sigma^2) \dots\dots\dots(12)$$

Jika sumber dari *noise* merupakan *thermal noise*, maka besarnya standar deviasi dapat dirumuskan sebagai berikut :

$$\sigma^2 = kTB \dots\dots\dots(13)$$

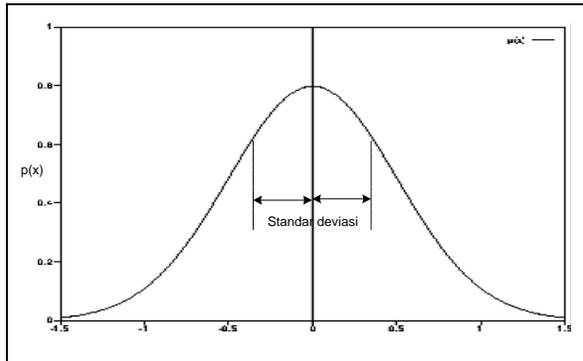
dimana :k = konstanta Boltzman ($1,38 \times 10^{-23} \text{ JK}^{-1}$); T = Temperatur (Kelvin); B = Bandwidth (Hz).

2.4.1 Noise Distribusi Kontinyu

Untuk *noise* dengan distribusi kontinyu, dinyatakan dengan persamaan berikut ini :

$$n(\mu, \sigma; x) = P(x) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left[-\frac{(x-\mu)^2}{2\sigma^2} \right] \dots\dots\dots(14)$$

dimana : μ = mean; σ = standar deviasi.



Gambar 5. Grafik PDF Noise Gaussian

Persamaan diatas umumnya digunakan untuk beberapa sumber *noise*, contohnya *white noise*. Pada umumnya nilai mean dari variabel acak adalah 0 (Sumber : Carlson 1986). Biasanya *noise* akan mempengaruhi amplitudo sinyal informasi, berikut diberikan sinyal informasi yang telah terkena *noise* terdistribusi *gaussian*.

Dengan mensubstitusi $t = (x - \mu) / \sigma$, maka persamaan (14) menjadi :

$$P(t) = \frac{1}{\sigma \sqrt{2\pi}} \exp \left[-\frac{t^2}{2} \right] \dots\dots\dots(15)$$

Jika menginginkan untuk mengetahui nilai probabilitas, misalnya *noise* dari sinyal adalah $n(t)$, dengan nilai $0 \leq t \leq z$, maka didapatkan :

$$P\{0 \leq t \leq z\} = \frac{1}{\sigma \sqrt{2\pi}} \int_0^z \exp^{-t^2/2} dt \dots\dots(16)$$

Contoh distribusi ini adalah distribusi normal atau *gaussian*.

2.4.2 Distribusi Diskrit.

Pada distribusi ini, *Probability Density Function (PDF)* perlu diubah dalam bentuk diskrit, jika variabel acak menggunakan nilai yang sifatnya diskrit untuk pdf, maka grafik pdf akan berbentuk garis lurus, contoh dari distribusi dalam bentuk diskrit adalah distribusi Poisson :

$$P(n) = P\{x = n\} = \frac{a^n}{n!} e^{-a} \dots\dots\dots(17)$$

dimana $n = 0, 1, 2, \dots$

Nilai sebuah variabel acak yang muncul pada peristiwa tidak dapat diprediksi. Hal yang berpengaruh dalam probabilitas suatu variabel acak adalah mean dan standar deviasi. Mean (η) suatu variabel acak didefinisikan sebagai berikut :

$$\eta = \int_{-\infty}^{\infty} xp(x)dx \dots\dots\dots(18)$$

atau dalam distribusi diskrit

$$\eta = \sum_{n=-\infty}^{\infty} np(n) \dots\dots\dots(19)$$

Pada contoh ini, diasumsikan bahwa mean dari distribusi *Gaussian* adalah 0, dan mean dari distribusi Poisson adalah a . Standar deviasi merupakan faktor penentu probabilitas dengan kisaran pada nilai mean. Standar deviasi didapatkan dari persamaan berikut :

