ComS 472
Homework 3

Sean Gordon

Oct 12, 2020

- 6.1 -

1) 3 colors for (SA), 2 for (WA), 1 for (NT, Q, NSW, V), 3 for (T) $\Rightarrow$
   $3 * 2 * 1 * 1 * 1 * 1 * 3 = 18$ solutions
2) 4:(SA), 3:(WA), 2:(NT, Q, NSW, V), 4:(T) $\Rightarrow 4 * 3 * 2 * 2 * 2 * 2 * 4 = 768$ solutions
3) 2:(SA), 1:(WA), 0:(NT, Q, NSW, V), 2:(T) $\Rightarrow 2 * 1 * 0 * 0 * 0 * 0 * 2 = 0$ solutions

---

- 6.6 -

1) Introduce variable D as a set of pair values (D_1, D_2). $A = D_1, B = D_2, and\ C = D_1 + D_2$

2) Constraints with more than 3 variables can be reduced in tiers, with n-1 variables reduced to n, n to n-1, ..., continuing until the set of constraints contains only binary constraints.

3) Unary constraints can be completely eliminated by moving the effects of the constraint into the domain of the variable it is affecting.

- 6.8 -

1) A1 $\rightarrow$ R ✓
2) H $\rightarrow$ R **X**, conflicts with A1 $\rightarrow$ backtrack
3) H $\rightarrow$ G ✓
4) A4 $\rightarrow$ R ✓
5) F1 $\rightarrow$ R ✓
6) A2 $\rightarrow$ R ✓
7) F2 $\rightarrow$ R ✓
8) A3 $\rightarrow$ R ✓
9) T $\rightarrow$ R **X**, conflicts with F1 $\rightarrow$ backtrack
9) T $\rightarrow$ G **X**, conflicts with H $\rightarrow$ backtrack
9) T $\rightarrow$ B ✓

- 6.11 -

Arcs:
$(SA \neq WA), (WA \neq SA), (SA \neq NT), (NT \neq SA), (SA \neq Q), (Q \neq SA),$
$(SA \neq NSW), (NSW \neq SA), (SA \neq V), (V \neq SA), (WA \neq NT), (NT \neq WA),$
$(NT \neq Q), (Q \neq NT), (Q \neq NSW), (NSW \neq Q), (NSW \neq V), (V \neq NSW)$

Initial Constraints:
SA={RGB}, WA={RGB}, NT={RGB}, Q={RGB}, NSW={RGB}, V={RGB}, T={RGB}

Set WA=green, V=red...

1)

C: SA={RGB}, WA={_G_}, NT={RGB}, Q={RGB}, NSW={RGB}, V={R__}, T={RGB}
A: All arcs, $(SA \neq WA), (NT \neq WA), (SA \neq V), (NSW \neq V)$ at the end for conciseness.
Action: Check and dequeue all consistent arcs.

2)

C: SA={RGB}, WA={_G_}, NT={RGB}, Q={RGB}, NSW={RGB}, V={R__}, T={RGB}
A: $(SA \neq WA), (NT \neq WA), (SA \neq V), (NSW \neq V)$
Action: $(SA \neq WA)$ is inconsistent. Remove G from SA, queue arcs with SA on right.

3)

C: SA={R_B}, WA={_G_}, NT={RGB}, Q={RGB}, NSW={RGB}, V={R__}, T={RGB}
A: $(NT \neq WA), (SA \neq V), (NSW \neq V), (WA \neq SA), (NT \neq SA), (Q \neq SA),$
$(NSW \neq SA), (V \neq SA)$
Action: $(NT \neq WA)$ is inconsistent. Remove G from NT, queue arcs with NT on right.

4)

C: SA={R_B}, WA={_G_}, NT={R_B}, Q={RGB}, NSW={RGB}, V={R__}, T={RGB}
A: $(SA \neq V), (NSW \neq V), (WA \neq SA), (NT \neq SA), (Q \neq SA), (NSW \neq SA),$
$(V \neq SA), (WA \neq NT), (Q \neq NT)$
Action: $(SA \neq V)$ is inconsistent. Remove R from SA, queue arcs with SA on right.

5)

C: SA={__B}, WA={_G_}, NT={R_B}, Q={RGB}, NSW={RGB}, V={R__}, T={RGB}
A: $(NSW \neq V), (WA \neq SA), (NT \neq SA), (Q \neq SA), (NSW \neq SA), (V \neq SA),$
$(WA \neq NT), (Q \neq NT)$
Action: $(NSW \neq V)$ is inconsistent. Remove R from NSW, queue arcs with NSW on right.

