

IMPLEMENTASI METODE *HAMMING CODE* DALAM MENDETEKSI DAN MENGOREKSI KESALAHAN TRANSMISI DATA PADA TRANSMISI DIGITAL

Devi Lianda¹, Amir D², Nasri³

^{1,2,3}Prodi Teknologi Rekayasa Jaringan Telekomunikasi
Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe
Email: devilianda995@gmail.com¹, amird@pnl.ac.id², nasrimt@gmail.com³

ABSTRAK

Dalam bidang telekomunikasi, kesalahan data yang terjadi pada saat transmisi dari pengirim ke penerima, dapat mengakibatkan perubahan nilai data yang dikirim, sehingga informasi yang diterima tidak sesuai dengan informasi aslinya. Oleh karena itu, diperlukan metode yang dapat mendeteksi serta mengoreksi kesalahan data. Pada penelitian ini dilakukan deteksi dan koreksi kesalahan data menggunakan metode *Hamming Code*. Serta menghitung *Reability* dan Efisiensi *Hamming Code*. Untuk mengamati deteksi dan koreksi kesalahan serta menghitung *Reability* dan efisiensi di uji dengan 7 skenario. Skenario pertama mensimulasikan 1 bit kesalahan data, skenario kedua mensimulasikan 2 bit kesalahan data, skenario ketiga mensimulasikan 3 bit kesalahan data, skenario keempat mensimulasikan 4 bit kesalahan data, skenario kelima mensimulasikan 5 bit kesalahan data, skenario keenam mensimulasikan 6 bit kesalahan data, dan skenario ketujuh mensimulasikan 7 bit kesalahan data. Hasil pengujian diperoleh bahwa pada skenario 1, jumlah kesalahan data yang terdeteksi sebanyak 1 bit. Sementara untuk skenario 2 sampai 6, jumlah kesalahan data terdeteksi sebanyak 1 bit. Sedangkan skenario 7, jumlah kesalahan data yang terdeteksi sebanyak 0 bit. Selanjutnya hasil perhitungan *Reability* untuk 1 bit kesalahan data adalah 1. Nilai *Reability* untuk 2 bit kesalahan data adalah 0.5. Nilai *Reability* untuk 3 bit kesalahan data adalah 0.33. Nilai *Reability* untuk 4 bit kesalahan data adalah 0.25. Nilai *Reability* untuk 5 bit kesalahan data adalah 0.2. Nilai *Reability* untuk 6 bit kesalahan data adalah 0.17. Serta nilai *Reability* untuk 7 bit kesalahan data adalah 0. Hasil perhitungan Efisiensi untuk 2 bit data asli sebesar 50%. Nilai Efisiensi untuk 4 bit data asli sebesar 50%. Nilai Efisiensi untuk 8 bit data asli sebesar 66.67%. Nilai Efisiensi untuk 16 bit data asli sebesar 76.19%. Nilai Efisiensi untuk 32 bit data asli sebesar 84.21%.

Kata kunci : *Hamming Code, Deteksi dan Koreksi Kesalahan Data, Reability, Efisiensi*

I. PENDAHULUAN

Dalam bidang telekomunikasi, kesalahan data yang terjadi pada saat transmisi dari pengirim ke penerima, dapat mengakibatkan perubahan nilai data yang dikirim, sehingga informasi yang diterima tidak sesuai dengan informasi aslinya. Oleh karena itu, diperlukan metode yang dapat mendeteksi serta mengoreksi kesalahan data [1][2][3].

Tujuan dari deteksi dan koreksi kesalahan data ini agar data yang ditransmisikan oleh pengirim dapat sampai ke penerima sesuai dengan informasi aslinya. Ada beberapa metode deteksi dan koreksi kesalahan data yang biasa digunakan dalam dunia telekomunikasi, diantaranya; *Hamming Code, Cyclic Code, BCH code, dan Reed-Solomon code*.

