

# Informed Search

---

Ali Akbar Septiandri

December 4, 2017

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**<sup>1</sup>

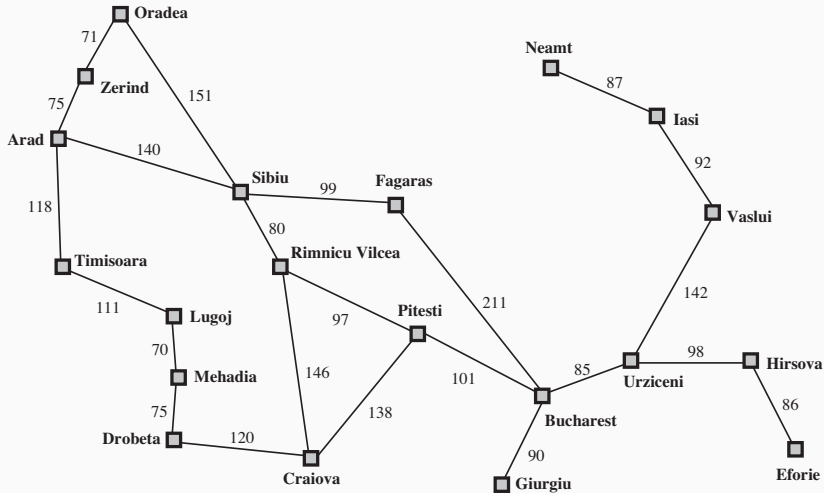
---

<sup>1</sup>Akan 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

# 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

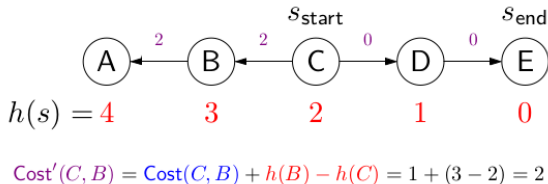
$$\text{PastCost}(s) + h(s)$$

## Algoritma

Jalankan A\* dengan perubahan ongkos simpel:

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

## Example



# Heuristik yang konsisten

Heuristik  $h$  **konsisten** jika

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

## Relaksasi

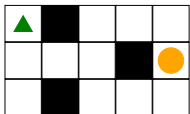
Rintangan membuat hidup lebih sulit.  
Bayangkan *dunia tanpa rintangan* untuk  
membantu membentuk heuristik.

# Closed form solution

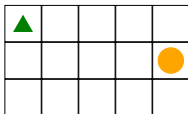


## Example: knock down walls

Goal: move from triangle to circle



Hard



Easy

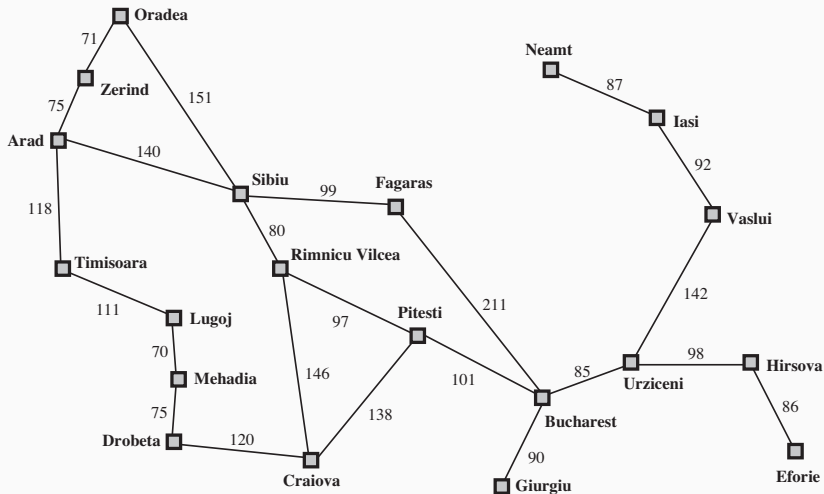
Heuristic:

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

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



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



## Pertemuan berikutnya

- Evolutionary algorithm
- Genetic algorithm

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

Terima kasih