$$a^2 = \int_{-\infty}^{\infty} (x - \eta)^2 p(x)dx \dots\dots(20)$$

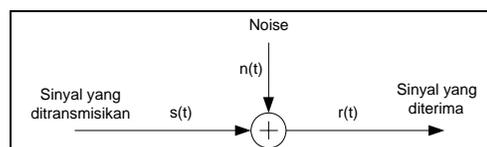
atau ditunjukkan dengan distribusi dalam bentuk diskrit

$$a^2 = \sum_{n=-\infty}^{\infty} (n - \eta)^2 p(n) \dots\dots(21)$$

Kuadrat dari standar deviasi disebut variance. Pada banyak kasus, keberadaan *noise* pada sinyal komunikasi dapat dimodelkan dengan nilai mean untuk variabel acak *Gaussian* sebesar 0, artinya amplitude pada waktu tertentu memiliki pdf sesuai dengan persamaan (14).

2.5 Kanal AWGN

Kanal *Additive White Gaussian Noise (AWGN)* merupakan salah satu model dari *Binary symmetric channel*. Pada gambar dibawah ini diberikan ilustrasi dari sebuah kanal AWGN. Kanal AWGN mempunyai *PDF* atau fungsi kerapatan seperti pada persamaan 14 dengan bentuk grafik seperti gambar 6. Ilustrasi dari kanal ini adalah sebagai berikut :



Gambar 6. Ilustrasi Kanal AWGN

Suatu sinyal yang ditransmisikan $s(t)$ mendapatkan *noise* $n(t)$ dari kanal *AWGN* (*Additive White Gaussian Noise*) sehingga sinyal yang diterima $r(t)$ adalah penjumlahan dari sinyal asli dengan *noise*, kondisi tersebut dapat dirumuskan pada persamaan berikut :

$$r(t) = s(t) + n(t) \dots\dots\dots(22)$$

dimana : $r(t)$ = Sinyal yang diterima ; $s(t)$ = Sinyal informasi yang dikirimkan ; $n(t)$ = Sinyal *noise*.

Untuk menghasilkan suatu bilangan random yang terdistribusi *gaussian*, dapat dicari dengan nilai random terdistribusi *rayleigh*. Jika nilai random terdistribusi *uniform* (nilai 0 sampai dengan 1) adalah U , maka variabel random terdistribusi *rayleigh* R dapat dinyatakan dengan persamaan :

$$R = \sqrt{2\sigma^2 \cdot \ln\left(\frac{1}{1-U}\right)} \dots\dots\dots(23)$$

$$R = \sigma \sqrt{2 \cdot \ln\left(\frac{1}{1-U}\right)} \dots\dots\dots(24)$$

Jika variabel random yang kedua adalah V , maka variabel random terdistribusi *gaussian* adalah sebagai berikut :

$$n(t) = R \cdot \text{Cos } V \dots\dots\dots(25)$$

Pada kanal *AWGN*, sinyal yang melewati kanal bisa mengalami *error* akibat pengaruh *additive noise* $n(t)$ yang mempunyai spektrum power sebesar $N_0/2$ Watt/Hertz. Dari autokorelasi dua buah sinyal terdistribusi *gaussian* dengan nilai mean sama dengan 0, maka dua buah sinyal tersebut bersifat *independent* dengan nilai varian :

$$\sigma^2 = \text{Es} \cdot N_0 \dots\dots\dots(26)$$

Hubungannya dengan SNR adalah besarnya SNR sama dengan E_b/N_0 . Jika diset nilai $E_s = E_b = 1$, maka besarnya SNR adalah sama dengan :

$$\text{SNR} = \frac{1}{N_0} \dots\dots\dots(27)$$

$$\text{SNR} = \frac{1}{\sigma^2} \dots\dots\dots(28)$$

Jadi, nilai standar deviasinya adalah :

$$\sigma = \frac{1}{\sqrt{\text{SNR}}} \dots\dots\dots(29)$$

2.6 Kode Hamming

Menurut Proakis (1989), kode Hamming termasuk ke dalam *linear blok code* (n,k) yang memiliki parameter :

$$n = 2^m - 1 \dots\dots\dots(30)$$

$$k = 2^m - m - 1 \dots\dots\dots(31)$$

$$m = n - k \dots\dots\dots(32)$$

dimana : n = Jumlah bit blok *codeword*; k = Jumlah bit informasi ; m = Jumlah bit *parity*

