CS 4341: Homework 2

Adam Camilli (aocamilli@wpi.edu)

February 10, 2018

Ch 6: CSP

1. **Problem 6.2:** Consider the problem of placing k knights on an $n \times n$ chessboard such that no two knights are attacking each other, where k is given and $k < n^2$.

   (a) Choose a CSP formulation. In your formulation, what are the variables?

   $$X = \text{set of all } k \text{ knights: } \{X_1, X_2, \ldots X_k\}$$

   (b) What are the possible values of each variable?

   $$D = (x, y) \text{ location of each variable of } X$$

   (c) What sets of variables are constrained, and how?

   Each knight X is constrained by the values of all other knights in X:
   Given $D(X_a) = (x, y, \mathit{occupied})$, and $D(X_b) = (i, j, \mathit{occupied})$,

   $$\Big(i = x \oplus j = y\Big) \wedge \Big(i = x \pm 1 \oplus j = y \pm 2\Big) \wedge \Big(i = x \pm 2 \oplus j = y \pm 1\Big)$$

   (d) Now consider the problem of putting *as many knights as possible* on the board without any attacks. Explain how to solve this with local search by defining appropriate ACTIONS and RESULT functions and a sensible objective function.

      i. Assign all k knights to random values (coordinates).
      ii. Go through X, change $D_i(X_i)$ to coordinates with lowest score. An (x, y) receives +1 for every knight attacking it and +1000 for every two knights on it.
      iii. After changing every knight in X in this way, if the sum of all scores of every spot on the board is greater than zero, repeat with $k - 1$ knights.

2. **Problem 6.9** Explain why it is a good heuristic to choose the variable that is *most* constrained but the value that is *least* constrained in a CSP search.

   The variables that are most constrained are the most likely to cause a failure. By selecting the most constrained variables and giving them the least constraining value available, failures will be detected as early as possible and resolved as quickly as possible.

3. **Problem 6.11** Use the AC-3 algorithm to show that arc consistency can detect the inconsistency of the partial assignment $\{WA = \mathit{green}, V = \mathit{red}\}$ for the problem shown in Figure 6.1.

   Here is one possible trace of AC-3:

   - $SA, WA =$ green : Remove $SA - WA$ and delete green from $D(SA)$.
   - Remove $SA - V$ and delete red from SA and leave only blue.
   - Remove $SA - NT$ and delete blue from NT.
   - Remove $NT - WA$ and delete green from NT and leave only red.

- Remove $SA - NSW$ and delete blue from NSW.
- Remove $NSW - V$ and delete red from NSW leave only green.
- Remove $Q - NT$ and delete red from Q.
- Remove $Q - NSW$ and delete red,green from Q.
- Remove $Q - SA$ and delete blue. **No assignment left for Q, inconsistency detected from partial assignment**.

4. This constraint satisfaction problem is a simplified version of Sudoku in a 4x4 matrix. The goal is to fill in each cell in the matrix with a number between 1 and 4 in such a way that no number is repeated on the same column or on the same row. To save you time, some cells have already been filled in with a value. The remaining ones have been named with a letter for easy reference. These letters, A, B, C, D, E, F and G, are the variables in the constraint satisfaction problem.

2	A	3	B
4	C	1	2
1	D	E	F
3	G	4	1

Variables: A-G
Domain: $\{1, 2, 3, 4\}$
Constraints: There is a constraint between each pair of cells P and Q that belong to the same column or to the same row of the matrix stating that the values assigned to the two cells cannot be equal.

Answer the questions below as if you were an agent following the CSP algorithms we studied in class.

(a) Fill in the table below (some values are provided as examples to guide you. For instance, A has two remaining values, 1 and 4, and it has constraints with four other variables B, C, D, G.):

Variable	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Remaining values	$\mathbf{1, 4}$	4	3	$\mathbf{2, 3, 4}$	2	$3, 4$	2
# of constraints with other variables	**four**	two	three	**five**	two	three	three

(b) Using the Minimum Remaining Values (MRV) heuristic, list the variable that the CSP search algorithm will select next. If there are ties, list all the variables that have the same MRV.

