Constraint Satisfaction Problems
Probability

Russell and Norvig: Chapter 6
CSE 240: Winter 2023
Lecture 11

Announcements

- Quiz 2 opens on Thursday after class
  - No time limit (should take ~30 minutes)
  - Deadline: 5pm on Friday.
  - Mixed multiple choice + open ended answers.
  - Open book open note
- Assignment 3 is out.

Agenda and Topics

- CSP Structure
  - Ordering
  - Structure
- Probability
- Assignment 3 QA

Ordering

Ordering: Minimum Remaining Values

- Variable Ordering: Minimum remaining values (MRV)
  - Choose the variable with the fewest legal left values in its domain
- Why min rather than max?
- Also called “most constrained variable”
  - “Fail-fast” ordering

Ordering: Least Constraining Value

- Value Ordering: Least Constraining Value
  - Given a choice of variable, choose the least constraining value
  - I.e., the one that rules out the fewest values in the remaining variables
  - Note that it may take some computation to determine this! (E.g., rerunning filtering)
- Why least rather than most?
- Combining these ordering ideas makes 1000 queens feasible

Iterative Improvement

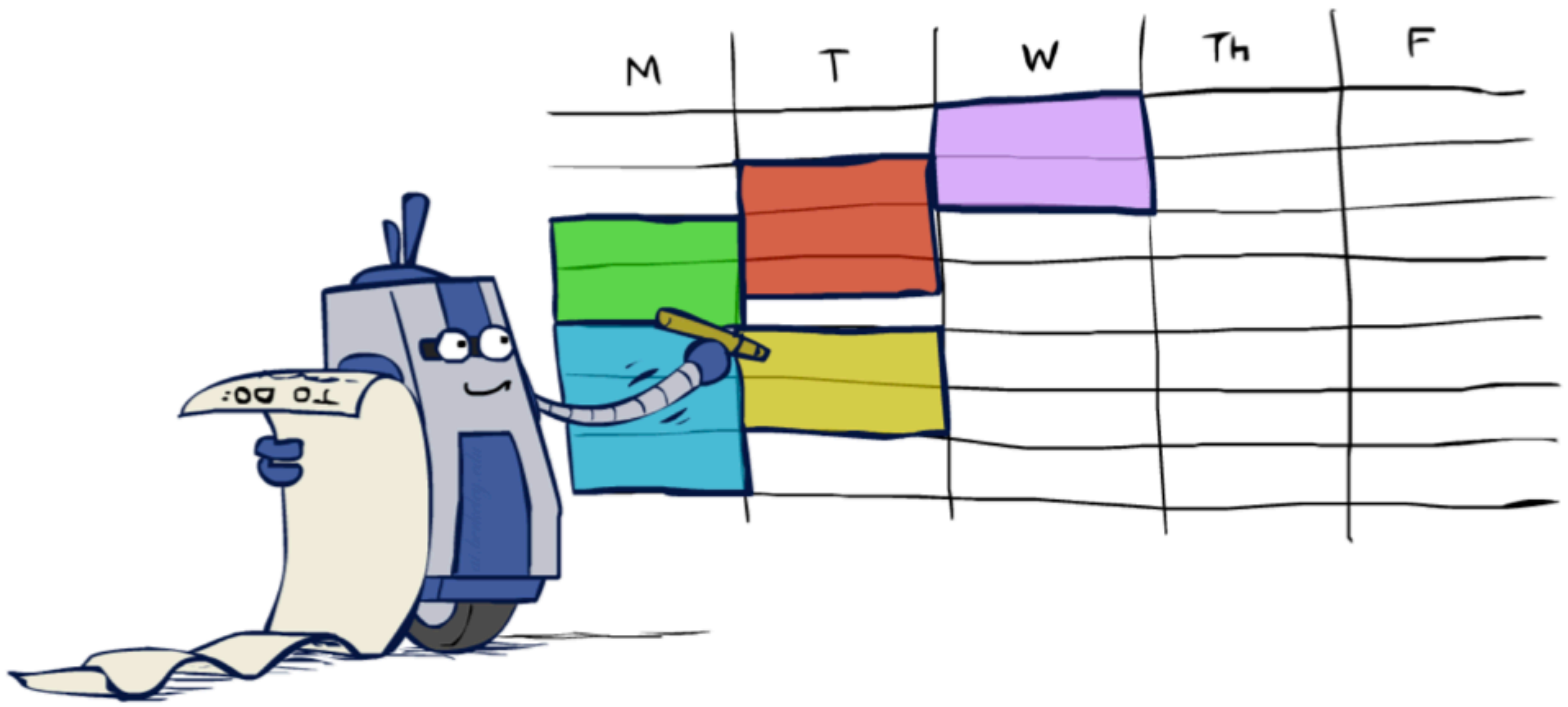
Iterative Algorithms for CSPs

- Local search methods typically work with “complete” states, i.e., all variables assigned
- To apply to CSPs:
  - Take an assignment with unsatisfied constraints
  - Operators reassign variable values

- Algorithm: While not solved,
  - Variable selection: randomly select any conflicted variable
  - Value selection: min-conflicts heuristic:
    - Choose a value that violates the fewest constraints
    - I.e., hill climb with h(x) = total number of violated constraints

