

OPTIMASI IRIGASI SAWAH DENGAN MENGGUNAKAN ALGORITMA GENETIK

Bilqis Amaliah¹, Chastine Fatichah², Agus Zainal A³,
Eka Gibran Hasany⁴, Maula N. M. Fachrurrozie⁵
Jurusan Informatika, FTIf

Institut Teknologi Sepuluh Nopember (ITS), Surabaya, Indonesia
bilqis@cs.its.ac.id¹, chastine@cs.its.ac.id², agus.za@its-sby.edu³,
gibran06@cs.its.ac.id⁴, maul_fruz@cs.its.ac.id⁵

Abstrak

Penelitian ini mengusulkan optimasi pengaturan distribusi air pada saluran irigasi sawah menggunakan *Generalized Minimal Spanning Tree* (GMST). GMST adalah optimasi jarak pada dengan spanning tree pada suatu graf yang terdiri dari beberapa cluster, dimana antar cluster dihubungkan minimal oleh sebuah node. Tiap cluster dapat terdiri dari satu atau lebih node. Kasus saluran irigasi sawah merupakan kasus khusus dari GMST, yaitu ada sumber air dan satu node dapat berada pada lebih dari satu cluster. Untuk menyelesaikan GMST khusus saluran irigasi sawah, Peneliti akan menggunakan metoda Algoritma Genetik (GA). Hasil yang ingin dicapai pada penelitian ini adalah minimum jumlah saluran irigasi sawah dan minimum panjang total saluran irigasi sawah yang bisa mengaliri semua petak sawah. Dengan demikian, masalah biaya dan tenaga untuk pembuatan saluran air bisa ditekan, dan hasilnya semua bidang sawah tetap bisa teraliri air.

Kata kunci : minimal spanning tree, general minimal spanning tree, algoritma genetik

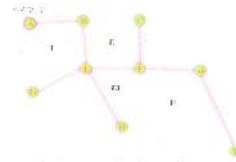
1. Pendahuluan

Kasus saluran irigasi sawah adalah kasus yang menarik untuk dijadikan penelitian di bidang optimasi. Sementara ini, para petani dalam masalah mengaliri sawahnya, tidak melihat, apakah sistem pengairannya optimal atau tidak, yang penting sawahnya teraliri oleh air. Kadangkala, satu petak sawah mendapat aliran air lebih dari satu sumber air, dan petak sawah lainnya hanya mendapat satu sumber air saja. Karena tidak adanya pengaturan yang optimal, maka air akan banyak terbuang percuma. Misal, ada satu petak sawah, seharusnya cukup hanya mendapat air dari satu saluran irigasi saja, tapi petak sawah tersebut mendapat lebih dari satu saluran irigasi, sehingga banyak air yang terbuang percuma.

Dari permasalahan diatas, maka Peneliti ingin mengajukan penelitian bagaimana caranya agar semua petak sawah teraliri air dengan jumlah saluran irigasi minimum dan panjang total saluran irigasi yang minimum pula. Untuk menjawab permasalahan ini, peneliti mengajukan sebuah metoda, yaitu metoda *generalized minimal spanning tree* yang disingkat menjadi GMST. Metoda ini merupakan generalisasi dari metoda *minimal spanning tree* (MST). Metode MST pada umumnya digunakan untuk menyelesaikan kasus transportasi. Namun peneliti sudah pernah menerapkan metode MST untuk menyelesaikan kasus lain yaitu deteksi tepi citra warna dan hasilnya lebih baik menggunakan metode *Vector Ranking* [8]. Pada usulan penelitian ini, digunakan metoda GMST yaitu generalisasi dari metode MST, untuk pengaturan distribusi air pada saluran irigasi sawah agar optimal. Berikut adalah contoh gambar aliran irigasi sawah dengan menggunakan *Minimal Spanning Tree* dan *Generalized Minimal Spanning Tree*



Gambar 1. Saluran Irigasi sawah dgn MST



Gambar 2. Saluran Irigasi sawah dgn GMST

Contoh lain dari penggunaan GMST adalah : mendesign jaringan pohon untuk fiber optik [7], Jaringan fiber optik di New York dan Dallas [6], Rencana Produksi [5].

GMST untuk kasus irigasi sawah, adalah kasus GMST khusus, karena adanya sebuah node yang dianggap sebagai sebuah sumber air. Sedangkan untuk contoh-contoh yang sudah diteliti sebelumnya [5,6,7] adalah kasus GMST umum, karena tidak memerlukan suatu node yang dianggap sebagai sumber.