Most Constrained: $B(2), C(3), E(2), G(3)$

Most Constraining: $B(2) : \Big(A(4), F(3)\Big), E(2) : \Big((D(5), F(3)\Big)$

(c) If the above was a tie, use the degree heuristic (i.e., variable with the most constraints on remaining variables) to break the tie. What variable would be selected? If a tie still remains, provide a systematic way to deal with the tie so that only one variable is selected. Explain your work.

Smallest Sum of Constraining Variables: B constrains A and F, which have a combined 7 constraints with other variables. This beats E, which constrains D and F for 8.

(d) Starting from the following possible values, use forward checking to propagate constraints. Show the propagation of just one constraint at a time neatly on a separate row in the table below, until no more constraints can be propagated. An example is provided on the 3rd row. You may not need all the rows provided here.

Variable	**A**	**B**	**C**	**D**	**E**	**F**	**G**
Possible values	$1, 4$	4	3	$2, 3, 4$	2	$3, 4$	2
Constraint between: **A and B**	1,4	4	3	2,3,4	2	3,4	2
Constraint between: **B and C**	1,4	4	3	2,3,4	2	3,4	2
Constraint between: **B and D**	1,4	4	3	2,3	2	3,4	2
Constraint between: **B and E**	1,4	4	3	2,3,4	2	3,4	2
Constraint between: **B and F**	1,4	4	3	2,3,4	2	3	2
Constraint between: **B and G**	1,4	4	3	2,3,4	2	3,4	2

Ch 7: Logical Agents

5. **Problem 7.4 (a,b,c,i,j)** Which of the following are correct?
   (a) $False \models True$: **Correct**
   (b) $True \models False$: **Incorrect**
   (c) $(A \wedge B) \models (A \leftrightarrow B)$: **Correct**
   (d) (i) $(A \vee B) \wedge (\neg C \vee \neg D \vee E) \models (A \vee B) \wedge (\neg D \vee E)$: **Incorrect**
   (e) (j) $(A \vee B) \wedge \neg(A \leftrightarrow B)$ is satisfiable: **Correct**

6. **Problem 7.10** Decide whether each of the following sentences is valid, unsatisfiable, or neither. Verify your decisions using truth tables or the equivalence rules of Figure 7.11 (page 253).
   (a) $Smoke \Rightarrow Smoke$: Valid

$Smoke$	$Smoke \rightarrow Smoke$
T	T
F	F

   (b) $Smoke \Rightarrow Fire$: Neither

$Smoke$	$Fire$	$Smoke \rightarrow Smoke$
T	T	T
T	F	F
F	T	T
F	F	T

   (c) $(Smoke \Rightarrow Fire) \Rightarrow (\neg Smoke \Rightarrow \neg Fire)$: Neither

$Smoke$	$Fire$	$\neg Smoke$	$\not{F}ire$	$Smoke \rightarrow Fire$	$\neg Smoke \rightarrow \neg Fire$	$(Smoke \Rightarrow Fire) \Rightarrow (\neg Smoke \rightarrow \neg Fire)$
T	T	F	F	T	T	T
T	F	F	T	F	T	T
F	T	T	F	T	F	F
F	F	T	T	T	T	T

(d) $Smoke \vee Fire \vee \neg Fire$: Valid

$Smoke$	$Fire$	$lnotFire$	$Smoke \vee Fire \vee \neg Fire$
T	T	F	T
T	F	T	T
F	T	F	T
F	F	T	T

(e) $((Smoke \wedge Heat) \Rightarrow Fire) \leftrightarrow ((Smoke \Rightarrow Fire) \vee (Heat \Rightarrow Fire))$: Valid

$Smoke$	$Heat$	$Fire$	$Smoke \wedge Heat$	$(Smoke \wedge Heat) \Rightarrow Fire$	$Smoke \Rightarrow Fire$	$Heat \Rightarrow Fire$	$(Smoke \Rightarrow Fire) \vee (Heat \Rightarrow Fire)$	$((Smoke \wedge Heat) \Rightarrow Fire) \leftrightarrow ((Smoke \Rightarrow Fire) \vee (Heat \Rightarrow Fire))$
T	T	T	T	T	T	T	T	T
T	T	F	T	F	F	F	F	T
T	F	T	F	T	T	T	T	T
T	F	F	F	T	F	T	T	T
F	T	T	F	T	T	T	T	T
F	T	F	F	T	T	F	T	T
F	F	T	F	T	T	T	T	T
F	F	F	F	T	T	T	T	T