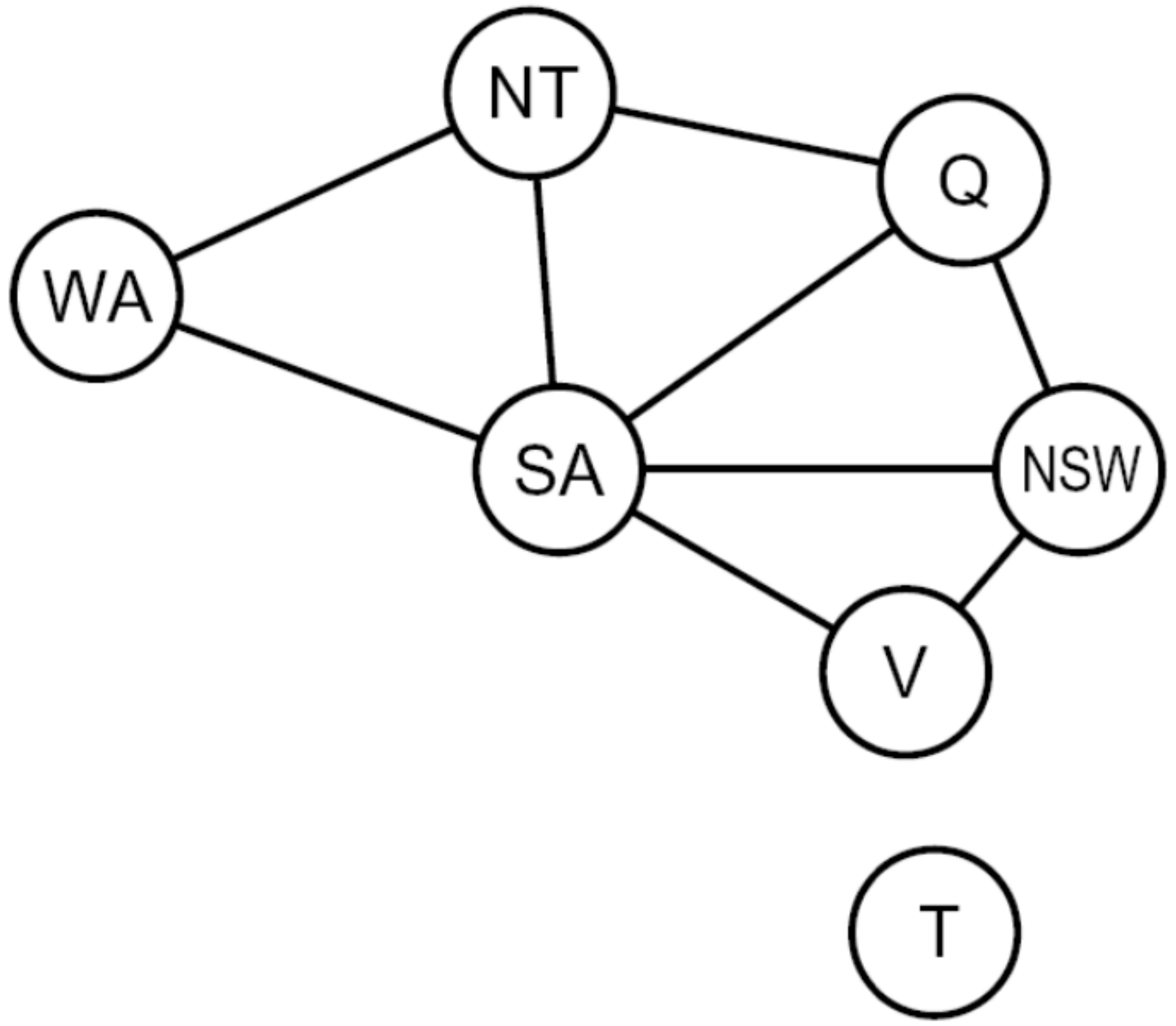
Summary: CSPs

- • CSPs are a special kind of search problem:
  - States are partial assignments
  - Goal test defined by constraints
- Basic solution: backtracking search
- Speed-ups:
  - Ordering
  - Filtering
  - Structure – turns out trees are easy!
- Iterative min-conflicts is often effective in practice



