Week 6: Delete Relaxation

COMP90054 – AI Planning for Autonomy

Key concepts

- Delete relaxation heuristic h^+
- The relationship between h^{max}, h^{add} and h^+

Problem 1

- prec:
- add:
- ~~del~~:

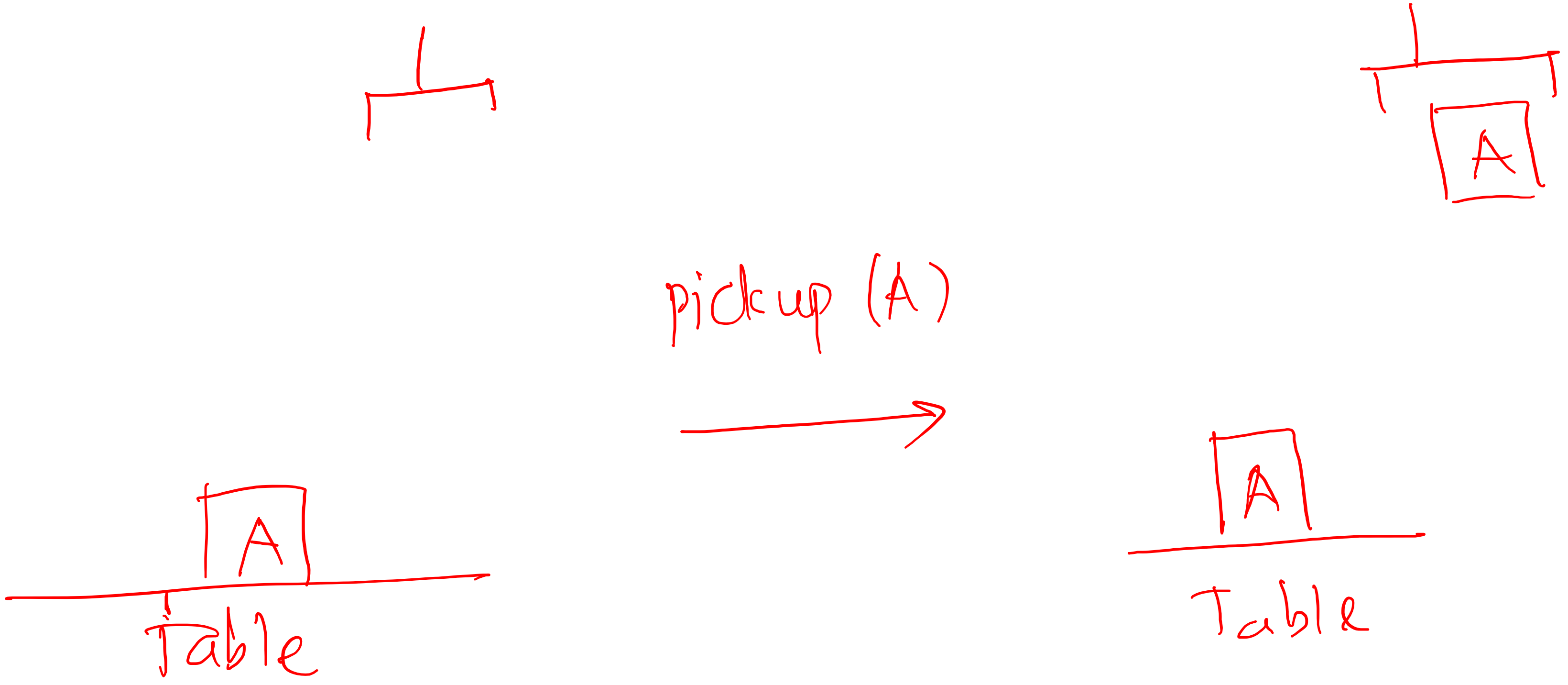
What is the (optimal) delete relaxation heuristic h^+?

Relaxing by **ignoring delete lists** **"What was once true remains true forever"**

Definition (Delete Relaxation).

