COMS 4701 - Homework 2 - Written

Albert Go ag4474

October 8, 2021

Question 1

1. First, S is added to the queue.
   - $[S]$ Then, S will be expanded, and nodes A, B, and C will be added to the queue.
   - $[A, B, C]$ S is removed and A will be expanded next, and node D will be added to the queue.
   - $[B, C, D]$ A is removed and B will be expanded next, and node E will be added to the queue.
   - $[C, D, E]$ B is removed and C will be expanded next, and node H will be added to the queue.
   - $[D, E, H]$ C is removed and D will be expanded next, but no nodes will be added because it has no children.
   - $[E, H]$ D is removed and E will be expanded next, but no nodes will be added because A was already explored.
   - $[H]$ E is removed and H will be expanded next, and F and G will be added to the queue.
   - $[F, G]$ F is removed and G will be expanded next, and no nodes will be added because D was already explored.
   - $[G]$ H is removed and G is explored and is discovered to be the goal
2. First, S is added to the queue. The stack in my problem is shown so that the element on the right most is to be pulled out first.
   - $[S]$ Then, S will be expanded, and nodes C, B, and A will be added to the stack.
   - $[C, B, A]$ S is removed and A will be expanded, and node D will be added to the stack.
   - $[C, B, D]$ A is removed and D will be expanded, and no nodes will be added because it has no children.
   - $[C, B]$ D is removed and B will be expanded, and node E will be added to the stack.
   - $[C, E]$ B is removed and E will be expanded, but no nodes will be added because A was already explored.
   - $[C]$ E is removed and C will be expanded, and node H will be added to the stack.
   - $[H]$ C is removed and H will be expanded, and G and F will be added to the stack.
   - $[G, F]$ H is removed and F is explored and no nodes will be added to the stack because D has already been explored.
   - $[G]$ F is removed and G is explored and is discovered to be the goal
3. First, S is added to the priority queue. The Priority Queue will be shown like a dictionary. The keys are the nodes and the values are the cost.
   - $\{S : 0\}$ Then, S will be expanded, and nodes B, C, and A will be added to the priority queue in that order.
   - $\{B : 2, C : 5, A : 6\}$ S is removed and B will be expanded next, and node E will be second in the priority queue.
   - $\{C : 5, E : 5, A : 6\}$ B is removed and C will be expanded next, and H will be added to the end of the priority queue..
   - $\{E : 5, A : 6, H : 7\}$ C is removed and E will be expanded next, but no nodes will be added because A is already in the queue and has a lesser cost than what it would be going from E to A.
   - $\{A : 6, H : 7]$ E is removed and A will be expanded next, and D will be added to the end of the priority queue.
   - $\{H : 7, D : 15\}$ A is removed and H will be expanded next, and node F and G will be added to the beginning of the priority queue.
   - $\{F : 9, G : 14, D : 15\}$ H is removed and F will be expanded, and D will be updated with a new priority value and moved up in the queue.
   - $\{D : 13, G : 14\}$ F is removed and D is explored next, and no nodes will be added because it has no children.
   - $\{G : 14\}$ H is removed and G is explored and is discovered to be the goal

4. First, S is added to the priority queue. The Priority Queue will be shown like a dictionary. The keys are the nodes and the values are the cost.
   - $\{S : 0\}$ Then, S will be expanded, and nodes B, C, and A will be added to the priority queue in that order.
   - $\{B : 3, C : 8, A : 11\}$ S is removed and B will be expanded next, and node E will be second in the priority queue.
   - $\{C : 5, E : 9, A : 11\}$ B is removed and C will be expanded next, and H will be added to the end of the priority queue..
   - $\{E : 9, A : 11, H : 14\}$ C is removed and E will be expanded next, but no nodes will be added because A is already in the queue and has a lesser cost than what it would be going from E to A.
   - $\{A : 11, H : 14]$ E is removed and A will be expanded next, and D will be added to the end of the priority queue.
   - $\{H : 14, D : 22\}$ A is removed and H will be expanded next, and node F and G will be added to the beginning of the priority queue.
   - $\{F : 9, G : 14, D : 22\}$ H is removed and F will be expanded, and D will be updated with a new priority value.
   - $\{G : 14, D : 20\}$ F is removed and G is explored and is discovered to be the goal