Berdasarkan jumlah *parity* bit kode Hamming dibedakan menjadi beberapa jenis:

Tabel.2 Jenis Kode Hamming Berdasarkan *Bit Parity*

m	(n,k)
3	(7,4)
4	(15,11)
5	(31,26)
6	(63,57)
7	(127,120)
8	(255,247)

Kode Hamming merupakan kode *single error correction* yaitu hanya ada koreksi kesalahan 1 bit yang mungkin per *codeword*. Struktur kode Hamming merupakan sub kelas pengkodean dengan rumus :

$$(n,k) = (2^m - 1, 2^m - m - 1) \dots\dots\dots(33)$$

dimana nilai m adalah integer positif. Matriks cek *parity* mengandung semua rangkaian dengan panjang m kecuali semua rangkaian yang mengandung 0. perbandingan kode ini dinyatakan dengan :

$$R = \frac{k}{n} = \frac{2^m - m - 1}{2^m - 1} \dots\dots\dots(34)$$

2.6.1 Generator Haming

Untuk menentukan matrik generator dari kode Hamming yaitu pertama dengan menentukan (n) dan (k) . Nilai (n) dan (k) diperoleh dengan persamaan 33. Setelah diperoleh nilai (n) dan (k) lalu menentukan nilai *polynomials* sesuai dengan tabel *polynomials* GF(2).

Tabel 3 *Polynomials* GF(2)

m	Polynomials
3	1 1 0 1
4	1 1 0 0 1
5	1 0 1 0 0 1
6	1 1 0 0 0 0 1
7	1 0 0 1 0 0 0 1

Setelah diketahui nilai dari *polynomial* sesuai dengan tabel *polynomials* GF(2) lalu disusun matrik dengan susunan sebagai berikut:

Contoh generator Hamming dengan $\text{parity}(m) = 3$.

$$n = 2^m - 1 = 7$$

$$k = 2^m - 1 - m = 4$$

- Gunakan *polynomials* dalam tabel 3 sebagai baris pertama dan menambahkan nilai 0 hingga n dengan nama $g(x)$.

$$g(x) = 1 1 0 1 0 0 0$$

- Membuat baris kedua dengan menggeser baris pertama kekanan, prosesnya $x \cdot g(x)$.

$$x \cdot g(x) = 0 1 1 0 1 0 0$$

- Membuat baris ke 3 dengan menggeser baris ke2 ke kanan dengan proses $x^2 \cdot g(x)$. Baris terus dilanjutkan sampai sejumlah k .

$$G = \begin{bmatrix} g(x) \\ x.g(x) \\ x^2.g(x) \\ x^3.g(x) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

Genetator yang terbentuk merupakan generator non sistematis. Untuk merubahnya menjadi sistematis yaitu dengan operasi baris elementer. Matrik dari G sistematis merupakan generator dari kode Hamming (7,4).

$$G = \begin{bmatrix} g(x) \\ x.g(x) \\ x^2.g(x)+g(x) \\ x^3.g(x)+g(x)+x.g(x) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Untuk generator kode Hamming dengan parity yang lain dapat diperoleh dengan cara yang sama.

