A constraint satisfaction problem (CSP) consists of
- a set of variables \( \{x_1, x_2, \ldots, x_i\} \);
- a finite set of domain \( D \) (possible values) for each variable;
- a set of constraints \( C \) restricting the values that the variables can simultaneously take

CSP is to assign values to variable so that all constraints are satisfied
CSP/COP – Approaches

- **Constraint programming techniques**
  - Systematic search on search trees + constraint based techniques
  - We’ll concentrate on CP in this module

- **Artificial intelligence search algorithms**
  - “move” within search space to promising solutions
  - Content of “Planning & Search” G52PAS
Constraint Programming

- Constraint programming techniques solve CSPs by using the combination of
  - Systematic search
  and
  - Constraint satisfaction techniques
Constraint Programming

- Systematic search
  - Generate and Test
  - Search tree
  - Backtracking

G53CLP – Constraint Logic Programming
Constraint Programming

- Constraint satisfaction techniques
  - Constraint propagation
    - Consistency check
  - Search strategies
    - Look back
    - Forward checking
    - Search orders
    - B & B (for constraint optimisation problems)
Systematic Search

- **Search space**
  - All possible states (solutions) which a search could arrive at

- Systematically search through a search space of all possible assignment of values to variables
  - It is complete
    - Either a solution is guaranteed
    - Or no solution exist
Systematic Search – G&T

- Generate-and-test (GT)
  - enumerate all possible combinations of values for variables one by one and see if they satisfy all constraints
Systematic Search

- Generate-and-test (GT)

REPEAT
- Select the next variable
- Assign a value to the variable
- If current assignment lead to a failure (dead-end: no values are consistent with previous values)
  - Backtrack (replace the value assigned for previous variable with a new value)

UNTIL a (no) complete solution is found
Systematic Search – Search Tree

- Search Tree represents the state of search
  - Node: partial solution
  - Branch: possible assignment of values to variables
  - Labelling: assign one value to a variable (taking one branch)
  - Dead-end: no further values can be assigned to the variable

We can now find the solution by searching on the search tree.
Systematic Search – Search Tree
Systematic Search – Search Tree

\[ \ldots nd \ldots \]

\[ nd^{*}(n-1)d \ldots \]

\[ nd^{*}(n-1)d^{*}(n-2)d \ldots \]

\[ n!d^n \]
Systematic Search – Search Tree

- How is the size of CSP measured
  - Number of variables
  - Size of domains for the variables
  - Number of constraints

- Number of leaves in the search tree $n! \times d^n$
Systematic Search – Search Tree

- Properties of CSP’s search tree

  - The depth of the tree is fixed
    - Solution will always be at the $n^{th}$ level
    - The graph coloring example

  - The size of domains
    - Branching factor – average number of branches
Systematic Search – Search Tree

- Properties of CSP’s search tree
  - Sub-trees are similar
    - This is useful for learning during the search
    - Same failure could be avoided at similar sub-trees
Systematic Search – Search Tree

Properties of CSP’s search tree

- The size of the search space for a problem is finite
  - Total number of leaves is fixed for a problem
  - However the internal nodes are different
  - Try the graph coloring problem search tree with different orderings of variables labelled
Systematic Search – Search Tree

- Depth first search
Systematic Search – Backtracking

- Backtracking
Systematic Search – Backtracking

- Revising past labels
  - Label one variable at a time
  - If current value is incompatible
    - Take an alternative value
  - If all values are tried
    - Un-assign the last variable

- Until
  - All variable labelled
  - No more label to backtrack to
Systematic search with backtracking
Tsang. (1996) Foundations of Constraint Satisfaction
Systematic Search – Backtracking

- Pure backtracking
  - is very inefficient
  - complexity is exponential
  - may explore branches which likely lead to infeasible solutions (dead-ends)
Systematic Search – Backtracking

- Pure backtracking

- Thrashing: repeat the same failed assignment
- Redundant: conflict values of variables not remembered
- Detection of conflict at later stage: after exploring large number of branches
Constraint Satisfaction Techniques

- Aim at
  - Avoid as much as possible backtrackings
  - Speed up the search

Idea search: backtrack free
Constraint Satisfaction Techniques

- Constraint propagation
  - Consistency enforcing
    - Arc consistency, path consistency

- Search strategies
  - Back and forward checking
  - Variable and Value ordering
  - Branch & bound (B&B)
    - Constraint optimisation techniques
Summary

- Solving CSP using CP
  - Systematic search
    - Search tree
    - Backtracking
  - Techniques
    - ...

G53CLP – Constraint Logic Programming

Dr R. Qu