Jurnal Sains dan Matematika Vol. 19(1) : 31-37 (2011)

Konstruksi Greedy Kode *Lexicographic* untuk Membangun Perluasan Kode Golay (24 12,8)

Aurora Nur Aini, Bambang Irawanto
Jurusan Matematika FMIPA UNDIP
Jalan Prof. Soedarto, SH. Tembalang Semarang 50275

ABSTRACT

Golay codes can be constructed by lexicographic codes. Lexicographic codes constructed by Greedy algorithm. Greedy construction is one type of Greedy algorithm. Given codes with minimum distance d and length n. To construct the greedy algorithm, the codeword with length n are processed in some fixed order, and the next codeword is inserted in the code when its distance from all codewords previously selected is $\geq d$. On Greedy construction, to set (n, k, d) codes, we only need to set $(n-k, k)$ codes with k iteration.

Keywords: linear codes, lexicographic codes, Golay, generator matrix

PENDAHULUAN

Pada komunikasi digital, sistem pengkodean pesan mempunyai peran yang sangat penting. Kode yang bisa digunakan dalam proses koreksi *error* antara lain kode Hamming yang mampu mengkoreksi satu kesalahan (*single error*), kode BHC yang mampu mengkoreksi dua kesalahan (*double error*), kode Golay yang mampu mengkoreksi tiga kesalahan (*triple error*), dan juga terdapat kode Reed Solomon yang mampu mengkoreksi *multiple error*. Kode Golay (23,12,7) merupakan kode dengan panjang $n = 23$, $k = 12$, dan jarak Hamming minimum $d = 7$. Kode tersebut dapat mendeteksi dan mengoreksi 3 kesalahan. Kode Golay dapat diperluas menjadi perluasan kode Golay (24,12,8), yaitu kode dengan panjang $n = 24$, $k = 12$, dan jarak Hamming minimum $d = 8$. Untuk membentuk perluasan kode Golay, dapat dilakukan dengan beberapa cara, salah satunya adalah dengan menggunakan kode lexicographic. Kode *lexicographic* merupakan kode yang dibentuk menggunakan algoritma Greedy. Algoritma Greedy tersebut terdiri atas konstruksi Greedy dan konstruksi *Lexicographic*. Untuk menghasilkan perluasan kode Golay dengan konstruksi Greedy dilakukan dengan mencari kode (12, 12), yatu kode dengan panjang $n = 12$ dan $k = 12$, dan diperlukan 12 iterasi.

KODE *LEXICOGRAPHIC*

Kode *lexicographic* atau yang sering disebut *lexicodes* merupakan kode yang dibangun dengan menggunakan algoritma Greedy (*Greedy algorithm*), menurut aturan *lexicographic*. Algoritma Greedy terbagi menjadi dua yaitu konstruksi Greedy dan konstruksi *lexicographic*. Pada paper ini hanya akan dibahas mengenai konstruksi greedy saja. Terdapat beberapa *lexicodes* hasil dari algoritma Greedy, antara lain kode *lexicographic* berbobot konstan (*constant weight lexicodes*) dimana semua vektor di S memiliki bobot konstan, dan kode greedy *lexicographic* (*lexicographic greedy codes*) dimana algoritma greedy di aplikasikan untuk membentuk barisan $V_n(Z_2)$, menjadikannya barisan yang berbeda dari barisan *lexicographic*. Generalisasi algoritma greedy dapat dilakukan untuk ruang vektor atas lapangan GF_q. Untuk kasus biner GF_2, algoritma Greedy menghasilkan kode linier sehingga k vektor basisnya dapat digunakan untuk membentuk 2^k kodekata. Pada bahasan selanjutnya, kode *lexicographic* yang dimaksud adalah kode greedy *lexicographic* biner. Pada algoritma Greedy ini syarat awal untuk membangun kode *lexicographic* adalah bobot vektor yang menjadi vektor basis *lexicographic* $wt(x) \geq d$. Dimana d adalah jarak hamming kode. Pada kode lexicographic digunakan teorema dan definisi di bawah ini :

Definisi 1

Suatu matriks generator G untuk suatu kode C (n, k) adalah matriks berukuran k x n dimana baris-barisnya merupakan basis ruang vektor C.
Bentuk matriks generator standar adalah G = [I_k A], dengan I_k adalah matriks identitas berukuran k x k.

