

Aplikasi Algoritme Greedy Best First Search dalam Optimalisasi Pergerakan Microbots

Akmal Mahardika Nurwahyu Pratama - 13521070

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
13521070@std.ste.itb.ac.id

Abstraksi—*Microbots* adalah robot dengan ukuran kecil yang kurang dari 1 mm. *Microbots* dikembangkan untuk kepentingan kesehatan dan bioteknologi, khususnya pada intervensi obat dalam tubuh manusia, dalam pembuluh darah. Dengan bantuan CT Scan, *microbots* dikendalikan dan diarahkan untuk menuju ke semua letak permukaan yang diduga memiliki kesalahan/penyakit. Meski terlihat sederhana, radiasi gelombang yang dihasilkan CT Scan dan pengendali dapat membahayakan tubuh bila terlalu lama. Oleh karena itu, pencarian solusi dibutuhkan agar pergerakan *microbots* optimal, yaitu seminimum mungkin, dengan algoritme yang cepat. Salah satu algoritme yang dapat dimanfaatkan adalah *Greedy Best First Search*.

Keywords—*microbots, greedy best first search, bioteknologi, GBFS, Medis*

I. PENDAHULUAN

Perkembangan teknologi saat ini tidak hanya berfokus pada otomatisasi, tetapi juga mendorong pengurangan ukuran perangkat. Hal ini tercermin dalam perkembangan robotika. Robot dikembangkan dengan tujuan mempermudah kegiatan manusia yang berulang, seperti otomatisasi. Namun, dalam bidang bioteknologi, pengembangan robot juga telah dilakukan untuk tujuan pengujian, pemeriksaan, dan pengobatan makhluk hidup.

Sejak zaman dulu, perkembangan manusia selalu dikenali berdasarkan alat-alat diciptakan. Berbagai alat telah diciptakan mulai abad 19, tetapi belum ada yang mampu untuk memikirkan sebuah robot. Istilah "robot" ("robota" yang berarti "buruh membosankan") pertama kali diperkenalkan oleh penulis Ceko, Karel Čapek, dalam drama panggungnya yang terkenal, "R.U.R" (Rossum's Universal Robots), yang dipentaskan pada tahun 1920. Sejak saat itu, minat terhadap pengembangan robot semakin meningkat. Selama Perang Dunia II, kebutuhan akan otomatisasi dalam produksi industri meningkat pesat, mendorong pengembangan robot yang dapat melakukan tugas-tugas manusia yang berbahaya atau monoton. Robot pertama yang serupa dengan gambaran modern muncul pada tahun 1950-an dan 1960-an, seperti Unimate yang digunakan untuk keperluan industri.[1]

Robot industri dibuat khusus untuk melakukan tugas-tugas fisik yang berulang dengan akurasi yang tinggi di lingkungan pembuatan barang atau bahan material. Mereka digunakan dalam berbagai sektor industri, seperti otomotif, elektronik, dan

produksi massa, untuk melakukan pekerjaan seperti pengelasan, perakitan, dan pemindahan material. Robot industri biasanya memiliki lengan yang fleksibel, dapat diprogram, dan sering dilengkapi dengan sensor untuk beroperasi secara efisien dan aman di sekitar manusia.

Selain robot industri, robot pelayanan juga semakin populer. Robot pelayanan dirancang untuk berinteraksi dengan manusia dalam berbagai konteks, seperti di lingkungan pelayanan kesehatan, perhotelan, atau sektor ritel. Mereka dapat memberikan bantuan dan informasi kepada pengguna, menerima pembayaran, mengambil pesanan, atau bahkan memberikan layanan perawatan pribadi. Robot pelayanan seringkali memiliki antarmuka yang intuitif, seperti layar sentuh atau pengenalan suara, untuk berkomunikasi dengan pengguna secara efektif.

Selain itu, terdapat juga robot medis yang digunakan dalam dunia kesehatan. Robot medis dapat membantu dalam prosedur bedah yang presisi, seperti robot-assisted surgery. Mereka memungkinkan dokter untuk mengendalikan perangkat melalui kendali jarak jauh dan memberikan akses yang lebih kecil dan lebih tepat dalam melakukan tindakan operasi. Robot medis juga dapat digunakan dalam proses diagnosa, rehabilitasi, atau perawatan pasien.

