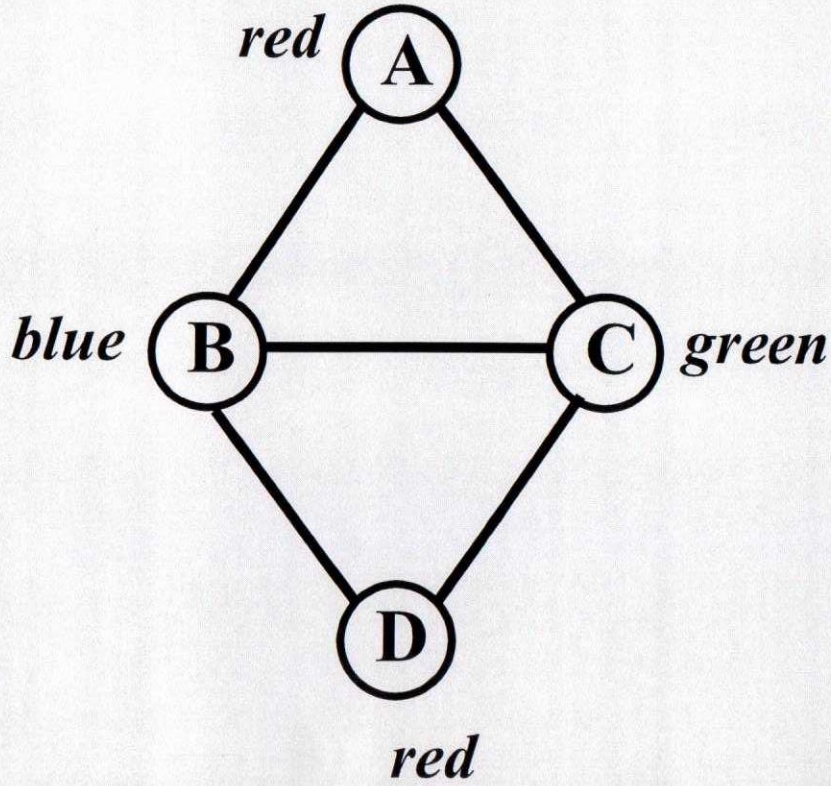
Constraint satisfaction problems (CSPs)

- Defined by:
  - A set of variables $x_1, x_2, \ldots, x_n$
  - A domain D_i for each variable x_i
  - Constraints $c_1, c_2, \ldots, c_m$
- A constraint is specified by
  - A subset (often, two) of the variables
  - All the allowable joint assignments to those variables
- Goal: find a complete, consistent assignment

Graph coloring

- Fixed number of colors; no two adjacent nodes can share a color



Cryptarithmetic puzzles

```
 T W O
 T W O +
--------
F O U R
```

E.g., setting F = 1, O = 4, R = 8, T = 7, W = 3, U = 6 gives 734+734=1468

Cryptarithmetic puzzles…

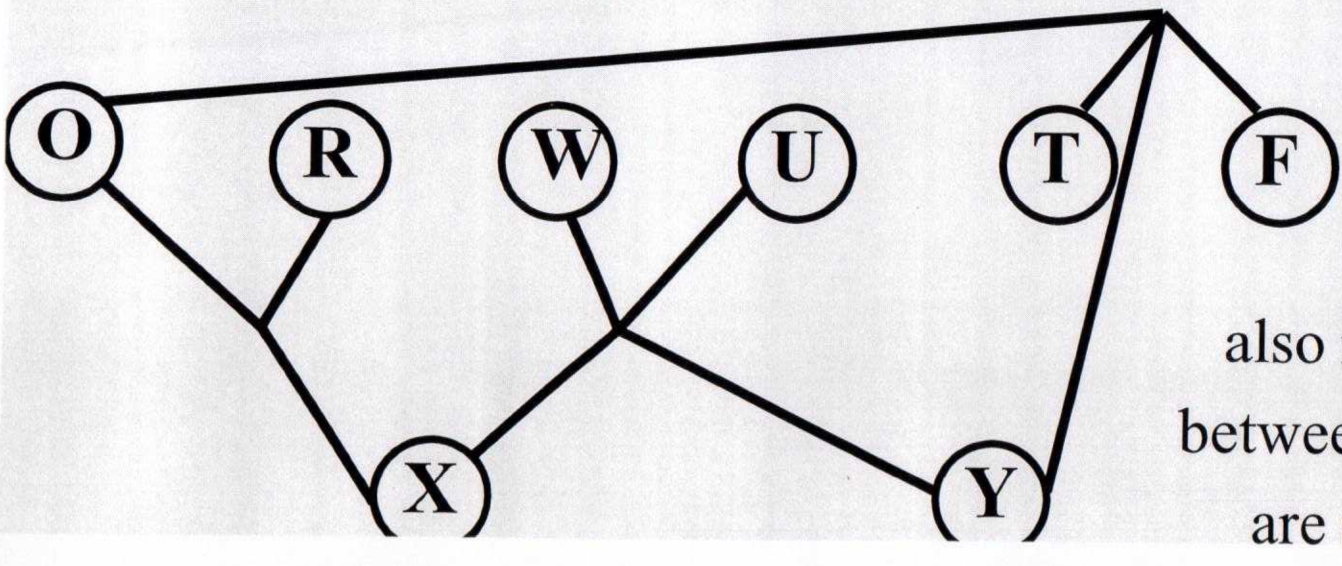
```
  T W O
  T W O +
---------
F O U R
```

