L07 Faster Constraint Satisfaction

William Mak

January 05 2015

1 n – queens problem

1. placing as many queens on a nxn board
2. Problem too big to solve exaclty
3. Let's define a neighbourhood
   work with complete assignments
   neighbourhood: possible assignments nearby
   all configs where only 1 queen is at a different location
4. initially you can be random
   random row but seperate columns

1.1 algorithm – Local search AKA(Hill climbing)

1. Either exhastively (if possible)
2. or randomly (when not)
3. Choose a config that is better (less broken constraints)
4. update current config
5. Repeat.

n queen
operations → move queens along column

1.2 problems wiht Local search

1. can't find better
2. still have unsatisfied constraints
3. this is possible because our initial setup is one that cannot result in the best solution
4. the solution? run k times and keep the best solution

1.3 Simulated Annealing

1. look around neighbourhood
   if better solution found, take it
   if not accept with probability
   $p \alpha e^{\Delta T/T}$
2. ΔT – always $(-)$ for worse solutions
   cost_current - cost_proposed
   if worse makes cost increase
   or cost_proposed-cost_current
   if cost decreases for worst solution
3. T is a parameter called "Temperature"
   starts large – decreases over time

1.4 Tabu Search

Always move towards best solution found in neighbourhood
keep list of recently visited config. Keep them out of neighbourhood