Dari keempat metode deteksi dan koreksi kesalahan data tersebut, maka pada penelitian skripsi ini akan diamati kinerja *Hamming Code* dalam mendeteksi dan mengoreksi kesalahan data yang ditransmisikan dari pengirim ke penerima. Metode deteksi dan koreksi kesalahan data *Hamming Code* ini dilakukan dalam bentuk simulasi menggunakan MATLAB. Kinerja deteksi dan koreksi kesalahan data *Hamming Code* akan diukur menggunakan parameter *Reability* dan Efisiensi. Dengan parameter ini deteksi dan koreksi kesalahan data saat transmisi dapat diketahui tingkat keandalannya.

Hal ini menarik untuk diteliti, oleh sebab itu pada penelitian skripsi ini pengusul mengajukan tema sebagai

bahan penelitian skripsi dengan judul “Implementasi Metode *Hamming Code* Dalam Mendeteksi dan Mengoreksi Kesalahan Transmisi Data Pada Transmisi Data Digital”. Kegiatan penelitian ini dilakukan di Laboratorium Jaringan Komputer Program Studi Teknologi Rekayasa Jaringan Telekomunikasi Jurusan Teknik Elektro Politeknik Negeri Lhokseumawe.

II. TINJAUAN PUSTAKA

A. Data

Data adalah kumpulan informasi atau fakta yang diperoleh melalui proses pencarian dan pengamatan yang akurat. Data dapat berupa kata-kata, kalimat, simbol, angka, dan elemen lainnya.

Penggunaan dan pemanfaatan data sudah mencakup banyak aspek. Data biasanya terdiri dari beberapa elemen data (*data item*). Elemen data adalah unit terkecil dari data yang ada artinya bagi pihak yang menggunakannya (*user*). Dalam suatu sistem basis data, elemen data ini disebut dengan *Field* [4].

B. Bit Kontrol (*Parity*)

Bit *parity* merupakan bilangan biner yang ditambahkan pada urutan bit-bit data yang ditransmisikan. Penambahan bit *parity* ini bertujuan untuk memastikan bahwa bit-bit yang ditransmisikan tidak mengalami perubahan nilai setelah sampai di penerima.

Bit *parity* digunakan sebagai bentuk paling sederhana dari kode pendeteksi kesalahan. Bit umumnya diterapkan ke unit terkecil dari protokol komunikasi, biasa oktat 8-bit, meskipun mereka juga diterapkan secara terpisah ke seluruh *string* pesan bit. Bit *parity* memastikan bahwa jumlah total 1-bit dalam *string* genap atau ganjil. Dengan demikian, ada dua jenis bit *parity*, bit *parity* genap dan bit *parity* ganjil [5].

1. Jenis Bit *Parity*

Terdapat dua jenis bit *parity* yang dapat ditambahkan pada urutan bit-bit data, yaitu:

a. *Parity* genap (*Even parity*)

Bit *parity* di set menjadi 1 apabila jumlah angka 1 dalam kesatuan tersebut (tidak termasuk bit *parity*) menjadi ganjil, sehingga menjadikan jumlah bit dalam kesatuan tersebut (termasuk bit *parity*) menjadi genap.

b. *Parity* ganjil (*Odd parity*)

Bit *parity* di set menjadi 1 apabila jumlah angka 1 dalam kesatuan bit tersebut (tidak termasuk bit *parity*) adalah genap, sehingga menjadikan jumlah bit dalam kesatuan tersebut (termasuk bit *parity*) menjadi ganjil.

2. Cara Kerja Bit *Parity*

Pihak pengirim akan menambahkan 1 bit *parity* pada data, untuk menggambarkan karakteristik dari data tersebut. Nilai dari bit *parity* (1 atau 0) tidak diperbolehkan secara sembarang. Dalam proses transmisi data tadi dikirim bersamaan (data dan bit *parity*). Pada terminal penerimaan, data dibaca dan didekodisasi dengan cara yang sama seperti saat menentukan nilai bit *parity* disisi pengirim. Lalu hasil dekodisasi tadi dibandingkan dengan bit *parity* yang dibawakan oleh pengirim. Apabila hasil pembacaan (dekodisasi) data terkirim sama dengan bit *parity*nya, maka data tersebut dapat dianggap benar. Sebaliknya, apabila diperoleh perbedaan nilai antara hasil dekodisasi dengan bit *parity*, maka data dapat di klasifikasi sebagai data yang *error*. Terminal penerima akan mengirim *request* pada terminal pengirim untuk mengirim ulang data yang *error*.

