

SNS COLLEGE OF ENGINEERING



Kurumbapalayam (Po), Coimbatore - 641 107

An Autonomous Institution

Accredited by NBA - AICTE and Accredited by NAAC - UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSE NAME : 19CS507- Artificial Intelligence

III YEAR /V SEMESTER

Unit 1- INTRODUCTION

Topic 7 : Constraint Satisfaction Problems



Constraint satisfaction problems (CSPs)

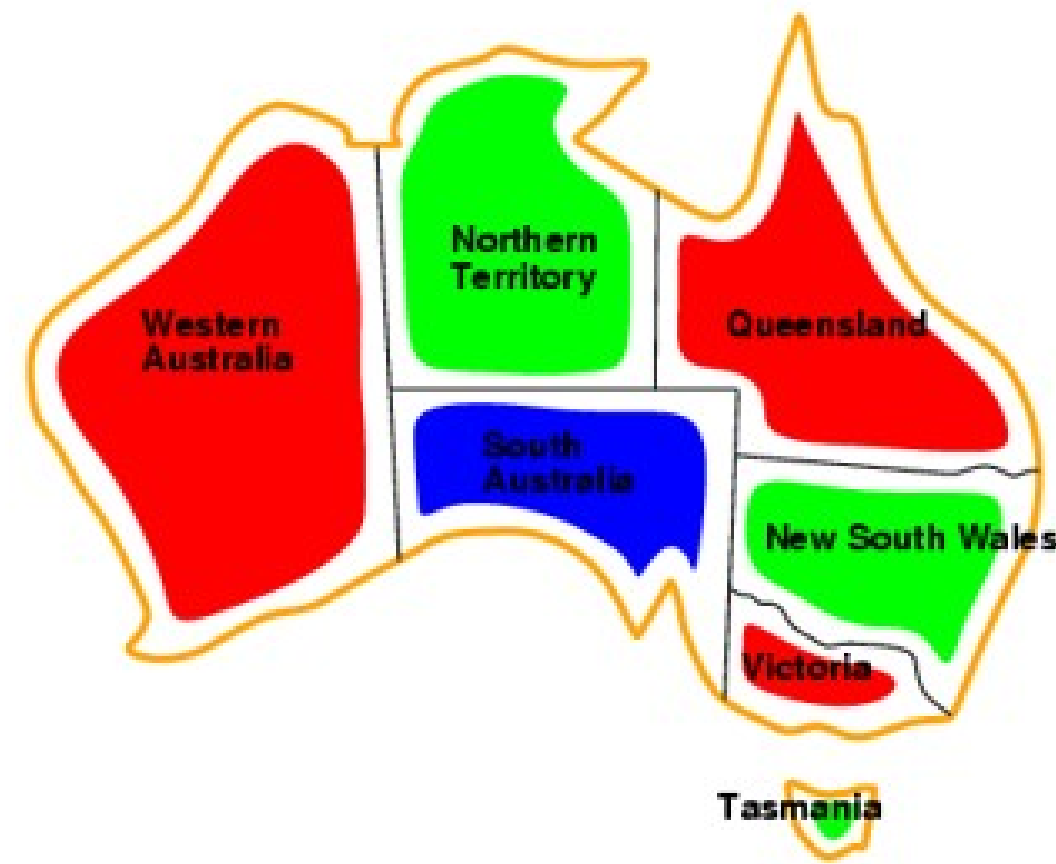
- Standard search problem: **state** is a "black box" – any data structure that supports successor function and goal test
- CSP:
 - **state** is defined by **variables** X_i with **values** from **domain** D_i
 - **goal test** is a set of **constraints** specifying allowable combinations of values for subsets of variables
- Simple example of a **formal representation language**
- Allows useful **general-purpose** algorithms with more power than standard search algorithms

Example: Map-Coloring



- **Variables** WA, NT, Q, NSW, V, SA, T
- **Domains** $D_i = \{\text{red, green, blue}\}$
- **Constraints:** adjacent regions must have different colors
- e.g., $WA \neq NT$, or (WA, NT) in $\{(\text{red, green}), (\text{red, blue}), (\text{green, red}), (\text{green, blue}), (\text{blue, red}), (\text{blue, green})\}$