(f) $(Smoke \Rightarrow Fire) \Rightarrow ((Smoke \wedge Heat) \Rightarrow Fire)$: Valid

$Smoke$	$Heat$	$Fire$	$Smoke \Rightarrow Fire$	$Smoke \wedge Heat$	$(Smoke \wedge Heat) \Rightarrow Fire$	$(Smoke \Rightarrow Fire) \Rightarrow ((Smoke \wedge Heat) \Rightarrow Fire)$
T	T	T	T	T	T	T
T	T	F	F	T	F	T
T	F	T	T	F	T	T
T	F	F	F	F	T	T
F	T	T	T	F	T	T
F	T	F	T	F	T	T
F	F	T	T	F	T	T
F	F	F	T	F	T	T

(g) $Big \wedge Dumb \wedge (Big \Rightarrow Dumb)$: Valid

Big	$Dumb$	$Big \Rightarrow Dumb$	$Big \vee Dumb \vee (Big \Rightarrow Dumb)$
T	T	F	T
T	F	T	T
F	T	F	T
F	F	T	T

7. **Problem 7.18** Consider the following sentence:

$$[(Food \Rightarrow Party) \vee (Drinks \Rightarrow Party)] \Rightarrow [(Food \wedge Drinks) \Rightarrow Party]$$

(a) Determine, using enumeration, whether this sentence is valid, satisfiable (but not valid), or unsatisfiable. **Valid**

$Food$	$Party$	$Drinks$	$Food \Rightarrow Party$	$Drinks \Rightarrow Party$	$(Food \Rightarrow Party) \vee (Drinks \Rightarrow Party)$	$Food \vee Drinks$	$(Food \wedge Drinks) \Rightarrow Party$	$[(Food \Rightarrow Party) \vee (Drinks \Rightarrow Party)] \Rightarrow [(Food \wedge Drinks) \Rightarrow Party]$
T	T	T	T	T	T	T	T	T
T	T	F	T	T	T	F	T	T
T	F	T	F	F	F	T	F	T
T	F	F	F	T	T	F	T	T
F	T	T	F	T	T	F	T	T
F	T	F	T	T	T	F	T	T
F	F	T	T	F	T	F	T	T
F	F	F	T	T	T	F	T	T

(b) Convert the left-hand and right-hand sides of the main implication into CNF, showing each step, and explain how results confirm your answer to (a).
Left hand side: $(Food \Rightarrow Party) \vee (Drinks \Rightarrow Party)$
CNF:

$$(\neg Food \vee Party) \vee (\neg Drinks \vee Party)$$

$$(\neg Food \vee Party \vee \neg Drinks \vee Party)$$

$$(\neg Food \vee \neg Drinks \vee Party)$$

Right hand side: $(Food \wedge Drinks) \Rightarrow Party$
CNF:

$$\neg(Food \wedge Drinks) \vee Party$$

$$(\neg Food \vee \neg Drinks) \vee Party$$

$$\neg(Food \vee \neg Drinks \vee Party)$$

Valid, both sides equal and therefore is is of the form $Q \Rightarrow Q$ which is true.

(c) Prove answer to (a) using resolution.

$$\neg[(Food \Rightarrow Party) \vee (Drinks \Rightarrow Party)] \vee [(Food \wedge Drinks) \Rightarrow Party]$$

$$\neg[(\neg Food \vee Party) \vee (\neg Drinks \vee Party)] \vee [\neg(Food \wedge Drinks) \vee Party]$$

$$[(Food \wedge \neg Party) \wedge (Drinks \wedge \neg Party)] \vee [\neg(Food \vee \neg Drinks) \wedge Party]$$

$$[Food \wedge Drinks \wedge \neg Party \vee \neg Food \vee \neg Drinks \wedge Party$$

Empty clause resolved, sentence valid.