(i) *For a STRIPS action a, by a^+ we denote the corresponding delete relaxed action, or short relaxed action, defined by $pre_{a^+} := pre_a$, $add_{a^+} := add_a$, and $del_{a^+} :=$*

P = <F, O, I, G> P = <F, O^+, I, G>



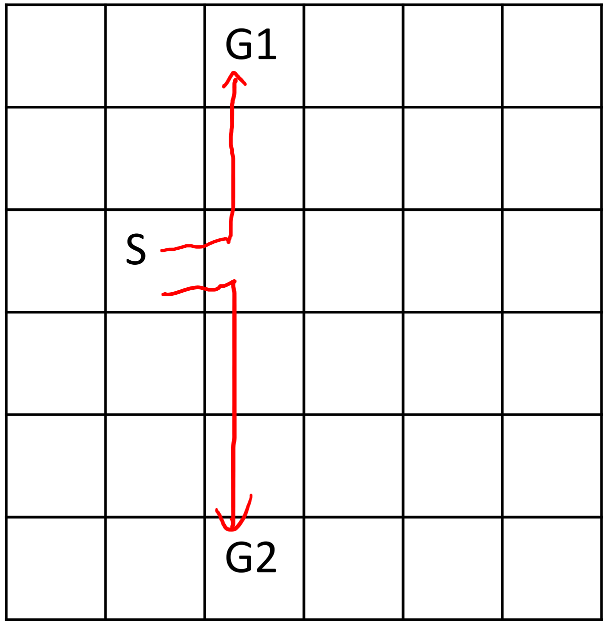
once true → forever true.

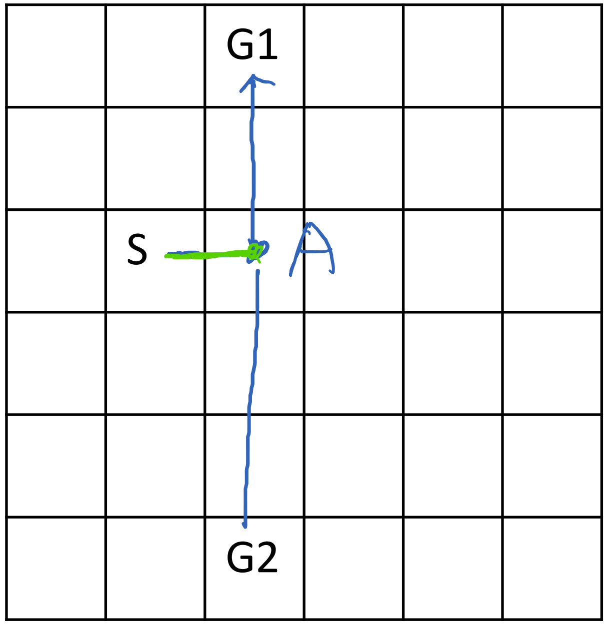
Problem 1

$h(s) = d(s, G1) + d(s, G2) = 3 + 4 = 7$

How would it be interpreted in pacman?

$h^+(s) = \underline{d(s, G1)} + \underline{d(A, G2)} = 3 + 3 = 6$





Minimum spanning tree:
Admissible, Not consistent

without delete relaxation.

Minimum Steiner tree:
Admissible, consistent

with delete relaxation

Problem 1

approximations of h^+

What is the relationship between h^{max}, h^{add} and h^+? What about h^*?

h* is the perfect heuristic (the optimal cost from the current state to the goal state)

h^+ is the **optimal delete relaxation** heuristic (not easy to compute)
h^+ is admissible

h^{max} is an approximation of h^+
h^{max} is admissible. h^{max} is very small.
h^{max} <= h^+ <= h*

h^{add} is an approximation of h^+
h^{add} is not admissible
h^{add} >= h^+

Problem 2: Computing h^{max} and h^{add}

Definition (h^{add}). *Let* $\Pi = (F, A, c, I, G)$ *be a STRIPS planning task. The additive heuristic* h^{add} *for* Π *is the function* $h^{\text{add}}(s) := h^{\text{add}}(s, G)$ *where* $h^{\text{add}}(s, g)$ *is the point-wise greatest function that satisfies* $h^{\text{add}}(s, g) =$

$$\begin{cases} 0 & g \subseteq s \\ min_{a \in A, g \in add_a} c(a) + h^{\text{add}}(s, pre_a) & |g| = 1 \\ \sum_{g' \in g} h^{\text{add}}(s, \{g'\}) & |g| > 1 \end{cases}$$