2.6.2 Encoding kode Hamming

Encoder kode Hamming akan membuat n bit output *codeword* yang terdiri dari k bit informasi dan m bit *parity*. Pada proses *encoding* kode Hamming, n bit output *codeword* dihasilkan dengan mengalikan k bit matriks informasi dengan generator matriks yang mempunyai ukuran matriks n x k. Proses tersebut dapat didefinisikan dengan persamaan sebagai berikut :

$$C = M . G \dots\dots\dots(35)$$

dimana : C = Output *codeword*; M= Matriks bit informasi; G = Matriks generator kode blok

2.6.3 Decoding kode Hamming

Untuk melakukan koreksi kesalahan pada kode Hamming harus diketahui *parity check matrix*nya yang dinyatakan dengan persamaan :

$$H = [I_{3,3} | -P_{4,3}] \dots\dots\dots(36)$$

Contoh matriks cek *parity* untuk kode Hamming (7,4) adalah sebagai berikut :

$$H = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}$$

Parity check matrix digunakan untuk mengetahui bentuk *error* dan *error syndrome* serta memperbaiki bit yang mengalami *error*. Untuk itu diperlukan *coset leader* yang bisa diperoleh dari *standar array*.

Coset leader digunakan untuk bentuk *error* yang jumlahnya sebanyak $2^{(n-k)}$. Untuk kode Hamming (7,4), jumlah *coset leadernya* adalah $2^{(7-4)} = 2^3 = 8$ termasuk *codeword* (0000000). Contoh *standar array* kode Hamming (7,4) adalah sebagai berikut (Wicker Stephen, 1995):

0000000	1010001	1110010	0100011	...	1111111	← Codeword
0000001	1010000	1110011	0100010	...	1111110	
0000010	1010011	1110000	0100001	...	1111101	
0000100	1010101	1110110	0100111	...	1111011	
0001000	1011001	1111010	0101011	...	1110111	
0010000	1000001	1100010	0110011	...	1101111	
0100000	1110001	1010010	0000011	...	1011111	
1000000	0010001	0110010	1100011	...	0111111	

Gambar 7. Standar Array Kode Hamming (7,4)

Dari bentuk *error* kode tersebut dapat dicari bentuk *syndrome error* dengan persamaan :

$$S = e . H^T \dots\dots\dots(37)$$

dimana :S= Bentuk *syndrome error*; e= Bentuk *error*; H^T= Transpose *parity check matrix*

Contoh bentuk *error* dan *syndrome error* kode Hamming (7,4) dapat dilihat pada table 4.

Untuk mengetahui *error* dalam *codeword*, dilakukan dengan membandingkan *syndrome error* dari kode masing-masing jenis Hamming dengan *syndrome error codeword* yang diterima. *Syndrome error* diperoleh dengan mengalikan *codeword* yang diterima dengan transpose matriks cek *parity*.

Tabel 4. Bentuk *Error* dan *Syndrome Error* Pada Hamming (7,4)

No.	Bentuk <i>error</i> (e)	<i>Syndrome error</i> (S)
1	0000000	000
2	0000001	101
3	0000010	111
4	0000100	011
5	0001000	110
6	0010000	001
7	0100000	010
8	1000000	100

Untuk memperbaiki *error*, dilakukan dengan menjumlahkan *codeword* yang diterima dengan bentuk *error* yang sesuai dengan *syndrome error*. Prosesnya dapat dilihat dengan persamaan :

$$v = e + r \dots\dots\dots(38)$$

dimana :v = *Codeword* setelah diperbaiki; e = Bentuk *error*; r = *Codeword* yang diterima. Untuk mendapatkan kembali bit-bit informasi dalam *codeword* dilakukan dengan membuang semua bit-bit *paritinya*.

2.7 Kemampuan Kode Hamming

Kode Hamming memiliki kemampuan mendeteksi dan koreksi kesalahan / *error* bit yang ditransmisikan, antara lain :

1. Mendeteksi semua kesalahan bit tunggal dan ganda. Dilakukan dengan membandingkan *codeword* hasil *encoding* dengan *codeword* hasil deteksi

decoding. Kemampuan mendeteksi *error* pada kode Hamming dapat dinyatakan dengan rumus :

$$e_d = d_{\min} - 1 \dots\dots\dots (39)$$

2. Mengkoreksi semua kesalahan bit tunggal. Jika terdeteksi adanya kesalahan bit dalam blok *codeword* pada proses *decoding*, maka dengan operasi XOR akan diperbaiki sebanyak 1 bit *error* yang terdeteksi. Kemampuan koreksi *error* Hamming dinyatakan dengan rumus :