Definisi 2
Mariks cek paritas (*parity check matrix*) H untuk kode linier C, dimana x adalah kodekata yang termasuk dalam kode C adalah
$H \cdot x^T = 0, \forall x \in C$
Hal ini menyatakan bahwa H dan x^T saling tegak lurus (orthogonal).
Dimensi matriks H adalah $(n - k)$ x n.

Hubungan yang untuk selanjutnya akan digunakan dalam masalah *lexicographic* adalah:
1. Jika G = [I_k A], maka H = [A^T I_{n-k}]
2. Jika matriks A telah diketahui, maka matriks generator G dapat dicari dengan menggabungkan matriks identitas I_k di sebelah kiri matriks A. Dengan cara yang sama, matriks H dapat dicari dengan menggunakan A^T.

Terdapat tiga parameter yang digunakan untuk konstruksi kode lexicographic, yaitu bobot kode ($wt(x)$), sindrom (s), dan koset *leader* (l). Bobot kode adalah banyaknya bit yang tidak sama dengan nol pada suatu kode, sedangkan koset leader adalah vektor dengan bobot terkecil pada setiap koset. Pada pembangunan kode *lexicographic*, setiap vektor yang berada pada koset yang sama memiliki sindrom yang sama, sehingga kita dapat menyebut sindrom sebagai sifat dari koset.

Definisi 3
Diberikan kode blok linier C dan vektor $a \in V_n(F)$, koset dari C adalah

$a + C = \{a + x \mid x \in C\}$

Vektor a dan b dikatakan berada pada koset yang sama jika $a - b \in C$.

Definisi 4
Sindrom s atas vektor $y \in V_n(Z_2)$, yang bersesuaian dengan kode (n,k,d) adalah vektor pada subruang $V_{n-k}(Z_2)$ yang didefinisikan sebagai berikut :
$s = \mathbf{H} \cdot y^T$
jika vektor y adalah kodekata, maka sindromnya selalu bernilai 0. Sehingga setiap vektor pada koset C memiliki sindrom 0.

Teorema 1
Pada masing-masing koset terdapat vektor tunggal m, yang disebut *lexicographically earliest vector* atau vektor paling awal *lexicographic*, yang terdiri dari bit informasi yang semuanya nol dan bit sisa (*redundant bit*) yang membentuk sindrom untuk koset tersebut. Jika diasumsikan matriks generator berada pada bentuk standar, *lexicographic earliest vector* merupakan bentuk :
$m = (00\ldots00|s_1 \ldots s_{n-k})$
Bukti :
Tanpa mengurangi sifat umum, diasumsikan kode blok berada pada bentuk standar, yang berarti, k bit pertama merupakan bit informasi. Hal ini untuk menunjukkan bahwa vektor m ada pada masing-masing koset. Asumsikan vektor y, dengan beberapa bit informasi tak nol, kodekata dari matriks generator selalu dapat ditambahkan ke y, sehingga akan membuat semua bit informasi dari y menjadi nol, oleh karena itu diperoleh m. Sesuai dengan definisi koset, y dan m berada pada koset yang sama, jadi dapat disimpulkan bahwa vektor m ada untuk masing-masing koset. Sindrom untuk koset ini adalah :
$s = H.\, y^T = H \cdot m^T = [A^T \mid I_{n-k}] \cdot [0\ldots0\ \ p_1\ldots p_{n-k}]^T$
Dimana $p_1, p_2, \ldots, p_{n-k}$ adalah bit sisa atas vektor m. Karena k bit pertama dari vektor m semua nol, k bit pertama matriks cek paritas tidak mempengaruhi perhitungan, jadi diperoleh
$s = [A^T \mid I_{n-k}] \cdot [0\ldots0\ \ p_1\ldots p_{n-k}]^T = [I_{n-k}] \cdot [p_1\ldots p_{n-k}]^T = [p_1\ldots p_{n-k}]^T$
Atau sindrom adalah semua bit sisa dari vektor m. Ketunggalan sindrom mempengaruhi ketunggalan vektor m.

KONSTRUKSI GREEDY
Pada konstruksi Greedy, jika diberikan barisan ruang vektor $V_n(Z_2)$ yang disusun secara *lexicographic*, dimana $\{0 < 1 < 10 < 11 < \ldots < 1\ldots1\}$, selanjutnya disebut barisan *lexicographic* (*lexicographic order*), konstruksi dimulai dengan kodekata yang semua bernilai nol, $S = \{0\}$ dan secara Greedy ditambahkan semua vektor yang

memiliki jarak d ke dalam S, hingga ditemukan M vektor pertama.

