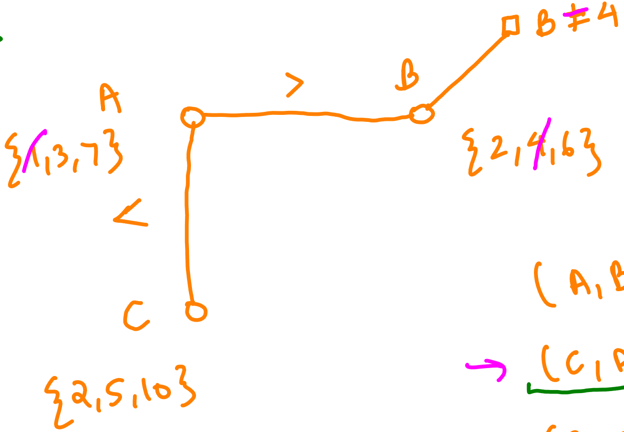
Finding a solution for a CSP

- The problem : $O(d^n)$ possible assignments.
- Reduce the number of legal values for different variables.
- Achieve node consistency

  $WA \neq Green$
- Achieve arc consistency

Binary Constraint and Arc consistency



(A, B)

(C, A)

(B, A)

(A, C)

AC-3 Algorithm

c 2c

function AC-3(csp) **returns** false if an inconsistency is found and true otherwise
  inputs: csp, a binary CSP with components (X, D, C)
  local variables: $queue$, a queue of arcs, initially all the arcs in csp

  while $queue$ is not empty **do**
    $(X_i, X_j) \leftarrow$ REMOVE-FIRST($queue$)
    if REVISE(csp, X_i, X_j) **then**
      if size of $D_i = 0$ **then return** $false$
      for each X_k **in** X_i.NEIGHBORS - $\{X_j\}$ **do**
        add (X_k, X_i) to $queue$
  return $true$

---

function REVISE(csp, X_i, X_j) **returns** true iff we revise the domain of X_i
  $revised \leftarrow false$
  for each x **in** D_i **do**
    if no value y in D_j allows (x,y) to satisfy the constraint between X_i and X_j **then**
      delete x from D_i
      $revised \leftarrow true$
  return $revised$

Binary Constraint and Arc consistency

~~(V, R)~~

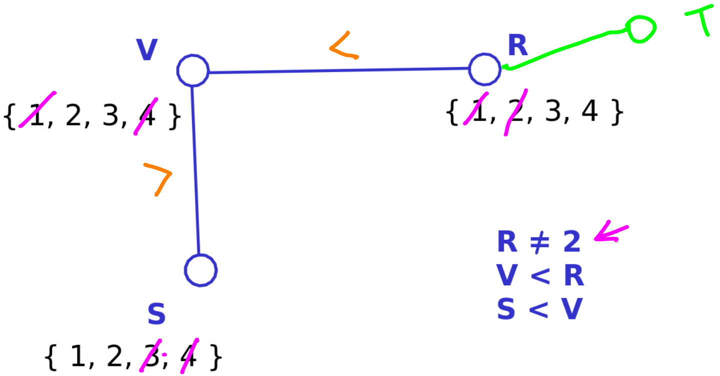
~~(R, V)~~

~~(V, S)~~

~~(S, V)~~

~~(S, V)~~

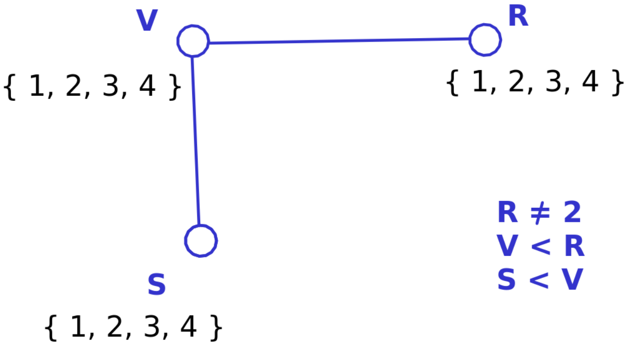
~~(R, V)~~



$R \neq 2$
$V < R$
$S < V$

(T, R)

AC-3 Algorithm Example



R ≠ 2
V < R
S < V

- The search space is reduced from $4^3 = 64$ to $2^3 = 8$.