$$e_c = \frac{d_{\min} - 1}{2} \dots\dots\dots (40)$$

2.8 Performansi Kode Blok

Menurut Oberg (2001), Probabilitas bit *error* dari kode blok pada kanal AWGN dapat dinyatakan dengan persamaan sebagai berikut :

$$P(e) = Q(\sqrt{2 R_c (E_b/N_0)}) \dots\dots\dots (41)$$

Nilai dari P(e) merupakan sinyal *bandpass* yaitu sinyal yang telah termodulasi. Untuk persamaan Q (Q *function*) dapat dinyatakan dengan persamaan :

$$Q(x) = 0,5 \cdot \text{erfc}\left(\frac{x}{\sqrt{2}}\right) \dots\dots\dots (42)$$

dimana : P(e) = Probabilitas bit *error*; Q = Q *function*; R_c = *Code rate* kode blok; E_b/N₀ = *Energi Bit to Noise Ratio* atau SNR (*Signal to Noise Ratio*)

2.9 Probability of Error (P(e)) dan Bit Error Rate (BER)

P(e) merupakan suatu nilai perhitungan secara matematis dari probabilitas bit *error* yang diharapkan pada suatu sistem. Sedangkan, BER adalah suatu nilai yang telah tercatat dalam suatu sistem yang menunjukkan nilai performansi dari sistem tersebut.

3. HASIL DAN PEMBAHASAN

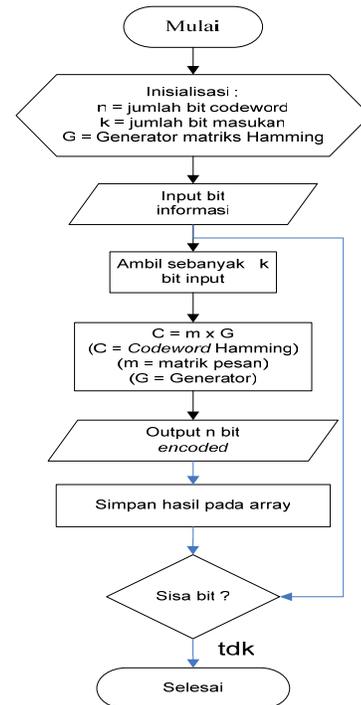
Berikut diuraikan mengenai proses simulasi *error control coding* dengan pengkodean Hamming baik pada *encoder* dan *decoder*. Disini juga membandingkan unjuk kerja dari model simulasi tanpa proses pengkodean dan menggunakan beberapa jenis kode Hamming, yang dibedakan atas jumlah bit paritinya.

Beberapa variabel yang digunakan untuk melakukan analisis pada simulasi antara lain :

1. Input informasi berupa : a. Bit acak sejumlah 1000.000 bit. b. String, c. Gambar digital (berwarna 128 x 128 *pixel*).
2. SNR = -10 dB sampai dengan 10 dB.

Besarnya *noise* yang dibangkitkan tergantung dari besarnya nilai mean dan varian sesuai dengan variasi SNR. Analisa hasil simulasi dilakukan dengan membandingkan masing-masing jenis kode Hamming untuk setiap jenis input yang diberikan. Input bit acak dilakukan dengan membangkitkan sejumlah 10⁶ bit acak, input string dilakukan dengan

menginputkan kalimat “*Program untuk mencari error pada suatu sistem telekomunikasi dengan menggunakan pengkodean hamming (7,4), hamming (15,11), hamming (31,26), hamming (63,57), hamming (127,120)*” yang semuanya terdiri dari 1218 bit. Sedangkan untuk input gambar, input gambar digital berwarna (RGB) adalah (128 x 128 *pixel*) x 3 warna x 8 (bit / warna / *pixel*) = 393216 bit.