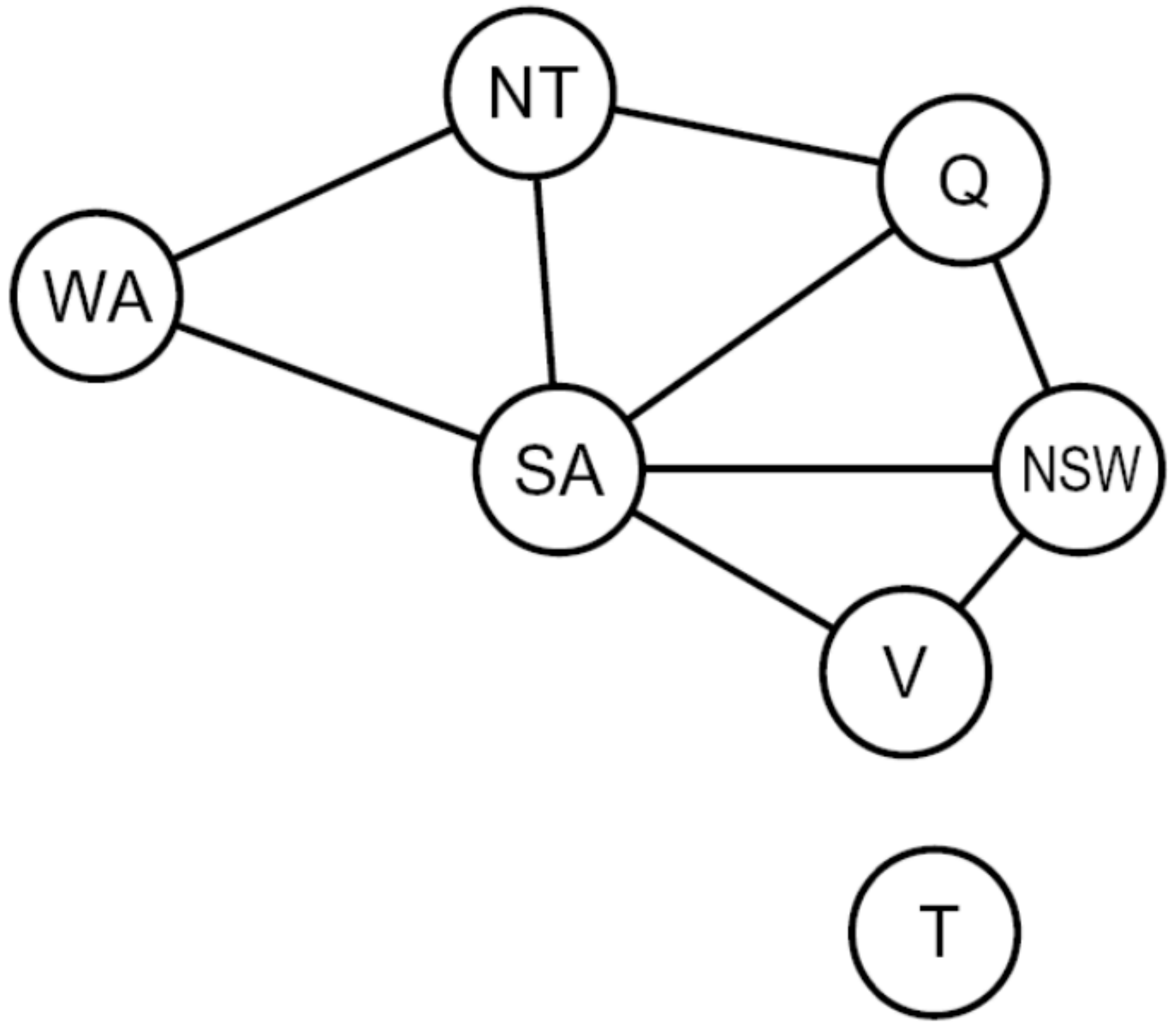
Structure

Problem structure



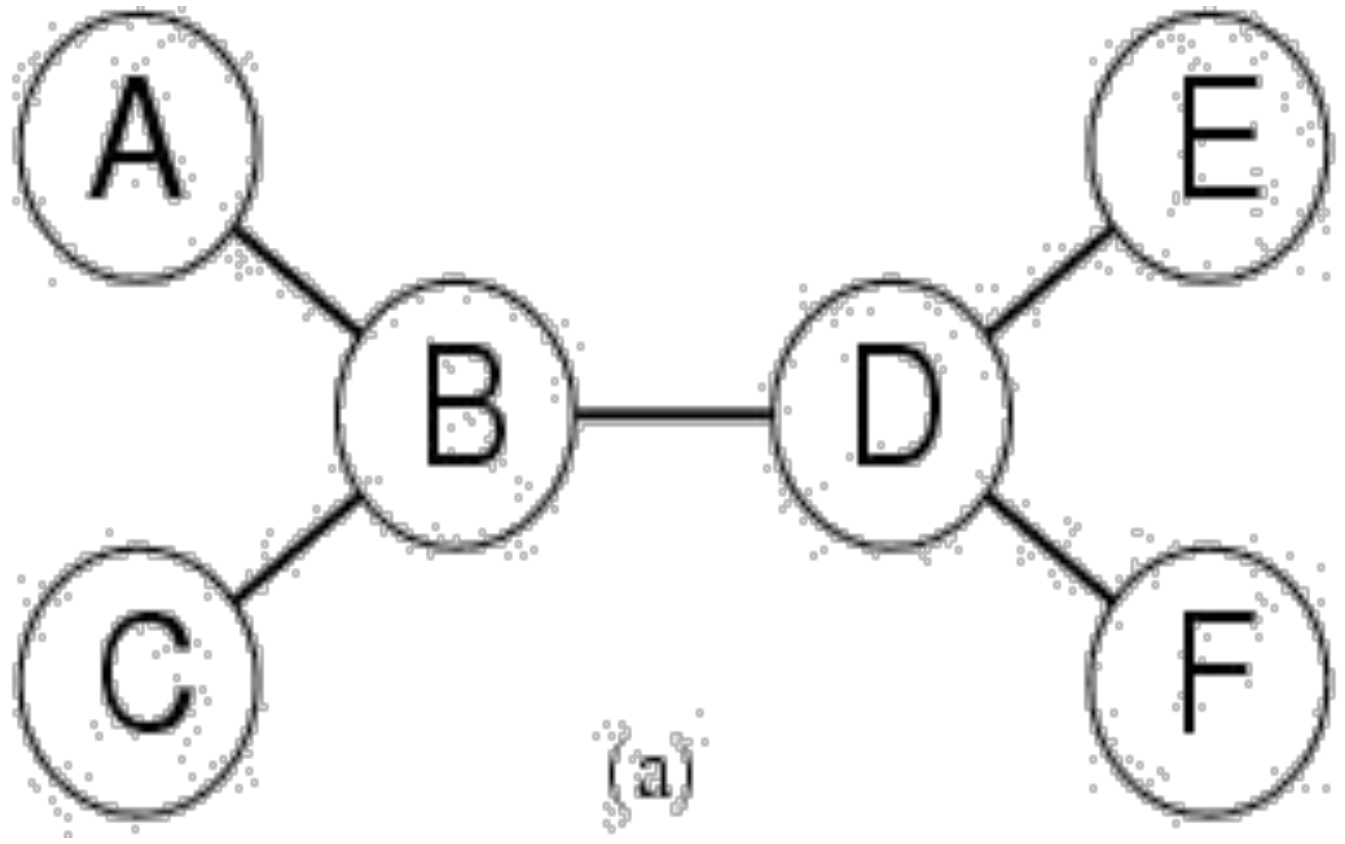
- How can the problem structure help to find a solution quickly?
- Subproblem identification is important:
  - Coloring Tasmania and mainland are independent subproblems
  - Identifiable as connected components of constrained graph.
- Improves performance