Trick: introduce auxiliary variables X, Y

$O + O = 10X + R$

$W + W + X = 10Y + U$

$T + T + Y = 10F + O$



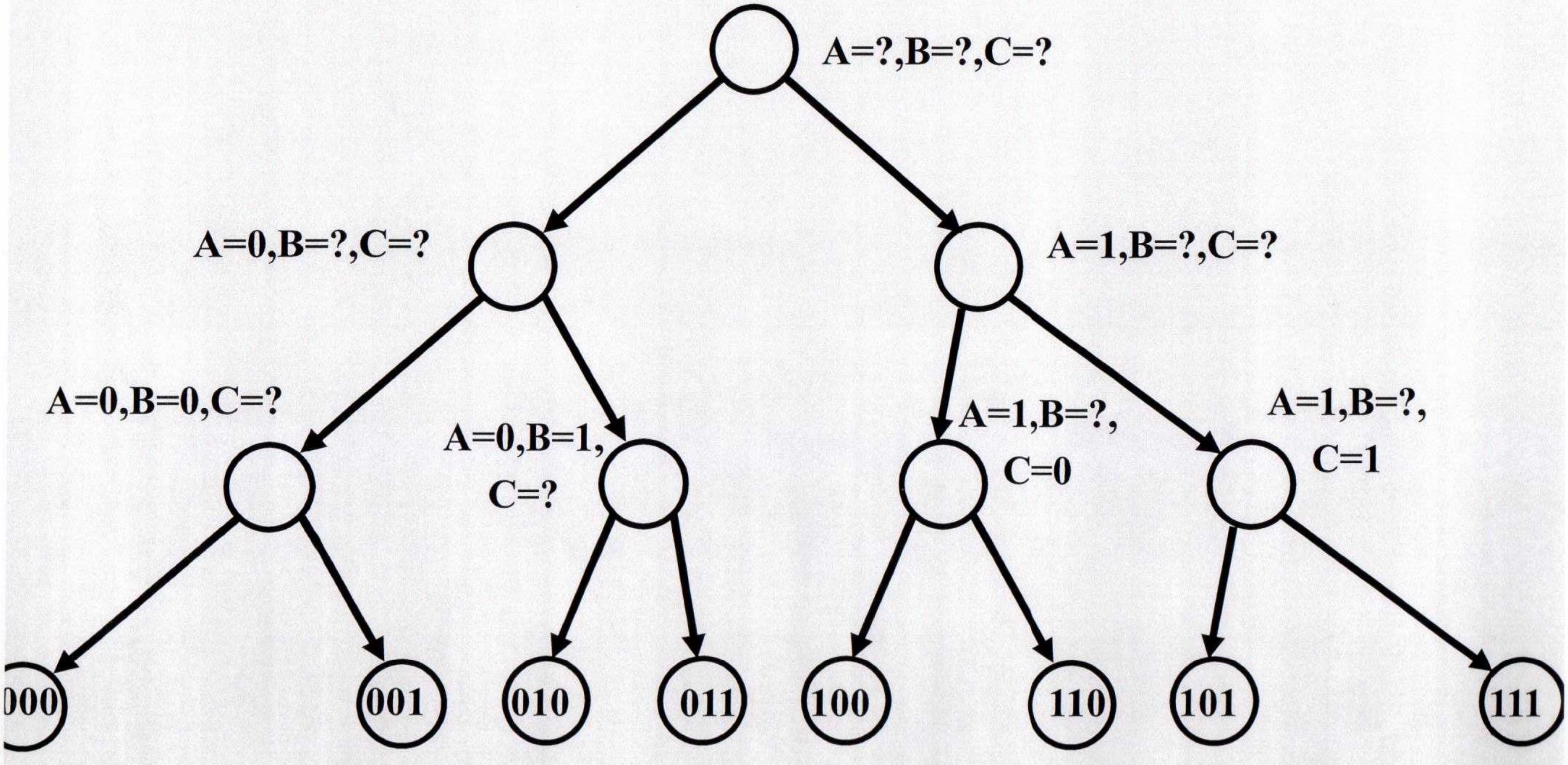
also need pairwise constraints between original variables if they are supposed to be different

Generic approaches to solving CSPs

- State: some variables assigned, others not assigned
- Naïve successors definition: any way of assigning a value to an unassigned variable results in a successor
  - Can check for consistency when expanding
  - How many leaves do we get in the worst case?
- CSPs satisfy commutativity: order in which actions applied does not matter
- Better idea: only consider assignments for a single variable at a time
  - How many leaves?

Choice of variable to branch on is still flexible!

- Do not always need to choose same variable at same level
- Each of variables A, B, C takes values in {0,1}



- Can you prove that this never increases the size of the tree?