6)

C: SA={__B}, WA={_G_}, NT={R_B}, Q={RGB}, NSW={_GB}, V={R__}, T={RGB}
A: $(WA \neq SA), (NT \neq SA), (Q \neq SA), (NSW \neq SA), (V \neq SA), (WA \neq NT),$
$(Q \neq NT), (SA \neq NSW), (Q \neq NSW), (V \neq NSW)$
Action: $(WA \neq SA)$ is consistent. Dequeue.

7)

C: SA={__B}, WA={_G_}, NT={R_B}, Q={RGB}, NSW={_GB}, V={R__}, T={RGB}
A: $(NT \neq SA), (Q \neq SA), (NSW \neq SA), (V \neq SA), (WA \neq NT), (Q \neq NT),$
$(SA \neq NSW), (Q \neq NSW), (V \neq NSW)$
Action: $(NT \neq SA)$ is inconsistent. Remove B from NT, queue arcs with NT on right.

8)

C: SA={__B}, WA={_G_}, NT={R__}, Q={RGB}, NSW={_GB}, V={R__}, T={RGB}
A: $(Q \neq SA), (NSW \neq SA), (V \neq SA), (WA \neq NT), (Q \neq NT), (SA \neq NSW),$
$(Q \neq NSW), (V \neq NSW), (SA \neq NT)$
Action: $(Q \neq SA)$ is inconsistent. Remove B from Q, queue arcs with Q on right.

9)

C: SA={__B}, WA={_G_}, NT={R__}, Q={RG_}, NSW={_GB}, V={R__}, T={RGB}
A: $(NSW \neq SA), (V \neq SA), (WA \neq NT), (Q \neq NT), (SA \neq NSW), (Q \neq NSW),$
$(V \neq NSW), (SA \neq NT), (SA \neq Q), (NT \neq Q), (NSW \neq Q)$
Action: $(NSW \neq SA)$ is inconsistent. Remove B from NSW, queue arcs with NSW on right.

10)

C: SA={__B}, WA={_G_}, NT={R__}, Q={RG_}, NSW={_GB}, V={R__}, T={RGB}
A: $(V \neq SA), (WA \neq NT), (Q \neq NT), (SA \neq NSW), (Q \neq NSW), (V \neq NSW),$
$(SA \neq NT), (SA \neq Q), (NT \neq Q), (NSW \neq Q)$
Action: $(V \neq SA)$ is consistent. Dequeue.

11)

C: SA={__B}, WA={_G_}, NT={R__}, Q={RG_}, NSW={_GB}, V={R__}, T={RGB}
A: $(WA \neq NT), (Q \neq NT), (SA \neq NSW), (Q \neq NSW), (V \neq NSW), (SA \neq NT),$
$(SA \neq Q), (NT \neq Q), (NSW \neq Q)$
Action: $(WA \neq NT)$ is consistent. Dequeue.

12)

C: SA={__B}, WA={_G_}, NT={R__}, Q={RG_}, NSW={_GB}, V={R__}, T={RGB}
A: $(Q \neq NT), (SA \neq NSW), (Q \neq NSW), (V \neq NSW), (SA \neq NT), (SA \neq Q),$
$(NT \neq Q), (NSW \neq Q)$
Action: $(Q \neq NT)$ is inconsistent. Remove R from Q, queue arcs with Q on right.

13)

C: SA={__B}, WA={_G_}, NT={R__}, Q={_G_}, NSW={_GB}, V={R__}, T={RGB}
A: $(SA \neq NSW), (Q \neq NSW), (V \neq NSW), (SA \neq NT), (SA \neq Q), (NT \neq Q),$
$(NSW \neq Q)$
Action: $(SA \neq NSW)$ is inconsistent. Remove B from SA, queue arcs with SA on right.

This leaves SA with no possible values, revealing an inconsistency in the partial assignment.

- 6.20 -

Variables: $S_1, S_2, ..., S_n$
Domain: Set of adjacent square pairs a domino can cover (s_1, s_2).
Constraints: No two dominos can cover the same square.

Variables: $D_1, D_2, ..., D_n$
Domain: Set of pairs (s, s'), with s = current square and s' = one of 4 adjacent squares.
Constraints: A square must only link to one other square. Square A can link to square B if square B can link to A.

A tree-structured constraint graph can be accomplished with a straight line of 6 squares.

A tree-structured graph requires there to be no loops. If the layout of the squares has no cycles, the resulting constraint graph will be tree-structured.

- 7.4 -

1) True: Because false will never be true, it doesn't matter what the second half of the equation is.

2) False: True is always true, but False will never be true, so by definition it is incorrect.

3) True: $A \wedge B$ is only true if A and B are logically equivalent, meaning the right side must be true.