Bidang-bidang seperti kecerdasan buatan, sensorika, dan otomatisasi semakin mendukung kemajuan dalam pengembangan robot. Dalam bidang bioteknologi, kebutuhan sebuah teknologi yang dapat melakukan pengujian, pemeriksaan, dan campur tangan di tingkat mikro, sehingga tercipta pengembangan robot mikro atau *microbots*.

Pengembangan robot mikro atau *microbots* memiliki tujuan awal yang khusus dalam bidang biologi, terutama pada aplikasi di bidang kesehatan. *Microbots* dirancang dengan ukuran sangat kecil, sebanding dengan ukuran sel-sel atau lebih kecil. Tujuan utama dari pengembangan *microbots* adalah untuk mengatasi tantangan dalam mengirimkan obat medis secara terarah dan spesifik ke dalam tubuh manusia. Selain aplikasi di bidang kesehatan, pengembangan *microbots* juga dapat diterapkan dalam bidang-bidang lain, seperti lingkungan, pertanian, dan teknologi laboratorium. *Microbots* dapat digunakan untuk melakukan pemantauan lingkungan yang presisi, mengumpulkan sampel tanah atau air, serta membantu dalam riset dan pengembangan di laboratorium.



Gambar 1 Robot medis untuk membantu operasi pada manusia

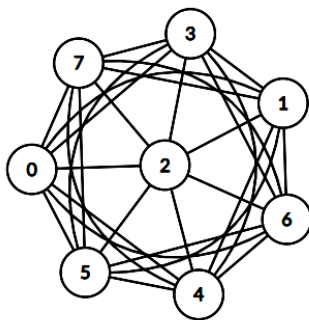
(sumber <https://www.alodokter.com/operasi-robotik-ini-yang-harus-anda-ketahui>)

II. LANDASAN TEORI

A. Graf

Graf, cabang ilmu matematika, merupakan gambaran struktur objek objek diskrit dan relasi antara objek tersebut. objek-objek tersebut dinyatakan dengan simpul (vertices atau nodes) dan hubungan antara objek dinyatakan dengan sisi (edge). Graf dapat di gambar dengan berbagai cara, salah satunya dengan sebuah titik atau lingkaran sebagai simpul dan garis sebagai sisi. Simpul biasanya diberi penanda sebuah angka atau huruf seperti gambar 2.

Dalam membahas graf, terdapat beberapa terminology. Salah satu terminology yang sering disebut, yaitu *ketetanggaan*. *Ketetanggaan* adalah dua simpul yang langsung terhubung (terdapat sisi yang menghubungkan). Terdapat juga istilah graf lengkap, yaitu graf dengan setiap simpul yang bertetangga dengan semua simpul secara langsung.



Gambar 2 Contoh graf lengkap dengan tujuh simpul

(sumber : dibuat dengan https://csacademy.com/app/graph_editor/)

B. Traversal Graf

Traversal graf merupakan teknik untuk mengunjungi simpul-simpul suatu graf secara sistematis. Graaf dapat direpresentasikan sebagai persoalan. Traversal graf adalah pencarian solusi persoalan tersebut.

Ketika mengunjungi simpul-simpul pada graf, traversal graf didekati dalam dua bentuk :

1) Graf Statis

Graf ini sudah terbentuk sebelum proses pencarian dilakukan. Graf ini direpresentasikan sebagai struktur data yang terdiri dari simpul dan sisi yang menghubungkan simpul-simpul tersebut. Setiap simpul dalam graf statis memiliki atribut atau informasi yang relevan, seperti nilai, label, atau jenis simpul tertentu. Graf statis sering digunakan ketika bentuk graf tidak berubah selama proses pencarian atau ketika informasi yang ada dalam graf sudah lengkap dan tidak perlu diperbarui.

