

Informed Search

Ali Akbar Septiandri

November 25, 2020

Universitas Al Azhar Indonesia

1. Uniform Cost Search
2. Informed Search
3. A*

Uniform Cost Search

- s_{start} : kondisi awal
- $Actions(s)$: kemungkinan aksi
- $Cost(s, a)$: ongkos aksi
- $Succ(s, a)$: suksesor
- $IsEnd(s)$: kondisi akhir?

Uniform cost search

- Ekspansi simpul dengan ongkos paling kecil

Uniform cost search

- Ekspansi simpul dengan ongkos paling kecil
- Implementasi: Gunakan antrean dengan prioritas

Uniform cost search

- Ekspansi simpul dengan ongkos paling kecil
- Implementasi: Gunakan antrean dengan prioritas
- Ekuivalen dengan BFS saat ongkosnya dianggap sama

Uniform cost search

- Ekspansi simpul dengan ongkos paling kecil
- Implementasi: Gunakan antrean dengan prioritas
- Ekuivalen dengan BFS saat ongkosnya dianggap sama
- Solusinya dijamin optimal

Dalam graf lengkap, algoritma ini ekuivalen
dengan algoritma **Dijkstra**¹

¹Akan dipelajari di strategi algoritma

Algoritma ini dikenal juga dengan nama **cheapest-first**

Tiga komponen

1. **Explored**: simpul yang sudah diketahui jalur optimalnya
2. **Frontier**: simpul yang sudah dilihat, tapi masih perlu dicari tahu jalan terpendeknya
3. **Unexplored**: simpul yang belum dilihat

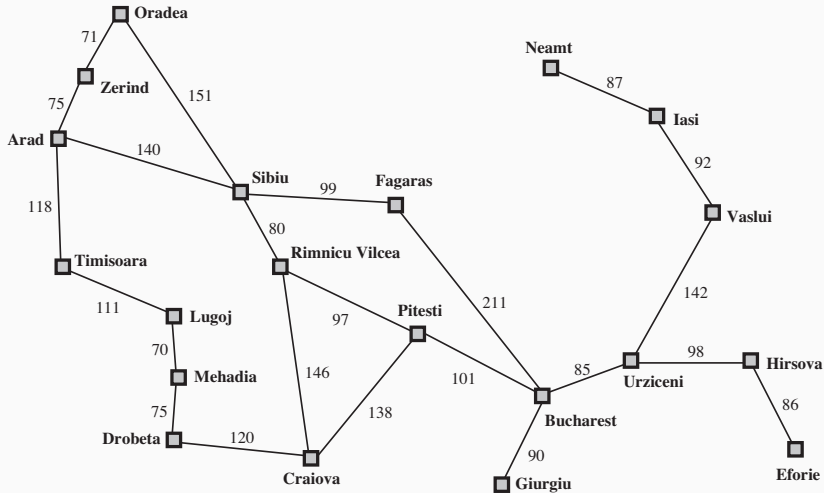


Figure 1: Peta Romania yang disederhanakan

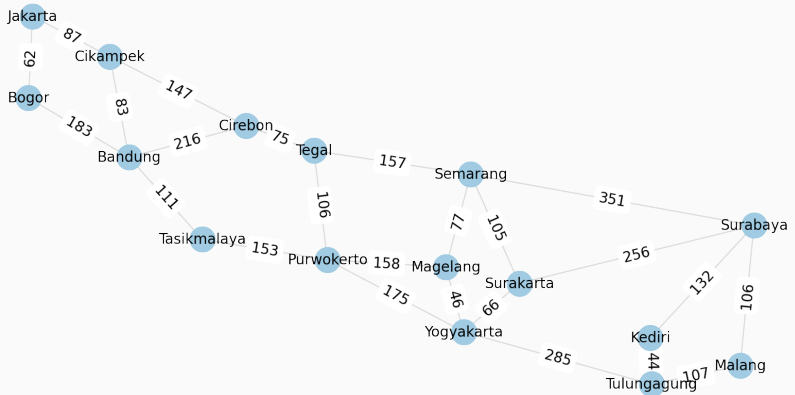


Figure 2: Peta Jawa yang disederhanakan

Algoritma

Tambahkan s_{start} ke **frontier** (antrean prioritas)

begin

repeat

if $IsEnd(s)$ **then**

 | **return** solution

end

 Tambahkan s ke dalam **explored**

for aksi $a \in Actions(s)$ **do**

 | Ambil suksesor $s' \leftarrow Succ(s, a)$

if $s' \in explored$ **then**

 | continue

end

 Perbarui **frontier** dengan s' dan prioritas $p + Cost(s, a)$

end

until $frontier = \emptyset$

end

Informed Search

Apakah UCS bisa diperbaiki?

- Perhatikan bahwa eksplorasi dilakukan hanya ke solusi yang dekat dengan titik awal

Apakah UCS bisa diperbaiki?

- Perhatikan bahwa eksplorasi dilakukan hanya ke solusi yang dekat dengan titik awal
- Bisa jadi, solusi tersebut **menjauhi titik tujuan**

Apakah UCS bisa diperbaiki?

