CP : Modelling

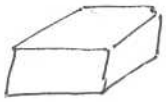
↓

Solving → Filtering/Pruning.

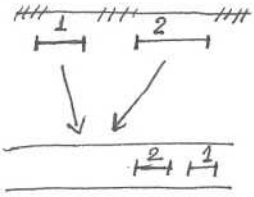
↓ → Branching.

Solutions

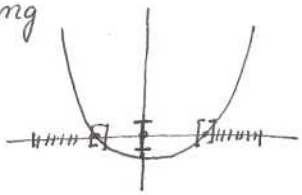
less noise in discrete solver than continuous solver.



constraints
variables.
domains.



Filtering



Filter a continuous domain

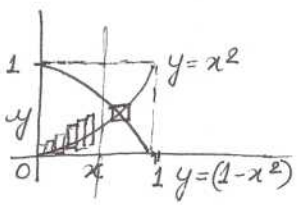
- split.
- converge.

Pruning (clipped part of Domains).

Filtering - Continuous domains
Pruning - Discrete domains
→ acheived through local consistency techniques. / propagation.

We have constraints $\{c_1, c_2 \ldots c_n\}$ where we check for consistency (local).

eg:



$c_1 : x = [0,1]$, $y = [0,1]$

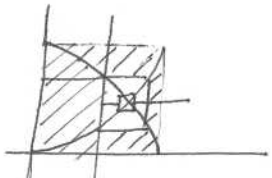
$c_2 : x = [0,1]$, $y = [0,1]$

$C = \{c_1 : y = x^2, \; c_2 : y = 1 - x^2\}$

$X = \{x, y\}$, $D = [0,1] \times [0,1]$

Branching :

split into y dimension. Not all solutions are consistent. So we filter out the inconsistent data.



Sometimes solvers generate instead of one solution, two solutions which are duplicates.

Global Consistency : ∈!

global constraints eg: all diff. Drawback: the set of global constraints have to fit into a pattern of any one.

Solving constraints → NP hard problem. (using heuristics the time is reduced)