

## TAHAP PENGENALAN KOTAK

Sebuah kotak berbentuk sudoku yang dimana memiliki 9 bilik. Setiap kotak harus memiliki nilai yang berbeda. Jumlah dari angka-angka yang tersusun secara horizontal dan vertikal harus memiliki nilai yang sama.

Dari deskripsi tersebut dapat dirancang sebuah terkhusus untuk sebuah sistem, yaitu randomizing dan scanning. Randomizing bertujuan untuk mengacak array yang sudah terdefiniskan terurut dari 1 sampai 9 demi menyesuaikan aturan yang dimana mengharuskan setiap kotak memiliki nilai yang berbeda. Randomizing juga dipecah menjadi 2 metode, yaitu metode shuffle dan metode specific random. Metode shuffle merupakan sebuah metode yang akan mengacak list secara tidak beraturan, sedangkan metode specific random merupakan sebuah metode yang akan mengacak list berdasarkan pemilihan 2 bilik secara acak.

Scanning bertujuan sebagai media evaluasi pada kotak yang telah tersusun oleh randomizing berdasarkan aturan yang dimana mengharuskan jumlah angka yang tersusun secara horizontal dan vertikal harus memiliki nilai yang sama. Untuk spesifikasi aturan tersebut, susunan horizontal meliputi bilik 4, 5, dan 6, sedangkan susunan vertikal meliputi bilik 2, 5, dan 8. Kedua variabel itu diambil selisihnya untuk dijadikan sebagai bahan evaluasi. Jika selisih yang didapat bernilai 0, maka dapat dipastikan bahwa nilai dari susunan horizontal dan vertikal adalah sama.

### CONTOH GAMBAR

Randomizing :

1	2	3
4	5	6
7	8	9

SHUFFLE  
⇒

6	7	4
3	5	1
2	9	8

6	7	4
3	5	1
2	9	8

SWITCH  
⇒

6	8	4
3	5	1
2	9	7

Metode Shuffle

Metode specific random

Scanning :

6	8	4
7	5	1
2	9	7

→ Horizontal

→ Vertikal

$$\text{Evaluasi} = |\text{Horizontal} - \text{Vertikal}|$$

$$= |(3+5+1) - (8+5+9)|$$

$$= |9 - 22|$$

$$= 13 \Rightarrow 13 \neq 0$$

# TAHAP PENYELESAIAN

Terdapat 3 Algoritma yang dapat menyelesaikan kotak berbentuk sudokan ini, yaitu Hill Climbing, Simulated Annealing, dan Genetic.

## HILL CLIMBING

Hill Climbing merupakan sebuah algoritma yang dimana pencarian solusi terbaik didasarkan pada konsep pencarian lokal di sekitar titik awal tertentu secara berulang. Hill Climbing bisa diibaratkan seperti pendakian gunung dengan kabut yang tebal dan pendaki mengalami amnesia.

Dalam penyelesaiannya, diperlukan inisiasi kotak yang sudah diacak menggunakan metode shuffle sebagai titik awal, serta inisiasi nilai awal evaluasi hasil scanning dari kotak tersebut. Selama nilai evaluasi belum mencapai 0, maka proses pencarian akan tetap berlangsung. Selama proses pencarian, kotak akan diacak lagi menggunakan metode specific random lalu hasilnya akan dijadikan sebagai inisiasi tetangga atau bisa disebut sebagai inisiasi langkah disekitar titik awal. Jika nilai evaluasi tetangga kurang dari nilai evaluasi titik awal, maka langkah selanjutnya adalah menuju ke tetangga tersebut. Hal ini akan terus berulang hingga target tercapai.

CONTOH :

3	4	6
8	5	1
7	9	2

eval = 4

INITIAL STATE



3	4	6
8	5	1
7	2	9

eval = 3

NEIGHBOR STATE

neighbor < current  
(3 < 4)  
----->

3	4	6
8	5	1
7	2	9

eval = 3

INITIAL STATE

looping



3	8	6
1	9	7
5	2	4

eval = 2

INITIAL STATE



3	6	8
1	9	7
5	2	4

eval = 0

NEIGHBOR STATE

neighbor < current  
(0 < 2)  
----->

3	6	8
1	9	7
5	2	4

eval = 0

INITIAL STATE

# SIMULATED ANNEALING

Simulated Annealing merupakan sebuah algoritma yang dimana prinsip terjadinya mengambil langkah-langkah acak dalam pencarian solusi, tetapi juga menerima langkah-langkah terburuk dengan probabilitas tertentu sehingga bisa keluar dari minimum lokal dan mencapai minimum global. Simulated Annealing bisa diibaratkan seperti pembentukan logam yang dimana logam ditempa dalam keadaan panas dan terus menerus ditempa seiring penurunan suhu hingga mendapatkan logam yang diharapkan.

Dalam penyelesaiannya, terdapat persamaan terhadap hill climbing. Namun, ada perbedaan dalam iterasi dan tata cara evaluasi. Iterasi tersebut menggunakan inisiasi suhu awal dan cooling rate yang dimana selama proses pencarian suhu akan terus berkurang hingga menjadi 0. Untuk evaluasi, nilai evaluasi tetangga akan dikurangi dengan nilai evaluasi saat ini sehingga membentuk nilai delta energi ( $\Delta E$ ) untuk perhitungan probabilitas. Rumus probabilitas adalah  $e^{(-\Delta E / \text{suhu})}$ . Inisiasi tetangga akan ditetapkan sebagai inisiasi saat ini jika  $\Delta E$  kurang dari 0 atau memenuhi probabilitas.

CONTOH :

Suhu awal = 1000  
cooling rate = 0,95

} Untuk ~~data~~ pengurangan suhu, suhu awal akan dikalikan dengan cooling rate

9	2	6
4	8	3
5	1	7

eval = 4

INITIAL STATE



9	2	6
4	8	1
5	3	7

eval = 0

NEW STATE

$$\begin{aligned}\Delta E &= \text{new} - \text{current} \\ &= 0 - 4 \\ &= -4\end{aligned}$$

$$\hookrightarrow \Delta E < 0 \dots -4 < 0$$

~~FINISH~~



9	2	6
4	8	1
5	3	7

eval = 0

NEW STATE

suhu x cooling rate

Pada proses ini, suhu akan terus berkurang walaupun sudah ditemukan solusi.

$\Rightarrow$  FINISH



# GENETIC ALGORITHM

Algoritma genetik merupakan sebuah algoritma yang bekerja layaknya teori evolusi dalam biologi dengan mengikuti konsep seleksi alam, rekombinasi genetik, mutasi, dan evaluasi.

Dalam penyelesaiannya, sebanyak jumlah populasi yang ditentukan akan dibuat dengan bantuan metode shuffle pada randomizing. Setelah itu, dibuat lah sebanyak jumlah generasi yang ditentukan untuk menciptakan populasi baru. Dalam pembuatan populasi baru, populasi awal akan diurutkan dari terkecil ke terbesar. Hasil dari pengurutan akan diambil 10 populasi dengan evaluasi hasil scanning terendah atau disebut sebagai 10 populasi teratas. 10 populasi terpilih akan dikombinasikan satu sama lain dan bahkan salah satu akan bermutasi jika memenuhi syarat mutation rate. Setelah proses tersebut selesai, populasi terbaru akan dievaluasi. Jika populasi tersebut memiliki nilai evaluasi 0, maka proses pencarian di berhentikan.

CONTOH GAMBAR :

Ketiga algoritma tersebut, sudah dibuatkan kode Python. Adapun, link github :

<https://github.com/MrRazorez/BoxAI-2023>