C. Deteksi dan Koreksi Kesalahan Data

Deteksi dan koreksi kesalahan data adalah dua proses berbeda namun saling berhubungan yang dapat terjadi ketika informasi dikirim. Deteksi kesalahan data adalah metode yang dapat mendeteksi apakah data telah rusak saat dikirim. Koreksi kesalahan data merupakan langkah yang lebih baik daripada deteksi kesalahan karena mencoba mengembalikan data pada keadaan semula.

Konsep dari *error detection and correction* adalah untuk mengurangi dampak kesalahan (*error*) yang bisa diakibatkan oleh masalah pengamatan dari informasi digital terhadap kesalahan yang kerap muncul karena pada saat data proses transmisi tersebut, justru mengakibatkan sistem menjadi salah pengertian melakukan interpretasi

terhadap simbol data-data yang akan diterjemahkan untuk kemudian diproses menjadi sebuah pesan [6]. Adapun beberapa metode yang dilakukan adalah :

1. *Backward Error Control* (BEC)

Metode dimana perangkat sebuah proses pengiriman akan segera mengirimkan sinyal kepada perangkat pengirim untuk melakukan pengiriman sinyal kepada.

2. *Forward Error Control* (FEC)

Metode dimana sebelum proses pengiriman data dilakukan terlebih dahulu dikodekan dengan suatu pembangkit kode (*encoder*), kemudian dikirim ke perangkat penerima.

D. *Hamming Code*

Algoritma *Hamming Code* ditemukan oleh Richard W. Hamming pada tahun 1940-an. Algoritma *Hamming Code* merupakan salah satu metode pendeteksi *error* yang mampu untuk mendeteksi beberapa *error*, namun hanya mampu mengoreksi satu *error*. Algoritma pendeteksi *error* ini sangat cocok digunakan pada situasi dimana terdapat beberapa *error* yang teracak [7][8]. *Hamming Code* bekerja dengan memberikan kode biner tambahan pada data yang berfungsi sebagai bit-bit pendeteksi kesalahan (bit *parity*). Bit-bit ini akan memberikan gambaran mengenai kondisi data yang sesungguhnya sehingga kesalahan (*error*) yang terjadi dapat dideteksi dengan mudah, karena terdapat suatu keterkaitan antara data dengan bit-bit pendeteksi kesalahan [9]. Metode *Hamming Code* menggunakan operasi logika *Exclusive-OR* (XOR) dalam proses pendeteksian *error* maupun proses pengoreksian *error*, sedangkan *input* dan *output* data dari Metode *Hamming Code* berupa bilangan biner [10].

E. *Reability*

Reability merupakan parameter untuk mengukur kehandalan atau kemampuan *Hamming Code* dalam mendeteksi dan memperbaiki kesalahan dalam data yang dikirim. *Reability* umumnya dilambangkan dengan R, dihitung sebagai rasio antara jumlah bit yang dapat dideteksi dengan total jumlah bit. Secara matematis, rumus *reability Hamming Code* dapat dinyatakan sebagai berikut:

$$R = \frac{\text{Jumlah bit } error \text{ yang dapat dideteksi}}{\text{Jumlah bit } error \text{ sebenarnya}} \quad (1)$$

Semakin tinggi nilai R, semakin baik kehandalan *Hamming Code* dalam mendeteksi dan mengoreksi kesalahan data. Hal ini menunjukkan bahwa sistem dapat dengan efektif mengidentifikasi dan memperbaiki kesalahan yang terjadi saat data dikirim. Sebaliknya, semakin rendah nilai R, maka semakin buruk kehandalan *Hamming Code* dalam mendeteksi dan mengoreksi kesalahan data. Hal ini menunjukkan bahwa sistem dapat mengalami kegagalan dalam mentransmisikan data dengan benar.

