Monday, Apr 3, 2023
Lecture #30
MSSC 6000

Announcements

* HW 4 due tonight, 11:59pm
* Fri, Apr. 7 - no class
  Mon, Apr. 10 - no lecture (home work day)
  no OH

Topic 12 - Simulated Annealing

Process:

Pick an initial temperature T. (How?)

```
x = random solution
best = x
Repeat:                                   (How long?)
    For a while:                          (How long?)
        s = tweak(x)                      (maximizing)
        Δ = score(s) - score(x)
        if Δ > 0:
            x = s
            if score(x) > score(best):
                best = x
        else:  #score worse
            r = random # in [0, 1]
            if r < e^(Δ/T):        0.1
                x = s  # take the sol. even though worse
    adjust the temp according to the
        cooling schedule
return best
```

Picking the initial temperature:

First decide on p_0: the initial probability with which you want a worsening move to be accepted.

What is a good value for p_0?
Different schools of thought
Depends on problem / search space / tweak etc

lots of little hills / fewer huge hills

$p_0 = 0.9$: the safest bet, if you have plenty of time, very random at the start
$p_0 = 0.5$: works pretty will in most cases
$p_0 = 0.2$: sometimes good, especially in slow cases, or landscapes with many short hills

How can we set an initial temp To that

gives us our desired p_0?

$$p = e^{\Delta/T}$$

We will approx. the average value of Δ for a tweak that gets worse, then use that to solve for T.

```
trials = []
while len(trials) < 1000:        (or whatever)
    x = random solution
    s = tweak(x)
    if score(s) < score(x):      (worsening)
        trials.append(score(s) - score(x))
avg = sum(trials) / 1000
```

$$p = e^{\Delta/T} \Rightarrow \ln(p) = \frac{\Delta}{T} \Rightarrow T = \frac{\Delta}{\ln(p)}$$

To pick T_0: $\frac{avg}{\ln(p_0)}$ ← our goal for p_0

Far from exact

(1) we used an average

(2) the average worsening in the actual process of running S.A. could be different than the average worsening from a random solution

Another way to pick T_0 to give a desired p_0:

* put a random temp. in your code
* run it, printing out at each cooling time, what % of worse solutions you accepted
* If it's not what you want, quit, change T_0, re-run

How long do we run at a fixed temp before cooling more?

Many possibilities:

* N tweak attempts in total

* K worsenings rejected
  or
  L worsenings accepted,
  whichever comes first

(Ex: 1000 accepted or 75,000 rejected)

How long do we cool the system before stopping?

* Run out of time (recommend printing each new best solution as you find it)
* No worsening solution has been accepted in some # of iterations (hill-climbing)
* Pre-set end temperature
  $T_f = 0.001 \cdot T_0$.

"checkpointing"