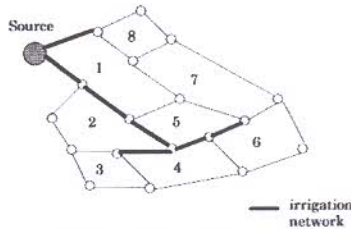
GMST adalah persoalan heuristik. Sehingga untuk menjawab persoalan ini, kami akan menggunakan pendekatan heuristik, yaitu untuk menentukan solusi yang baik atau mendekati optimal dari suatu masalah optimisasi. Sebuah heuristik tidak menjamin suatu solusi yang paling optimal, melainkan mendekatinya.

2. Masalah Irigasi Sawah

Tujuan dari penelitian ini adalah membangun perangkat lunak untuk optimasi distribusi air pada saluran irigasi sawah menggunakan *Generalized Minimal Spanning Tree* (GMST).

Hasil dari penelitian ini adalah aplikasi yang interaktif berupa graf dimana user menginputkan node, edge (saluran irigasi sawah) dan bobot (panjang saluran irigasi sawah). Output yang dihasilkan minimum jumlah saluran irigasi sawah dan minimum panjang total saluran irigasi sawah yang bisa mengalir semua petak sawah.

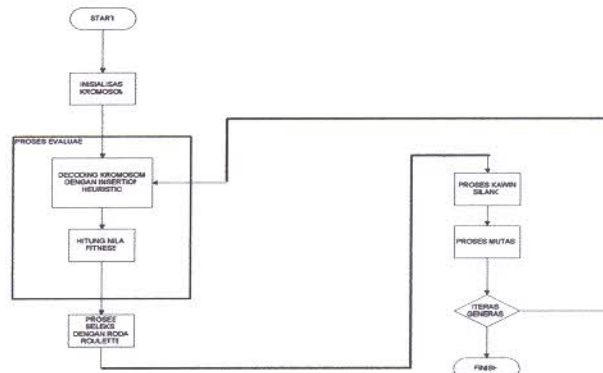
Saluran irigasi sawah adalah contoh dari clustering, dimana sawah dibagi menjadi beberapa bagian. Pada saluran irigasi sawah, satu cluster terdiri dari beberapa node dan satu node dapat menjadi milik banyak cluster, node adalah sudut sawah. Optimisasi saluran irigasi sawah adalah kasus khusus dari algoritma formal, karena harus ada satu sumber air dan satu node dapat berada pada banyak cluster.



Gambar 3. Contoh irigasi sawah

3. Perancangan

Algoritma genetika merupakan suatu metode pencarian yang didasarkan pada mekanisme dari seleksi dan genetika natural. Blok diagram dari mekanisme kerja algoritma genetika untuk kasus GMST ini adalah seperti yang terlihat pada gambar 4.



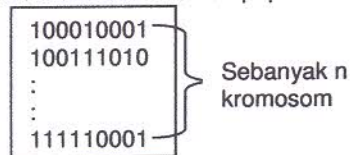
Gambar 4. Diagram alur algoritma genetik

a. INISIALISASI KROMOSOM

Algoritma dimulai dengan pembentukan sejumlah alternatif pemecahan yang disebut populasi. Pembentukan populasi awal dalam algoritma genetika dilakukan secara acak sebanyak n kromosom. Kromosom berisikan informasi solusi dari sekian banyak alternatif solusi masalah yang dihadapi.

Kromosom untuk masalah GMST diilustrasikan dengan menggunakan coded string yang unik, disimbolkan X dengan panjang string n (jumlah node) dan disebut individual. Kita ambil binary coded string : $X_i = 1$ jika tree merentang node i (pada graf sawah, hal ini berarti vertex yang teraliri air), dan $X_i = 0$ jika sebaliknya (pada graf sawah, hal ini berarti vertex yang tidak teraliri air), dengan $i=1, \dots, n$ vertex/ node pintu air.

Berikut adalah contoh populasi :



Gambar 5. Contoh kromosom- kromosom dalam populasi

Bit pertama pada kromosom mewakili vertex pertama, dan selalu bernilai 1 pada semua kromosom, karena merupakan vertex sumber air yang harus selalu teraliri air.

b. PROSES EVALUASI

Sebelum dihitung nilai fitness dari suatu kromosom, kromosom harus di-decoding terlebih dahulu, menggunakan **Insertion Heuristic** :

- Langkah 1: pilih node i dari T , $T = T \setminus \{i\}$, $ST = \{i\}$.
- Langkah 2: Temukan node $j \in T$ dan paling dekat dengan ST , dan jalur (ST, j) menunjukkan jarak terpendek dari j ke ST .
 $ST := ST \cup \text{PATH}(ST, j)$, $T = T \setminus \{j\}$
- Langkah 3: Jika $T = \emptyset$ maka solusinya adalah ST , selesai. Namun jika tidak, lakukan langkah 2.