Pada konstruksi greedy untuk kode *lexicographic*, secara umum menggunakan langkah-langkah seperti pada contoh di atas, namun ada beberapa sifat dari barisan *lexicographic* yang dapat mempermudah dalam melakukan konstruksi, yaitu :

1. Hubungan antara barisan *lexicographic* dengan barisan bilangan asli. Oleh karena itu, pencarian atas basis barisan *lexicographic* untuk membentuk *lexicographic earliest vector* dilakukan dengan menggunakan operasi penambahan biasa.
2. Jika $\Lambda(k,d)$ menyatakan matriks generator untuk algoritma Greedy, himpunan bit informasi untuk $\Lambda(k,d)$ adalah matriks identitas k x k. Hal ini berarti k bit dari keseluruhan n bit yang harus dicari sudah diketahui, sehingga kita hanya perlu menitik beratkan pencarian untuk menemukan matriks A, untuk kemudian menggabungkan matriks identitas k x k tersebut dengan matriks A.
3. Adanya korespondensi satu-satu antara sindrom dan anggota pertama *lexicographic* atau *lexicographic earliest member* dari sebuah koset. Pernyataan tersebut berarti *lexicographic earliest member* berada pada suatu koset, sementara setiap kodekata yang berada pada koset tersebut memiliki sindrom yang sama.

MEMBANGUN PERLUASAN KODE GOLAY (24, 12, 8) DENGAN MENGGUNAKAN ALGORITMA GREEDY

Konstruksi greedy untuk kode *lexicographic* dapat digunakan untuk membentuk kode Hamming (7, 4, 3) dan perluasan kode Golay (24, 12, 8) . Langkah – langkah untuk membangun kode *lexicographic* dengan menggunakan konstruksi Greedy adalah sebagai berikut :

1. Membentuk bit informasi, yaitu matriks identitas berukuran k x k.
2. Mencari semua (n-k) tupel untuk menemukan (n-k) tupel pertama yang sesuai untuk menjadi vektor basis *lexicographic*.

Pada ruang dimensi n, jika vektor basis *lexicographic* dinotasikan dengan $\Lambda(k,d)$, maka vektor basis selanjutnya untuk $\Lambda(k+1,d)$ harus memenuhi pertidaksamaan jarak minimum :

$$d(x,c_i) \geq d,\ \forall c_i \qquad (**)$$

Dimana vektor c_i merupakan kombinasi linier dari t vektor basis atas $\Lambda(k,d)$ dan merupakan kodekata n dimensi.

Pertidaksamaan tersebut berlaku untuk kodekata berdimensi n, sedangkan yang diperlukan adalah kodekata berdimensi n-k. Sehingga pertidaksamaan di atas perlu diadaptasi untuk sub ruang dimensi n-k. Dimana untuk dapat membentuk matriks generator kode *lexicographic*, kode berdimensi (n-k) tersebut digabungkan dengan matriks identikas I_k.

Vektor (n-k) - tupel yang dicari adalah $x \in V_{n-k}(Z_2)$. Untuk mengetahui apakah vektor $x \in V_{n-k}(Z_2)$ sesuai untuk menjadi vektor basis *lexicographic*, terdapat aturan sebagai berikut :

1. $wt(x) \geq (d-1)$
2. $d(x,a_i) \geq (d-1-t)\ \forall a_i$

Dimana a_i adalah vektor dari bit sisa kodekata c_i .

Maka gabungan antara x dan matriks identitas yang bersesuaian akan memenuhi pertidaksamaan jarak minimum untuk kodekata berdimensi n (**) di atas. Dan vektor x tersebut merupakan vektor basis *lexicographic* berikutnya.

Pertidaksamaan dalam aturan konstruksi Greedy di atas selalu benar untuk $t \geq (d-1)$, karena $(d-1-t) \leq 0$. Jika c_i merupakan kombinasi linier dari t vektor basis maka jumlahan t bit informasi adalah himpunan 1, sehingga jarak antara x dan c_i adalah $d(x,c_i) = d(x,a_i) + 1 + t$.

