CS 2710/ISSP 2160: Homework 2 Answer key

1 Hill Climbing Search, Constraint Representation (25 pts)

(a) A possible better valuation would be the #dependency conflicts for a given work permutation. If we a permutation of $\{1, 2, 3, 4, 5\}$ where job 1 depends on 2,3,4 then the valuation would be 3.

A possible better successor function might be to only allow a swap for a dependent job. So in the previous example, only job 1 would be allowed to swap.

(b) One way to define the problem is as follows: We have five foods $\mathrm{vp}, \mathrm{swf}, \mathrm{salmon}, \mathrm{pl}, \mathrm{cp}$ representing veal parmesan, grilled swordfish, pan seared salmon, pork loin, and chicken parmesan, respectively. They can be served on one of five days: $\mathrm{W}, \mathrm{R}, \mathrm{F}, \mathrm{Sa}, \mathrm{Su}$. Let VP represent the days veal parmesan is served, etc. The initial domain for each food item is $\{\mathrm{W}, \mathrm{R}, \mathrm{F}, \mathrm{Sa}, \mathrm{Su}\}$.

The constraints are:

- Something must be served everyday: $(\mathrm{W} = \mathrm{VP} \vee \mathrm{W} = \mathrm{SWF} \vee \mathrm{W} = \mathrm{SALMON} \vee \mathrm{W} = \mathrm{PL} \vee \mathrm{W} = \mathrm{CP}) \wedge (\mathrm{R} = \mathrm{VP} \vee \mathrm{R} = \mathrm{SWF} \vee \mathrm{R} = \mathrm{SALMON} \vee \mathrm{R} = \mathrm{PL} \vee \mathrm{R} = \mathrm{CP}) \wedge (\mathrm{F} = \mathrm{VP} \vee \mathrm{F} = \mathrm{SWF} \vee \mathrm{F} = \mathrm{SALMON} \vee \mathrm{F} = \mathrm{PL} \vee \mathrm{F} = \mathrm{CP}) \wedge (\mathrm{Sa} = \mathrm{VP} \vee \mathrm{Sa} = \mathrm{SWF} \vee \mathrm{Sa} = \mathrm{SALMON} \vee \mathrm{Sa} = \mathrm{PL} \vee \mathrm{Sa} = \mathrm{CP}) \wedge (\mathrm{Su} = \mathrm{VP} \vee \mathrm{Su} = \mathrm{SWF} \vee \mathrm{Su} = \mathrm{SALMON} \vee \mathrm{Su} = \mathrm{PL} \vee \mathrm{Su} = \mathrm{CP})$
- Grilled swordfish is never served on Friday: $\mathrm{SWF} \neq \mathrm{F}$
- Pork loin cannot be served the day after veal parmesan: $(\mathrm{W} = \mathrm{VP} \Rightarrow \mathrm{R} \neq \mathrm{PL}) \wedge (\mathrm{R} = \mathrm{VP} \Rightarrow \mathrm{F} \neq \mathrm{PL}) \wedge (\mathrm{F} = \mathrm{VP} \Rightarrow \mathrm{Sa} \neq \mathrm{PL}) \wedge (\mathrm{Sa} = \mathrm{VP} \Rightarrow \mathrm{Su} \neq \mathrm{PL})$
- Pan seared salmon and veal parmesan must be served on the weekend: $(\mathrm{Sa} = \mathrm{SALMON} \vee \mathrm{Su} = \mathrm{SALMON}) \wedge (\mathrm{Sa} = \mathrm{VP} \vee \mathrm{Su} = \mathrm{VP})$
- Chicken parmesan is always served within two days of grilled swordfish: $(\mathrm{W} = \mathrm{SWF} \Rightarrow (\mathrm{R} = \mathrm{CP} \vee \mathrm{F} = \mathrm{CP})) \wedge (\mathrm{R} = \mathrm{SWF} \Rightarrow (\mathrm{F} = \mathrm{CP} \vee \mathrm{Sa} = \mathrm{CP})) \wedge (\mathrm{F} = \mathrm{SWF} \Rightarrow (\mathrm{Sa} = \mathrm{CP} \vee \mathrm{Su} = \mathrm{CP})) \wedge (\mathrm{Sa} = \mathrm{SWF} \Rightarrow (\mathrm{Su} = \mathrm{CP}))$

