Foundations of Artificial Intelligence

16. State-Space Search: Greedy BFS, A*, Weighted A*

Malte Helmert

University of Basel

March 29, 2021

Foundations of Artificial Intelligence

March 29, 2021 — 16. State-Space Search: Greedy BFS, A*, Weighted A*

State-Space Search: Overview

Chapter overview: state-space search

- 5.–7. Foundations
- 8.–12. Basic Algorithms
- 13.–19. Heuristic Algorithms
  - 13. Heuristics
  - 14. Analysis of Heuristics
  - 15. Best-first Graph Search
  - 16. Greedy Best-first Search, A*, Weighted A*
  - 17. IDA*
  - 18. Properties of A*, Part I
  - 19. Properties of A*, Part II

16.1 Introduction

What Is It About?

In this chapter we study last chapter's algorithms in more detail:

- greedy best-first search
- A*
- weighted A*

16.2 Greedy Best-first Search

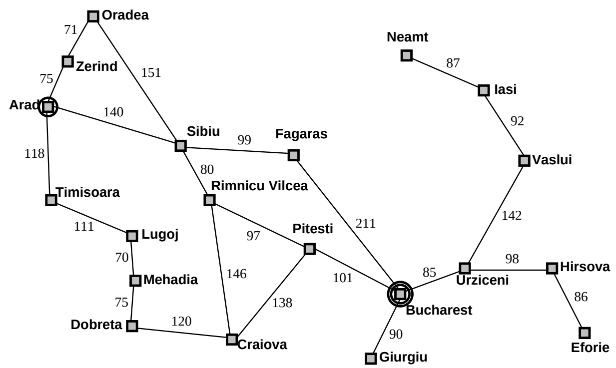
Greedy Best-first Search

Greedy Best-first Search

only consider the heuristic: $f(n) = h(n.\text{state})$

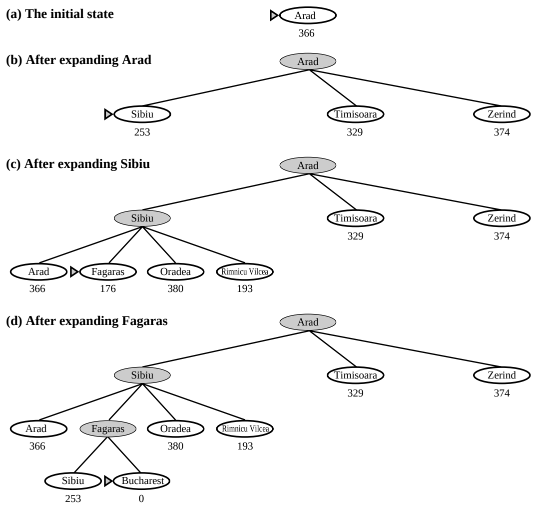
Note: usually without reopening (for reasons of efficiency)

Example: Greedy Best-first Search for Route Planning



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

Example: Greedy Best-first Search for Route Planning



Greedy Best-first Search: Properties

- complete with safe heuristics (like all variants of best-first graph search)
- suboptimal: solutions can be arbitrarily bad
- often very fast: one of the fastest search algorithms in practice
- monotonic transformations of h (e.g. scaling, additive constants) do not affect behaviour (Why is this interesting?)

16.3 A*

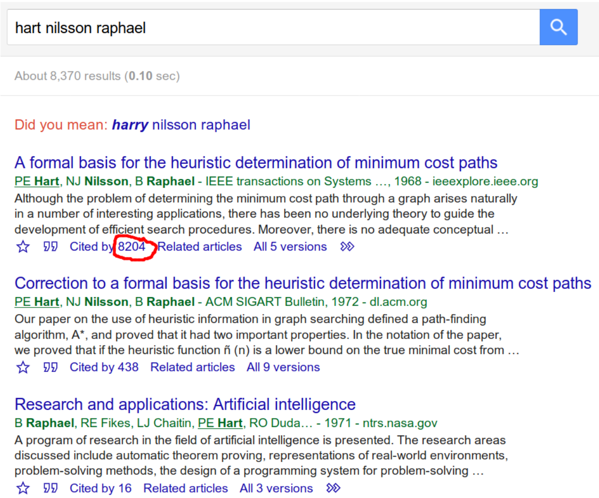
A*

A*

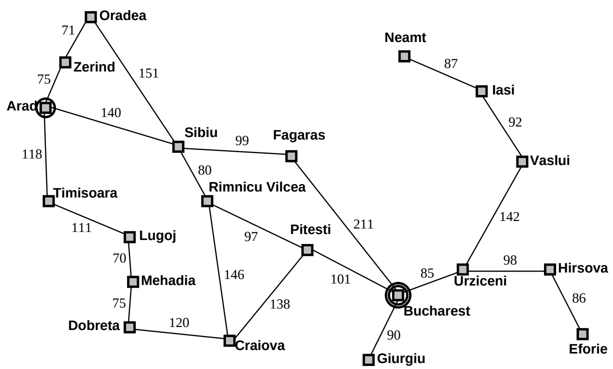
combine greedy best-first search with uniform cost search:
$f(n) = g(n) + h(n.\text{state})$