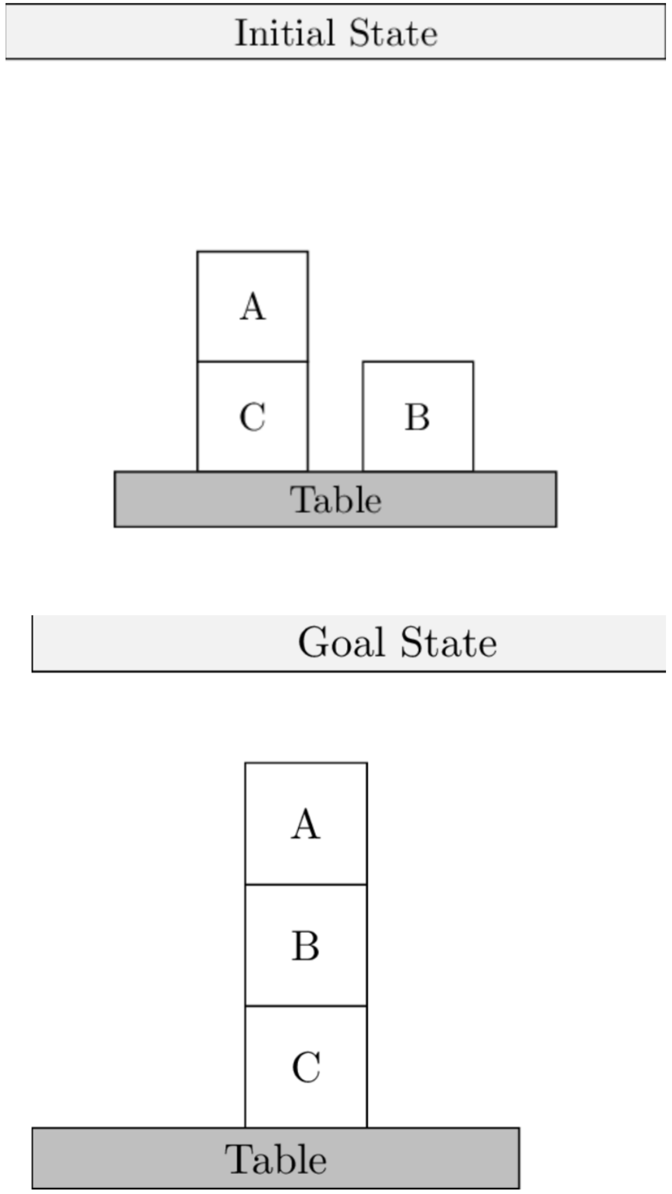
Definition (h^{max}). *Let* $\Pi = (F, A, c, I, G)$ *be a STRIPS planning task. The max heuristic* h^{max} *for* Π *is the function* $h^{\text{max}}(s) := h^{\text{max}}(s, G)$ *where* $h^{\text{max}}(s, g)$ *is the point-wise greatest function that satisfies* $h^{\text{max}}(s, g) =$

$$\begin{cases} 0 & g \subseteq s \\ min_{a \in A, g \in add_a} c(a) + h^{\text{max}}(s, pre_a) & |g| = 1 \\ \max_{g' \in g} h^{\text{max}}(s, \{g'\}) & |g| > 1 \end{cases}$$

Problem 2: Computing h^{max} and h^{add}

Initial state
I = {on(A, C), onTable(C), onTable(B), clear(A), clear(B), handFree}