Example: Map-Coloring

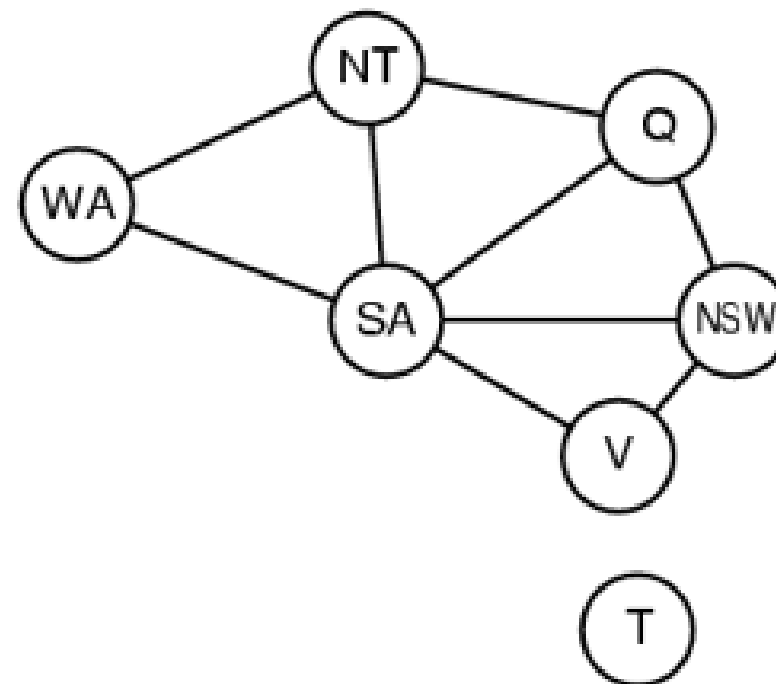


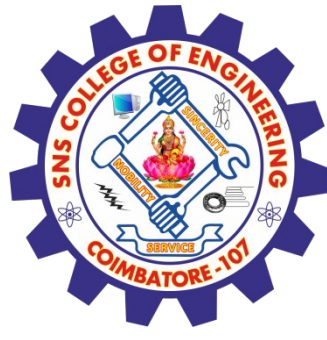
- Solutions are **complete** and **consistent** assignments
- e.g., WA = red, NT = green, Q = red, NSW = green, V = red, SA = blue, T = green



Constraint graph

- **Binary CSP:** each constraint relates two variables
- **Constraint graph:** nodes are variables, arcs are constraints





Varieties of CSPs

- Discrete variables
 - finite domains:
 - n variables, domain size $d \rightarrow O(d^n)$ complete assignments
 - e.g., Boolean CSPs, incl. Boolean satisfiability (NP-complete)
 - infinite domains:
 - integers, strings, etc.
 - e.g., job scheduling, variables are start/end days for each job
 - need a constraint language, e.g., $StartJob_1 + 5 \leq StartJob_3$
- Continuous variables
 - e.g., start/end times for Hubble Space Telescope observations
 - linear constraints solvable in polynomial time by LP



Varieties of constraints

- **Unary** constraints involve a single variable,
 - e.g., $SA \neq \text{green}$
- **Binary** constraints involve pairs of variables,
 - e.g., $SA \neq WA$
- **Higher-order** constraints involve 3 or more variables,
 - e.g., cryptarithmic column constraints



Real-world CSPs

- Assignment problems
 - e.g., who teaches what class
- Timetabling problems
 - e.g., which class is offered when and where?
- Transportation scheduling
- Factory scheduling
- Notice that many real-world problems involve real-valued variables



Standard search formulation (incremental)

Let's start with the straightforward approach, then fix it

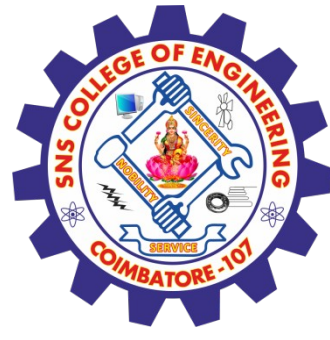
States are defined by the values assigned so far

- **Initial state:** the empty assignment $\{ \}$
- **Successor function:** assign a value to an unassigned variable that does not conflict with current assignment
 - fail if no legal assignments
- **Goal test:** the current assignment is complete

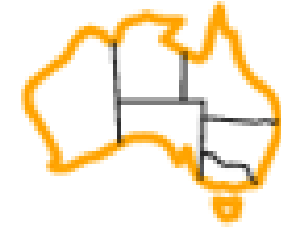


Backtracking search

- Variable assignments are **commutative**, i.e.,
[WA = red then NT = green] same as [NT = green then WA = red]
- \Rightarrow Only need to consider assignments to a single variable at each node
- Depth-first search for CSPs with single-variable assignments is called **backtracking** search
- Can solve n -queens for $n \approx 25$