Problem structure



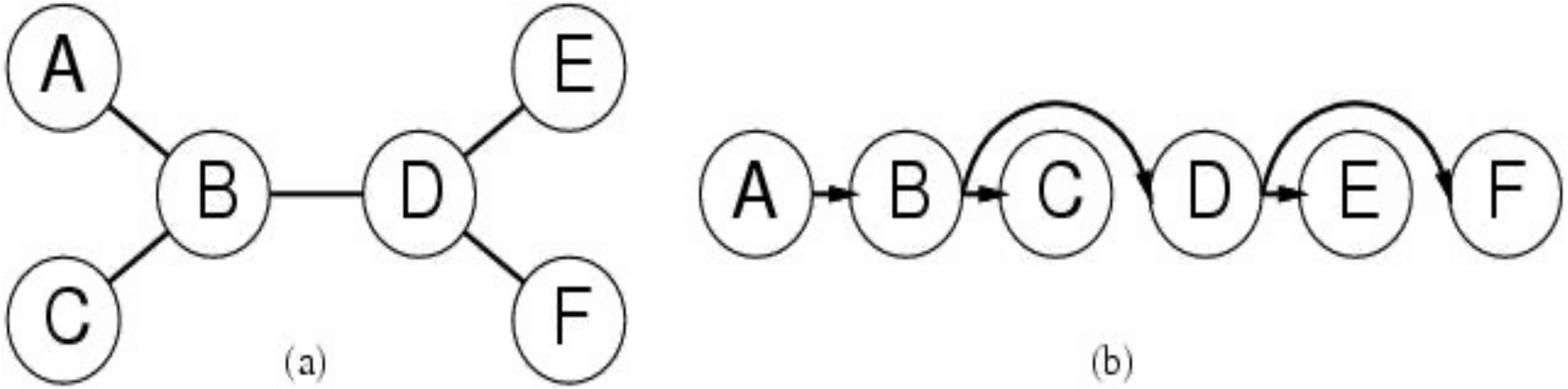
- Suppose each problem has c variables out of a total of n.
- Worst case solution cost is $O(n/c\ d^c)$, i.e. linear in n
  - Instead of $O(d^n)$, exponential in n
- E.g. n= 80, c= 20, d=2
  - 2^{80} = 4 billion years at 1 million nodes/sec.
  - $4 * 2^{20}$= 4 seconds at 1 million nodes/sec

Tree-structured CSPs



- Theorem: if the constraint graph has no loops then CSP can be solved in $O(nd^2)$ time
- Compare difference with general CSP, where worst case is $O(d^n)$

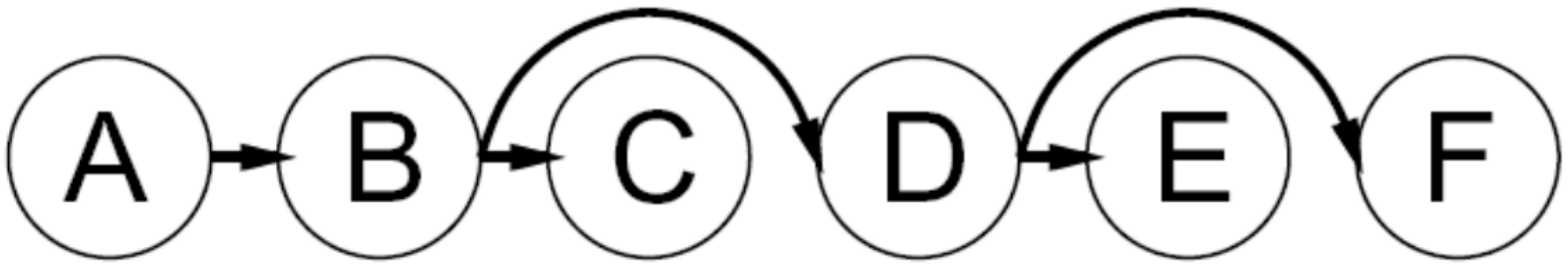
Tree-structured CSPs



- Any tree-structured CSP can be solved in time linear in the number of variables.
  - Choose a variable as root, order variables from root to leaves such that every node's parent precedes it in the ordering. (label var from X_1 to X_n)
  - For j from n down to 2, apply REMOVE-INCONSISTENT-VALUES(Parent(X_j),X_j)
  - For j from 1 to n assign X_j consistently with Parent(X_j)

Runtime: $O(nd^2)$ (why?)