Time complexity of AC-3 algorithm

$2c,\quad d^2$

$(x_k, x_i) \leftarrow d$

$\underbrace{2c \times d}\, \times d^2 \qquad O(cd^3)$

- Domain size atmost d and c binary constraints.

$\rightarrow (x_k, x_i)$

Time complexity of AC-3 algorithm

- Domain size atmost d and c binary constraints.
- Each arc (X_k, X_i) can be inserted into the queue at most d times.

Time complexity of AC-3 algorithm

- Domain size atmost d and c binary constraints.
- Each arc (X_k, X_i) can be inserted into the queue at most d times.
- Arc consistency can be checked in $O(d^2)$ time.

Time complexity of AC-3 algorithm

- Domain size atmost d and c binary constraints.
- Each arc (X_k, X_i) can be inserted into the queue at most d times.
- Arc consistency can be checked in $O(d^2)$ time.
- Worst case time complexity of AC-3 algorithm $O(cd^3)$.

$O(d^n)$

A stronger notion of consistency

Can we use node and arc consistency to detect that the map cannot be coloured using two colors: $\{red, blue\}$?

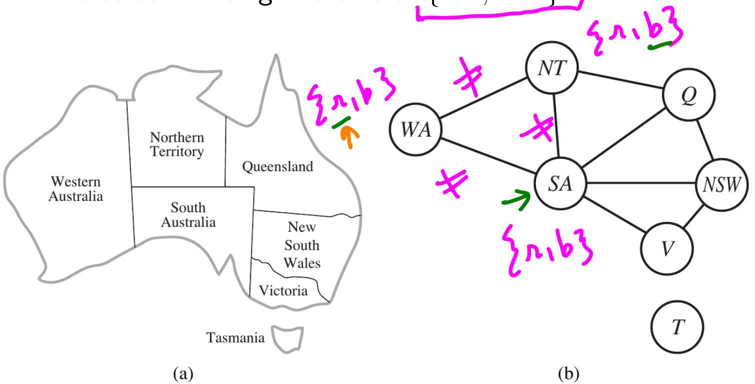


Figure 6.1 (a) The principal states and territories of Australia. Coloring this map can be viewed as a constraint satisfaction problem (CSP). The goal is to assign colors to each region so that no neighboring regions have the same color. (b) The map-coloring problem represented as a constraint graph.

Path consistency

- A two-variable set $\{X_i, X_j\}$ is path-consistent with respect to a third variable X_m if, for every *consistent* assignment $\{X_i = a, X_j = b\}$, there is a *consistent* assignment to X_m such that the constraints on $\{X_i, X_m\}$ and $\{X_m, X_j\}$ are satisfied.

Path consistency

- A two-variable set $\{X_i, X_j\}$ is path-consistent with respect to a third variable X_m if, for every *consistent* assignment $\{X_i = a, X_j = b\}$, there is a *consistent* assignment to X_m such that the constraints on $\{X_i, X_m\}$ and $\{X_m, X_j\}$ are satisfied.
- We can detect that the australian map cannot be colored using two colors.

k-consistency

- A CSP is k-consistent if, for any consistent assignment to $k-1$ variables, there is a consistent assignment for the k^{th} variable.

k-consistency

- A CSP is k-consistent if, for any consistent assignment to $k-1$ variables, there is a consistent assignment for the k^{th} variable.
- 2-consistency is same as

k-consistency

- A CSP is k-consistent if, for any consistent assignment to $k-1$ variables, there is a consistent assignment for the k^{th} variable.
- 2-consistency is same as arc consistency.

k-consistency

- A CSP is k-consistent if, for any consistent assignment to $k-1$ variables, there is a consistent assignment for the k^{th} variable.
- 2-consistency is same as arc consistency.
- 3-consistency is same as

k-consistency

- A CSP is k-consistent if, for any consistent assignment to $k-1$ variables, there is a consistent assignment for the k^{th} variable.
- 2-consistency is same as arc consistency.
- 3-consistency is same as path consistency.