8. Convert each of the following sentences to clausal form.

(a) $P \Rightarrow Q$

$$\neg P \vee Q$$

(b) $L \wedge M \Rightarrow P$

$$\neg L \vee \neg M \vee P$$

(c) $B \wedge L \Rightarrow M$

$$\neg B \vee \neg L \vee M$$

(d) $A \wedge B \Rightarrow L$

$$\neg A \vee \not B \vee L$$

(e) A

$$A$$

(f) B

$$B$$

Ch 8: First Order Logic

9. **Problem 8.10** Consider a vocabulary with the following symbols:

$$Occupation(p, o) : \text{Predicate. Person } p \text{ has occupation } o.$$

$$Customer(p_1, p_2) : \text{Predicate. Person } p_1 \text{ is customer of person } p_2.$$

$$Boss(p_1, p_2) : \text{Predicate. Person } p_1 \text{ is a boos of person } p_2.$$

$$Doctor, Surgeon, Lawyer, Actor : \text{Constants denoting occupations.}$$

$$Emily, Joe : \text{Constants denoting people.}$$

Use these symbols to write the following assertions in first-order logic:

(a) Emily is either a surgeon or lawyer.

$$Occupation(Emily, Surgeon) \vee Occupation(Emily, Lawyer)$$

(b) Joe is an actor, but he also holds another job.

$$\exists o (o \neq Actor) \wedge Occupation(Joe, Actor) \wedge Occupation(Joe, o)$$

(c) All surgeons has doctors.

$$\forall p Occupation(p, Surgeon) \Rightarrow Occupation(p, Doctor)$$

(d) Joe does not have a lawyer (i.e., is not a customer of any lawyer)

$$\neg \exists p Occupation(p, Lawyer) \wedge Customer(Joe, p)$$

(e) Emily has a boss who is a lawyer.

$$\exists p Boss(p, Emily) \wedge Occupation(p, Lawyer)$$

(f) There exists a lawyer all of whose customers are doctors.

$$\exists p_1 Occupation(p_1, Lawyer) \wedge \forall p_2 Customer(p_2, p_1) \Rightarrow Occupation(p_2, Doctor)$$

(g) Every surgeon has a lawyer.

$$\forall p_1 Occupation(p_1, Surgeion) \Rightarrow \exists p_2 Occupation(p_2, Lawyer) \wedge Customer(p_1, p_2)$$

10. **Problem 8.11** Complete the following exercises about logical sentences:

(a) Translate into *good,natural* English (no x's or y's!):

$\forall x, y, l \; SpeaksLanguage(x, l) \wedge SpeaksLanguage(y, l) \Rightarrow Understands(x, y) \wedge Understands(y, x)$

People who speak the same language understand each other.

(b) Explain why this sentence is entailed by the sentence

$$\forall x, y, l\ SpeaksLanguage(x, l) \wedge SpeaksLanguage(y, l) \Rightarrow Understands(x, y)$$

Since both people speak the same language, the sentence becomes true, implying both people can understand each others' language. Implication condition is valid for both sentences, therefore they entail each other.

(c) Translate into first-order logic the following sentences:

i. Understanding leads to friendship.

$$\exists x, y\ Understands(x, y) \Rightarrow Friends(x, y)$$

ii. Friendship is transitive.

$$\forall x, y, z\ Friends(x, y) \wedge Friends(y, z) \Rightarrow Friends(x, z)$$

11. **Problem 8.20** Arithmetic assertions can be written in first-order logic with the predicate symbol ¡, the function symbols + and ×, and the constant symbols 0 and 1. Additional predicates can also be defined with biconditionals.

(a) Represent the property "x is an even number"

$$S = \exists b(x \approx b + b)$$

(b) Represent the property "x is prime"

$$S = (1 < x) \wedge \forall b\Big((b|x) \rightarrow \big((b \approx 1) \vee (b \approx a)\big)\Big)$$

(c) Goldbach's conjecture is the conjecture (unproven as yet) that every even number is equal to the sum of two primes. Represent this conjecture as a logical sentence.

