Max O’Cull
0028687211
CS 471
Homework 2
02/04/19

1. Solve the cryptarithmetic problem in Figure 6.2 by hand, using the strategy of backtracking with forward checking and the MRV and least-constraining-value heuristics.

Max O'Cull
0028687211
CS 471
Homework 2
02/04/19

$$\begin{array}{r} TWO \\ +\ TWO \\ \hline FOUR \end{array}$$

$O + O = R + 10(C_1)$
$W + W + C_1 = U + 10(C_2)$
$T + T + C_2 = O + 10(C_3)$
$F = C_3$
$F, C_3 \neq 0$
$\text{Alldiff}(O, W, T, F, U, R)$

Domains:
$F, C_3 = \{1\}$ (largest sum $= 9+9+1 = \underline{1}9$)
$C_1, C_2 = \{0, 1\}$
$T = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$W, O, U, R = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Step	Backtrack:	
1	$F = 1$	Select by MRV
		Remove 1's by forward checking (FC)
		$T = \{2, 3, 4, 5, 6, 7, 8, 9\}$
		$W, O, U, R = \{0, 2, 3, 4, 5, 6, 7, 8, 9\}$
2	$C_3 = 1$	Select by MRV
		Remove 2, 3, 4 in T by FC
		$T = \{5, 6, 7, 8, 9\}$
3	$C_2 = 0$	Select by MRV
		Assign 0 by LCV
		Remove 3, 5, 7, 9 in O
		and 5, 6, 7, 8, 9 in W by FC
		$W = \{0, 2, 3, 4\}$
		$O = \{0, 2, 4, 6, 8\}$
4	$C_1 = 0$	Select by MRV
		Assign 0 by LCV
		Remove 6, 8 in O
		and 1, 3, 5, 7, 9 in U by FC
		$O = \{0, 2, 4\}$
		$U = \{0, 2, 4, 6, 8\}$

→

Max O'Cull
0028687211
CS 471

5 $O = 2$

Select by MRV
Assign 2 by LCV
Remove 2's by FC

$T = \{6\}$
$R = \{4\}$
$O = \{2\}$
$U = \{0, 4, 6, 8\}$
$W = \{0, 3, 4\}$

6 $W = 3$

$$\begin{array}{r} 732 \\ +\ 732 \\ \hline 12U4 \end{array}$$

Select by MRV
Assign 3 by LCV
Incorrect, backtrack to step 3.

7 $C_1 = 1$

Assign 1
Remove 0,2,4 in O and 0,2,4,6,8 in U by FC

$O = \{6, 8\}$
$U = \{3, 5, 7, 9\}$

8 $O = 6$

Assign 6
Remove all 6's by FC

$T = \{8\}$
$O = \{6\}$
$R = \{2\}$

9 $W = 3$

$$\begin{array}{r} 836 \\ +\ 836 \\ \hline 1672 \end{array}$$

Assign 3 by LCV
Solution Found

Max O'Cull
0028687211
CS 471
Homework 2
02/04/19

2. Show how a single ternary constraint such as A + B = C can be turned into three binary constraints by using an auxiliary variable. You may assume finite domains. (Hint: Consider a new variable that takes on values that are pairs of other values, and consider constraints such as X is the first element of the pair Y). Show how constraints with more than three variables can be treated similarly.

   Let there be an auxiliary variable k which has a domain containing all valid 3-tuples which would satisfy the aforementioned constraint. The constraint $k[A] \in A$ is only satisfied if the tuple v within the domain of k is consistent with a value a contained in the domain of A, thus $v[A]=a$. We can then define two more constraints, $k[B] \in B$ and $k[C] \in C$ following the same rules.

   Following the methodology from above, one can observe that a larger constraint containing n variables can then be reduced to a set of n constraints between an auxiliary variable and the original variables.