F. Efisiensi

Efisiensi merupakan parameter untuk mengukur seberapa baik *Hamming Code* memanfaatkan bit untuk transmisi data. Efisiensi diukur berdasarkan rasio bit data terhadap total bit dalam blok kode, termasuk bit paritas. Semakin tinggi rasio bit data, semakin efisien *Hamming Code* dalam menggunakan ruang bit untuk data. Secara matematis, rumus efisiensi *Hamming Code* dapat dinyatakan sebagai berikut:

$$E = \frac{k}{n} \times 100\% \tag{2}$$

Dimana:

E = Efisiensi *Hamming Code*

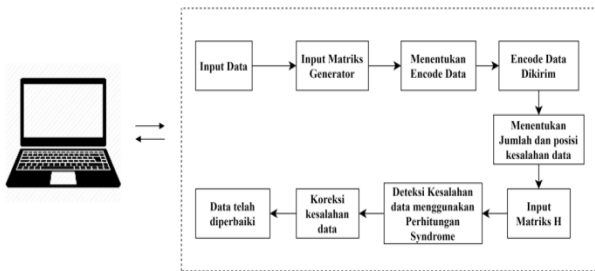
k = Jumlah bit data

n = jumlah total bit setelah penambahan bit *parity*

G. METODOLOGI

A. Alat dan Bahan

Sebelum melakukan penelitian, maka dilakukan persiapan alat dan bahan yang dibutuhkan untuk menunjang proses penelitian seperti gambar 1.



Gbr 1 Diagram Blok Proses Deteksi dan Koreksi Kesalahan Data

Uji deteksi dan koreksi kesalahan transmisi data, dilakukan uji dengan 7 macam skenario. Skenario pertama mensimulasikan 1 bit kesalahan data, skenario kedua mensimulasikan 2 bit kesalahan data, skenario ketiga mensimulasikan 3 bit kesalahan data, skenario keempat mensimulasikan 4 bit kesalahan data, skenario kelima mensimulasikan 5 bit kesalahan data, skenario keenam mensimulasikan 6 bit kesalahan data, dan skenario ketujuh mensimulasikan 7 bit kesalahan data.

Pada skenario ini, bit-bit kesalahan data akan dideteksi dan dikoreksi menggunakan metode *Hamming Code* dengan jumlah bit kesalahan yang berbeda, yaitu 1, 2, 3, 4, 5, 6, dan 7 bit. Hasilnya dibahas pada tabel I.

B. Teknik Pengumpulan Data

Teknik pengumpulan data dilakukan dengan metode eksperimen, seperti pada gambar 1 data diperoleh melalui proses simulasi deteksi dan koreksi kesalahan transmisi data yang dimulai dengan menginput data. Menginput matriks generator, lalu menentukan *encode data* melalui perhitungan matriks generator dikalikan dengan data asli. Selanjutnya *encode data* dikirim menghasilkan *received data*, lalu mensimulasikan bit kesalahan data supaya dapat dideteksi. Selanjutnya, menginput matriks H. Lalu, deteksi kesalahan data dengan menggunakan perhitungan *syndrome*. Apabila kesalahan data terdeteksi, maka

kesalahan data akan dikoreksi selanjutnya data dikirimkan ke penerima.

Adapun data-data yang dikumpulkan berupa data deteksi dan koreksi kesalahan, data-data yang dikumpulkan dapat dilihat pada lampiran 2, 3, 4, 5, 6, 7, dan 8.

C. Teknik Pengolahan Data

Data-data hasil pengukuran diolah dan ditampilkan dalam bentuk tabel, antara lain; pengolahan data tentang deteksi dan koreksi kesalahan data, pengolahan data perhitungan *Reability* dan Efisiensi.

D. Metode Analisis Data

Adapun metode analisis yang digunakan pada penelitian ini menggunakan metode kuantitatif, dimana data-data diukur menggunakan persamaan *Reability* dan persamaan Efisiensi. Kemudian data tersebut dianalisis dan ditampilkan dalam data hasil perhitungan *Reability* dan Efisiensi seperti dijelaskan pada tinjauan pustaka dan hasilnya diperlihatkan pada borang data tabel II dan III.