- trade-off between path cost and proximity to goal
- $f(n)$ estimates overall cost of cheapest solution from initial state via n to the goal

A*: Citations

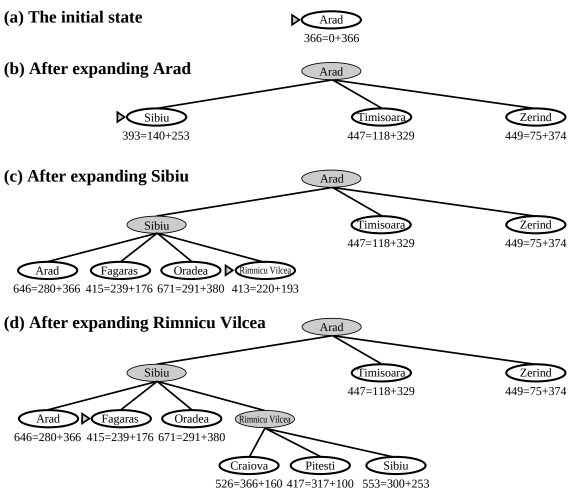


Example: A* for Route Planning

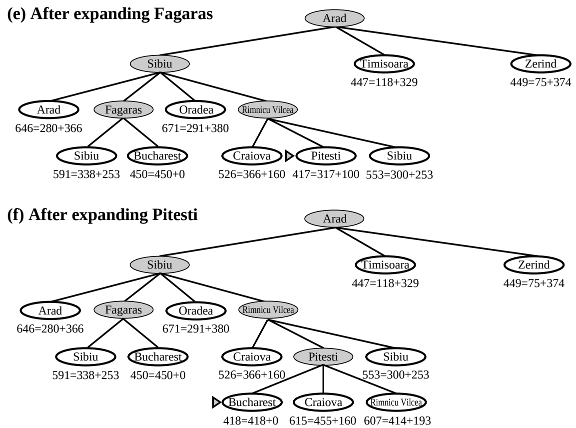


Arad	366
Bucharest	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Example: A* for Route Planning



Example: A* for Route Planning



A*: Properties

- complete with safe heuristics (like all variants of best-first graph search)
- with reopening: optimal with admissible heuristics
- without reopening: optimal with heuristics that are admissible and consistent

$\rightsquigarrow$ proofs: Chapters 18 and 19

A*: Implementation Aspects

some practical remarks on implementing A*:

- common bug: reopening not implemented although heuristic is not consistent
- common bug: duplicate test "too early" (upon generation of search nodes)
- common bug: goal test "too early" (upon generation of search nodes)
- all these bugs lead to loss of optimality and can remain undetected for a long time

16.4 Weighted A*

Weighted A*

Weighted A*

A* with more heavily weighted heuristic:

$f(n) = g(n) + w \cdot h(n.\mathit{state})$,

where weight $w \in \mathbb{R}_0^+$ with $w \geq 1$ is a freely choosable parameter

Note: $w < 1$ is conceivable, but usually not a good idea (Why not?)

Weighted A*: Properties

weight parameter controls "greediness" of search:

- $w = 0$: like uniform cost search
- $w = 1$: like A*
- $w \to \infty$: like greedy best-first search

with $w \geq 1$ properties analogous to A*:

- *h* admissible:
  found solution guaranteed to be at most w times as expensive as optimum when reopening is used
- *h* admissible and consistent:
  found solution guaranteed to be at most w times as expensive as optimum; no reopening needed

(without proof)

16.5 Summary

Summary

best-first graph search with evaluation function f:

- $f = h$: greedy best-first search
  suboptimal, often very fast
- $f = g + h$: A*
  optimal if h admissible and consistent or if h admissible and reopening is used
- $f = g + w \cdot h$: weighted A*
  for $w \geq 1$ suboptimality factor at most w under same conditions as for optimality of A*