2) Graf Dinamis

Graf dinamis terbentuk saat proses pencarian dilakukan. Graf ini tidak tersedia sebelum pencarian dimulai, melainkan dibangun secara bertahap selama proses pencarian solusi. Pada awalnya, graf dinamis bisa hanya memiliki simpul awal atau simpul awal yang diketahui, dan kemudian sisi-sisi tambahan ditambahkan saat pencarian berlanjut. Biasanya, graf dinamis digunakan ketika bentuk graf berubah selama proses pencarian, atau ketika struktur graf perlu disesuaikan dengan informasi baru yang ditemukan saat pencarian berlangsung.

Terdapat dua algoritma yang dapat dipakai untuk melakukan traversal graf, yaitu

1) Pencarian Tanpa Informasi (*Uninformed/Blind Search*)

Metode pencarian yang tidak memiliki informasi tambahan tentang hal yang berhubungan dengan tujuan pencarian kecuali simpul tujuan. Metode ini sering digunakan ketika tidak ada informasi tambahan yang tersedia untuk membantu dalam pencarian solusi. Metode ini mengandalkan struktur dasar graf atau pemetaan langkah-langkah pencarian untuk menemukan solusi. Beberapa contoh metode pencarian tanpa informasi termasuk Depth-First Search (DFS), Breadth-First Search (BFS), Depth Limited Search, Iterative Deepening Search, dan Uniform Cost Search.

2) Pencarian dengan Informasi (*Informed Search*)

Pencarian dilakukan dengan berbasis heuristik, melibatkan penggunaan informasi tambahan yang diberikan kepada algoritma pencarian. Dalam pencarian berbasis heuristik, langkah-langkah pencarian yang diambil lebih cenderung menuju keadaan yang dianggap lebih dekat dengan tujuan akhir. Contoh metode pencarian dengan informasi termasuk Best First Search dan algoritma A*. Metode-metode ini membantu mempercepat pencarian solusi dengan menggunakan informasi tambahan yang relevan tentang lingkungan atau tujuan.

C. Greedy Best First Search

Greedy Best First Search (GBFS) adalah algoritma pencarian yang menggunakan pendekatan heuristik untuk mencari simpul terbaik yang memiliki estimasi harga paling rendah menuju tujuan. Algoritme ini berfokus pada pengambilan keputusan yang didasarkan pada nilai heuristik

dari simpul saat ini, tanpa mempertimbangkan informasi tentang biaya yang telah ditempuh sebelumnya.

Fungsi estimasi tersebut adalah $h(n)$. Fungsi estimasi ini menjadi harga yang akan dipilih minimumnya berdasarkan simpul-simpul yang dapat dituju, sehingga terbentuk $f(n) = h(n)$

$f(n)$: total harga sebuah simpul ekspan

$h(n)$: fungsi heuristik

GBFS akan memilih simpul hidup dengan $f(n)$ minimum saat itu, $f(n)$ saat minimum disebut local minima atau *plateau*.

Terdapat beberapa hal yang perlu diketahui tentang istilah yang digunakan dalam algoritme, yaitu:

- **Simpul hidup**
Simpul hidup adalah simpul dalam graf yang dapat dikunjungi atau dieksplorasi selama proses pencarian. Pada saat pencarian berlangsung, simpul hidup merupakan simpul yang belum dikunjungi dan memiliki potensi untuk menjadi bagian dari jalur solusi. Setiap simpul hidup akan dievaluasi atau dianalisis selama pencarian untuk menentukan langkah berikutnya.
- **Simpul ekspan**
Simpul ekspan adalah simpul yang sedang dikunjungi atau sedang dieksplorasi dalam proses pencarian. Simpul ekspan adalah simpul yang sedang dipertimbangkan sebagai langkah selanjutnya dalam upaya mencari solusi.
- **Simpul tujuan**
Simpul tujuan adalah simpul ekspan yang menyatakan solusi dari suatu persoalan. Simpul tujuan biasanya mencerminkan kondisi atau konfigurasi yang dicari dalam masalah yang sedang diselesaikan. Ketika pencarian mencapai simpul tujuan, itu berarti solusi telah ditemukan dan pencarian dapat dihentikan. Simpul tujuan juga bisa merupakan simpul yang memenuhi kriteria tertentu atau mencapai tujuan yang telah ditentukan sebelumnya.