III. HASIL DAN PEMBAHASAN

A. Pengujian Deteksi dan Koreksi Kesalahan Data

Berdasarkan pengujian deteksi dan koreksi kesalahan data menggunakan metode *Hamming Code*, pengujian dilakukan dengan jumlah bit kesalahan data yang berbeda, yaitu; 1, 2, 3, 4, 5, 6, dan 7 bit. Hasilnya dapat dilihat pada lampiran 2, 3, 4, 5, 6, 7, dan 8.

TABEL I
Hasil simulasi deteksi dan koreksi kesalahan data

Data	Encode Data	Jumlah Kesalahan Data	Posisi Kesalahan Data	Received Data	Jumlah Deteksi Kesalahan Data	Koreksi Kesalahan Data
1011	1011001	1	3	1001001	1	1011001
		2	2 dan 3	1101001	1	1101101
		3	2, 4, dan 7	1110000	1	1111000
		4	2, 3, 4, dan 5	1100101	1	1101101
		5	1, 2, 3, 4, dan 5	0100101	1	1011001
		6	1, 2, 3, 5, 6, dan 7	0101110	1	0101010
		7	1, 2, 3, 4, 5, 6, dan 7	0100110	0	0100110

Berdasarkan tabel 1, pengujian deteksi dan koreksi kesalahan data dengan skenario yang dijelaskan di atas, diketahui bahwa deteksi dan koreksi kesalahan data yang berhasil berada pada pengujian dengan jumlah kesalahan data sebanyak 1 bit, jumlah kesalahan data yang terdeteksi sebanyak 1 bit, karena hasil perhitungan *syndrome* sebesar [1 0 1]. Pada pengujian dengan jumlah kesalahan data sebanyak 2, 3, 4, 5, dan 6 bit, jumlah kesalahan data terdeteksi sebanyak 1 bit. Sedangkan deteksi dan koreksi kesalahan data yang tidak berhasil berada pada pengujian dengan jumlah kesalahan data sebanyak 7 bit, jumlah kesalahan data yang terdeteksi sebanyak 0 bit, karena hasil perhitungan *syndrome* sebesar [0 0 0].

Hasil pengujian memberikan informasi bahwa *Hamming Code* dapat mendeteksi dan mengoreksi kesalahan transmisi data dengan kesalahan tunggal. Serta dapat mendeteksi kesalahan data dengan kesalahan lebih dari 1 bit, tetapi tidak dapat mengoreksi kesalahan data tersebut.

B. *Reability*

Reability merupakan parameter untuk mengukur kehandalan atau kemampuan *Hamming Code* dalam mendeteksi dan memperbaiki kesalahan dalam data yang dikirim. *Reability* dari *Hamming Code* dapat dihitung menggunakan persamaan 1.

TABEL II
Reability Hamming Code

Parameter yang diketahui		Parameter yang dihitung
Jumlah bit <i>error</i> sebenarnya	Jumlah bit <i>error</i> yang dideteksi	<i>Reability</i>
1	1	1
2	1	0.5
3	1	0.3333
4	1	0.25
5	1	0.2
6	1	0.1667
7	0	0

Berdasarkan tabel 2, dapat diketahui bahwa nilai *Reability* untuk 1 bit kesalahan data adalah 1, yang berarti semua kesalahan berhasil dideteksi. Namun, nilai *Reability* menurun secara signifikan untuk kesalahan lebih dari 1 bit. Nilai *Reability* untuk 2 bit kesalahan data adalah 0.5. Nilai *Reability* untuk 3 bit kesalahan data adalah 0.33. Nilai *Reability* untuk 4 bit kesalahan data adalah 0.25. Nilai *Reability* untuk 5 bit kesalahan data adalah 0.2. Nilai *Reability* untuk 6 bit kesalahan data adalah 0.17. Serta nilai *Reability* untuk 7 bit kesalahan data adalah 0. Nilai-nilai tersebut, menunjukkan bahwa semakin banyak jumlah bit kesalahan data, maka kehandalan *Hamming Code* dalam mendeteksi dan mengoreksi kesalahan data semakin rendah.

C. Efisiensi

Efisiensi merupakan parameter untuk mengukur seberapa baik *Hamming Code* memanfaatkan bit untuk transmisi data. Efisiensi dari *Hamming Code* dapat dihitung menggunakan persamaan 2.