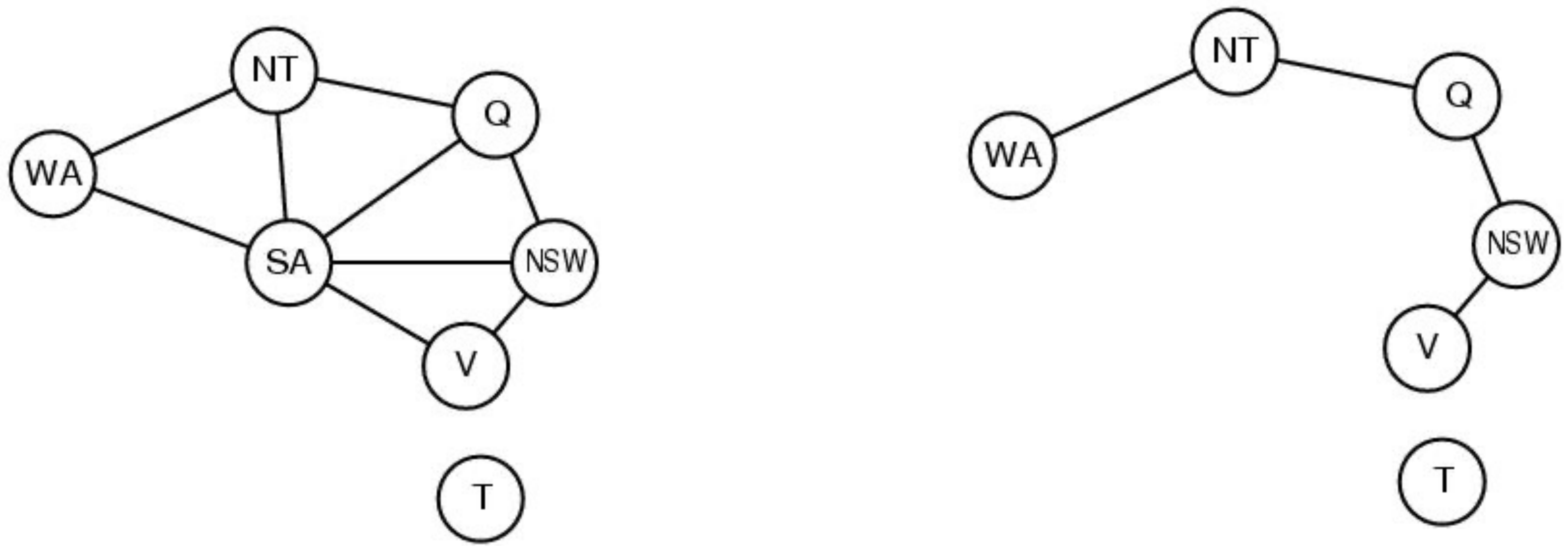
Tree-Structured CSPs



- Claim 1: After backward pass, all root-to-leaf arcs are consistent
- Proof: Each $X \rightarrow Y$ was made consistent at one point and Y's domain could not have been reduced thereafter (because Y's children were processed before Y)
- Claim 2: If root-to-leaf arcs are consistent, forward assignment will not backtrack
- Proof: Induction on position

Improving Structure

Nearly tree-structured CSPs



- Conditioning: instantiate a variable, prune its neighbors' domains
- Cutset conditioning: instantiate (in all ways) a set of variables such that the remaining constraint graph is a tree
- Cutset size c gives runtime $O(d^c(n-c)d^2)$ very fast for small c

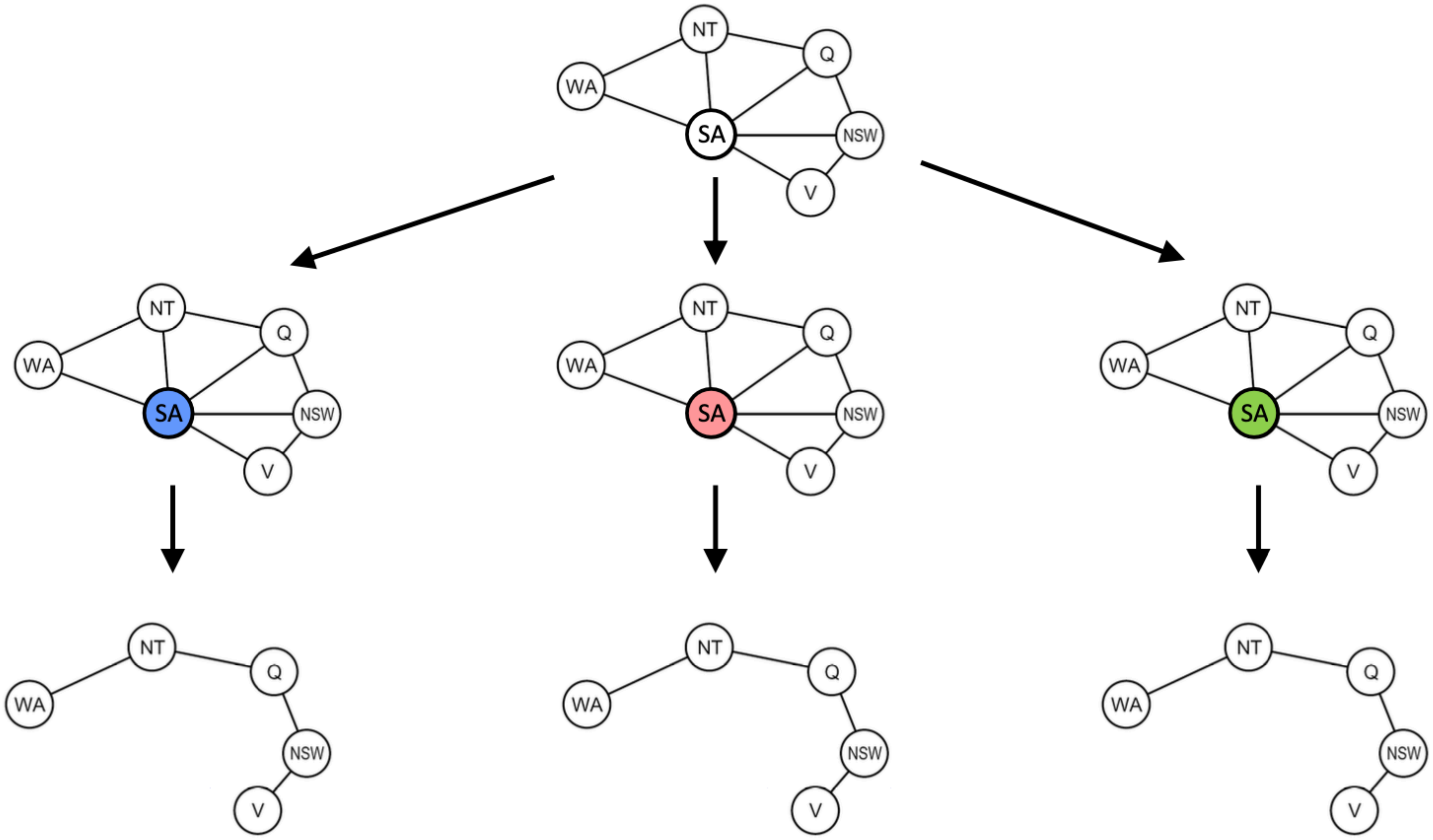
Cutest Conditioning

Choose a cutset

Instantiate the cutset (all possible ways)