Greedy Best First Search (GBFS) memiliki kelebihan. Kelebihannya adalah kecepatan pencariannya yang relatif cepat. Algoritme ini hanya fokus pada simpul dengan nilai heuristik terendah menuju tujuan sehingga memungkinkan untuk mengurangi ekspansi tidak perlu. Hal ini memungkinkan GBFS menemukan solusi yang memadai dalam waktu singkat.

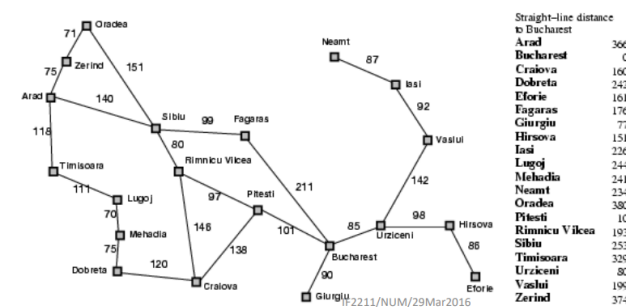
Namun, GBFS juga memiliki kelemahan. GBFS kurang mampu untuk mengubah jalur pencarian jika simpul dengan nilai heuristik lebih rendah menghasilkan jalur yang lebih panjang. Ini bisa mengakibatkan GBFS terjebak dalam jalur yang tidak optimal atau bahkan tidak menemukan solusi. Selain itu, GBFS rentan terhadap masalah dengan heuristic yang tidak akurat. Jika fungsi heuristik tidak memberikan estimasi yang tepat tentang jarak menuju tujuan, GBFS dapat menghasilkan solusi yang jauh dari optimal.

Dalam penggunaan GBFS, perlu mempertimbangkan antara kecepatan dan kualitas solusi. Jika kecepatan penting, GBFS dapat menjadi pilihan yang baik.

GBFS bekerja dengan cara dengan langkah sebagai berikut

1. Inisiasi, siapkan wadah simpul hidup dan jalur berbentuk larik
2. Masukkan simpul permulaan kedalam larik simpul hidup
3. Keluarkan simpul dari larik simpul hidup yang memiliki nilai harga, $f(n)$, yang paling kecil. Simpul ini menjadi simpul ekspan. Masukkan simpul ini kedalam jalur.
4. Jelajahi simpul ekspan, dapat kan simpul-simpul yang terhubung dengan simpul ekspan.
5. Cek untuk setiap simpul yang terhubung dengan simpul ekspan apakah merupakan simpul tujuan, jika benar, masukkan simpul tujuan tersebut kedalam jalur kemudian hentikan iterasi. Jika bukan, periksa apakah simpul tersebut sudah ada dalam larik simpul hidup atau larik simpul ekspan, jika belum masukkan simpul kedalam larik simpul hidup.
6. (opsional) Evaluasi kembali nilai heuristik untuk setiap simpul dalam larik simpul hidup.
7. Ulangi dari poin ke 2

Pada poin ke-lima, pencarian dihentikan karena sudah memenuhi syarat, yaitu mencapai simpul tujuan. Poin ke-enam merupakan opsional karena bergantung pada fungsi heuristik apa yang dipakai dan apakah nilai heuristik simpul berubah setiap bergerak atau tidak. Meskipun ada cara lain yaitu dengan menyimpan hasil perhitungan dan evaluasi heuristik setiap simpul kedalam sebuah larik, seperti pada gambar 3, terdapat nilai heuristik untuk setiap simpul ke simpul tujuan (Bucharest).



Gambar 3 Graf yang memiliki informasi heuristic jarak garis lurus ke simpul tujuan (Bucharest)

(sumber :

<https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Route-Planning-Bagian1-2021.pdf>)

D. Microbots dalam Pembuluh Darah Manusia

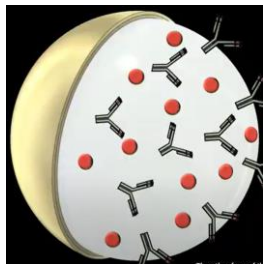
Pembuluh darah manusia berbentuk tabung yang terhubung ke seluruh bagian dalam tubuh. Ukuran dari pembuluh darah rata-rata 60 mikrometer untuk selain pembuluh darah kapiler (pembuluh darah kapiler memiliki urang yang sangat kecil).