Gambar 8. Flowchart *Encoding* Kode Hamming

3.1 Channel Encoder Haming

Proses pembuatan simulasi pada bagian *encoder* dapat dilihat pada alur analisis gambar 10

3.2 Modulator

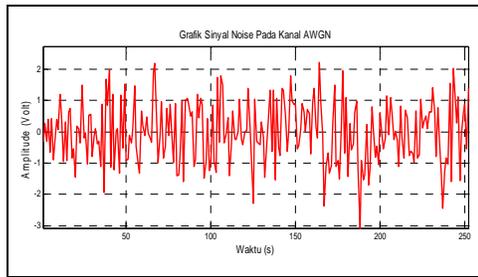
Teknik modulasinya menggunakan modulator BPSK dengan sinyal yang dihasilkan seperti berikut:

$$S_1(t) = - \sqrt{\frac{E_b}{N_0}} \sin(2\pi f_c t), \text{ untuk bit 0}$$

$$S_2(t) = + \sqrt{\frac{E_b}{N_0}} \sin(2\pi f_c t), \text{ untuk bit 1}$$

3.3 Kanal AWGN

Pada kanal AWGN, besarnya *noise* yang dibangkitkan tergantung dari besarnya mean dan varian sesuai dengan perubahan SNR. Gambar 8. adalah contoh *noise* yang dibangkitkan dan sinyal *noisy* (sinyal informasi yang sudah bercampur dengan *noise*) pada kanal AWGN untuk nilai varian *gaussian* (σ²) = 1.



Gambar 9. Sinyal Noise Pada Kanal AWGN Dengan SNR = -10dB

3.4 Demodulator

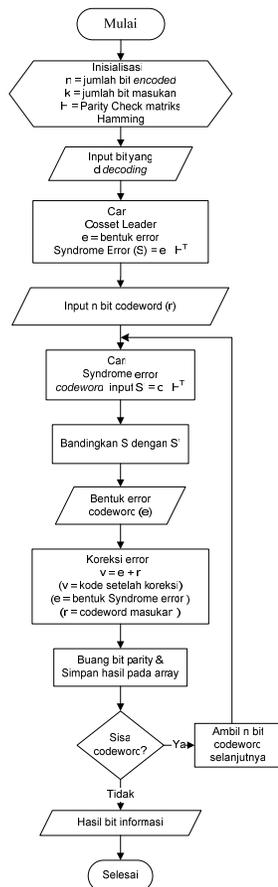
Proses pada demodulator dilakukan dengan mengalikan sinyal yang diterima $r(t)$ dengan sinyal carrier $c(t)$. Selanjutnya sinyal tersebut dikodekan seperti berikut:

$$\int_0^{kTc} r(t)c(t)dt = x$$

$x = 1$ jika $x \geq 0$, dan $x = 0$ jika $x < 0$

3.5 Channel Decoder Hamming

Alur analisis proses simulasi pada sisi receiver dapat dilihat pada gambar berikut :



Gambar 10. Flowchart Decoding Kode Hamming

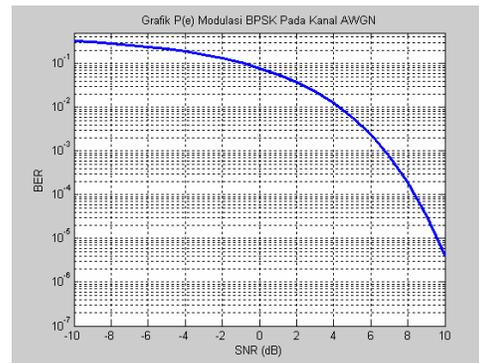
3.6 Perhitungan Probability of Error (P(e))

Probability of error $P(e)$ untuk modulasi BPSK pada kanal AWGN dapat dihitung dengan persamaan sebagai berikut :

$$P(e) = 0,5 \operatorname{erfc} \left[\frac{\sqrt{2} \cdot \text{SNR}}{\sqrt{2}} \right]$$

$$\text{atau } P(e) = 0,5 \operatorname{erfc} \left(\sqrt{\text{SNR}} \right)$$

Nilai probability of error $(p(e))$ modulasi BPSK pada kanal AWGN dapat dilihat pada gambar grafik berikut :