Question 2

1. The Apriori algorithm will discover Mac → PC. This association does exceed the minimum supply threshold with a $supp(Mac \rightarrow PC) = 40\%$ and exceeds the minimum confidence.

$$conf(Mac \rightarrow PC) = \frac{P(Mac \cap PC)}{P(Mac)} \approx 67\% \tag{1}$$

2. To determine independence, we must calculate the $Interest(Mac \rightarrow PC)$

$$Interest(Mac \rightarrow PC) = \frac{P(Mac \cap PC)}{P(Mac) * P(PC)} \approx 0.89 \tag{2}$$

   Since $Interest(Mac \rightarrow PC) \approx 0.89 < 1$, Mac and PC are negatively dependent.

Question 3

(a) Each square with a variable, without considering constraints, can be any number between 1 and 4 inclusive. As a result, the size of the state space is 4^{12}—4 numbers to choose and 12 variables.

(b) $A \neq 2, 4$; $B \neq 3, 4$; $C \neq 4$; $D \neq 4$; $E \neq 2, 4$; $F \neq 3, 4$; $G \neq 2, 4$; $H \neq 2, 3$; $I \neq 2, 3, 4$; $J \neq 2, 3, 4$; $K \neq 2, 3$; $L \neq 3, 4$

(c) There is a tie if we choose to assign the first variable using the minimum remaining values heuristic. Letters I and J would be first because each of them can only be assigned a value of 1.

(d) Figure 1 below depicts my thought process for reducing the domains of the problem through unary constraints and arc consistency.

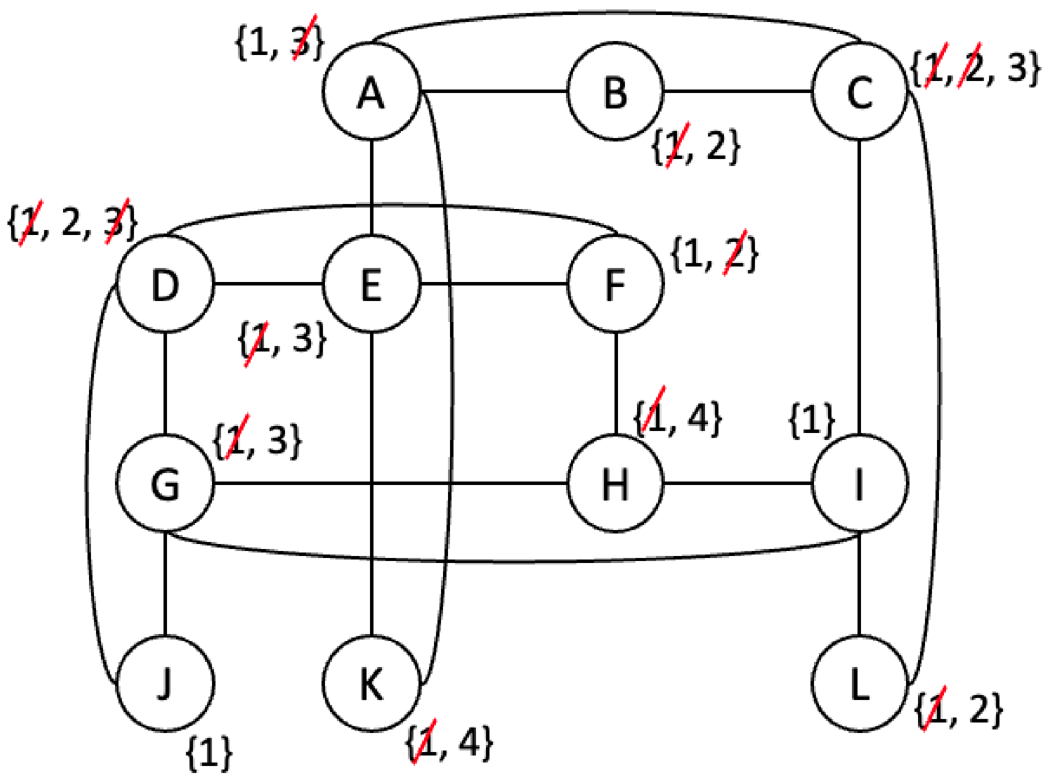

Figure 1: Arc consistency tree.