Goal state
G = {on(A,B), on(B,C), onTable(C)}

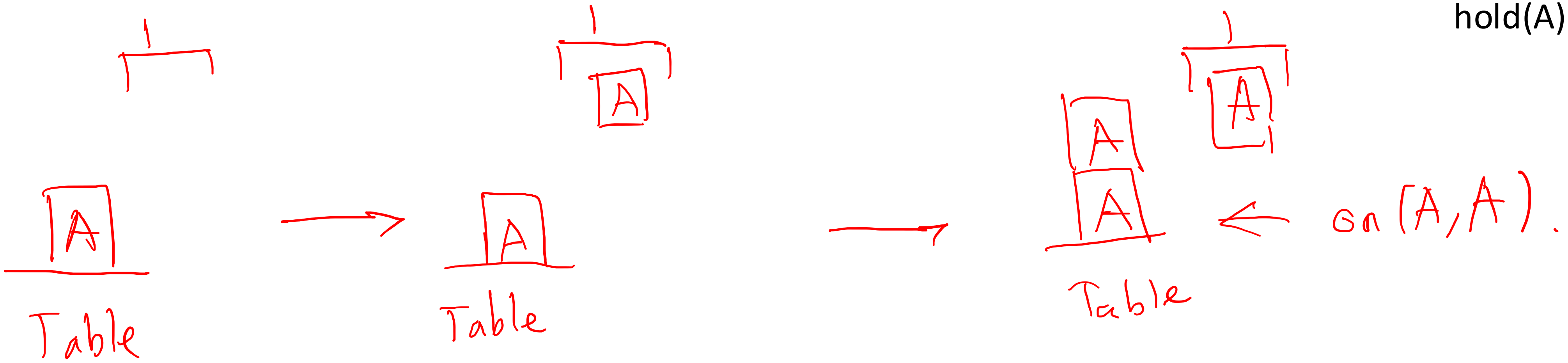


Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1																			
2																			

I = {on(A, C), onTable(C), onTable(B), clear(A), clear(B), handFree}

c(A) = clear(A)
onTable(A) = onT(A)
hold(A) = holding(A)



Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	?	0						0								0	0

Which actions can we take to make **clear(C)** True?

Problem 2

Once true → forever true

Which actions can we take to make **clear(C)** True?

putdown(C)
stack(C, A)
stack(C, B)
unstack(A, C)
unstack(B, C)
stack(C, C)
unstack(C, C)

- unstack(x, C) $x \in \{A, B, C\}$
- putdown(C)
- stack(C, y) $y \in \{A, B, C\}$

Define Operators

O = {

pickup(x)

- Prec: onTable(x), clear(x), handFree
- Add: holding(x)
- Del: onTable(x), clear(x), handFree

unstack(x, y)

- Prec: on(x, y), clear(x), handFree
- Add: holding(x), clear(y)
- Del: on(x, y), clear(x), handFree

putdown(x)

- Prec: holding(x)
- Add: clear(x), onTable(x), handFree
- Del: holding(x)

stack(x, y)

- Prec: holding(x), clear(y)
- Add: clear(x), on(x,y), handFree
- Del: clear(y), holding(x)

}

Problem 2

action cost = 1

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	?	0						0								0	0

h^{add} = action cost + ***sum***(heuristic of preconditions)

h^{max} = action cost + ***max***(heuristic of preconditions)

putdown(C)
stack(C, A)
stack(C, B)
unstack(A, C)
unstack(B, C)
stack(C, C)
unstack(C, C)

unstack(x, y)
- Prec: on(x, y), clear(x), handFree
- Add: holding(x), clear(y)
- Del: on(x, y), clear(x), handFree

putdown(x)
- Prec: holding(x)
- Add: clear(x), onTable(x), handFree
- Del: holding(x)

stack(x, y)
- Prec: holding(x), clear(y)
- Add: clear(x), on(x,y), handFree
- Del: clear(y), holding(x)

Problem 2

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	?	0						0								0	0

putdown(C) = 1 + hold(C) = 1 + ∞ = ∞

1 + hold(C) = ∞

stack(C, A) = 1 + hold(C) + clear(A) = 1 + ∞ + 0 = ∞

1 + max(hold(C), clear(A)) = 1 + ∞ = ∞

stack(C, B) = 1 + hold(C) + clear(B) = 1 + ∞ + 0 = ∞

1 + max(hold(C), clear(B)) = 1 + ∞ = ∞

unstack(A, C) = 1 + on(A, C) + clear(A) + handFree = 1 + 0 + 0 + 0 = 1

1 + max(on(A, C), clear(A), handFree) = 1

unstack(B, C) = 1 + on(B, C) + clear(B) + handFree = 1 + ∞ + 0 + 0 = ∞

1 + max(on(B, C), clear(B), handFree) = ∞

stack(C, C) = 1 + hold(C) + clear(C) = 1 + ∞ + ∞ = ∞

1 + max(hold(C), clear(C)) = 1 + ∞ = ∞

unstack(C, C) = 1 + on(C, C) + clear(C) + handFree = 1 + ∞ + ∞ + 0 = ∞

1 + max(on(C, C), clear(C), handFree) = 1 + ∞ = ∞

unstack(x, y)
- Prec: on(x, y), clear(x), handFree
- Add: holding(x), clear(y)
- Del: on(x, y), clear(x), handFree

putdown(x)
- Prec: holding(x)
- Add: clear(x), onTable(x), handFree
- Del: holding(x)

stack(x, y)
- Prec: holding(x), clear(y)
- Add: clear(x), on(x,y), handFree
- Del: clear(y), holding(x)

