Yuchen (Bobby) Zhang, zhan2175
Prof. Bareinboim
CS 47100, Spring 2018
Purdue University
Feb 8, 2018

Homework 1

Problem 1

(a)

UNIV→BRNG→STEW→HAAS→REC→PMU→LWSN→WTHR→HEAV→ELLT

(b)

HEAV→
HEAV→PMU→WTHR→
HEAV→PMU→STEW→WTHR→ELLT→REC→SC→
HEAV→PMU→STEW→BRNG→UNIV→WTHR→ELLT→LWSN→REC→BRNG→SC→LWSN→
HEAV→PMU→STEW→BRNG→HAAS

(c)

LWSN→ELLT→WTHR→HEAV→PMU

(d)

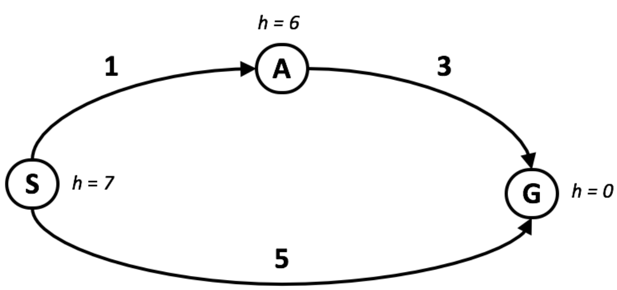
LWSN→SC→WTHR→HEAV→PMU

Problem 2

(a)

DFS performs better when the goal falls on further down left. Please refer to the graph above. The DFS will follow along the left path and directly hit the goal, while BFS will iterate through every depth then reach the goal.

(b)



Please refer to the graph above. The heuristic search will follow the path $S \to G$, since $f(A) = g(A) + h(A) = 1 + 6 = 7$, while $f(G) = g(G) + h(G) = 5 + 0 = 5$, G will be chosen. The total cost will be 5. But the actual optimal goal is the path $S \to A \to G$ with the total cost of 4.

(c)

Proof for addmissible:

Since h_i, where $i = 1...k$, are consistent, they are also admissible, i.e., $\forall i \forall n$ $h_i(n) \leq g(G) - g(n)$ in the system, and certainly $max(h_i) \leq g(G) - g(n)$. By definition, $h^*(n) = max_{i=1}^{k} h_k(n) = max(h_i(n)) \leq g(G) - g(n)$. Therefore h^* is admissible.

Proof for consistency (by contradiction):

For some points A and B, set $g(B) - g(A) = c$, where c is a constant. Assume a maximum number p such that for the first p consistent heuristics, h_p^* is consistent. We have

$$h_p^*(A) - h_p^*(B) \leq c$$

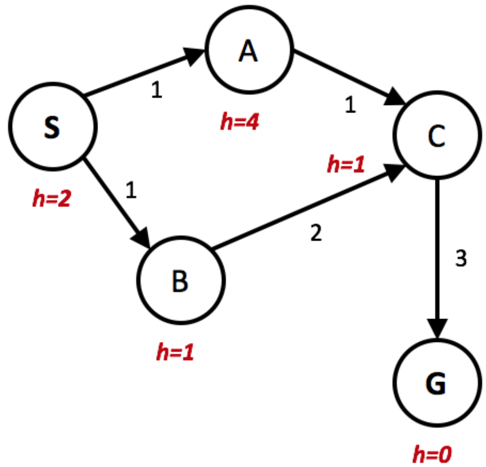
Then we take the $p+1$ consistent heuristics into consideration and h_{p+1}^* becomes inconsistent. Only when $h_p^*(A) < h_{p+1}(A)$ and $h_p^*(B) > h_{p+1}(B)$ is possible. Since

1) $h_p^*(A) > h_{p+1}(A)$, $h_p^*(B) > h_{p+1}(B) \Rightarrow h_p^*(A) - h_p^*(B) \leq c$ consistent;
2) $h_p^*(A) < h_{p+1}(A)$, $h_p^*(B) < h_{p+1}(B) \Rightarrow h_{p+1}(A) - h_{p+1}(B) \leq c$ consistent;
3) $h_p^*(A) > h_{p+1}(A)$, $h_p^*(B) < h_{p+1}(B) \Rightarrow h_p^*(A) - h_{p+1}(B) \leq c$ consistent.

Since inconsistency, we have $h_{p+1}(A) - h_{p+1}(B) > h_{p+1}(A) - h_p^*(B) > c$, which contradict to the fact the the $p+1$ consistent heuristics.

Therefore, h^* is also consistent.

(d)



All the nodes have $h(n) \leq h^*(n)$, which satisfies the admissible property. The path found by the heuristic search is $S \rightarrow B \rightarrow C \rightarrow G$ with total cost of 6, while the optimal path is $S \rightarrow A \rightarrow C \rightarrow G$ with total cost of 5.

Problem 3

(a)

Variables: $A\alpha$, $B\zeta$, $C\gamma$, $D\beta$, $E\omega$, $F\mu$
Domain: $\{D8TO, 9GA, UROK\}$
Constraints: Let $t \in T$ represent the time. $R = \{R_D, R_9, R_U\}$ represents the three robots. R_{i_x} represents robot i preform task x. Then the constraints can be expressed as below.

$$(\exists t)(\exists x, y, z \xrightarrow{\text{bijective}} A\alpha, D\beta, E\omega) R_{D_x} \wedge R_{9_y} \wedge R_{U_z}$$

$$(\forall R_i \in R)\neg(R_{i_B} \wedge R_{i_E})$$

$$(\forall R_i \in R)\neg(R_{i_F} \wedge R_{i_C})$$

(b)

	A	B	C	D	E	F
D8TO				X	X	X
9GA			X			
RUOK		X		X		

(c)

We can order our choice of tasks first by the minimum remaining value. For example, we can easily see that the task D can only be assigned to $9GA$. We can also apply arc consistency. For example, since $9GA$ perform task D, it cannot perform task A and E. Then we know task E can only be assigned to $RUOK$. And so on...

Problem 4

(a)

- C_4 and C_5 cannot be in the same room
- C_2 and C_3 cannot be in the same room
- C_1 and C_6 cannot be in the same room
- C_6 and C_7 cannot be in the same room
- R_A can have conference 1,4,5,6
- R_B can have conference 1,2,3,6,7
- R_C can have conference 2,3,5,7

(b)

	1	2	3	4	5	6	7
A		X	X	O	X		X
B				X	X		
C	X			X	O	X	

R_A can have conference 1,6
R_B can have conference 1,2,3,6,7
R_C can have conference 2,3,7

(c)

The conference room R_A should be considered first since it only has two remaining values.