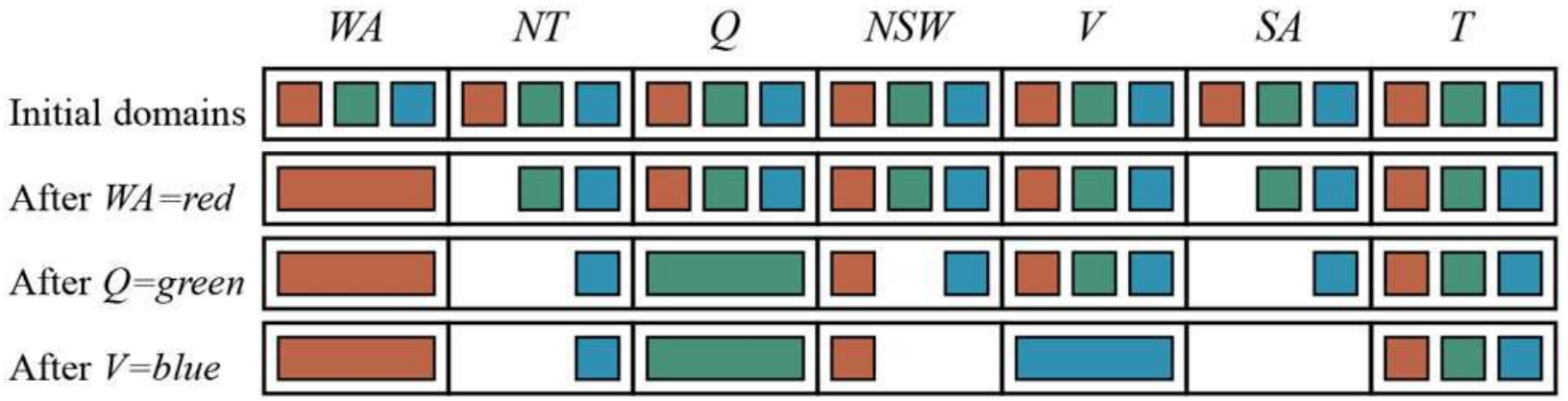
CONSTRAINT PROPAGATION: INFERENCE IN CSPS

INTERLEAVING SEARCH AND INFERENCE

- Forward checking: Whenever a variable X is assigned, the forward-checking process establishes arc consistency for it: for each unassigned variable Y that is connected to X by a constraint, delete from Y's domain any value that is inconsistent with the value chosen for X.



MAINTAINING ARC CONSISTENCY

- After a variable X_i is assigned a value, the INFERENCE procedure calls AC-3, but instead of a queue of all arcs in the CSP, we start with only the arcs (X_j, X_i) for all X_j that are unassigned variables that are neighbors of X_i.

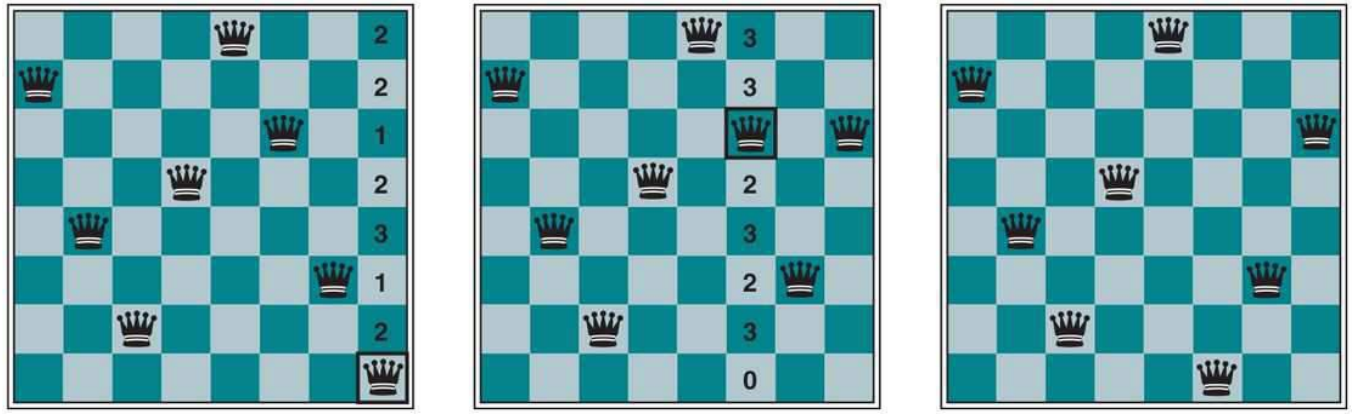
INTELLIGENT BACKTRACKING: LOOKING BACKWARD

- Fixed variable ordering Q, NSW, V, T, SA, WA, NT.
- Partial assignment: {Q = red,NSW = green,V = blue,T = red}
- Conflict set: {Q = red,NSW = green,V = blue,T = red}

- Conflict-directed backjumping

$$\text{conf}(X_i) \leftarrow \text{conf}(X_i) \cup \text{conf}(X_j) - \{X_i\}$$

LOCAL SEARCH FOR CSPs



MIN-CONFLICTS HEURISTIC

```
function MIN-CONFLICTS(csp, max_steps) returns a solution or failure
  inputs: csp, a constraint satisfaction problem
          max_steps, the number of steps allowed before giving up

  current ← an initial complete assignment for csp
  for i = 1 to max_steps do
      if current is a solution for csp then return current
      var ← a randomly chosen conflicted variable from csp.VARIABLES
      value ← the value v for var that minimizes CONFLICTS(csp, var, v, current)
      set var = value in current
  return failure
```