

6.2 Algoritma Genetika (Genetic Algorithm)

Genetic Algorithm (GA) adalah algoritma pencarian yang didasarkan pada mekanisme **seleksi alamiah** dan **genetika alamiah** [GOL89].

GA dimulai dengan serangkaian solusi awal (kromosom), yang disebut populasi. Populasi ini akan ber-evolusi menjadi populasi yang berbeda melalui serangkaian iterasi. Pada akhir iterasi, GA mengembalikan anggota populasi yang terbaik sebagai solusi untuk problem tersebut. Pada setiap iterasi (generasi), proses evolusi yang terjadi adalah sebagai berikut:

1. Dua anggota populasi (*parent*) dipilih berdasarkan pada suatu distribusi populasi. Kedua anggota ini kemudian dikombinasikan atau dikawinkan melalui operator **crossover** (pindah silang) untuk menghasilkan anak (*offspring*).
2. Dengan probabilitas yang rendah, anak ini akan mengalami perubahan oleh operator **mutasi**.
3. Apabila anak sesuai untuk populasi tersebut, suatu skema penggantian (*replacement scheme*) diterapkan untuk memilih anggota populasi yang akan digantikan oleh anak.
4. Proses ini terus berulang sampai dicapai kondisi tertentu, misalnya sampai jumlah iterasi tertentu.

Kromosom : serangkaian nilai gen.

Gen : mempunyai nilai berupa suatu alfabet S, misalnya {0,1}.

Solusi : direpresentasikan oleh suatu kromosom.

Skema : pola gen yang menggambarkan suatu subset dari string yang memiliki kesamaan pada posisi gen tertentu.

Orde : banyaknya simbol khusus pada suatu skema.

Defining length : jarak antara simbol khusus paling kiri dan paling kanan.

Misal: 10* adalah skema yang mewakili himpunan kromosom yang mempunyai nilai 1 pada bit pertama dan 0 pada bit kedua. Skema tersebut memiliki orde 2 dan *defining length*-nya adalah 1.

Suatu kromosom dengan panjang n dapat juga dipandang sebagai *instance* (anggota) dari 2^n skema. Misal, kromosom 101 adalah suatu instance dari 8 skema, yaitu ***, 1**, *0*, **1, 10*, 1*1, *01, dan 101. Algoritma Genetika tidak secara eksplisit dalam menangani skema, namun konsep skema sangat penting dalam analisis perilaku Algoritma Genetika dalam pemecahan suatu masalah.

6.2.1 Sifat-Sifat Khusus GA [GOL89]:

1. GA bekerja dengan sebuah pengkodean dari kumpulan parameternya sendiri.
2. GA mencari dari sebuah populasi titik-titik, bukan dari sebuah titik tunggal.
3. GA hanya menggunakan informasi hasil (fungsi objektif), bukan menurunkan dari bantuan pengetahuan lain.
4. GA menggunakan kemungkinan aturan transisi, bukan aturan-aturan deterministik.