Kode Golay (23,12,7) merupakan kode dengan panjang n = 23, k = 12, dan jarak Hamming minimum d = 7. Kode tersebut dapat mendeteksi dan mengoreksi 3 kesalahan. Kode Golay tersebut dapat diperluas menjadi perluasan kode Golay (24,12,8), yaitu kode dengan panjang n = 24, k = 12, dan jarak Hamming minimum d = 8.

Langkah pertama untuk membentuk perluasan kode Golay (24, 12, 8) dengan menggunakan konstruksi Greedy kode *lexicographic* adalah dengan membentuk matriks identitas berukuran k x k, atau dalam hal ini berukuran 12 x 12.

Selanjutnya akan dibangun matriks A berukuran (n-k) x k dengan cara mencari $\Lambda(12,8)$.

Untuk n = 12, dapat dibentuk barisan ruang vektor :
000000000000, 000000000001, 000000000010, 000000000011, … , 111111111110, 111111111111.
Matriks A dicari dengan langkah-langkah sebagai berikut :

1. Iterasi pertama. Menentukan S = {0} = [000000000000]
2. Iterasi kedua, akan dicari $\Lambda(1,8)$ dari S, yaitu vektor sesuai urutan *lexicographic* yang memenuhi syarat konstruksi Greedy, yaitu
   - $wt(x) \geq (d-1)$
   - $d(x, a_i) \geq (d-1-t)\ \forall a_i$

     Menurut barisan *lexicographic*, ditemukan vektor [000001111111] sebagai kandidat vektor basis *lexicographic* yang pertama.
   - $wt(000001111111) \geq (8-1)$

     7 = 7 (terpenuhi)
   - $d(000001111111, 000000000000) \geq (8-1-0)$

     7 = 7 (terpenuhi)

   Jadi vektor tersebut memenuhi sebagai vektor basis *lexicographic* dan dapat dituliskan $\Lambda(1,6) = [0\ 0\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 1\ 1]$
3. Iterasi ketiga, akan dicari $\Lambda(2,6)$

   Untuk dapat menemukan vektor basis *lexicographic* selanjutnya, dilakukan dengan cara menemukan kodekata sesuai urutan *lexicographic* yang memenuhi syarat konstruksi Greedy. Ditemukan vektor [001110001111] sebagai kandidat vektor basis *lexicographic* selanjutnya. Maka,
   - $wt(x) \geq (d\text{-}1)$

     $wt(001110001111) \geq (8\text{-}1)$

     7 = 7 (terpenuhi)
   - $d(x, a_i) \geq (d-1-t)\ \forall\ a_i$

     untuk i = 1, $d(001110001111, 000001111111) \geq (8-1-1)$

     6 = 6 (terpenuhi)

     Karena vektor [001110001111] memenuhi pertidaksamaan di atas, maka vektor tersebut menjadi vektor basis *lexicographic* selanjutnya. Jadi

$$\Lambda(2,6) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

4. Iterasi keempat, akan dicari $\Lambda(3,6)$

   Dengan cara yang serupa pada iterasi ketiga, ditemukan vektor [010111100011] sebagai kandidat vektor basis *lexicographic* selanjutnya.

   Akan dicek apakah vektor yang ketiga tersebut layak menjadi vektor basis *lexicographic*. Maka vektor tersebut harus memenuhi syarat berikut :
   - $wt(x) \geq (d\text{-}1)$

     $wt(010111100011) \geq (8\text{-}1)$

     7 = 7 (terpenuhi)
   - $d(x, a_i) \geq (d-1-t)\ \forall\ a_i$

     untuk i = 1

   - $d(010111100011, 000001111111) \geq (8-1-1)$

   6 = 6 (terpenuhi)
   - $d(010111100011, 001110001111) \geq (8-1-1)$

     6 = 6 (terpenuhi)

     untuk i = 2
   - $d(010111100011, 001111110000) \geq (8-1-2)$

     5 = 5 (terpenuhi)

   Karena vektor ketiga tersebut memenuhi pertidaksamaan di atas, maka vektor tersebut menjadi vektor basis *lexicographic* selanjutnya. Jadi

$$\Lambda(3,6) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

5. Untuk menemukan $\Lambda(12,8)$ diperlukan 13 iterasi. Hasil dari terasi-iterasi di bawah ini ditemukan dengan cara yang serupa dengan iterasi sebelumnya.
   - Pada konstruksi $\Lambda(4,8)$ ditemukan vektor [011010110101] sebagai vektor basis keempat.
   - Pada konstruksi $\Lambda(5,8)$ ditemukan vektor [011101011001] sebagai vektor basis kelima.
   - Pada konstruksi $\Lambda(6,8)$ ditemukan vektor [100111010101] sebagai vektor basis keenam.
   - Pada konstruksi $\Lambda(7,8)$ ditemukan vektor [101011101001] sebagai vektor basis ketujuh.