3. Answer the following questions regarding local search.

   a. $False \vDash True$

      Correct, since False is never true, RHS is never compared to LHS.

   b. $True \vDash False$

      Incorrect, since False is not valid in all models of True.

   c. $(A \wedge B) \vDash (A \Leftrightarrow B)$

      Correct, since LHS is true when A and B are true

   d. $A \Leftrightarrow B \vDash A \vee B$

      Incorrect, because when A is true and B is true, LHS is true, but RHS is not.

   e. $A \Leftrightarrow B \vDash \neg A \vee B$

      Correct, because when B is true but A is false, RHS becomes true but LHS is false.

   f. $(A \wedge B) \Rightarrow C \vDash (A \Rightarrow C) \vee (B \Rightarrow C)$

      Incorrect, since if only B is true, then LHS is true, but RHS is false.

   g. $(C \vee (\neg A \wedge \neg B)) = ((A \Rightarrow C) \wedge (B \Rightarrow C))$

      Correct, the truth values of these are equivalent for all values of A, B, and C.

   h. $(A \vee B) \wedge (C \vee \neg D \vee E) \vDash (A \vee B)$

      Correct, if all values are false, then both LHS and RHS are always false.

   i. $(A \vee B) \wedge (\neg C \vee \neg D \vee E) \vDash (A \vee B) \wedge (\neg D \vee E)$

Max O'Cull
0028687211
CS 471
Homework 2
02/04/19

Incorrect, when only B and D are true, LHS is true, but RHS is false.

j. $(A \vee B) \wedge \neg(A \Rightarrow B)$ is satisfiable

Correct, satisfied when A is true and B is false.

k. $(A \Leftrightarrow B) \wedge (\neg A \vee B)$ is satisfiable

Correct, satisfied when both A and B are false, or when both A and B are true. This can be simplified as biconditional.

l. $(A \Leftrightarrow B) \Leftrightarrow C$ has the same number of models as $(A \Leftrightarrow B)$ for any fixed set of proposition symbols that includes A, B, C

Incorrect, because $A \Leftrightarrow B$ will have 4 models, whereas $(A \Leftrightarrow B) \Leftrightarrow C$ will have 8, in any arrangement of A, B, and C.

4. According to some, a person who likes cats (C), likes dogs (D) if he/she likes all animals (A), but otherwise does not. Which of the following are correct representations of this assertion?

   a. $(C \wedge D) \Leftrightarrow A$

      Incorrect, because you don't have to like cats and dogs in order to like all animals (we're interpreting in a shallow manner, according to Piazza post #76).

   b. $C \Rightarrow (D \Leftrightarrow A)$

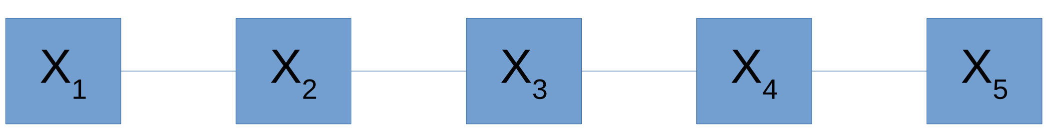
      Correct, because if one likes cats, then they like dogs if and only if they like all animals.

   c. $C \Rightarrow ((A \Rightarrow D) \vee \neg D)$

      Incorrect, because this statement is true no matter what.

5. This question considers representing satisfiability (SAT) problems as CSPs.

   a. Draw the constraint graph corresponding to the SAT problem $(\neg X_1 \vee X_2) \wedge (\neg X_2 \vee X_3) \wedge \ldots \wedge (\neg X_{n-1} \vee X_n)$ for $n = 5$



   b. How many solutions are there for this general SAT problem as a function of n?

Max O'Cull
0028687211
CS 471
Homework 2
02/04/19

This problem has 6 satisfying solutions. One for all values of X being false, and one solution for each X. So in the case of n variables, there would be $n+1$ solutions.

c. Suppose we apply BACKTRACKING-SEARCH (page 215) to find all solutions to a SAT CSP of the type given in (a). (To find all solutions to a CSP, we simply modify the basic algorithm so it continues searching after each solution is found.) Assume that variables are ordered X1 , ..., Xn and false is ordered before true. How much time will the algorithm take to terminate? (Write an O() expression as a function of n.)

The Backtracking-Search algorithm is a form of recursive Depth First Search (DFS). This SAT problem generates a search tree containing $n!*2^n$ leaves. Normally in a backtracking search algorithm, a single solution is found, but in this one, all solutions must be found, thus the DFS will examine every leaf of the tree. Therefore, this algorithm runs in the same time as worst-case DFS, thus the runtime should be $O(n!*2^n)$.