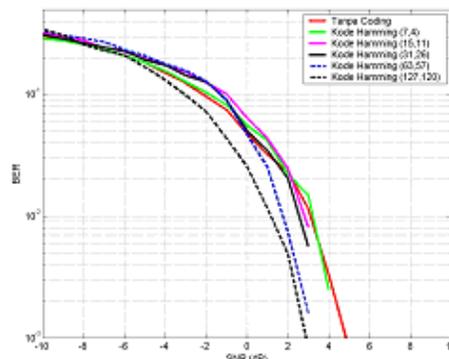
Gambar 11. Grafik Hasil Perhitungan $p(e)$ Modulasi BPSK

3.6 Analisa BER Untuk Semua Simulasi dan Semua Input

Berikut akan disampaikan analisa BER dan hasil dari simulasi dari semua tipe kode Hamming yang diimplementasikan untuk semua input yang diberikan yaitu input acak, string dan gambar digital.

3.6.1 Hasil Simulasi Analisa BER untuk Input Acak

Gambar 14 menunjukkan bahwa, kinerja dari masing-masing kode Hamming mengalami peningkatan sesuai dengan penambahan jumlah bit parity. Nilai BER terkecil untuk SNR yang sama diperoleh saat menggunakan kode Hamming (127,120) atau kode Hamming dengan bit parity = 7.



Gambar 12. Grafik Hasil BER Seluruh Simulasi Dengan Input Acak

Disini juga terlihat bahwa penggunaan kode Hamming meningkatkan kinerja suatu sistem telekomunikasi.

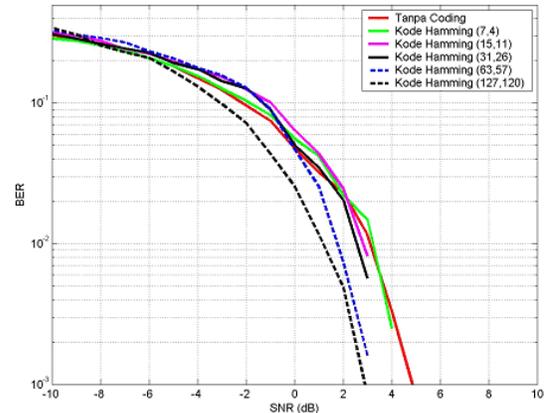
3.6.2 Hasil Simulasi dan Analisa BER Untuk Input String pada Setiap Kode Hamming

Berikut ini adalah perbandingan jumlah bit *error* dan BER untuk input string pada hasil simulasi:

Tabel 5. Perbandingan Jumlah Bit Error Pada Input String

SNR (dB)	Tanpa Pengkodean	Kode Hamming				
		(7,4)	(15, 11)	(31, 26)	(63, 57)	(127,120)
-10	369	351	386	377	403	417
-9	345	337	365	349	373	372
-8	319	312	336	321	354	312
-7	284	287	295	298	329	280
-6	252	254	272	278	285	254
-5	222	223	240	235	253	204
-4	182	190	214	212	217	160
-3	151	154	187	174	191	120
-2	117	126	152	155	156	88
-1	91	100	123	109	110	53
0	58	68	78	61	57	31
1	39	51	53	42	31	14
2	28	27	30	25	9	6
3	14	18	10	7	2	1
4	4	3	0	0	0	0
5	1	0	0	0	0	0
6	0	0	0	0	0	0

Pada kode Hamming (127,120) terjadi bit error hanya 1 bit pada SNR 3 dB. Jumlah bit error yang lebih banyak diberikan oleh kode Hamming yang lainnya.