Pada dinding dari pembuluh darah terdapat bagian yang digunakan untuk menyerap sebuah nutrisi yang dibawa oleh sel darah merah. Bagian inilah yang sering menjadi sarang penyakit. Tidak hanya itu, beberapa bakteri atau tumor dapat

tumbuh di dinding tersebut. Tumor tersebut dapat berupa gundukan nutri yang tidak baik sehingga dapat menyebabkan aliran darah tersumbat. Tumor juga bisa berupa kematian cell di tandai dengan mengumpulnya sel darah putih pada pembuluh darah tersebut. Hal tersebut dapat dilihat melalui CT Scan.

CT Scan, singkatan dari Computerized Tomography Scan, adalah sebuah metode pencitraan medis yang menggunakan sinar-X dan teknologi komputer untuk menghasilkan gambar atau vide detail dari bagian dalam tubuh. Dalam konteks pembuluh darah manusia, CT Scan dapat membantu dalam mendeteksi adanya penyumbatan, peradangan, atau kelainan struktural pada pembuluh darah. Namun, CT Scan juga memiliki risiko jika dilakukan lama. Sinar-X dapat meningkatkan risiko kerusakan sel dan berpotensi menyebabkan kerusakan DNA. Oleh karena itu, penting bagi tenaga medis untuk memastikan bahwa CT Scan dilakukan dengan dosis sinar-X yang aman dan sesuai dengan kebutuhan medis yang mendesak

Microbots yang digunakan dapat memanfaatkan amoeba yang ditemplei dengan partikel magnetis. *Micorbots* lain yang bisa digunakan yaitu robot berbentuk bola seperti pada gambar 4 yang baru ini dikembangkan, bagian magnetis ada pada bagian terluar. Dengan konsep kerja yang sama, yaitu bagian magnetis digunakan sebagai reseptor untuk melakukan gerakan. Gerakan tersebut dikendalikan oleh medan magnet yang diatur sehingga bergerak sesuai keinginan seperti pada gambar 5. Karena gerakan bergantung pada medan magnet, gerakan *microbot* hanya berada pada permukaan. Baik pada amoeba atau robot bola, keduanya ditemplei dengan kandungan untuk melakukan pengobatan. Kandungan ini nantinya akan dilepaskan dan berpengaruh pada suatu area.



Gambar 4 *Microbots* ciptaan untuk bergerak dalam pembuluh darah.

(sumber : <https://www.youtube.com/watch?v=IviRupap7SY>)



Gambar 5 Pergerakan *microbots* yang diatur oleh medan magnet

(sumber : <https://www.youtube.com/watch?v=IviRupap7SY>)

III. ANALISIS KASUS DAN SOLUSI

A. Analisis Kasus

Tugas pengontrol (seseorang atau program) adalah untuk mengunjungi bagian-bagian yang terdapat masalah dan menyebarkan kandungan tersebut. Kandungan yang dapat robot keluarkan tidak hanya sekali. Bagian yang terdapat masalah biasanya memiliki anomali seperti warna putih atau hitam secara lambat bergerak (seperti darah tersumbat), bentuk dinding pembuluh darah yang mengalami anomali, dan lainnya. Akan tetapi, dengan teknologi sekarang masih sulit untuk menentukan dengan mata telanjang sehingga bagian yang mengalami masalah masih merupakan dugaan.

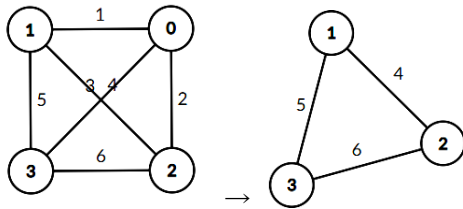
B. Pemetaan Masalah Menjadi Graf

Sebelum menggunakan algoritma *Greedy Best First Search*. Pemetaan masalah menjadi sebuah graf harus dilakukan. Pemetaan tersebut harus jelas sehingga solusi dapat tepat terapkan. Oleh karena itu hal yang pertama dilakukan adalah dengan asumsi yang tidak berlebihan maupun disamakan. Asumsi-asumsi pada pemetaan masalah adalah sebagai berikut:

1. Terdapat suatu fungsi untuk memetakan gambar/video menjadi matriks 0/1 dengan 1 adalah daerah yang diduga mengalami kesalahan. Dan matrix posisi yang berkorespondensi satu-satu ke matriks 0/1.
2. *Microbots* akan diinjeksi pada bagian tengah dari daerah-daerah yang memiliki masalah. Hal ini maksudnya *microbots* tidak diinjeksi secara asal dan menunggu robot tersebut hingga sampai pembuluh darah yang dituju.
3. *Microbots* memiliki kecepatan yang hanya berdasarkan medan magnet dan tidak terpengaruh arus aliran darah. Artinya medan magnet cukup tepat dalam mengendalikan robot.
4. Disebabkan *microbot* hanya bergerak di dinding pembuluh darah, dapat diasumsikan bahwa letak/koordinat suatu masalah dipetakan menjadi dua dimensi.
5. *Microbots* dapat bergerak bebas ke arah manapun dalam permukaan/dinding pembuluh darah. Karena pergerakan yang menggelinding (robot bola) atau fleksibel (amoeba) pergerakan hanya dibatasi dinding dan medan magnet.

Graf lengkap dapat dibentuk dari asumsi-asumsi tersebut. *Microbots* dapat bergerak bebas kemanapun sehingga dalam *microbots* dapat bergerak ke semua arah dan dari arah manapun.

Ilustrasi pada gambar 6, misal 0 adalah simpul awal *microbot* dan simpul yang digambarkan merupakan simpul yang dicurigai. Rute terbentuk adalah $0 \rightarrow 1 \rightarrow 2 \rightarrow 3$ dengan jarak tempuh $1+4+6 = 11$ jika heuristik menggunakan jarak euclidean (jarak garis lurus).



Gambar 6 Contoh graf yang terbentuk
(sumber : dibuat dengan

https://csacademy.com/app/graph_editor/)

Sehingga masalah yang dihadapi adalah bagaimana menentukan rute tercepat ke semua simpul yang dicurigai pada graf lengkap. Nilai heuristik yang dapat diambil sama dengan jarak dari simpul ke simpul lain dalam simpul hidup

C. Analisis Solusi Graf

Graf diselesaikan dengan algoritme *Greedy Best First Search*. Penggunaan GBFS membutuhkan informasi berupa koordinat tiap simpul sehingga dapat mengetahui jaraknya. Penggunaan GBFS pada masalah ini akan mengarah ke solusi yang optimal.

Misal sebuah himpunan simpul $V = \{v_1, v_2, \dots, v_n\}$ dan himpunan jarak heuristik $H = \{h_1, h_2, \dots, h_n\}$ dengan h_1, h_2, \dots, h_n masing masing adalah himpunan jarak heuristik simpul n ke setiap simpul. Pengambilan simpul selanjutnya dari simpul v_n akan menggunakan minimum dari h_n . Dengan demikian, total jarak yang ditempuh, misal S , adalah

$$S = \min(h_1) + \min(h_2) + \dots + \min(h_n)$$

Metode GBFS dipastikan dapat menemukan jarak S minimum dengan cepat dan tidak perlu mengunjungi simpul yang sama lebih dari satu kali.

D. Analisis Dampak

Penggunaan algoritme ini tentunya dapat meminimumkan waktu pasien berada dalam CT Scan dan medan magnet pengendali. Kedua hal tersebut memancarkan gelombang radiasi dengan intensitas yang kecil. Oleh karena itu diperlukan dosis yang tepat agar gelombang radiasi tidak merusak tubuh. Dengan meminimalkan waktu penggunaan gelombang dari medan magnet pengendali, jumlah radiasi yang dipancarkan ke tubuh dapat diminimumkan.

IV. IMPLEMENTASI DAN PENGUJIAN

Untuk implementasi code dapat dilihat pada <https://github.com/akmaldika/Strategy-Algorithm-Paper-Project>

Tabel I merupakan spesifikasi lingkungan pengujian berlangsung.