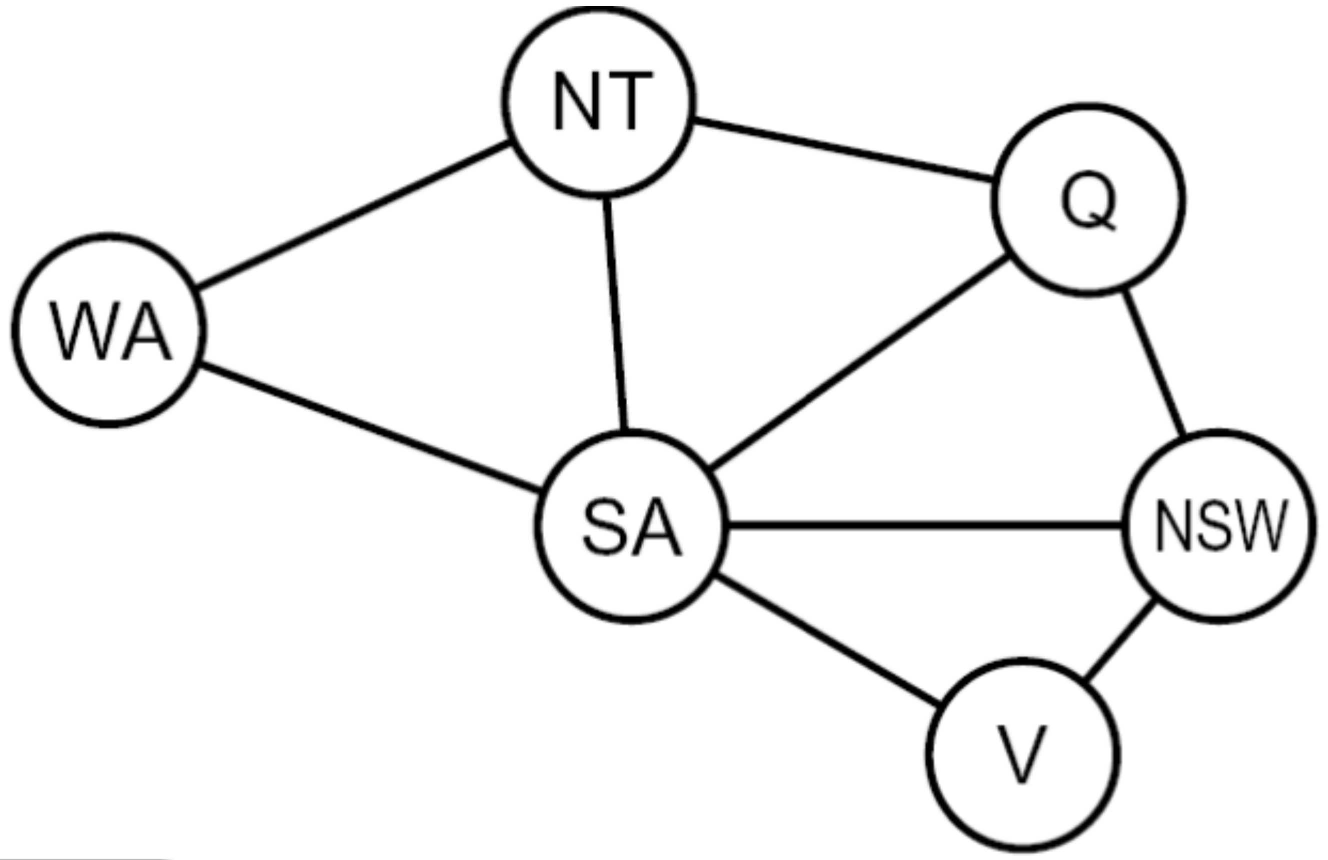
Compute residual CSP for each assignment

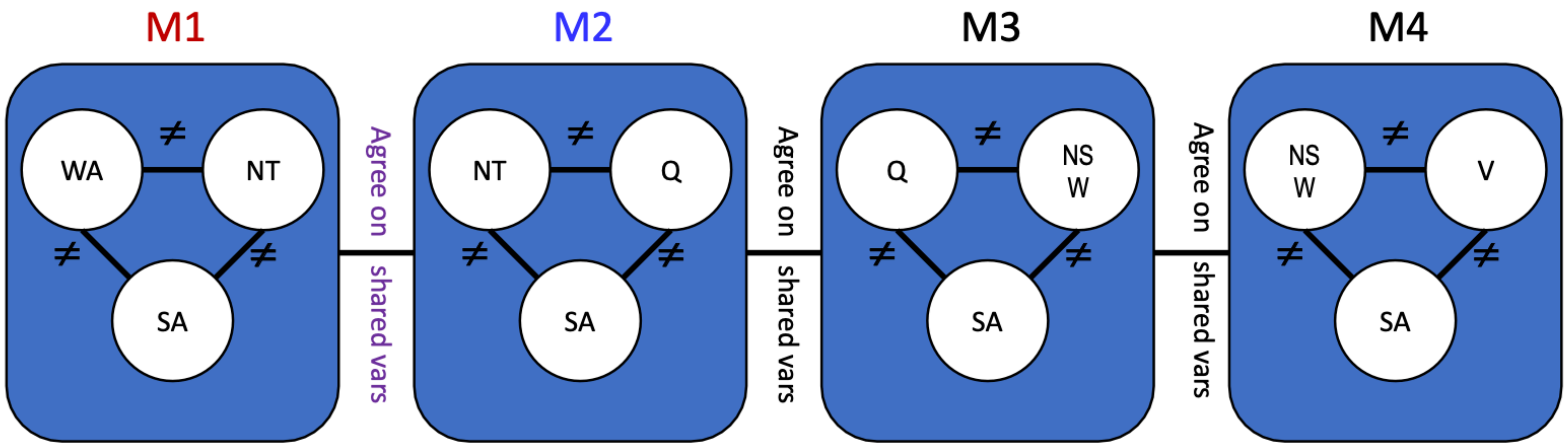
Solve the residual CSPs (tree structured)



