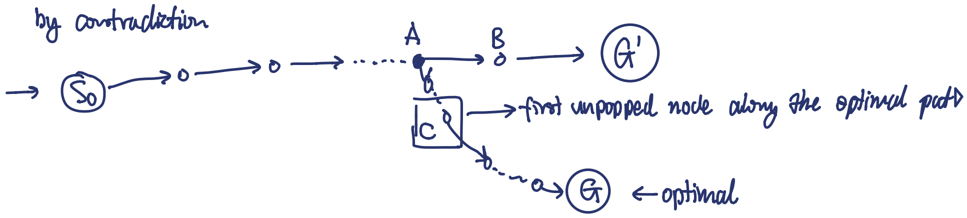
Def: h is admissible if $\forall s \quad 0 \le h(s) \le h^*(s)$

Theorem: A^* using admissible h finds the optimal path

proof idea: by contradiction

Suppose :



but really $S_0 \to G$ is optimal

It must be a state where two paths diverge. one goes to G'
one goes to G

① B is poped, not C $\longrightarrow f(B) \le f(C)$

② G' suboptimal $\longrightarrow g(G') > g(G)$

$\longrightarrow f(G') > f(G) \qquad (h(G') = h(G) = 0)$

③ h admissible $\longrightarrow \forall s \quad h(s) \le h^*(s)$

$\longrightarrow \forall s \quad g(s) + h(s) \le g(s) + h^*(s)$

$\longrightarrow \forall s \quad f(s) \le f(G)$

③ $\to f(C) \le f(G) \xrightarrow{①} f(G') \le f(G)$ contradiction, since G' assumed suboptimal.

Def: h dominates h' if: ① $h(s) \le h^*(s)$, $h'(s) \le h^*(s)$, $\forall s$

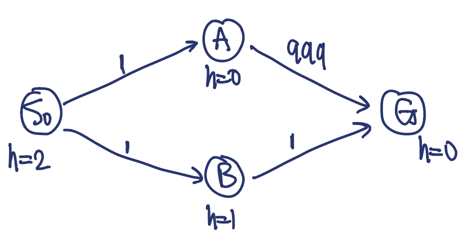
② $h(s) \ge h'(s)$, $\forall s$

③ $\exists s \quad h(s) > h(s')$

The better the h, the better the memory usage / time.

Best h: h^*

Ex. 1



Do not do goal check while generating states

A^*

PQ

~~S_0: $f(s) = g(s) + h(s) = 0 + 2 = 2$~~

~~A $f(s) = g(s) + h(s) = 1 + 0 = 1$~~

~~B $f(s) = g(s) + h(s) = 1 + 1 = 2$~~

G $f(s) =$ ~~$g(s) + h(s) = 1000 + 0 = 1000$~~

$g(s) + h(s) = 2 + 0 = 2$ ☆ when popped G, find goal.

need to update parent ptr!

Closed

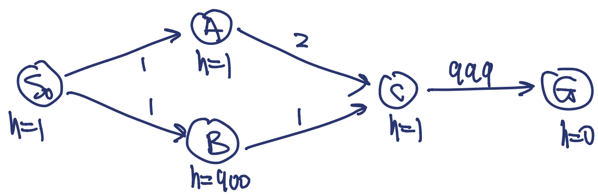
S_0 : 2

A: 1

B: 2

need to add path pointers

Ex. 2



A* PQ

~~$S_0: f=g+h=0+1=1$ ϕ~~

~~$A: f=g+h=1+1=2$ S_0~~

~~$B: f=g+h=1+400=401$ S_0~~

~~$C: f=g+h=3+1=4$ A~~

G $f=g+h=$ ~~$1002+0=1002$ C~~

$1001+0=1001$ ② C

~~C $f=g+h=2+1=3$ B~~

Closed

S_0	1	ϕ
A	2	S_0
~~C~~	~~4~~	~~A~~
B	401	S_0
C	3	B
G	1001	C

$G \to C \to B \to S_0$

P2. A* search implementation