Tabel I Spesifikasi pengujian code dijalankan

Spesifikasi	Keterangan
Sistem Operasi	Windows 11 Home Single Language 22H2
Prosesor	11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz 2.80 GHz
Ram	8 GB

Compiler	Phyton 3.11.2
----------	---------------

Pengujian dilakukan dengan mengetes seberapa lama waktu yang diperlukan untuk menjelajahi simpul-simpul. Untuk setiap pengujian kasus, akan ditampilkan (dapat melihat gambar 7 untuk melihat keluaran pada setiap eksekusi:

- Matriks 0/1 berkorespondensi matriks ukuran $n \times n$ posisi yang dihasilkan secara random dengan posisi diantara 1 – 999 micrometer
- Posisi dan kecepatan microbots dibuat sama yaitu pada posisi (500,500) micrometer dengan kecepatan 100 micrometer/detik.
- Letak koordinat dengan matriks 0/1 bernilai 1 dan jarak euclidean dari posisi awal (graf awal terbentuk)
- Hasil rute yang dilalui
- Total waktu yang dibutuhkan *microbots*
- Total waktu yang dibutuhkan compiler

```

Matrix 0/1:
0 1
1 0
Position Matrix:
(856.522, 271.784) (786.629, 802.023)
(153.296, 294.362) (285.197, 454.649)
Positions of 1s: coordinates = distance from microbots:
(0, 1): (786.629, 802.023) = 416.382 µm
(1, 0): (153.296, 294.362) = 403.101 µm
Greedy Best First Search:
(500.0, 500.0) -> (153.296, 294.362) -> (786.629, 802.023)
Time microbot takes: 14.541 µs
Time elapsed: 0.000998 s

```

Gambar 7 Contoh keluaran pada setiap eksekusi
(sumber: dokumentasi pribadi)

Hal yang diatur dari program hanya ukuran matriks.

Berikut merupakan pengujian yang dilakukan. Untuk setiap pengujian, hanya akan dilampirkan waktu eksekusi dan waktu microbots secara rata rata.

1. Pengujian $n = 3$ dengan 3000 kali pengujian
Berikut merupakan hasil uji :

```

Time microbot takes: 23.681 s
Time elapsed: 0.0 s

```

Gambar 8 Salah satu hasil uji untuk $n = 3$

- Rata-rata microbot menggapai semua simpul = 10.70696155 detik
- Rata-rata waktu eksekusi = 23.8921 micordetik

2. Pengujian $n = 5$ dengan 1000 kali pengujian
Berikut merupakan hasil uji :

```

Time microbot takes: 125.927 s
Time elapsed: 0.0 s

```

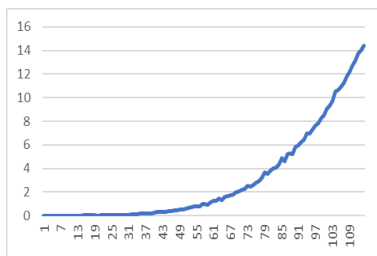
Gambar 9 Salah satu hasil uji untuk $n = 5$

- Rata-rata microbot menggapai semua simpul = 17.36911463 detik
 - Rata-rata waktu eksekusi = 58.2318 micordetik
3. Pengujian n = 10 dengan 1000 kali pengujian Berikut merupakan hasil uji :

```
Time microbot takes: 382.769 s
Time elapsed: 0.0 s
```

Gambar 9 Salah satu hasil uji untuk n = 10

- Rata-rata microbot menggapai semua simpul = 96.0921 detik
 - Rata-rata waktu eksekusi = 34.82494 micordetik
4. Pengujian dengan n = 100 dengan 1000 kali pengujian Berikut merupakan hasil uji :
- Rata-rata microbot menggapai semua simpul = 5575.784 detik
 - Rata-rata waktu eksekusi = 0.780702415 detik
5. Pengujian dengan n menaik dari 1 hingga 100 Berikut merupakan hasil waktu eksekusi dalam grafik



Gambar 9 Grafik kenaikan waktu eksekusi terhadap ukuran matriks

V. KESIMPULAN DAN SARAN

Algoritme *Greedy Best First Search* dapat diaplikasikan pada hasil scan untuk menunjukkan jalan selanjutnya. Algoritma ini akan memberikan waktu berbentuk parabola sehingga perkiraan kompleksitas $O(n^2)$.