Tree Decomposition

- Idea: create a tree-structured graph of mega-variables
- Each mega-variable encodes part of the original CSP
- Sub-problems overlap to ensure consistent solutions





{(WA=r,SA=g,NT=b),
(WA=b,SA=r,NT=g),
…}

{(NT=r,SA=g,Q=b),
(NT=b,SA=g,Q=r),
…}

Agree: (M1,M2) ∈
{((WA=g,SA=g,NT=g), (NT=g,SA=g,Q=g)), …}

CE 11: Tree CSPs

Why do we prefer to change nearly tree-structured CSPs to tree-structured CSPs?

CSE 240 Status

- We’re done with Part I Search and Planning!
- Part II: Probabilistic Reasoning
  - Diagnosis
  - Speech recognition
  - Tracking objects
  - Robot mapping
  - Genetics
  - Error correcting codes
  - lots more!

Uncertainty

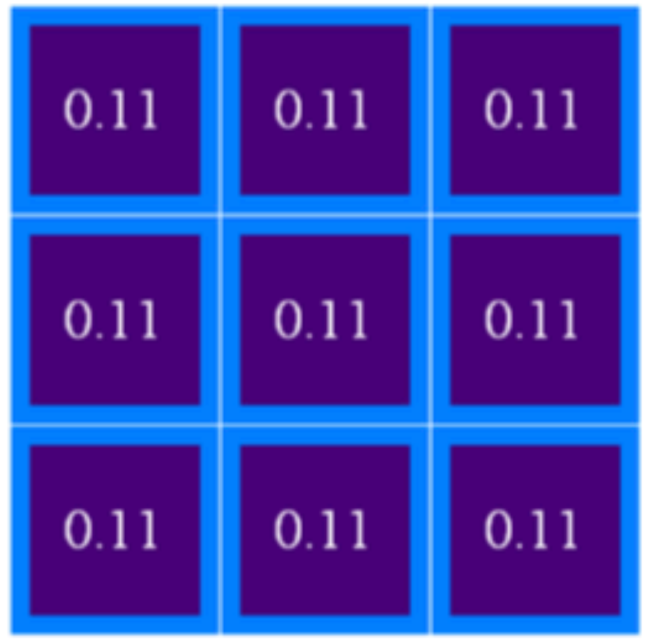
General situation:

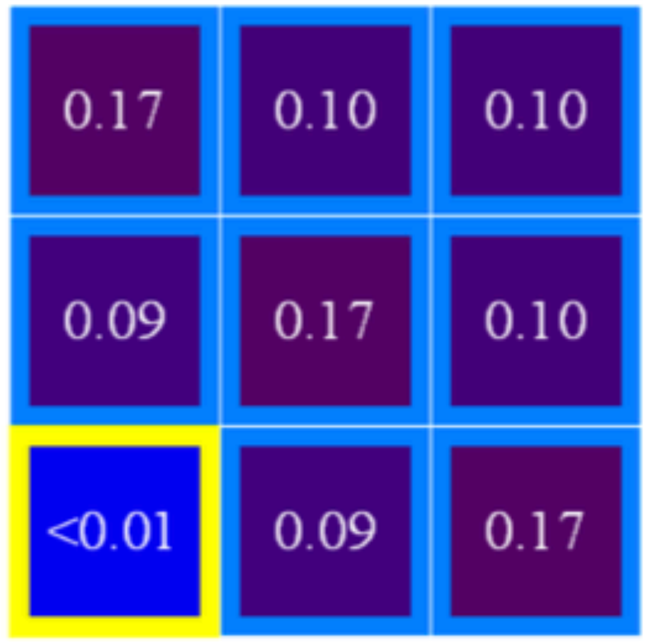
- **Observed variables (evidence)**: Agent knows certain things about the state of the world (e.g., sensor readings or symptoms)

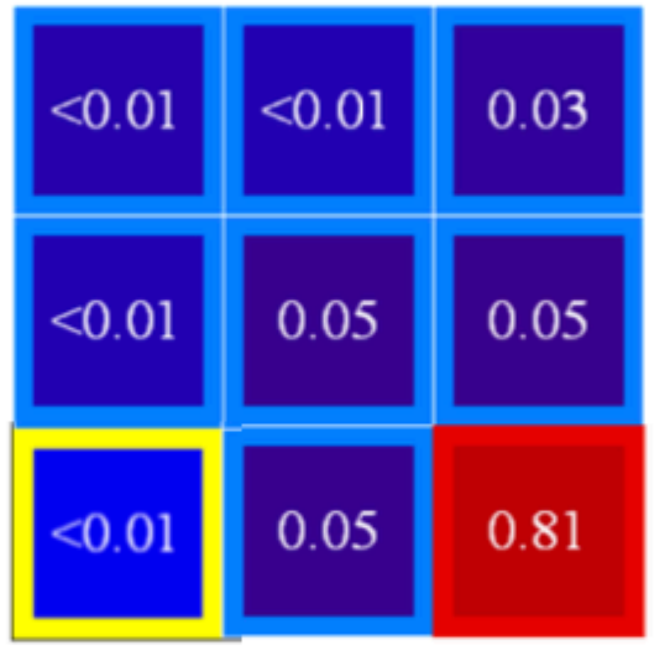
- **Unobserved variables**: Agent needs to reason about other aspects (e.g. where an object is or what disease is present)

- **Model**: Agent knows something about how the known variables relate to the unknown variables