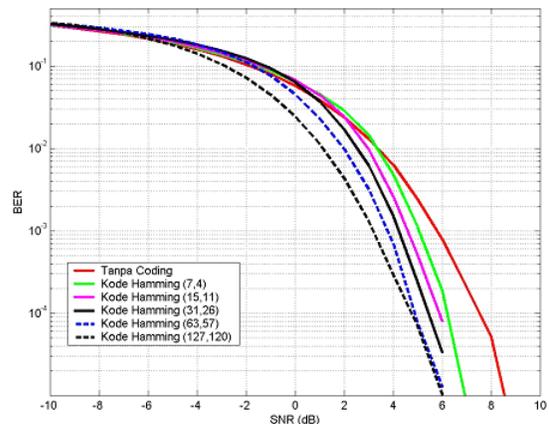


Gambar 13. Grafik Hasil BER Seluruh Simulasi Dengan Input String

Gambar 15 menunjukkan peningkatan kinerja dari kode Hamming sesuai penambahan bit *parity* BER terendah pada keadaan SNR yang sama dihasilkan oleh kode Hamming (127,120) atau pada kode Hamming dengan bit *parity* = 7.

3.6.3 Analisa BER Untuk Input Gambar pada Masing-Masing Kode Hamming

Gambar 16 menunjukkan perbandingan jumlah *bit error* dan BER untuk input gambar pada semua



Gambar 14. Grafik Hasil BER Seluruh Simulasi Dengan Input Gambar

Dari grafik gambar 16 diatas dapat dilihat kinerja dari masing-masing kode Hamming mengalami peningkatan sesuai dengan perubahan penambahan bit *parity*. BER paling kecil pada keadaan SNR yang sama diperoleh pada saat menggunakan kode Hamming (127,120). Nilai BER pada saat tanpa menggunakan pengkodean lebih besar dibandingkan dengan pada saat menggunakan kode Hamming manapun.

4. KESIMPULAN

Dari hasil analisa dan pembahasan dapat disimpulkan sebagai berikut :

1. Secara garis besar proses *error control coding* pada kode Hamming ada dua yaitu *encoding* dan *decoding*. Pada *encoder* bit informasi dikalikan dengan matrik generator, menghasilkan *codeword* yang akan ditransmisikan. Sedangkan pada *decoder* proses pertama menentukan matrik *parity* kemudian menentukan *syndrome error*. Terdapat dua jenis *syndrome error* yaitu *syndrome error* oleh kode Hamming dan *syndrome error* yang berasal dari *codeword* yang diperoleh dari perkalian *transpose* matrik *parity* dengan *codeword*. Selanjutnya membandingkan kedua *syndrome error* tersebut sehingga diperoleh bentuk *error* dari bit informasi yang dikirimkan. Kemudian bentuk *error* tersebut dijumlahkan dengan *codeword* yang diterima pada *decoder*. Akhirnya, membuang bit terakhir sesuai dengan jumlah bit *parity*nya hingga diperoleh bit informasi semula.
2. Kinerja dari pengkodean Hamming mengalami peningkatan seiring dengan penambahan bit *parity*. Dengan bit *parity* 3, 4, 5, 6, dan 7 untuk ketiga input yang diberikan (input bit acak, string dan gambar), semakin bertambah bit *parity* diperoleh nilai BER semakin kecil berarti kinerja dari suatu kode Hamming semakin meningkat.

5. DAFTAR PUSTAKA

- [1]. Bateman. 1998. **Digital Communications**. Prentice Hall, Inc: Singapura.
- [2]. Carlson. 1986. **Communication Systems 3th Edition, An Introduction to Signals and Noise in Electrical Communication**. Mc.Graw-Hill. Inc: Singapura.
- [3]. Daniel, J. 1983. **Error Control Coding : Fundamentals and Applications**. Prentice Hall International (UK): USA.
- [4]. Keiser Gerd. 2000. **Optical Fiber Communication, 3rd Edition**. McGraw-Hill. Inc: Singapura.
- [5]. Proakis, 1989. **Digital Communication, International Edition**. McGraw-Hill: New York.
- [6]. Rhee Man Young. 1989. **Error Correction Coding Theory**. McGraw-Hill: Singapura.
- [7]. Oberg Tommy. 2001. **Modulation, Detection and Coding**. John Wiley & Sons, Ltd: England.
- [8]. Wicker Stephen. 1995. **Error Control Systems for Digital Communication and Storage**. Prentice Hall: New Jersey.