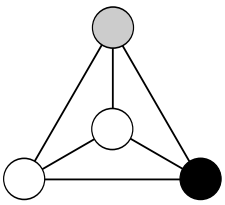
CS 188
Fall 2019

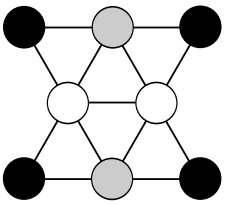
Midterm Review CSPs Solutions

Q1. CSPs

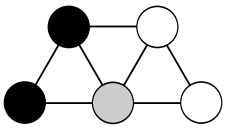
In this question we are considering CSPs for map coloring. Each region on the map is a variable, and their values are chosen from {black, gray, white}. Adjacent regions cannot have the same color. The figures below show the constraint graphs for three CSPs and an assignment for each one. None of the assignments are solutions as each has a pair of adjacent variables that are white. For both parts of this question, let the score of an assignment be the number of satisfied constraints (so a higher score is better).



(1)



(2)



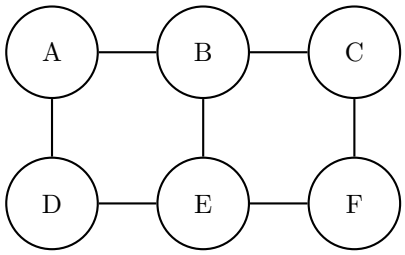
(3)

(a) Consider applying Local Search starting from each of the assignments in the figure above. For each successor function, indicate whether each configuration is a local optimum and whether it is a global optimum (note that the CSPs may not have satisfying assignments).

Successor Function	CSP	Local optimum?		Global Optimum?	
Change a single variable	(1)	**Yes**	No	**Yes**	No
	(2)	**Yes**	No	Yes	**No**
	(3)	Yes	**No**	Yes	**No**
Change a single variable, or a pair of variables	(1)	**Yes**	No	**Yes**	No
	(2)	Yes	**No**	Yes	**No**
	(3)	Yes	**No**	Yes	**No**

Q2. Worst-Case Backtracking

Consider solving the following CSP with standard backtracking search where we enforce arc consistency of all arcs before every variable assignment. Assume every variable in the CSP has a domain size $d > 1$.



(a) For each of the variable orderings, mark the variables for which backtracking search (with arc consistency checking) could end up considering more than one different value during the search.

(i) Ordering: A, B, C, D, E, F

■ A ■ B ☐ C ☐ D ☐ E ☐ F

(ii) Ordering: B, D, F, E, C, A

☐ A ■ B ☐ C ☐ D ☐ E ☐ F

Since we are enforcing arc consistency before every value assignment, we will only be guaranteed that we won't need to backtrack when our remaining variables left to be assign form a tree structure (or a forest of trees). For the first ordering, after we assign A and B, the nodes C, D, E, F, form a tree. For the second ordering, after we assign B, the nodes A, C, D, E, F form a tree.

(b) Now assume that an adversary gets to observe which variable ordering you are using, and after doing so, gets to choose to add one additional binary constraint between any pair of variables in the CSP in order to maximize the number of variables that backtracking could occur in the worst case. For each of the following variable orderings, select which additional binary constraint should the adversary add. Then, mark the variables for which backtracking search (with arc consistency checking) could end up considering more than one different value during the search when solving the modified CSP.

(i) Ordering: A, B, C, D, E, F

The adversary should add the additional binary constraint:

○ AC ○ AE ○ AF ○ BD
○ BF ○ CD ○ CE ● DF

When solving the modified CSP with this ordering, backtracking might occur at the following variable(s):

■ A ■ B ■ C ■ D ☐ E ☐ F

By adding the edge DF, now only after we assign A, B, C, D, the remaining nodes E, F form a tree.

(ii) Ordering: B, D, F, E, C, A

The adversary should add the additional binary constraint:

○ AC ○ AE ○ AF ○ BD
○ BF ○ CD ● CE ○ DF

When solving the modified CSP with this ordering, backtracking might occur at the following variable(s):

☐ A ■ B ☐ C ■ D ☐ E ■ F

By adding the edge CE, now only after we assign B, D, F, the remaining nodes A, C, E form a tree.

Q3. CSPs: Midterm 1 Staff Assignments

(a) We will model this problem as a constraint satisfaction problem (CSP). Our variables correspond to each of the staff members, J, F, N, D, M, B, K, and the domains are the questions 1, 2, 3, 4, 5, 6. After applying the unary constraints, what are the resulting domains of each variable? (The second grid with variables and domains is provided as a back-up in case you mess up on the first one.)

B					5	
D	1	2	3	4		6
F	1	2	3			
J	1	2	3	4	5	6
K	1	2	3			
N	1	2	3	4	5	
M	1		3		5	

(b) If we apply the Minimum Remaining Value (MRV) heuristic, which variable should be assigned first?

Brad – because he has the least values left in his domain.

(c) Normally we would now proceed with the variable you found in (b), but to decouple this question from the previous one (and prevent potential errors from propagating), let's proceed with assigning Michael first. For value ordering we use the Least Constraining Value (LCV) heuristic, where we use *Forward Checking* to compute the number of remaining values in other variables domains. What ordering of values is prescribed by the LCV heuristic? Include your work—i.e., include the resulting filtered domains that are different for the different values.

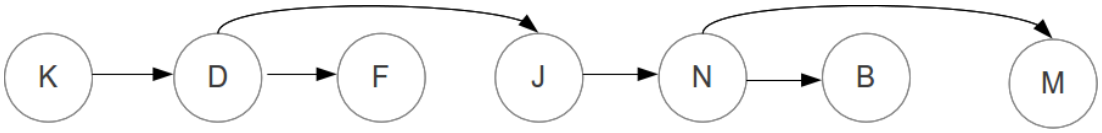
Michael's value will be assigned as 5, 3, 1, in that order.

Why these variables? They are the only feasible variables for Michael. Why this order? This is the increasing order of the number of constraints on each variable.

The only binary constraint incolving Michael is "Nick (N) must work on a question that's before Michael (M)'s question." So, only Nick's domain is affected by forward checking on these assignments, and it will change from {1, 2, 3, 4, 5} to {1, 2, 3, 4}, {1, 2}, and { } for the assignments 5, 3, 1, respectively.

(d) Realizing this is a tree-structured CSP, we decide not to run backtracking search, and instead use the efficient two-pass algorithm to solve tree-structured CSPs. We will run this two-pass algorithm after applying the unary constraints from part (a). Below is the linearized version of the tree-structured CSP graph for you to work with.

(i) First Pass: Domain Pruning. Pass from *right to left* to perform Domain Pruning. Write the values that remain in each domain below each node in the figure above.



1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5
6	6	6	6	6	6	6

Remaining values in each domain after the domain pruning right-to-left pass:
Kyle: 1
Donahue: 1,2
Ferguson: 1,2,3
Judy: 2,3,4,5,6
Nick: 1,2,3,4
Brad: 5
Michael: 1,3,5

(ii) Second Pass: Find Solution. Pass from *left to right*, assigning values for the solution. If there is more than one possible assignment, choose the highest value.

Assigned Values after the left-to-right pass:

Kyle: 1
Donahue: 2
Ferguson: 3
Judy: 6
Nick: 4
Brad: 5
Michael: 5