- Pada konstruksi $\Lambda(8,8)$ ditemukan vektor [101100110011] sebagai vektor basis kedelapan.
- Pada konstruksi $\Lambda(9,8)$ ditemukan vektor [110010011011] sebagai vektor basis kesembilan.
- Pada konstruksi $\Lambda(10,8)$ ditemukan vektor [110100101101] sebagai vektor basis kesepuluh.
- Pada konstruksi $\Lambda(11,8)$ ditemukan vektor [111001000111] sebagai vektor basis kesebelas.
- Pada konstruksi $\Lambda(12,8)$ ditemukan vektor [111111111110] sebagai vektor basis keduabelas. Sehingga dengan menggabungkan vektor-vekor basis yang diperoleh dapat dibentuk :

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

Untuk memperoleh matriks generator untuk perluasan kode Golay (24, 12, 8) maka matriks generator lexicodes tersebut digabungkan dengan matriks identitas I12 sehingga menghasilkan perluasan kode Golay sebagai berikut :

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

SIMPULAN

Kode *Lexicographic* dapat dibentuk dengan menggunakan algoritma Greedy. Algoritma Greedy tersebut terbagi menjadi konstruksi Greedy dan konstruksi *lexicographic*. Kode *lexicographic* tersebut dapat digunakan untuk membangun beberapa kode, misalnya perluasan kode Golay (24, 12, 8).

Pada konstruksi Greedy, untuk membentuk kode (n, k) yaitu kode dengan panjang n dan dimensi k, maka hanya perlu dicari kode (n-k, k) karena k x k bit pertama kode (n,k) tersebut adalah matriks identitas. Jumlah iterasi untuk menghasilkan kode (n-k, k) adalah sebanyak k iterasi.

DAFTAR PUSTAKA

[1] Acosta, Kishion, dkk.2001. *Lexicographic and non-Lexicographic Greedy Codes*. University of California, Los angeles.http://www.wyki.org/Lexicographic_code, diakses bulan November 2008

[2] Anton, Howard.1984. *Aljabar Linier Elementer*. Erlangga : Jakarta.

[3] Conway, J.H & N.J.A. Sloane. *Lexicographic Codes : Error - Correcting*

codes from Game Theory. IEEE Transaction on Information Theory, May 1986.http://www.research.att.com/~njas/doc/lex.pdf, diakses bulan November 2008

[4] Gilbert, Jimmie & Linda Gilbert. *Element of Modern Algebra*.

[5] Harjito, Drs,dkk.2006. *Buku Ajar Aljabar 1*. Universitas Diponegoro.

[6] Hillman, Abraham.P & Gerald L. Alexanderson. 1993. *Abstract Algebra, a First Undergraduate Course, fifth edition*. PWS Publishing Company : Boston MA.

[7] Kanemasu, Melisa. *Golay Codes*. http://www-math.mit.edu/phase2/UJM/vol1/MKANEM~1.pdf.

[8] Rizki. K, Ikhsan.2008. *Membangun Kode Golay (24, 12, 8) dengan Matriks Generator*. Jurusan Matematika, FMIPA. Universitas Diponegoro.

[9] Spasov, Dejan. 2006. *Implementing The Lexicografik Construction*. Boston University. http://nislab.bu.edu/nislab/projects/lexicode/project%20paper.pdf, diakses bulan November 2008

[10] Suryadi, H.S & S. Harini Machmudi. 1984. *Teori dan Soal Pendahuluan Aljabar Linier*. Ghalia Indonesia : Jakarta.

[11] Trachtenberg, Ari. *Designing Lexicographic Codes with Given Trellis Complexity*. IEEE Transaction on Information Theory, January 2002. http://people.bu.edu/trachten/ diakses bulan November 2008

[12] Widyaningsih, Santi. 2008. *Deteksi dan Koreksi Error pada pesan digital dengan kode Hamming*. Jurusan Matematika, FMIPA. Universitas Diponegoro.

[13] Vanstone, S.A & Oorschot, P.C. 1989. *An Introduction to Error Correcting Codes with Applications*. London : Kluwer Academic Publisher.