2 Constraint satisfaction problems (25 pts)

(a) Backtracking search. See Figure 1

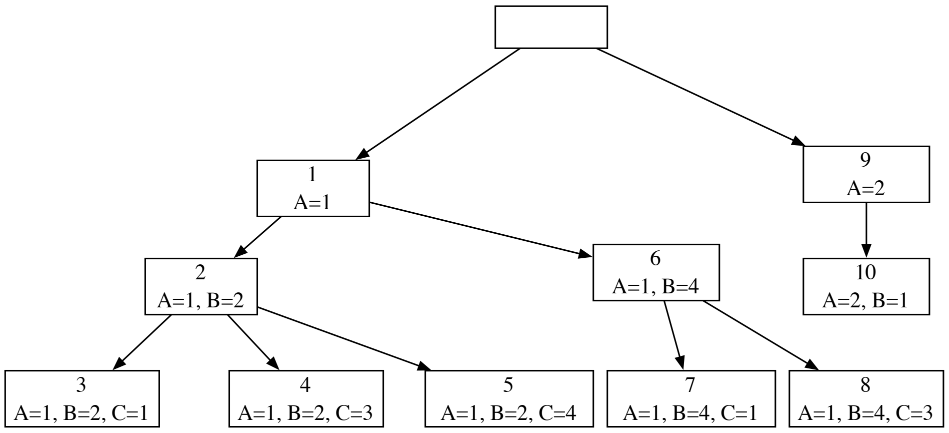


Figure 1: Problem 2-part (a)

(b) (Solution based on Aimee's) Most constrained variable: Assigning the most constrained variable first reduces the branching factor. For example, we may start by assigning B which has a domain $\{1, 2, 4\}$.
Most constraining variable: Among the most-constrained variable, we may assign variables that participate in a larger number of constraints first. This would reduce the branching factor at subsequent tree levels by restricting the values that can be assigned to the variables. If we continue the above case and choose $B = 2$ and set $E = 1$, then A, C, and Ds possible remaining values are $\{3, 4\}$. Because D is our most constraining variable in that it appears in the most constraints with A and C ($D = A$ and $D < C$), then Ds value is chosen next. If $D = 3$, then Cs domain is then reduced to nothing. So backtrack and set $D = 4$. This forces A to be 4 and limits C's domain to $\{3\}$. As you can see, choosing the most constraining variable, D, we were able to eliminate future values of A and C, thus reducing the future branching factor.
Least-constraining value: When assigning a value to a given variable, we may choose the value that allows the most flexibility in assigning the remaining variables. For example, if we initially select $E = 1$, this only removes one value from each of the domains of A, B, C, D ($\{2, 3, 4\}$). This would allow us more freedom in future choices since only one value was eliminated from each variable's domain.

(c) See table.

Current Assignment	Domains				
being considered	A	B	C	D	E
$A = 1$	1	2,4	1,3,4	1	
$A = 2$	2	1,4	1,3,4	2	1
$A = 2, B = 1$	2	1	3,4	2	
$A = 2, B = 4$	2	4	1,3	2	1
$A = 2, B = 4, C = 1$	2	4	1	2	
$A = 2, B = 4, C = 3$	2	4	3		1
$A = 3$	3	1,2,4	1,3,4	3	1,2
$A = 3, B = 1$	3	1	3,4	3	
$A = 3, B = 2$	3	2	1,3,4	3	1
$A = 2, B = 2, C = 1$	3	2	1	3	
$A = 3, B = 2, C = 3$	3	2	3		1
$A = 3, B = 2, C = 4$	3	2	4		1
$A = 3, B = 4$	3	4	1,3	3	1,2
$A = 3, B = 4, C = 1$	3	4	1	3	
$A = 3, B = 4, C = 3$	3	4	3		1,2
$A = 4$	4	1,2	1,3,4	4	1,2,3
$A = 4, B = 1$	4	1	3,4	4	
$A = 4, B = 2$	4	2	1,3,4	4	1
$A = 4, B = 2, C = 1$	4	2	1	4	
$A = 4, B = 2, C = 3$	4	2	3	4	1
$A = 4, B = 2, C = 3, D = 4$	4	2	3	4	1
$A = 4, B = 2, C = 3, D = 4, E = 1$	4	2	3	4	1

3 Adversarial search

See Figure 2.

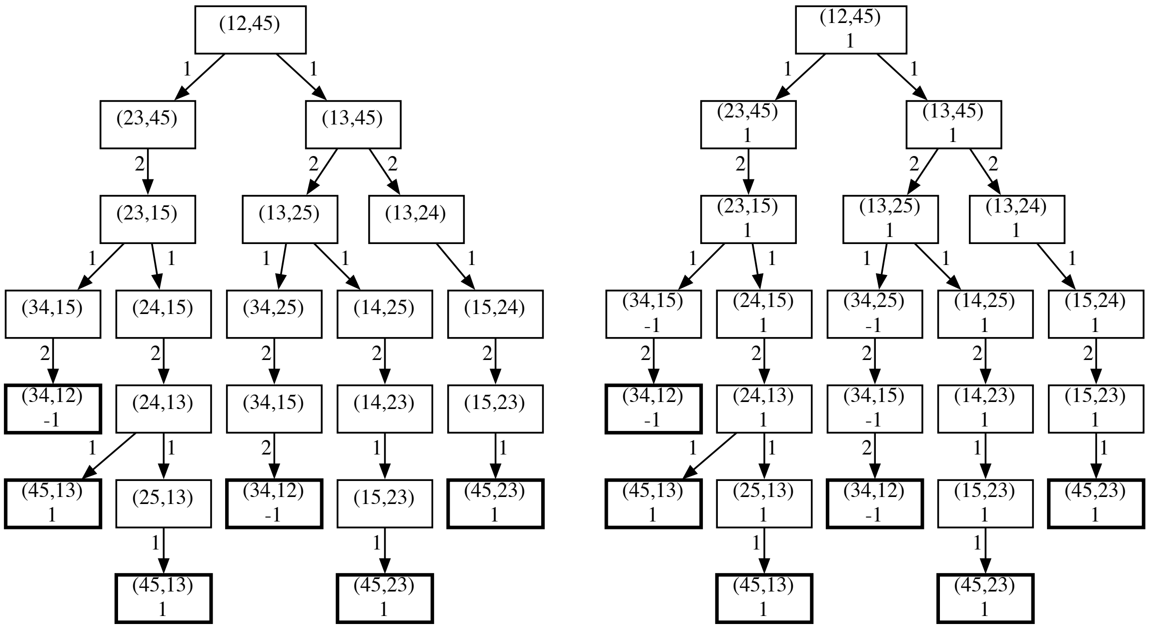
4 Logic (25 pts)