Algoritme GBFS ini efektif untuk meminimalkan jarak yang harus ditempuh apabila harus mengunjungi simpul setiap graf lengkap. Penerapan GBFS dengan menambahkan kasus lain dapat membuat algoritme ini menjadi lebih akurat. Akurat yang dimaksud adalah keakuratan berdasarkan ilmu-ilmu kesehatan dan bioteknologi atau ilmu lain yang ingin memanfaatkan teknik yang sama.

Kajian dari makalah ini dapat dikembangkan lebih lanjut dalam merancang *microbots* dan perkembangannya. Sebagai contoh, *microbots* sekarang yang hanya dapat bergerak dalam permukaan mungkin bisa dikembangkan untuk dapat bergerak secara tiga dimensi, memungkinkan pergerakan *microbots* lebih cepat dan bebas. Dengan adanya makalah ini, diharapkan bahwa penyelesaian berbagai masalah dapat dipandang dari sudut yang berbeda sehingga menghasilkan strategi algoritma yang lebih efisien dan optimal dalam proses komputasi berbagai penyelesaian masalah.

UCAPAN TERIMA KASIH

Puji dan syukur penulis panjatkan ke hadirat Allah SWT atas limpahan Rahmat, karunia, dan hidayah-Nya sehingga penulis dapat menyelesaikan penulisan makalah ini dengan baik. Penulis berterima kasih kepada keluarga dan teman-teman yang telah memberi dukungan kepada penulis dalam menyelesaikan makalah ini. Penulis juga berterima kasih kepada dosen pengampu mata kuliah Strategi Algoritme Semester 4 2022/2023 kelas K02, Ibu Dr. Nur Ulfa Maulidevi, S.T., M.Sc. atas pengajaran yang beliau berikan sehingga saya bisa memahami materi, terutama materi Greedy Best First Search ini dengan lebih baik. Terakhir, penulis memohon maaf apabila di dalam penulisan makalah ini terdapat kesalahan baik disengaja, maupun tidak disengaja. Penulis berharap makalah ini dapat bermanfaat bagi banyak orang.

REFERENCES

- [1] Website <https://elib.unikom.ac.id/files/disk1/494/jbptunikompp-gdl-asephendra-24693-4-unikom-a-i.pdf> diakses pada 22 Mei 2023 pukul 15:00 GMT +7
- [2] Website https://www.researchgate.net/publication/263547830_Robotics_in_the_s_mall diakses pada 22 Mei 2023 pukul 15:20 GMT+7
- [3] Munir. Rinaldi, "Graf.", Informatika. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf> (diakses pada 22 Mei 2023)
- [4] Munir. Rinaldi, "Graf.", Informatika. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian2.pdf> (diakses pada 22 Mei 2023)
- [5] Munir. Rinaldi, "Breadth/Depth First Search (BFS/DFS) (bagian 1).", Informatika. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/BFS-DFS-2021-Bag1.pdf> (diakses pada 22 Mei 2023)
- [6] Munir. Rinaldi, "Penentuan Rute (Route/Path Planning) Bagian 1: BFS, DFS, UCS, Greedy Best First Search.", Informatika. <https://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2020-2021/Route-Planning-Bagian1-2021.pdf> (diakses pada 22 Mei 2023)
- [7] Chemical & Engineering News, "Microrobots roll against blood flow to deliver drugs", <https://www.youtube.com/watch?v=IviRupap7SY> (diakses pada 22 Mei 2023)
- [8] Biba, Jacob, 2022, "Microrobotics: Tiny Robots and Their Many Uses" <https://builtin.com/robotics/microrobotics> (diakses pada 22 Mei 2023)
- [9] NHS, "CT Scan", <https://www.nhs.uk/conditions/ct-scan/> (diakses pada 22 Mei 2023)

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 22 Mei 2023

Akmal Mahardika Nurwahyu Pratama
13521070

