CS-GY 6613 HW3
Runze Li

nyu.edu

Q1 (a) If using DFS algo to solve constrain satisfaction problems,
the time complexity is $O(n!d^n)$,
the space complexity is $O(n^2d)$
In CSP, there are n variables, each with domain size d.

(b) If using Backtracking-Search algo to solve constrain satisfaction problems,
the time complexity is $O(d^n)$
the space complexity is $O(n)$
In CSP. there are n variables, each with domain size d.

Q2 When using the minimum remaining value and degree heuristics for selecting variables, we first use MRV and if there are the same variables with minimum remaining values, then use degree heuristics. So the tree is

[ ]

↓

[NT = blue]

↙ ↘

[NT = blue, SA = red] [NT = blue, SA = green]

↓

[NT = blue, SA = red, Q = green]

↓

[NT = blue, SA = red, Q = green, NSW = blue]

↓

[NT = blue, SA = red, Q = green, NSW = blue, WA = green]

↓

[NT = blue, SA = red, Q = green, NSW = blue, WA = green, V = green]

↙ ↓ ↘

[NT = blue, SA = red, Q = green, NSW = blue, WA = green, V = green, T = red] [NT = blue, SA = red, Q = green, NSW = blue, WA = green, V = green, T = green] [NT = blue, SA = red, Q = green, NSW = blue, WA = green, V = green, T = blue]

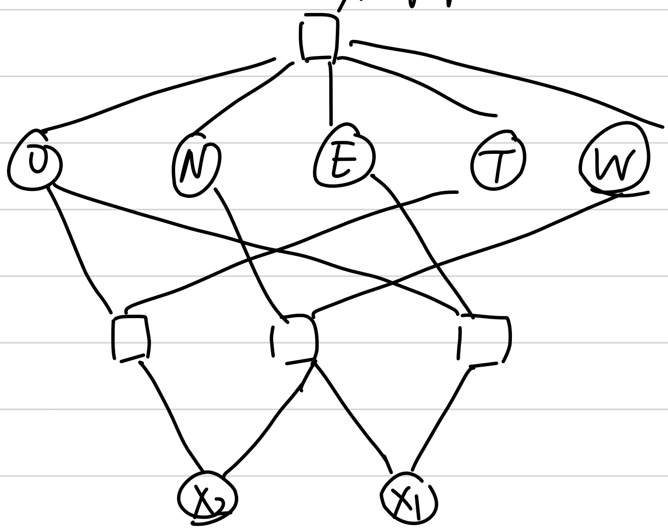
⇑

(complete)

Finally, we can get one solution {NT = blue, SA = red, Q = green, NSW = blue, WA = green, V = green, T = red}

Q3. Given the specifications of the cryptarithmetic problem, we can get the constraint hypergraph.

Constraint hypergraph



Domains

$O: \{2,4\}$

$T: \{2,3,4,5,6,7,8,9\}$

$N, E, W: \{0,1,2,3,4,5,6,7,8,9\}$

$x_1, x_2: \{0,1\}$

Constraints:

$\text{Alldiff}(O, N, E, T, W)$ (global constraint)

$$\left.\begin{array}{l} E+E = 10x_1 + O \\ x_1 + N + N = 10x_2 + W \\ x_2 + O + O = T \end{array}\right\} \text{(Higher-order constraints)}$$

If there's a tie using MRV, select the variable that is involved in the largest number of constraint with other unassigned variables.

Level 1:

From MRV, x_1, x_2, O have the fewest legal values of 2

From Degree heuristic, O have 6 unassigned neighbors while x_1, x_2 have 5.

Thus we choose O.

Level 2:

From MRV, x_1, x_2 have the fewest legal values of 2.

From Degree heuristic, both x_1 and x_2 have 4 unassigned neighbors.

We can choose x_1.

Level 3:

From MRV, E has the fewest legal value of 1 because of $E+E = 10x_1 + O$

Thus we choose E

Level 4:

From MRV, x_2 has the fewest legal values of 2

Thus we choose x_2

Level 5:

From MRV. T has the fewest legal value of 1 because of $x_2 + O + O = T$

Thus we choose T.

Level 6:

From MRV, both N and W have 7 values and they have 1 unassigned neighbors.

So we choose N.

Level 7:

From MRV, W has 1 legal value.

Thus we choose W.

Finally, we can determine the variable selected $\{O, x_1, E, x_2, T, N, W\}$

Also, if we choose W in level 6, we can get another sequence $\{O, x_1, E, x_2, T, W, N\}$.

What's more, if we choose x_2 in level 2, the process is similar:

Level 3. choose T (only 1 legal value)

Level 4: choose x_1 (2 legal values)

Level 5: choose E (only 1 legal values)

Level 6: choose N or W (both have same values and same unassigned neighbors)

Level 7: choose another

So we can get two more sequences $\{O, x_2, T, x_1, E, N, W\}$ and $\{O, x_2, T, x_1, E, W, N\}$.

Eventually, we can get 4 possible lists of variables above.

Q4 (a)

	B	C	D
initial	RG	R	RGB
After FC	G	R	GB

(b)

	B	C	D
initial	RG	R	RGB
After AC-3	G	R	B

The initial content of the queue is

(B, C), (C, B). (C, D), (D, C), (B. D) (D B)

Pop (B, C), Revise : true, add (D. B) to queue

queue. (C. B). (C. D). (D. C) (B, D). (D. B) (D. B)

B	C	D
~~R~~G	R	RGB

Pop (C. B). Revise = false

queue : (C. D). (D. C) (B. D) (D. B) (D. B)

B	C	D
G	R	RGB

Pop (C. D), Revise = false

queue : (D. C) (B. D) (D. B) (D. B)

B	C	D
G	R	RGB

Pop (D. C), Revise = true add (B. D) to queue

queue : (B. D) (D. B) (D. B) (B. D)

B	C	D
G	R	~~R~~GB

Pop (B. D) Revise = false

queue. (D. B) (D. B) (B. D)

B	C	D
G	R	GB

Pop (D. B) Revise = true add (C. D) to queue

queue : (D. B) (B. D) (C. D)

B	C	D
G	R	~~G~~B

Pop (D. B) . (B. D). (C. D) and their Revise are all false.

Finally, the queue is empty. and the algorithm stops.

The domain value of each region after the algorithm stops is as follows.

After AC-3,

B	C	D
G	R	B