(a) Let P: I have a sweet tooth, Q: I like chocolate, R: I like cake, S: I like danish, T: I'm chocoholic.
KB: $(P \wedge Q) \vee (Q \wedge R)$
Rules: $R \Rightarrow S$, $S \Rightarrow P$, $(P \wedge Q) \Rightarrow T$
Goal: T.

(b) CNF representation
KB:

- $(P \vee Q) \wedge (P \vee R) \wedge Q \wedge (Q \vee R)$ (By application of the distributive property).
  Using the absorption law, this is equivalent to:
- $(P \vee R) \wedge Q$

Rules:



(a) Game search tree

(b) Tree with Minimax values for player 1

Figure 2: Problem .

- $(\neg R \vee S) \wedge (\neg S \vee P) \wedge (\neg(P \wedge Q) \vee T)$
- $(\neg R \vee S) \wedge (\neg S \vee P) \wedge (\neg P \vee \neg Q \vee T)$ (De Morgan's law, associativity)

Goal: T

(c) Resolution: We rewrite the knowledge base as a sequence of disjunctive sentences (the conjunction connective is implicit) ending with the negated theorem:

1. $(P \vee R)$
2. Q
3. $(\neg R \vee S)$
4. $(\neg S \vee P)$
5. $(\neg P \vee \neg Q \vee T)$
6. $\neg T$

   Now we apply resolution: Resolving 5 and 6 yields

5'. $(\neg P \vee \neg Q)$

   Resolving 3 and 4 yields

3'. $\neg R \vee P$

   1 and 3' give:

1'. P

1' and 5' give:

1". : $\neg Q$

Resolving 1" and 2 results in a contradiction thus proving the theorem T.

d. Representation using FOPL: A possible solution is to keep the propositions P and T from part (a), introduce a predicate Like to specify the foods I like, and define the domain of foods such that $\text{Food} = \{\text{Cake}, \text{Danish}, \text{Chocolate}\}$. The predicate Like is formally a mapping $\text{Like} : \text{Food} \rightarrow \{\text{True}, \text{False}\}$. Now we can rewrite the sentences as follows:

- KB: $(P \wedge \text{Like}(\text{Chocolate})) \vee (\text{Like}(\text{Chocolate}) \wedge \text{Like}(\text{Cake}))$
- Rules:
  - $\text{Like}(\text{Cake}) \Rightarrow \text{Like}(\text{Danish})$
  - $\text{Like}(\text{Danish}) \Rightarrow P$
  - $P \wedge \text{Like}(\text{Chocolate}) \Rightarrow T$
- Goal: T.

e. Let People denote the domain of people. A possible solution is to replace the predicate Like with a new predicate $\text{Likes} : \text{People} \times \text{Food} \rightarrow \{\text{True}, \text{False}\}$. Also define the predicate $P : \text{People} \rightarrow \{\text{True}, \text{False}\}$ that specifies people with a sweet tooth, and $T : \text{People} \rightarrow \{\text{True}, \text{False}\}$ that specifies people who are chocoholic. Now we can rewrite the sentences as follows:

- KB: $\forall X \in \text{People} : (P(X) \wedge \text{Likes}(X, \text{Chocolate})) \vee (\text{Likes}(X, \text{Chocolate}) \wedge \text{Likes}(X, \text{Cake}))$
- Rules:
  - $\forall X \in \text{People} : \text{Likes}(X, \text{Cake}) \Rightarrow \text{Likes}(X, \text{Danish})$
  - $\forall X \in \text{People} : \text{Likes}(X, \text{Danish}) \Rightarrow P(X)$
  - $\forall X \in \text{People} : P(X) \wedge \text{Likes}(X, \text{Chocolate}) \Rightarrow T(X)$
- Goal: $\forall X \in \text{People} : T(X)$. (Everyone is a chocoholic!)