Problem 2

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1? / 1	0						0								0	0

putdown(C) = ∞

stack(C, A) = ∞

stack(C, B) = ∞

unstack(A, C) = 1

unstack(B, C) = ∞

stack(C, C) = ∞

unstack(C, C) = ∞

min(putdown(C), stack(C, A), stack(C, B), stack(C, C), unstack(A, C), unstack(B, C), unstack(C, C)) = 1

Problem 2

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1	0						0								0	0

Summary

1. Find all actions that make the predicate become True
2. Calculate h^{add} and h^{max} of all actions

h^{add} = action cost + ***sum***(heuristic of preconditions)
h^{max} = action cost + ***max***(heuristic of preconditions)

3. Get the minimum heuristic value

Definition (h^{add}). *Let* $\Pi = (F, A, c, I, G)$ *be a STRIPS planning task. The additive heuristic* h^{add} *for* Π *is the function* $h^{\text{add}}(s) := h^{\text{add}}(s, G)$ *where* $h^{\text{add}}(s, g)$ *is the point-wise greatest function that satisfies* $h^{\text{add}}(s, g) =$

$$\begin{cases} 0 & g \subseteq s \\ min_{a \in A, g \in add_a} c(a) + h^{\text{add}}(s, pre_a) & |g| = 1 \\ \sum_{g' \in g} h^{\text{add}}(s, \{g'\}) & |g| > 1 \end{cases}$$

Definition (h^{max}). *Let* $\Pi = (F, A, c, I, G)$ *be a STRIPS planning task. The max heuristic* h^{max} *for* Π *is the function* $h^{\text{max}}(s) := h^{\text{max}}(s, G)$ *where* $h^{\text{max}}(s, g)$ *is the point-wise greatest function that satisfies* $h^{\text{max}}(s, g) =$

$$\begin{cases} 0 & g \subseteq s \\ min_{a \in A, g \in add_a} c(a) + h^{\text{max}}(s, pre_a) & |g| = 1 \\ \max_{g' \in g} h^{\text{max}}(s, \{g'\}) & |g| > 1 \end{cases}$$

Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1	0	1	1	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
2													?						

pickup(x)
- Prec: onTable(x), clear(x), handFree
- Add: holding(x)
- Del: onTable(x), clear(x), handFree

unstack(x, y)
- Prec: on(x, y), clear(x), handFree
- Add: holding(x), clear(y)
- Del: on(x, y), clear(x), handFree

putdown(x)
- Prec: holding(x)
- Add: clear(x), onTable(x), handFree
- Del: holding(x)

stack(x, y)
- Prec: holding(x), clear(y)
- Add: clear(x), on(x,y), handFree
- Del: clear(y), holding(x)

Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1	0	1	1	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
2													$\overset{?}{3}$ / 2						

h^{add} = action cost + ***sum***(heuristic of preconditions)

h^{max} = action cost + ***max***(heuristic of preconditions)

stack(B,C) = 1 + hold(B) + c(C) = 1 + 1 + 1 = 3

stack(B,C) = 1 + max(hold(B), c(C)) = 1 + 1 = 2

stack(x, y)

- Prec: holding(x), clear(y)
- Add: clear(x), on(x,y), handFree
- Del: clear(y), holding(x)

Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1	0	1	1	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
2	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	∞	∞	∞	2	0	0

h^{add} / h^{max}

Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1	0	1	1	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
2	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	∞	∞	∞	2	0	0
3	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	3	3	4 / 3	2	0	0
4	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	3	3	4 / 3	2	0	0

h^{add} / h^{max}

stop when converge (2 rows have the same values)

Problem 2: Computing h^{max} and h^{add}

Iter	c(A)	c(B)	c(C)	hand Free	h(A)	h(B)	h(C)	on(A, A)	on(A,B)	on(A,C)	on(B,A)	on(B,B)	on(B,C)	on(C,A)	on(C,B)	on(C,C)	onT(A)	onT(B)	onT(C)
0	0	0	∞	0	∞	∞	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
1	0	0	1	0	1	1	∞	∞	∞	0	∞	∞	∞	∞	∞	∞	∞	0	0
2	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	∞	∞	∞	2	0	0
3	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	3	3	4 / 3	2	0	0
4	0	0	1	0	1	1	2	2	2	0	2	2	3 / 2	3	3	4 / 3	2	0	0

h^{add} / h^{max}

G = {on(A,B), on(B,C), onTable(C)}

h^{add}(s0) = 2 + 3 + 0 = 5

$h^{max}(s0) = \max(2, 2, 0) = 2$