Contoh penerapannya untuk kromosom 100010001. Kromosom tersebut jika direpresentasikan dalam graf bisa digambarkan seperti berikut :



Gambar 5. Representasi kromosom dalam graf sawah

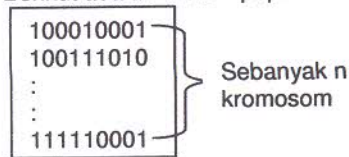
Tampak pada gambar di atas, graf belum membentuk suatu tree, yang berakibat belum bisa terbentuk suatu saluran air. Permasalahan berikutnya adalah bagaimana membentuk saluran dari node 1 ke node 5 dan ke node 9. Untuk melakukannya, digunakan metode Insertion heuristic yang algoritmanya telah dijelaskan sebelumnya.

a. INISIALISASI KROMOSOM

Algoritma dimulai dengan pembentukan sejumlah alternatif pemecahan yang disebut populasi. Pembentukan populasi awal dalam algoritma genetika dilakukan secara acak sebanyak n kromosom. Kromosom berisikan informasi solusi dari sekian banyak alternatif solusi masalah yang dihadapi.

Kromosom untuk masalah GMST diilustrasikan dengan menggunakan coded string yang unik, disimbolkan X dengan panjang string n (jumlah node) dan disebut individual. Kita ambil binary coded string : $X_i = 1$ jika tree merentang node i (pada graf sawah, hal ini berarti vertex yang teraliri air), dan $X_i = 0$ jika sebaliknya (pada graf sawah, hal ini berarti vertex yang tidak teraliri air), dengan $i=1, \dots, n$ vertex/node pintu air.

Berikut adalah contoh populasi :



Gambar 5. Contoh kromosom- kromosom dalam populasi

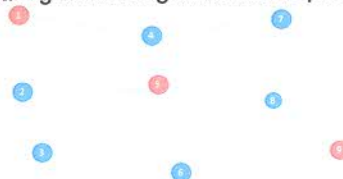
Bit pertama pada kromosom mewakili vertex pertama, dan selalu bernilai 1 pada semua kromosom, karena merupakan vertex sumber air yang harus selalu teraliri air.

b. PROSES EVALUASI

Sebelum dihitung nilai fitness dari suatu kromosom, kromosom harus di-decoding terlebih dahulu, menggunakan **Insertion Heuristic** :

- Langkah 1: pilih node i dari T , $T = T \setminus \{i\}$, $ST = \{i\}$.
- Langkah 2: Temukan node $j \in T$ dan paling dekat dengan ST , dan jalur (ST, j) menunjukkan jarak terpendek dari j ke ST .
 $ST := ST \cup \text{PATH}(ST, j)$, $T = T \setminus \{j\}$
- Langkah 3: Jika $T = \emptyset$ maka solusinya adalah ST , selesai. Namun jika tidak, lakukan langkah 2.

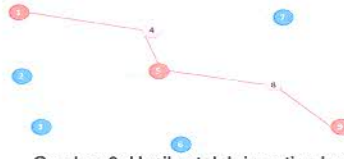
Contoh penerapannya untuk kromosom 100010001. Kromosom tersebut jika direpresentasikan dalam graf bisa digambarkan seperti berikut :



Gambar 5. Representasi kromosom dalam graf sawah

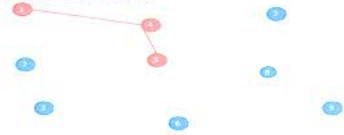
Tampak pada gambar di atas, graf belum membentuk suatu tree, yang berakibat belum bisa terbentuk suatu saluran air. Permasalahan berikutnya adalah bagaimana membentuk saluran dari node 1 ke node 5 dan ke node 9. Untuk melakukannya, digunakan metode Insertion heuristic yang algoritmanya telah dijelaskan sebelumnya.

Hasil dari insertion heuristic pada contoh kromosom 100010001 digambarkan pada gambar berikut :



Gambar 6. Hasil setelah insertion heuristic

Tampak bahwa, vertex ke-4 dan ke-8 harus dipilih untuk mendapatkan suatu saluran air yang benar. Maka kromosom 100010001 berubah menjadi 100110011. Sangat berguna jika Steiner tree yang diperoleh dari metode Insertion Heuristic ditingkatkan lagi dengan menghapus **redundant edges** (edge yang berlebih). Hasil dari penghapusan redundant edge pada contoh kromosom 100110011 digambarkan pada gambar berikut :



Gambar 7. Hasil penghapusan redundant edge

Tampak pada gambar, edge yang menghubungkan vertex 5 dan vertex 8 dihapus sehingga sudah tidak termasuk dalam saluran lagi. Meskipun edge tadi dihapus, hal ini tidak menjadi masalah karena cluster sawah kanan bawah tetap dapat teraliri air yaitu dari vertex 5. Hal yang sama juga dilakukan pada edge yang menghubungkan vertex 8 dan vertex 9.

