

CS322 Fall 1999  
Module 6 (Constraint Satisfaction Problems)  
Assignment 6

Due: 1:30pm, Wednesday 20 October 1999.

The aim of this assignment is to learn about constraint satisfaction problems.

### Question 1

In this question you will look at backtracking versus arc consistency for solving CSP problems.

Consider a scheduling problem, where there are five variables  $A, B, C, D,$  and  $E,$  each with domain  $\{1, 2, 3, 4\}.$  Suppose the constraints are:  $E - A$  is odd,  $A < D, D < C, E > B, A \neq B, E \neq C, E \neq D.$

- (a) Show how backtracking can be used to solve this problem, using the variable ordering  $A, B, C, D, E.$  To do this you should draw the search tree generated to find all answers. Indicate clearly the satisfying assignments. How many leaves are in the search tree generated?

To indicate the search tree, write it in text form with each branch on one line. For example, suppose we had variables  $X, Y$  and  $Z$  with domains  $t, f,$  and constraints  $X \neq Y, Y \neq Z.$  The corresponding search tree can be written as:

```
X=t Y=t failure
      Y=f Z=t solution
          Z=f failure
X=f Y=t Z=t failure
          Z=f solution
          Y=f failure
```

Hint: this may be large! It may be easier to write a program to generate it (but then again, it may not be easier; try it by hand first).

- (b) Is there a different variable ordering that results in a smaller tree? Give a variable ordering that results in the smallest tree. How many leaves are on this tree? Explain why you think this is the optimal ordering. (A good explanation is more important than actually finding the optimal ordering).
- (c) Show how arc consistency can be used to solve this problem. To do this you need to:
- i) draw the constraint graph,
  - ii) show which elements of a domain are deleted at each step, and which arc is responsible for removing the element,
  - iii) show explicitly the constraint graph after arc consistency has stopped.
  - iv) show how splitting domains can be used to solve this problem. Assume we split the alphabetically first variable that has more than one element in its domain for any arc-consistent network. Include all arc consistency steps.

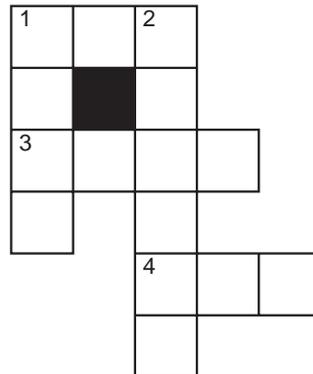


Figure 1: Simple Crossword

## Question 2

Consider the crossword puzzle shown in Figure 1.

Suppose that each word position is a variable, with domain a set of words. The value of the word position is the word that goes into that position.

Suppose we have the following variables and domains:

Variable	Domain
1-across	{ant, big, bus, car, has}
3-across	{book, buys, hold, lane, year}
4-across	{ant, big, bus, car, has}
1-down	{book, buys, hold, lane, year}
2-down	{beast, ginger, search, symbol, syntax}

The constraints are that the intersecting words have to have the same letter in the intersecting positions.

- This is not arc consistent. Give one domain value that can be pruned. Explain which arc can be used to prune it any why.
- Give the domains where arc consistency stops (without domain pruning).
- Show how hill climbing can be used for this problem. Suppose a neighbour is obtained by changing the value of one of the variables to the alphabetically next or previous domain element. The heuristic function to be maximized is the number of satisfied constraints, and you always choose a neighbour with the maximal heuristic value (even if it is the same as the current node).
  - Show the sequence of assignments and corresponding heuristic values when we start with the assignment:  $1\text{-across}=\text{ant}$ ,  $3\text{-across}=\text{book}$ ,  $4\text{-across}=\text{ant}$ ,  $1\text{-down}=\text{book}$ ,  $2\text{-down}=\text{brown}$ .
  - Suppose Alex suggested using the domains pruned by arc consistency, but otherwise the same. Is this a good idea? Explain. Does it work well?
  - Suppose Joe suggested picking any domain element as long as it improves the heuristic value. Why might this be better or worse than the original solution? Does it work well?

## Question 3

For each question in this assignment, say how long you spent on it. Was this reasonable? What did you learn?