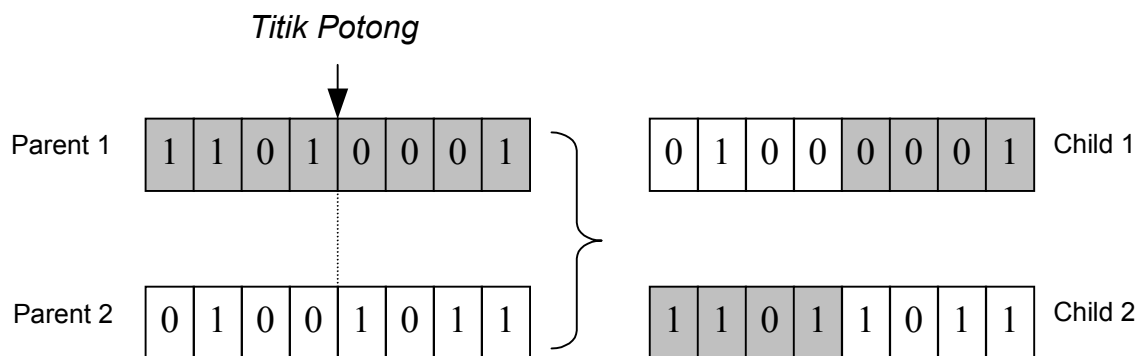
6.2.2 Operator-operator GA

6.2.2.1 Reproduksi

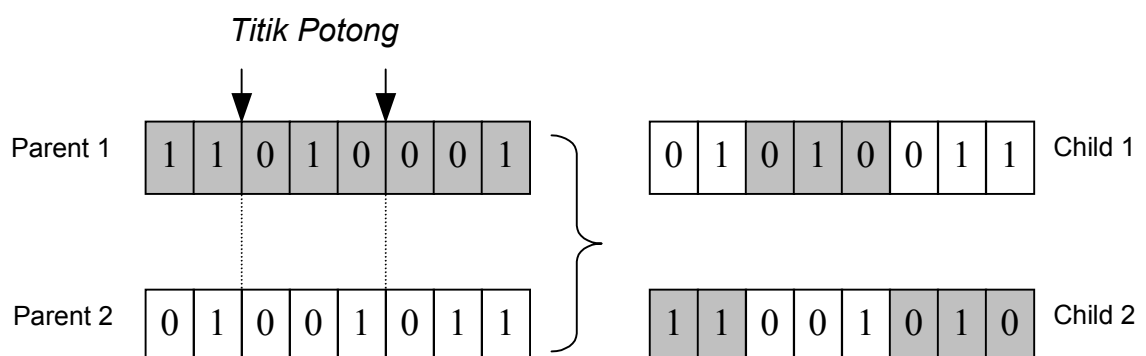
- Reproduksi merupakan sebuah proses dimana sebuah string individu disalin kembali menjadi individu-individu baru yang siap menjadi orang tua.
- Mekanisme reproduksi menggunakan **cross over** dan **mutasi**. Pemilihan individu orang tua bergantung pada nilai fungsi **fitness** dari individu-individu tersebut di dalam populasinya.
- Kegunaan dari pemilihan orang tua dalam GA adalah untuk memberikan kesempatan bereproduksi terhadap seluruh anggota populasi yang terbaik. Ada banyak cara untuk menyelenggarakan pemilihan orang tua, salah satunya adalah pemilihan orang tua **Roulette Wheel**.

6.2.2.2 Pindah Silang (cross over)

- **Pindah Silang Satu Titik**



- **Pindah Silang Dua Titik**



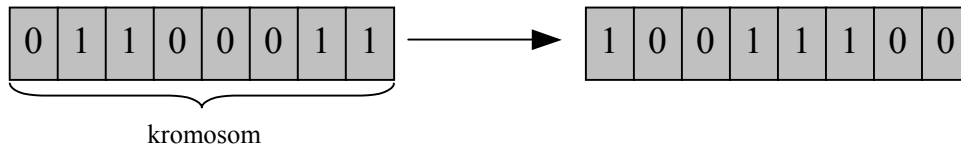
Operator pindah silang dapat dilakukan dengan lebih dari dua titik. Tetapi jumlah titik potong yang semakin banyak akan memperendah kualitas solusi yang didapatkan. Hal ini disebabkan operasi pindah silang terlalu sering merusak kromosom yang baik.

6.2.2.3 Mutasi

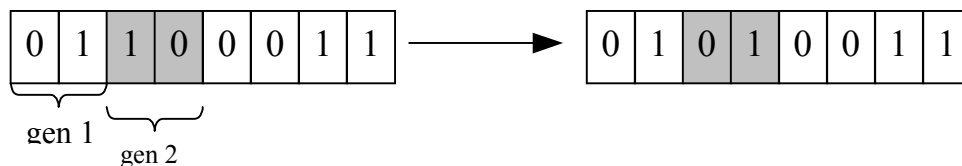
Mutasi diperlukan untuk mengembalikan informasi bit yang hilang akibat *cross over*. Mutasi diterapkan dengan probabilitas sangat kecil. Jika mutasi dilakukan terlalu sering, maka akan menghasilkan individu yang lemah karena konfigurasi bit pada kromosom yang unggul akan dirusak.

Berdasarkan bagian yang termutasi, proses mutasi dapat dibedakan atas tiga bagian:

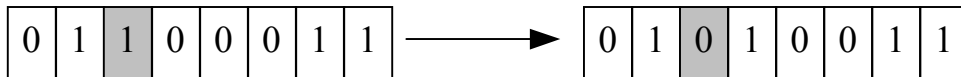
- Tingkat kromosom: semua gen dalam kromosom berubah.



- Tingkat gen: semua bit dalam satu gen akan berubah. Misal gen 2 yang mengalami mutasi.



- Tingkat bit: hanya satu bit yang berubah.



