

**Perbandingan Algoritma Greedy Search dan Algoritma Depth-First-Search pada Pencarian Kota dengan Graph Romania Problem*****Comparison of Greedy Search Algorithm and Depth-First-Search Algorithm on City Search with Romanian Graph Problem***

Muhammad Yudha Syuhada*, Sudirman

Universitas Sumatera Utara, STMIK & AMIK Logika Medan

*E-mail Korespondensi : yudha.shuhada@gmail.com**Abstrak**

Pada jurnal ini, akan dibahas penggunaan algoritma Greedy Search dan Depth-First-Search untuk mencari sebuah kota (*final state*) dari kota asal (*initial state*). Kedua algoritma akan dibandingkan berdasarkan total *cost* dari path yang terbentuk oleh pencarian masing-masing algoritma. *Path* yang digunakan adalah Romania Problem yang mana *path* ini sering digunakan pada kasus-kasus kecerdasan buatan. Jurnal ini juga mendiskusikan perbandingan antara kinerja kedua algoritma yang mana secara umum akan menunjukkan algoritma mana yang lebih efisien dalam menemukan solusi.

Kata kunci: *Greedy-Search, DFS, Romania Problem***Abstract**

In this paper, we will discuss the use of Greedy Search and Depth-First-Search algorithms to find a city (final state) of the original city. Both algorithms will be compared based on the total cost of the path formed by the search of each algorithm. The path used is the Romanian Problem where this path is often used in cases of artificial intelligence. The journal also discusses the comparison between the performance of the two algorithms which will generally indicate which algorithm is more efficient in finding solutions.

Keywords: *Greedy-Search, DFS, Romania Problem*

9. expand child dari X
10. masukkan X dalam closed
11. hapus child dari X yang ada di closed atau open
12. end
13. return(gagal)
14. end
15. end

Dapat dilihat dari *pseudocode* diatas bahwa memang terbukti DFS akan selalu *complete* apabila *graph* memiliki sifat *infinite state space* karena lambat laun, DFS akan mengekskansi seluruh node yang ada.

HASIL DAN PEMBAHASAN

Pengujian akan dilakukan dengan membandingkan total *cost* dari kedua algoritma pencarian. Dimana akan diambil 10 sampel *initial* dan *final state* yang kemudian akan dibandingkan hasilnya. Hasil pengujian dapat dilihat pada Tabel.1.

| Initial | Final | Total Cost | |
|-----------|-----------|------------|------|
| | | Greedy | DFS |
| Arad | Bucharest | 575 | 607 |
| Lasi | Bucharest | Fail | 319 |
| Mehadia | Bucharest | 874 | 434 |
| Sibiu | Neamt | Fail | 716 |
| Lugoj | Pitesti | 403 | 403 |
| Bucharest | Craiova | Fail | 625 |
| Giurgiu | Evorie | 359 | 359 |
| Oradea | Zerind | 71 | 1374 |
| Fagaras | R.Vilcea | 179 | 409 |
| Drobeta | Sibiu | 671 | 669 |

Tabel.1. Tabel Perbandingan Cost

SIMPULAN

Dari hasil pengujian, dapat disimpulkan bahwa pada *graph* Romania Problem, algoritma DFS akan selalu menemukan solusi jika dibandingkan dengan algoritma Greedy. Ini sesuai dengan teori dimana DFS akan selalu menemukan solusi dikarenakan DFS lambat laun akan mengekskansi seluruh node pada *graph*. Namun, untuk beberapa kasus tertentu, DFS sangat tidak efisien dimana pada kasus Oradea-Zerind, yang mana Zerind merupakan *child* langsung dari Oradea namun DFS mengekskansi Sibiu karena Sibiu merupakan node *child* pertama pada *adjacency list* yang digunakan.

Algoritma Greedy adalah algoritma yang tidak menjamin menemukan solusi, dapat dilihat pada Tabel.1, bahwa ada tiga kasus algoritma ini tidak menemukan solusi walaupun di kasus lain, algoritma ini dapat sangat efisien tergantung dari *graph* yang digunakan.

Secara umum, DFS lebih dapat digunakan untuk menemukan solusi daripada Greedy. Namun ada beberapa kasus DFS tidak lebih baik dari algoritma Greedy.

DAFTAR PUSTAKA

- Annu Malik, Anju Sharma, Mr. Vinod Saroha, "Greedy Algorithm", *International Journal of Scientific and Research Publications*, Volume 3, Issue 8, 2013.
- Jon Freeman, "Parallel Algorithms for Depth-First Search", *University of Pennsylvania Departement of Computer and Information Science Technical Report No. MS-CIS-91-71*, 1991

Prama, Irvan, dkk. *Makalah Algoritma Greedy untuk Mencari Lintasan Terpendek*, Departemen Teknik Informatika ITB, 2005.