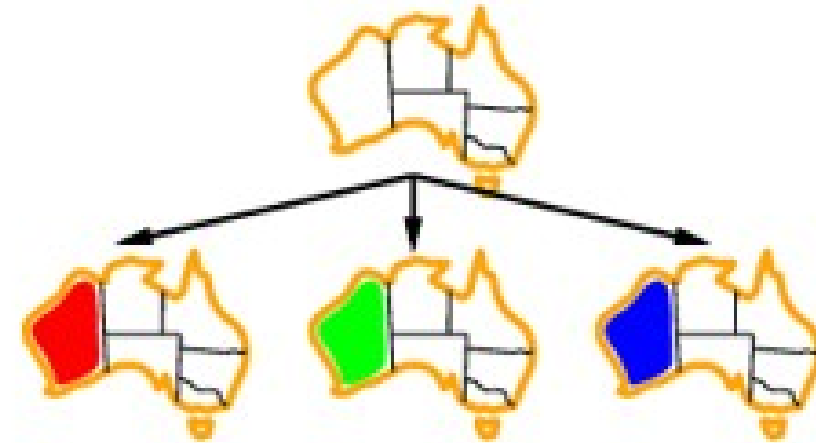


Backtracking example



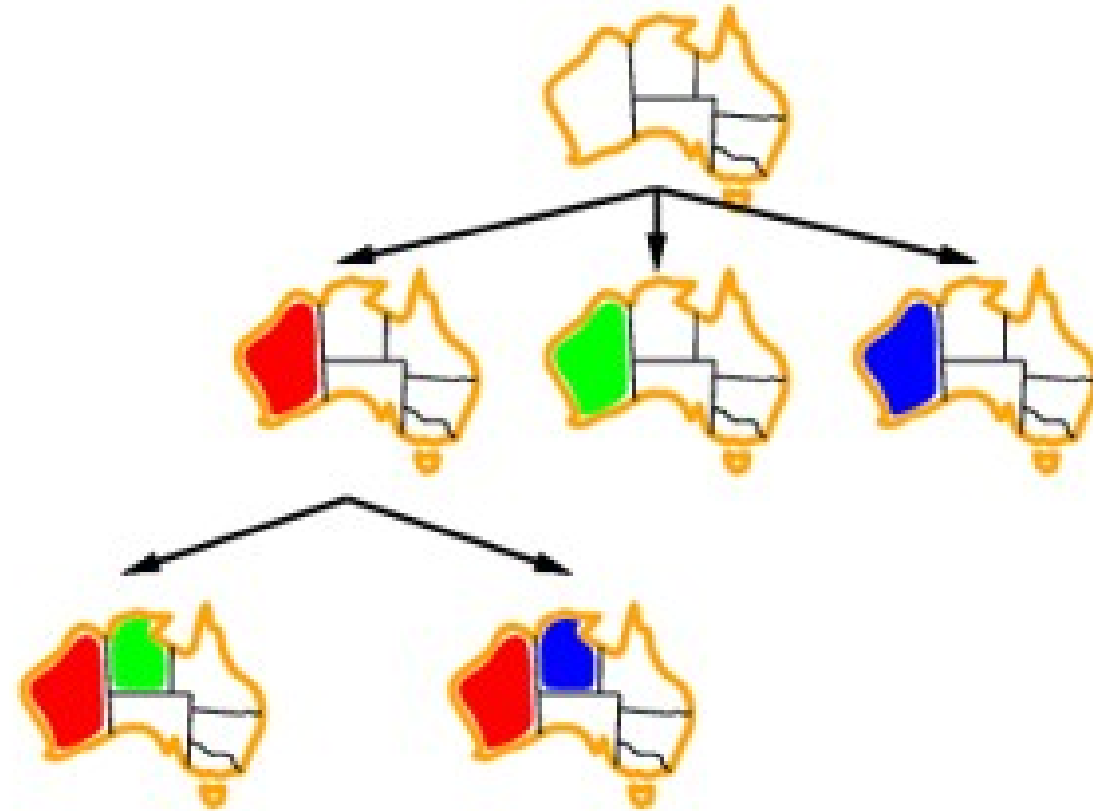


Backtracking example

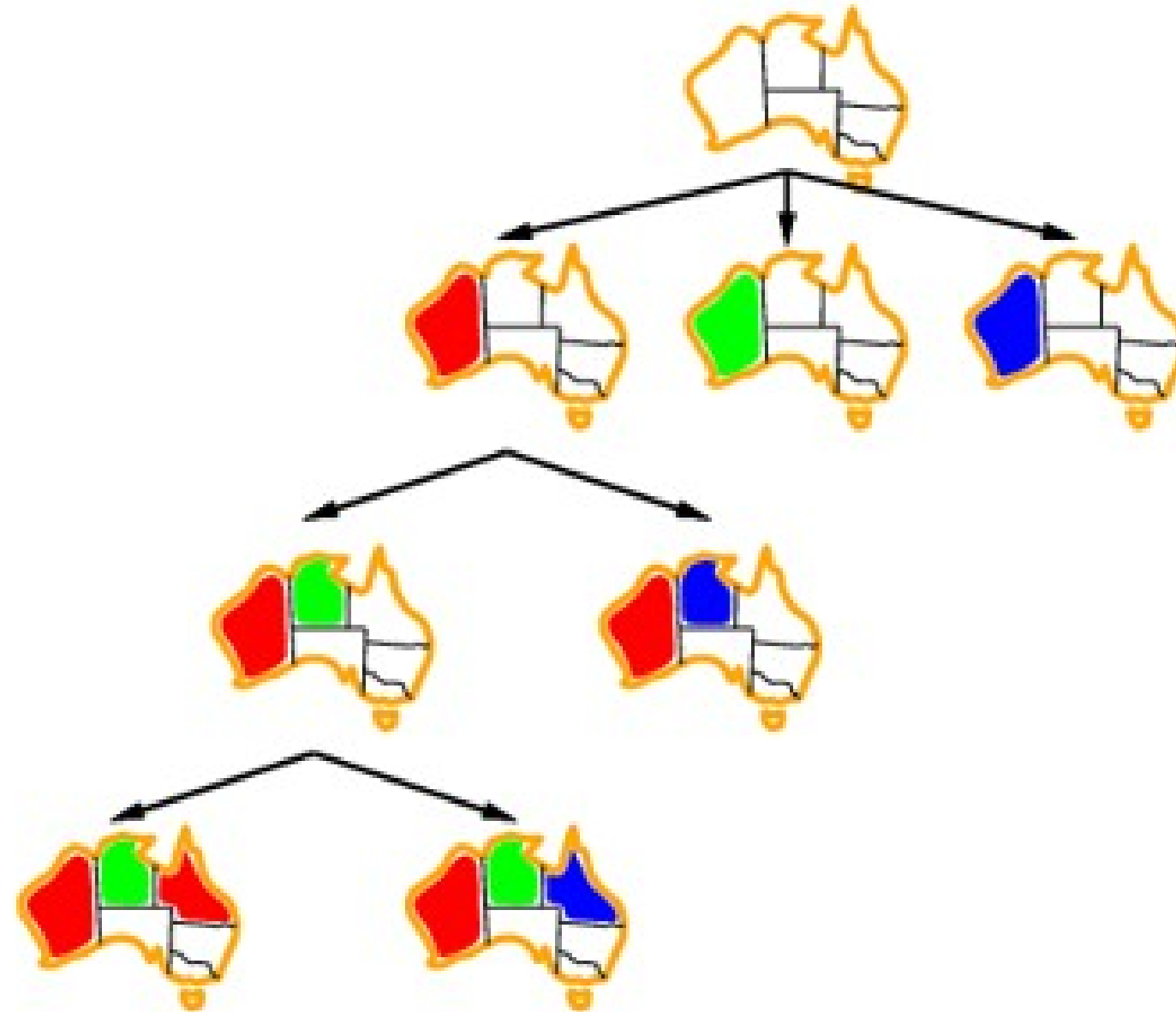


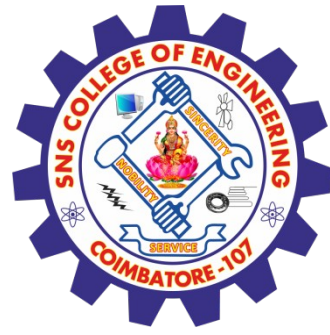


Backtracking example



Backtracking example





REFERENCES



1. S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach", Prentice Hall, Third Edition, 2009.

THANK YOU