4) False: $A \Leftrightarrow B$ can be true when A=B=False, but $A \vee B$ is false.

5) True: $A \Leftrightarrow B$ is only true when A==B, but on the right A!=B so $A \vee B$ must be true.

6) True: The right side is only false when A and B are true but C is false, which couldn't happen because the left side would also be false.

7) True: By truth table, the left is equivalent to the right.

8) True: $A \vee B$ must be true to reach the right side, which is also $A \vee B$.

9) False: If (A or B = true) and (C=false) but (D=true) and (E=false), this fails.

10) True: A=true, B=false

11) True: A=B=true

12) True: (...) $\Leftrightarrow C$ relies on what is within the parentheses, and can never have more or less models than it because it does not add anything.

- 7.6 -

1) By definition of entailment, True $\models \alpha$ when α is true, but not when α is false. Therefore, for True $\models \alpha$ to be true, α must be true already.

2) It doesn't matter what α is, because False is never True.

3) Via truth table, whenever $A \models B$ is true, $A \Rightarrow B$ is also true. $A \models B$ does not depend on $A \Rightarrow B$, but the assertion holds.

4) Via truth table, whenever $A \Leftrightarrow B$ is true, $A \equiv B$ is also true. $A \equiv B$ does not depend on $A \Leftrightarrow B$, but the assertion holds.

5) LHS is valid only if A and B are true, which then invalidates the RHS.

- 7.7 -

1) For $A \wedge B$ to be true, A and B must be true. Then, if just one of A or B $\models$ Y, $A \wedge B$ must also $\models$ Y.

2) For (A $\wedge$ B) to be true, A and B must be true. Then, if (A $\wedge$ B) $\models$ Y, one or both of A or B must also model Y.

3) For (B $\vee$ Y) to be true, B or Y or both must be true. Then, if A is true, and one or both of B or Y is true, A $\models$ either B or Y or both.

- 7.15 -

Clauses:

$S1 : A \Leftrightarrow (B \vee E)$		$S4 : E \Rightarrow B$	$\neg E \vee B$
$S2 : E \Rightarrow D$	$\neg E \vee D$	$S5 : B \Rightarrow F$	$\neg B \vee F$
$S3 : C \wedge F \Rightarrow \neg B$	$(\neg C \vee \neg F \vee \neg B)$	$S6 : B \Rightarrow C$	$\neg B \vee C$

With material implication and de Morgan's law, S3 ($C \wedge F \Rightarrow \neg B$) becomes $(\neg C \vee \neg F \vee \neg B)$.
$\neg C$ can be removed using S6 ($B \Rightarrow C$), becoming $(\neg F \vee \neg B)$.
$\neg F$ can be removed using S5 ($B \Rightarrow F$), becoming $(\neg B)$.

We can break clause S1 ($A \Leftrightarrow (B \vee E)$) into two parts: $(A \Rightarrow (B \vee E))$ and $((B \vee E) \Rightarrow A)$.
Through material implication, $(A \Rightarrow (B \vee E))$ is then transformed into $(\neg A \vee B \vee E)$.
Using clause S4 ($E \Rightarrow B$) we can remove E from $(\neg A \vee B \vee E)$, becoming $(\neg A \vee B)$.
Because we have proven $\neg B$, we can remove B from the sentence, becoming $(\neg A)$.

Thus, $\neg A$ and $\neg B$ are proven separately, so $(\neg A \wedge \neg B)$ must also be valid.

- 7.16 -

1) Through implication elimination $(\neg X \vee Y) = (X \Rightarrow Y)$, so $(\neg P_1 \vee ... \vee \neg P_m \vee Q)$ is equivalent to $(P_1 \vee ... \vee P_m) \Rightarrow Q$.
Then, through de Morgan's law $\neg(P_1 \vee ... \vee P_m)$ becomes $(P_1 \wedge ... \wedge P_m)$.
Thus, $(\neg P_1 \vee ... \vee \neg P_m \vee Q) \equiv (P_1 \wedge ... \wedge P_m) \Rightarrow Q$

2) Any clause can be arranged as $(\neg P_1 \vee ... \vee \neg P_m \vee Q_1 \vee ... \vee Q_n)$.
We can consolidate Q_1 *to* Q_n as Q, turning the arrangement into $(\neg P_1 \vee ... \vee \neg P_m \vee Q)$.
Implication elimination and de Morgan's law can then be applied as above, resulting in $(P_1 \wedge ... \wedge P_m) \Rightarrow Q$.
Q can then be expanded, with a final result of $(P_1 \wedge ... \wedge P_m) \Rightarrow (Q_1 \wedge ... \wedge Q_n)$.

3) Wat
