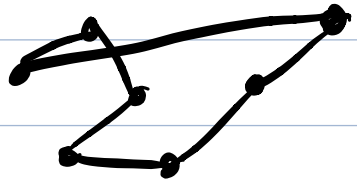
Monday, April 26 Lecture #38/42

→ Homework 6 Questions?
→ No final exam

Topic #19- Greedy Randomized Adaptive Search Procedure (GRASP)

Single-Solution MH

Idea: (1) Build a greedy solution, but not being as fully greedy as poss., so you have choices that you can decide between.



(2) From this greedy solution, perform Hill Climbing (single tweak, steepest ascent)
(3) Repeat from the start

Back when we did H-C w/ R.R.

starting from a fully greedy sol and then H-C
⇒ pretty bad

Only real question: how do we create these greedy-ish solutions.

Regular Greedy: Build up a solution bit-by-bit, picking the best new component to add at each point in time.

GRASP: At each step, compile some of the best next components, and then randomly pick one of them.

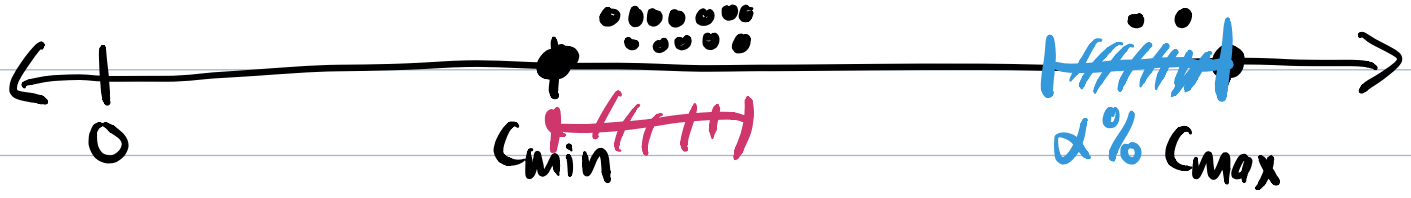
↳ Restricted Candidate List (RCL)

Two options for picking the RCL:

Option 1: Decide ahead of time on a percentage p, and just put the top p% of options in the

RCL. Good values depend on the problem, 10% - 30% is pretty reasonable.

Option 2: Consider all possible next components and the value they would add. Let c_{min} and c_{max} to be the smallest and largest.



Form the RCL by adding all poss. in the top α% of this range

This is all components with a value $\geq c_{max} - \alpha \cdot (c_{max} - c_{min})$

Option 2 is usually better.

What should α be?

(a) A fixed value, maybe 10% - 30%.

(b) Each iteration, randomly pick α

from some range, for example [0%, 30%].

(c) Make α an <u>adaptive parameter</u>. It will continuously adjust itself depending on the quality of solutions it finds.

One way to do this:

- Pick a finite set of α values $\{\alpha_1, \alpha_2, \ldots, \alpha_m\} = \{0\%, 1\%, 2\%, \ldots, 30\%\}$

- Each will have a probability p_i of being chosen with each restart. Initial probabilities: $p_i = \frac{1}{m}$.

- Keep track of B = best score ever
  A_i = average score of all solutions found using α_i as the α value

- Every iteration, recompute the p_i

as follows. Define $q_i = \frac{A_i}{B}$

(higher when avg. score using x_i is better, always < 1)

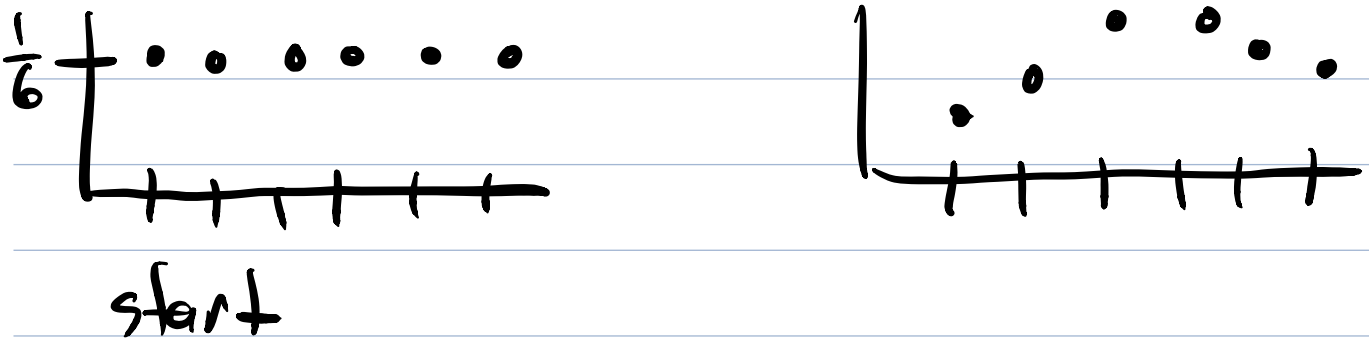
Then define $p_i = \frac{q_i}{q_1 + q_2 + \cdots + q_m}$.

$q_1 = 0.5 \quad q_2 = 0.9 \quad q_3 = 0.2$

$p_1 = \frac{0.5}{1.6} \quad p_2 = \frac{0.9}{1.6} \quad p_3 = \frac{0.2}{1.6}$

$p_1 + p_2 + p_3 = 1$ ✓

So this will automatically tune the probability distribution for the x's.



Once the RCL is formed, how do

we pick which component to use next?

Standard GRASP: pick from the RCL with uniform probability.

There are many interesting ways that bias some components over others.