TABEL III
Efisiensi Hamming Code

Parameter yang diketahui		Parameter yang dihitung
k	n	E (%)
2	4	50
4	8	50
8	12	66.67
16	21	76.19
32	38	84.21

Berdasarkan Tabel 3, dapat diketahui bahwa nilai Efisiensi untuk 2 bit data asli sebesar 50%. Nilai Efisiensi untuk 4 bit data asli sebesar 50%. Nilai Efisiensi untuk 8 bit data asli sebesar 66.67%. Nilai Efisiensi untuk 16 bit data asli sebesar 76.19%. Nilai Efisiensi tertinggi sebesar 84.21% dengan jumlah bit data asli sebanyak 32 bit. Nilai-nilai tersebut, menunjukkan bahwa Semakin banyak bit

yang ditransmisikan, maka semakin efisien *Hamming code* memanfaatkan bit untuk transmisi data.

IV. KESIMPULAN

Berdasarkan analisa yang telah dilakukan dapat disimpulkan sebagai berikut:

1. *Hamming Code* dapat mendeteksi dan mengoreksi kesalahan transmisi data dengan kesalahan tunggal. Serta dapat mendeteksi kesalahan data dengan kesalahan lebih dari 1 bit, tetapi tidak dapat mengoreksi kesalahan data tersebut.
2. Semakin banyak jumlah bit yang salah, maka kehandalan *Hamming Code* dalam mendeteksi dan mengoreksi kesalahan data semakin rendah.
3. Semakin banyak bit yang ditransmisikan, maka semakin efisien *Hamming code* memanfaatkan bit untuk transmisi data.

REFERENSI

- [1] Achmad Fauzi, Rizka Putri Rahayu, *Implementation Of The Hamming Code Method In Bit Data Improvement Transmission Procces*. Teknik Informatika-STMik Kaputama. 2021.
- [2] Nur Hamid, Nurul Jannah, *Implementasi Teori Koding Dengan Kode Hamming Yang Diperluas Pada Steganografi*. Pendidikan Matematika-Universitas Nurul Paiton Probolinggo. 2023.
- [3] Achmad Fauzi, Nurhayati, dkk, *Bit Error Detection And Correction With Hamming Code Algorithm*. Jurusan Teknik Komputer-Institut Teknologi Medan.
- [4] Mahyuzar Masri, Hermansyah Alam, dkk, *Simulasi Byte Error Correction Dengan Menggunakan Hamming Code*". Teknik Informatika-Institut Teknologi Medan. 2019.
- [5] Misdem Sembiring, Fauzi Haris Simbolon, "Perancangan Perangkat Lunak Pembelajaran Algoritma Hamming Code Dalam Mencari Bit Error Pada Komunikasi Data". Teknologi Informasi dan Komunikasi-AMIK Medan Business Polytechnic. 2021.
- [6] Ahmad Alfi Albar Lubis, Dr. Poltak Sihombing, M.Kom, dkk, "Perancangan Error Detection System and Error Correction System Menggunakan Metode Hamming Code Pada Pengiriman Data Text. Ilmu Komputer-Universitas Sumatera Utara.
- [7] Deli Alvinda, Achmad Fauzi, dkk, *Pengecekan Bit Error Pada Media Transmisi Pengiriman Gambar*

- Menggunakan Metode *Hamming Code*.** Teknik Informatika-STMIK Kaputama. 2022.
- [8] Rafsan Al Mamun, Md. Motaharul Islam, dkk, ***Error Detection And Correction For Onboard Satellite Computers Using Hamming Code***. Engineering and Technology-World Academy of Science. 2020.
- [9] Rahmaddeni, Koko Harianto, **Konsep Pendeteksian dan Pengkoreksian Kerusakan Gambar Hasil Pengiriman Melalui Facebook.** Teknik Informatika-STMIK Amik Riau.
- [10] Sony Bahagia Sinaga, Satria Yudha Prayogi, dkk, **Implementasi *Cyclic Redundancy Check* dalam Mendeteksi Bit *Error* pada Transmisi Data.** Teknologi Komunikasi-Universitas Islam Sumatera Utara. 2022.