6.2.3 Algoritma Genetika Sederhana

Random populasi awal

Repeat

Pilih parent1 dan parent2 dari populasi;

Offspring = crossover(parent1, parent2);

If Mutable **then** Mutation(offspring);

If Suited(offspring) **then**

Replace(population, offspring);

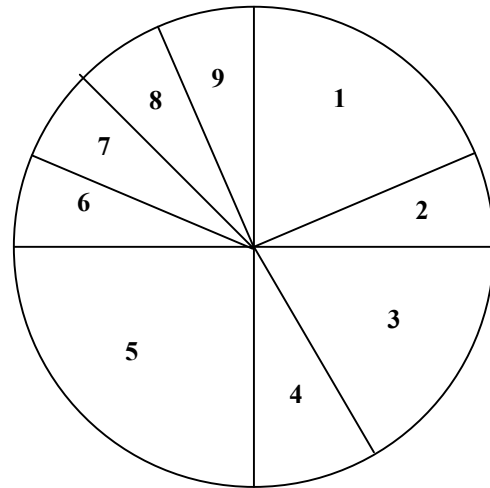
...

Gambar 6.11 Struktur Umum Steady State GA.

6.2.4 Skema Pemilihan Orang Tua

Pemilihan orang tua didasarkan pada *proportional selection scheme*, yaitu: Setiap kromosom dipilih sebagai *parent* berdasarkan suatu probabilitas yang proporsional terhadap nilai *fitness* masing-masing. Salah satu teknik yang sering digunakan adalah *Roulette Wheel*.

String	Nilai Fitness
S1	0.6
S2	0.2
S3	0.6
S4	0.2
S5	0.8
S6	0.2
S7	0.2
S8	0.2
S9	0.2
Jumlah	3.2



Catatan: Penentuan fungsi fitness bergantung pada masalah yang akan dioptimasi.

6.2.5 Skema Penggantian Individu dalam Populasi

1. Selalu mengganti anggota populasi yang memiliki nilai fitness terkecil [GEN97].
2. Membandingkan anak dengan kedua orang tua (*preselection*). Apabila anak memiliki nilai fitness yang lebih baik daripada salah satu atau kedua orang tua, anak menggantikan orang tua yang memiliki nilai fitness terendah sebagai anggota populasi. Skema ini dapat menjaga keanekaragaman yang lebih baik dibanding skema sebelumnya. Dengan mengganti orang tua yang memiliki *Hamming Distance* (beda bit) yang lebih sedikit, penggantian tersebut dapat diharapkan tidak cepat menghilangkan keragaman.

6.2.6 Kriteria Penghentian

1. Mengambil suatu nilai sebagai batas iterasi. Apabila batas iterasi tersebut dicapai, iterasi dihentikan dan solusi terbaik pada populasi dilaporkan sebagai solusi terbaik pencarian.
2. Menghitung kegagalan penggantian anggota populasi yang terjadi secara berurutan sampai jumlah tertentu. Hal ini berarti bahwa anggota-anggota populasi adalah kemungkinan-kemungkinan solusi yang terbaik dari semua kemungkinan solusi pada ruang pencarian yang ada. Sehingga iterasi lanjutan yang dilakukan tidak akan banyak membantu lebih jauh untuk mendapatkan solusi yang lebih baik.

6.2.7 Kelemahan Algoritma genetika [GOL89]:

- Lebih lama menemukan solusi karena memproses beberapa solusi sekaligus.
- GA dapat mengalami konvergensi yang terlalu cepat sehingga tidak ada jaminan solusi yang didapatkan adalah solusi optimal.

Hipotesa *Building Blocks*:

Operasi terhadap suatu kromosom dapat juga dipandang sebagai operasi implisit terhadap 2^n skema. Sehingga dapat dikatakan bahwa selain memproses kromosom secara eksplisit, GA juga secara implisit memproses skema. Ini disebut sebagai *implicit parallelism* atau *intrinsic parallelism*.

***Building Block Hypothesis* by Goldberg:**

GA secara implisit akan memilih skema berorde rendah dan memiliki nilai fitness tinggi, kemudian menggenerate skema-skema dengan orde yang lebih tinggi dan nilai fitness tinggi dari skema-skema berorde rendah tersebut melalui operasi pindah silang. Proses inilah yang menjadi kelebihan Algoritma Genetika [GOL89].

Agar skema-skema bernilai fitness tinggi dapat dikombinasikan satu sama lain menjadi skema berorde lebih tinggi, skema tersebut harus tetap terjaga selama proses pindah silang (tidak terpotong). Sehingga dapat dikatakan bahwa semakin pendek *defining length* dari suatu skema, kemungkinan skema tersebut terpotong oleh operasi pindah silang semakin kecil, demikian pula sebaliknya.