CSP with Arc consistency.

\[ \begin{array}{cccccccc}
N & Q & S & V & W & X & Y \\
\text{g} & b & r & g & b & g & \text{rb} \\
\end{array} \]

Inconsistency:
Should check: for each value in Q's domain, if not consistent with W (and Q's other neighbors), remove it!

Ex: Variables: A, B, C, E

- \( D_A: 4, 5, 6, 7, 3 \)
- \( D_B: 4, 5, 6, 8, 9, 3 \)
- \( D_C: 3, 5, 6, 7, 9, 3 \)
- \( D_E: 3, 1, 3 \)

Constraints: \( A = B, B = C, E \neq A \)

Arcs: \( (A, B), (B, A), (B, C), (C, B), (E, A), (A, E), (A, B) \)

1. \( (A, B) \)
   - 4 \( \times \) 5 \( \vee \) 8 \x 9 \x
   - \( \Rightarrow D_A: 4, 5, 6, 3 \)

2. \( (B, A) \)
   - 4 \( \times \) 5 \( \vee \) 6 \x 8 \x 9 \x
   - \( \Rightarrow D_B: 4, 5, 6, 3 \)

Now what? Need to update A again.

Reinsert \( (A, B) \).
AC-3 ( )

1. Add all arcs (u, v) + (v, u) to queue Q.

2. while (true)

3. if Q is empty

4. return success (check domains for assignment)

5. (x, y) = Dequeue()

6. Update Dx to make consistent with Dy

7. Remove (x, y)

8. if Dx is empty

9. return failure

10. If Dx shrinks away force other domains to shrink

11. For all other neighbors z of x

When to quit?

1. Success: No more arcs (Step 3).

2. Failure: Some variable has empty domain (Step 6).

Continue with example:

Q: (C, B) (E, A) (A, E) (A, B)
4) (C, B)  
   \[ 3 \times 5 \cup 6 \cup 7 \times 9 \times \]  
   \[ D_e = 3563 \]  
   \[ D_a = 24563 \ (\text{no change}) \]  

6) (E, A)  
   \[ 4 \times 5 \cup 6 \cup \]  
   \[ D_e = 3153 \ (\text{no change}) \]  
   \[ D_a = 75163 \]  

Queue is empty.  
Look at domain:  
\[ D_a = D_b = D_c = 3563 \]  
\[ D_e = 0153 \]  

Do we have a solution? No, must still search.  

\[
\begin{array}{c}
A = 5 \\
B = 6 \\
C = 1 \\
E = 1 \\
\end{array}
\]

\[ A = 5 \]  
\[ A = 6 \]  
\[ \text{SKIP.} \]

\[ D_b = 3456 \
\begin{array}{c}
A = 5 \\
B = 6 \\
C = 1 \\
E = 1 \\
\end{array}
\]

Forward check:  
\[ B = 5 \]  
\[ C = 5 \]  
\[ \vdots \]  
\[ E = 5 \]  
\[ \vdots \]  
\[ C = 5 \]
Complexity of AC3

Depends on:

Revise(X, Y)

for each value in D_X

if no value in D_Y is consistent with v,
delete v from D_X.

d: maximum domain size

C: number of constraints/arc

Revise(): O(d^2)

Max # of times Revise() called?

At most once per arc in queue, BUT

some arcs may be reinserted.

Max # times an arc may be reinserted?

If each time it is reinserted, at least one domain value
is deleted ⇒ d times

Revise called C times for each arc, then at
most d times for each reinserted arc

⇒ O(d^2(C + d)) = O(d^3).
Try AC3.

Vars: X, Y, Z

D_x = \{1, 2, 3\} \quad D_y = \{1, 2, 3\} \quad D_z = \{1, 2, 3\}

Constraints: \(X \neq Y, X \neq Z, Y \neq Z\)

Q: (x, y) (y, x) (x, z) (z, x) (y, z) (z, y)

1. (x, y)  \quad 2. (x, z)  \quad 3. (x, z)

1 \checkmark  \quad 2 \checkmark  \quad 1 \checkmark  \quad 2 \checkmark

2. (y, x)

1 \checkmark  \quad 2 \checkmark

AC3 does not detect inconsistencies!

Solution: combine AC3 w/ backtracking + forward checking

\[
\begin{pmatrix}
  x & y & z \\
  1 & 2 & 3
\end{pmatrix}
\]

\(x = 1\) \quad \(y = 2\) \quad \(z = 3\)

\(x = 1\) \quad \(y = 2\) \quad \(z = 3\)

\(D_x = \{1, 2, 3\}\)

\(D_y = \{1, 2, 3\}\)

\(D_z = \{1, 2, 3\}\)

1. Forward check: \(z = 0\)

\(x = 1\) \quad \(y = 2\) \quad \(z = 3\)

Backtrack to \(x = 2\).