Title: Search orders in CSPs

Context: Finite domains, binary constraints

Material: Foundations of Constraint Satisfaction, Edward Tsang
Chapter 6

Foundations of Constraint Satisfaction

CSCE421/821, Fall 2001

www.cse.unl.edu/~choueiry/F01-421-821/

Berthe Y. Choueiry (Shu-we-ri)
Ferguson Hall, Room 104
choueiry@cse.unl.edu, Tel: (402)472-5444

Motivation for ordering heuristics

- In BT, less of undoing-labels needs to be done under some orderings than others

- In lookahead:
  - failure could be detected earlier under some orderings than others
  - larger portions of the solution space can be pruned off under some orderings than others

- When searching for one solution, value ordering may speed up search as branches that have a better chance to reach a solution are explored first

Heuristics

Variable ordering: Fail first principal (FFP)

- Least domain (LD)
- Minimal width ordering (MWO)
- Minimal bandwidth ordering (MBO)
- Maximal cardinality ordering (MCO)

Value ordering:

- Min-conflict heuristic: orders values according to the conflicts in which they are involved with the future variables
- Cruciality [Keng and Yu'89]
- Promise [Geelen'94], etc.

$\rightarrow$ static, dynamic?

Fail first principal (FFP)

Aims at recognizing dead-ends as soon as possible so that search effort can be saved.

Several FFP measures:

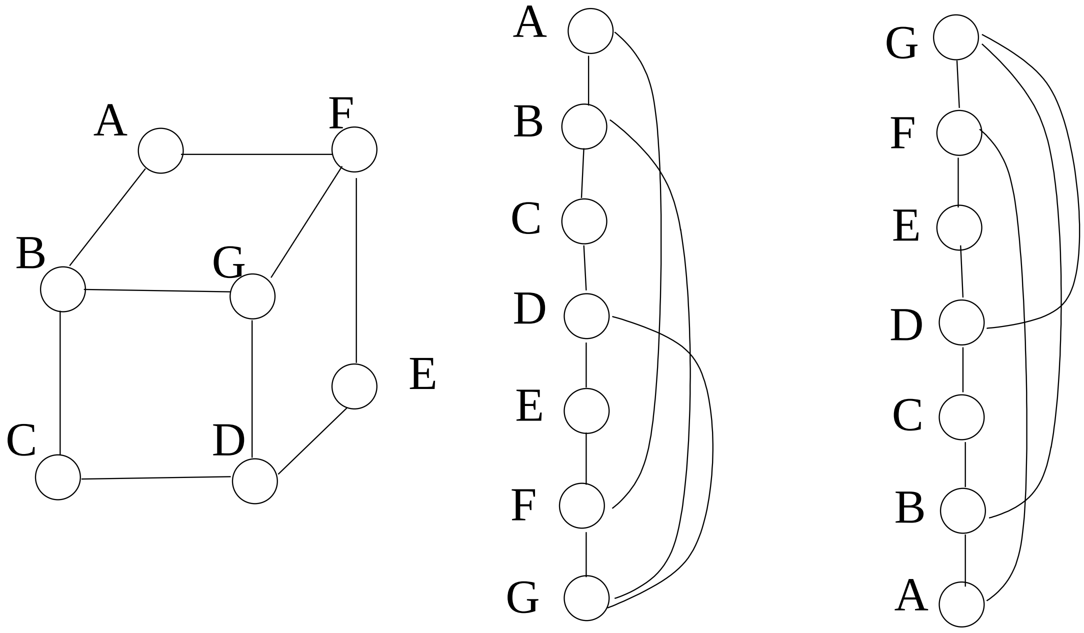
- smallest domain size first (least domain, LD)
- smallest of ratio of domain size to degree of node
- etc.

Simple, cheap and effective
Suitable for both static and dynamic ordering

Width of graph: A graph-theoretic criterion

- Constraint graph
- Ordering of the nodes *how many possible orderings?*
- Width of an ordering
- Width of the graph (independent of the ordering)

Minimal width ordering



Reduces the chance of backtracking: variables that have more unassigned values depending on them are labeled first. Variables at the front of the ordering are in general more constrained..

Finding minimum width ordering: $\mathcal{O}(n^2)$.

Freuder'82

Procedure Width of a graph G:
Remove from the graph all nodes not connected to any others
Set $k \leftarrow 0$
Do while there are nodes left in the graph:
    Set $k \leftarrow (k+1)$
    Do while there are nodes not connected to more than k others:
        Remove such nodes from the graph, along
        with any edges connected to them.
Return k
end

The minimal width ordering of the nodes is obtained by taking the nodes in the reverse order than they were removed.

Minimal bandwidth ordering

- Localizes/confines backtracking:
  the smaller the bandwidth, the sooner one could backtrack to relevant decisions
- Finding minimum bandwidth ordering is NP-hard :—(
- Is there an ordering of a *given* bandwidth k?
  $\rightarrow \mathcal{O}(n^{k+1})$ (polynomial)

Note:
(Zabih'90) finds a correlation between bandwidth and the possibility to decompose the CSP (conjunctive decomposition)

Maximum Cardinality Ordering

An approximation of min. width ordering, $\mathcal{O}(n)$

Choose a node arbitrarily. Among the remaining nodes, choose the one that is connected to the maximum number of already chosen nodes, break ties arbitrarily, repeat...
Reverse the final order.

Ordering heurstics: how, when?

How:

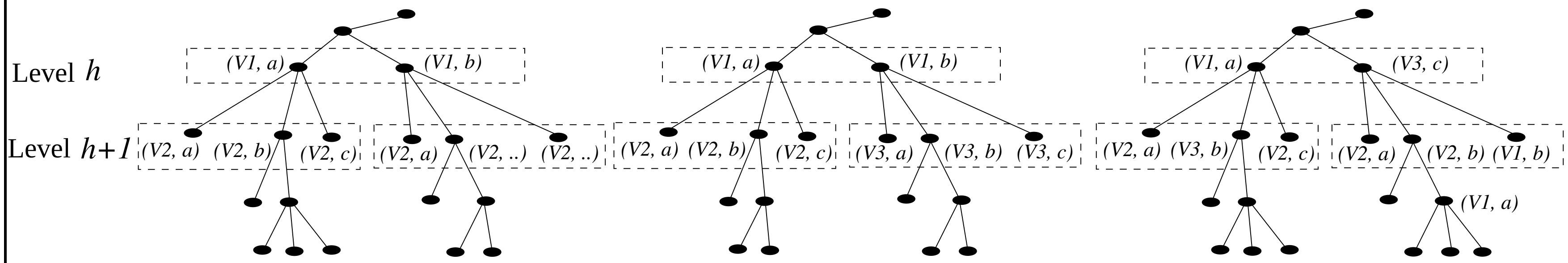
- Static variable, value ordering
- Dynamic variable (static value)
- Dynamic variable, dynamic value (dynamic vvp)

When:

- Finding one solution
- Finding all solutions

Ordering heurstics: careful in search

Choueiry & Beckwith, 2001



Static variable, static value

Dynamic variable, static value

Dynamic variable-value pair