- Perhatikan bahwa eksplorasi dilakukan hanya ke solusi yang dekat dengan titik awal
- Bisa jadi, solusi tersebut **menjauhi titik tujuan**
- Perlu memanfaatkan **heuristik**, i.e. semakin dekat titik tersebut dengan tujuan, semakin baik

Heuristik $h(s)$ adalah **estimasi** apapun untuk
FutureCost(s)

A*



Eksplorasi dengan urutan

$$\text{PastCost}(s) + h(s)$$

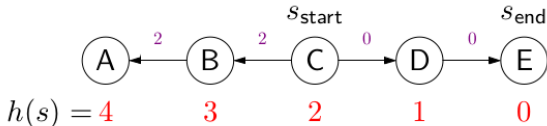
Carian A*

Algoritma

Jalankan A* dengan perubahan ongkos simpel:

$$\text{Cost}'(s, a) = \text{Cost}(s, a) + h(\text{Succ}(s, a)) - h(s)$$

Example



$$\text{Cost}'(C, B) = \text{Cost}(C, B) + h(B) - h(C) = 1 + (3 - 2) = 2$$

Heuristik h **konsisten** jika

- $Cost'(s, a) = Cost(s, a) + h(Succ(s, a)) - h(s) \geq 0$
- $h(s_{end}) = 0$

Relaksasi

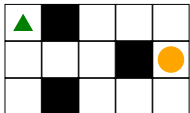
Rintangan membuat hidup lebih sulit.
Bayangkan *dunia tanpa rintangan* untuk
membantu membentuk heuristik.

Closed form solution

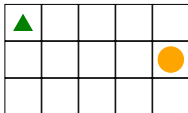


Example: knock down walls

Goal: move from triangle to circle



Hard



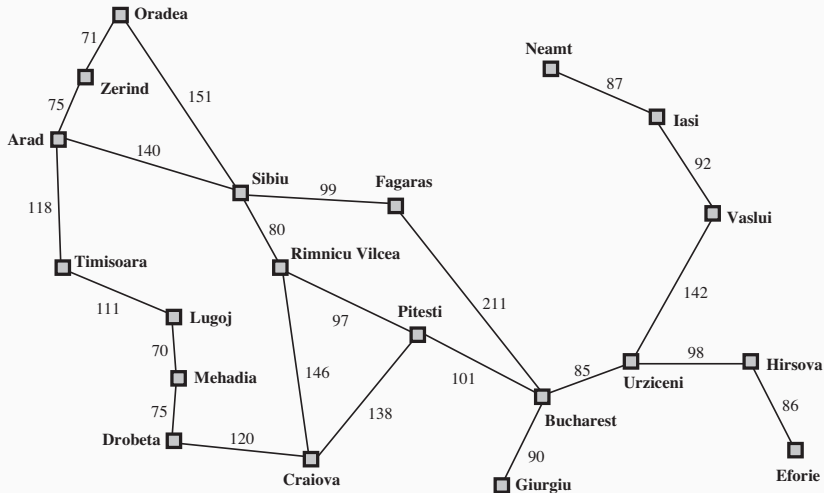
Easy

Heuristic:

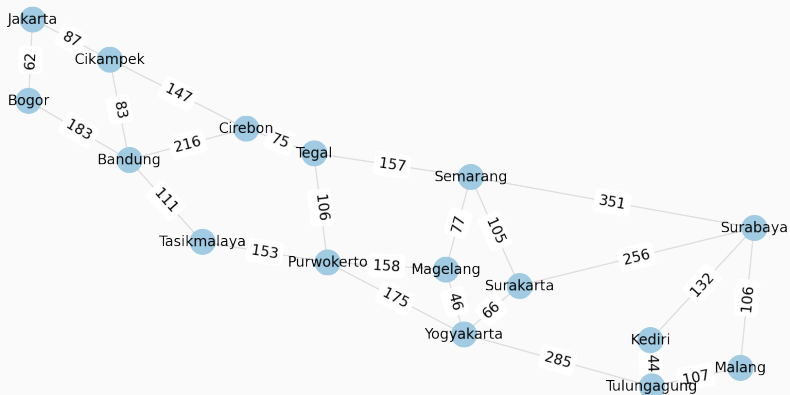
$$h(s) = \text{ManhattanDistance}(s, (2, 5))$$

$$\text{e.g., } h((1, 1)) = 5$$

Arad	366	Mehadia	241
Bucharest	0	Neamt	234
Craiova	160	Oradea	380
Drobeta	242	Pitesti	100
Eforie	161	Rimnicu Vilcea	193
Fagaras	176	Sibiu	253
Giurgiu	77	Timisoara	329
Hirsova	151	Urziceni	80
Iasi	226	Vaslui	199
Lugoj	244	Zerind	374



Bandung	570	Purwokerto	389
Bogor	662	Semarang	259
Cikampek	589	Surabaya	0
Cirebon	464	Surakarta	215
Jakarta	659	Tasikmalaya	503
Kediri	98	Tegal	402
Magelang	281	Tulungagung	100
Malang	33	Yogyakarta	266



Pertemuan berikutnya

- Evolutionary algorithm
- Genetic algorithm

Beberapa materi dari salindia ini diadaptasi
dari **Caltech CS154** dan **Stanford CS221**.

Terima kasih