Dalam menentukan nilai fitness dari suatu kromosom, bisa didapat dari **total weight of the tree** (berat total suatu tree) dan **penalty cost**.

Penalty cost meningkatkan kesesuaian pada **infeasible tree** (saat beberapa subset tidak tercover). Nilainya sama dengan αK , dengan α adalah **penalty factor** dan K adalah jumlah subset yang tidak tercover.

Secara umum, fungsi untuk mendapatkan nilai fitness ditunjukkan pada rumus di bawah ini, dengan W adalah total weight tree :

$$F = W + \alpha.K$$

c. PROSES SELEKSI

Menggunakan metode pemutaran roda roulette sebanyak n kali, dengan n adalah jumlah kromosom dalam populasi.

d. PROSES KAWIN SILANG

Parent 1: 011101001 Child 1: 01110010
 Parent 2: 101110010 Child 2: 10101001

Crossover merupakan pertukaran bagian bit antara string dari 2 parent untuk memperoleh 2 **offspring string** (string dari child dari parent tersebut). Kami telah menunjukkan 1-point crossover, crossover point dipilih secara acak dari string tersebut.

e. PROSES MUTASI

Mutasi terjadi dengan beberapa kemungkinan khusus pada tiap bit dari suatu string. Operator **bitwise complement mutation** mengubah nilai suatu bit menjadi nilai lawannya.

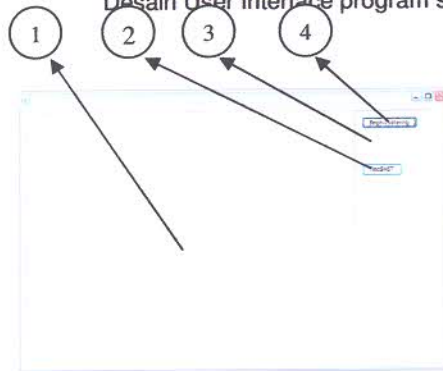
f. PROSES ITERASI GENERASI

Keseluruhan proses ini diulang- ulang untuk menghasilkan **successive generations** (hasil yang bagus). String dengan **least cost** (cost terendah) akan disimpan dan di-update

4. Analisa dan Uji Coba

Dalam tahap analisa dan uji coba program ini, spesifikasi komputer yang digunakan adalah processor Core Duo 2.0GHz dan RAM 1.5GB.

Desain User Interface program seperti dijelaskan pada gambar berikut ini :



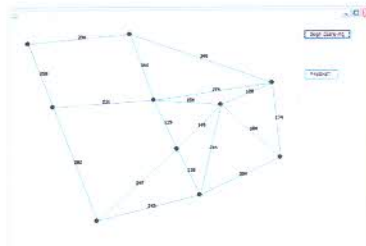
Gambar 8.Desain User Interface program

Keterangan :

1. Ruang untuk input graf dan output GMST
2. Tombol untuk melakukan pemasukkan vertex- vertex pada cluster
3. Tombol untuk menambahkan cluster
4. Tombol untuk melakukan pencarian GMST

Program akan kita coba dengan data input

seperti di bawah ini :



Gambar 9.Contoh graf sawah untuk data input

Dengan parameter :

Jumlah Node	10
Jumlah Edge	18
Jumlah Cluster	9
α	3620

Hasilnya seperti pada tabel di bawah ini:

Jenis GA	n kromosom = 500 n generasi = 50	n kromosom = 600 n generasi = 100
GA1	658 (2'12'')	658 (4'07'')
GA2	842(2'21'')	664 (4'29'')
GA3	989 (2'23'')	1025 (4'46'')
GA4	842 (2'17'')	897 (4'31'')

Keterangan :

1. Nilai pada tabel => total edge(waktu komputasi)
2. GA1 = probabilitas CrossOver dan mutasi adalah 0,1
3. GA2 = probabilitas CrossOver dan mutasi adalah 0,25
4. GA3 = probabilitas CrossOver dan mutasi adalah 0,75
5. GA4 = probabilitas CrossOver dan mutasi adalah 0,9