• Probabilistic reasoning gives us a framework for managing our beliefs and knowledge







Random Variables

- A random variable is some aspect of the world about which we (may) have uncertainty
  - R = Is it raining?
  - D = How long will it take to drive to work?
  - L = Where am I?
- We denote random variables with capital letters
- Like variables in a CSP, random variables have domains
  - R in {true, false}   (sometimes write as {+r, ¬r})
  - D in $[0, \infty)$
  - L in possible locations, maybe {(0,0), (0,1), …}

Probability Distributions

- Unobserved random variables have distributions

$$P(T)$$

T	P
warm	0.5
cold	0.5

$$P(W)$$

W	P
sun	0.6
rain	0.1
fog	0.3

- A distribution is a TABLE of probability values
- A probability (lower case value) is a single number

$$P(W = rain) = 0.1 \qquad P(rain) = 0.1$$

- Must have:

$$\forall x \, P(x) \geq 0 \qquad \sum_x P(x) = 1$$

Joint Distributions

- A joint distribution over a set of random variables: $X_1, X_2, \ldots, X_n$ specifies a real number for each assignment (or outcome):

$$P(X_1 = x_1, X_2 = x_2, \ldots X_n = x_n)$$

$$P(x_1, x_2, \ldots x_n)$$

  - Size of distribution if n variables with domain sizes d?
  - Must obey:

$$P(x_1, x_2, \ldots x_n) \geq 0$$

$$\sum_{(x_1, x_2, \ldots x_n)} P(x_1, x_2, \ldots x_n) = 1$$

- For all but the smallest distributions, impractical to write out

$$P(T, W)$$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Probabilistic Models

- A probabilistic model is a joint distribution over a set of random variables
- Probabilistic models:
  - (Random) variables with domains Assignments are called outcomes
  - Joint distributions: say whether assignments (outcomes) are likely
  - Normalized: sum to 1.0
  - Ideally: only certain variables directly interact
- Constraint satisfaction probs:
  - Variables with domains
  - Constraints: state whether assignments are possible
  - Ideally: only certain variables directly interact

Distribution over T,W

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Constraint over T,W

T	W	P
hot	sun	T
hot	rain	F
cold	sun	F
cold	rain	T

Events

- An event is a set E of outcomes

$$P(E) = \sum_{(x_1 \ldots x_n) \in E} P(x_1 \ldots x_n)$$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

- From a joint distribution, we can calculate the probability of any event
  - Probability that it's hot AND sunny? 0.4
  - Probability that it's hot? 0.5
  - Probability that it's hot OR sunny? 0.7
- Typically, the events we care about are *partial assignments*, like P(T=hot)

Marginal Distributions

- Marginal distributions are sub-tables which eliminate variables
- Marginalization (summing out): Combine collapsed rows by adding

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$$P(t) = \sum_w P(t, w)$$

$P(T)$

T	P
hot	0.5
cold	0.5

$$P(w) = \sum_t P(t, w)$$

$P(W)$

W	P
sun	0.6
rain	0.4

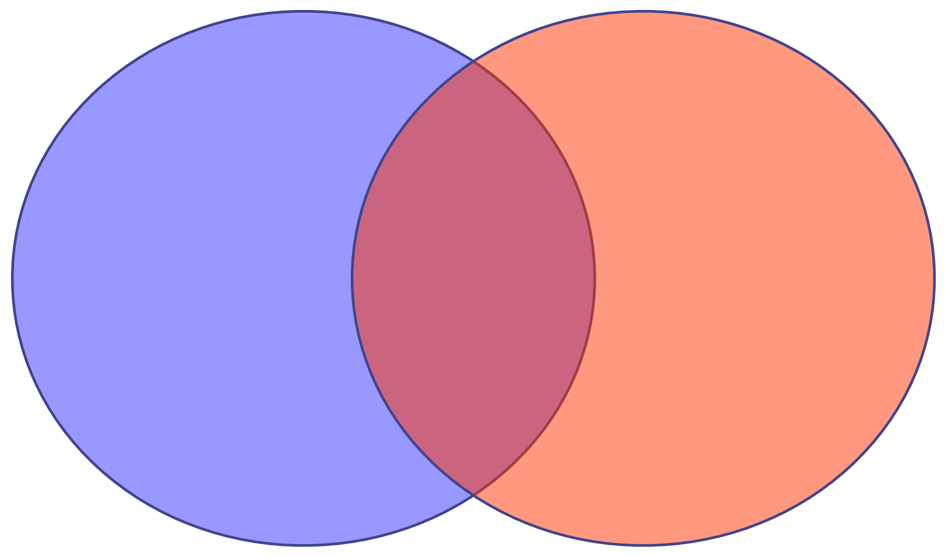
$$P(X_1 = x_1) = \sum_{x_2} P(X_1 = x_1, X_2 = x_2)$$

Conditional Probabilities

- A simple relation between joint and conditional probabilities
  - In fact, this is taken as the definition of a conditional probability

$$P(a|b) = \frac{P(a,b)}{P(b)}$$

$P(a,b)$



$P(a)$ $P(b)$

$P(T,W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$$P(W = r|T = c) = ???$$

0.3/(0.2 + 0.3) = 0.6

Conditional Distributions

- Conditional distributions are probability distributions over some variables given fixed values of others

Conditional Distributions

$$P(W|T) \begin{cases} P(W|T = hot) \\ P(W|T = cold) \end{cases}$$

$P(W|T = hot)$

W	P
sun	0.8
rain	0.2

$P(W|T = cold)$

W	P
sun	0.4
rain	0.6

Joint Distribution

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Question: Conditional Probabilities

- P(hot | sun)?
- P(cold | sun)?
- P(rain | hot) ?

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Summary and Next Time

- Constraint Satisfaction Problems
  - Ordering
  - Structure
- Probability Review

- This week
  - Naives Bayes