Let two prime numbers be X_a, Y_a.

$$S = X_a + Y_b$$

12. **Problem 8.23** For each of the following sentences in English, decide if the following accompanying first-order logic sentence is a good translation. If not, explain why not and correct it. (Some sentences may have more than one error!)

(a) No two people have the same social security number.

$$\neg\exists x, y, n person(x) \wedge person(y) \Rightarrow \big[HasSS\#(x, n) \wedge HasSS\#(y, n)\big]$$

This is is not a good representation, it implies that no people exist. Correction:

$$\neg\exists x, y, n person(x) \wedge person(y) \wedge \neg(x = y) \Rightarrow \big[HasSS\#(x, n) \wedge HasSS\#(y, n)\big]$$

(b) John's social security number is the same as Mary's.

$$\exists n HasSS\#(John, n) \wedge HasSS\#(Mary, n)$$

This is a good representation.

(c) Everyone's social security number has nine digits.

$$\forall x, n person(x) \Rightarrow [HasSS\#(x, n) \wedge Digits(n, 9)]$$

This is not a good representation. It presumes everyone has a number. Correction:

$$\forall x, n(person(x) \wedge HasSS\#(x, n)) \Rightarrow Digits(n, 9)$$

Ch 9: Inference from First Order Logic

13. **Problem 9.9** Suppose you are given the following axioms:

$$1.0 \leq 3$$

$$2.7 \leq 9$$

$$3.\forall x \ x \leq x$$

$$4.\forall x \ x \leq x + 0$$

$$5.\forall x \ x + 0 \leq x$$

$$6.\forall x, y \ x + y \leq y + x$$

$$7.\forall w, x, y, z \ w \leq y \land x \leq z \Rightarrow y + z$$

$$8.\forall x, y, z \ x \leq y \land y \leq z \Rightarrow x \leq z$$

(a) Give a backward-chaining proof of sentence 2.

$$8.7 + 0 \leq 7 \land 7 \leq (3 + 9) \Rightarrow (7 + 0) \leq 3 + 9$$

$$6.9 + 3 \leq 3 + 9$$

$$7.7 \leq 9 \land 0 \leq 3 \Rightarrow 7 + 0 \leq 3 + 9$$

$$1.0 \leq 3$$

$$2.7 \leq 9$$

(b) Give a forward-chaining proof of sentence 2.

$$7.7 \leq 9 \land 0 \leq 3 \Rightarrow 7 + 0 \leq 9 + 3$$

$$6.9 + 3 \leq 3 + 9$$

$$4.7 \leq 7 + 0$$

$$8.7 + 0 \leq 9 + 3 \land 9 + 3 \Rightarrow 7 + 0 \leq 3 + 9$$

14. **Problem 9.6** Write down logical representations for the following sentences, suitable for use wuth Generalized Modus Ponens:

(a) Horses, cows, and pigs are mammals.

$$Horse(x) \Rightarrow Mammal(x)$$

$$Cow(x) \Rightarrow Mammal(x)$$

$$Pig(x) \Rightarrow Mammal(x)$$

(b) An offspring of a horse is a horse.

$$Offspring(x, y) \land Horse(y) \Rightarrow Horse(x)$$

(c) Bluebeard is a horse.

$$Horse(Bluebeard)$$

(d) Bluebeard is Charlie's parent.

$$Parent(Bluebeard, Charlie)$$

(e) Offspring and parent are inverse relations.

$$Offspring(x, y) \Rightarrow Parent(y, x)$$

$$Parent(x, y) \Rightarrow Offspring(y, x)$$

(f) Every mammal has a parent.

$$Mammal(x) \Rightarrow Parent(G(x), x)$$

15. **Problem 9.13a** Use the sentences you wrote in **9.6** to answer a question by using a backward-chaining algorithm.

    (a) Draw the proof tree generated by an exhaustive backward-chaining algorithm for the query $\exists h Horse(h)$ where clauses are matched in the order given.

$$h = \text{Charlie}, y = \text{Bluebeard}$$