Wednesday, April 14 Lecture #33/42

PSO

What if your particles run away?

→ Bounds like $-2\pi \leq x, y \leq 2\pi$

→ Constraints

Options if your particle moves into an invalid space.

1) If the new position you would go to is not allowed, just don't go there.
2) If the partice would go into an invalid spot, just delete it and randomly add brand new particle to the swarm.

Pro: More exploration

Make sure you keep track of the global best.

One way to reduce the freq. of this is to set maximum allowable speed for each particle.

* Sometimes people add another term to the velocity, in between "personal best" and "global best." For every particle, randomly pick a few (3-10) other particles to be its "informants", then add a term

$$r_3 \cdot \delta \cdot (\text{[best sol from any informant]} - \text{current position})$$

* One interesting thing: PSO has no tweaking, nothing like H-C.

Pros / Cons:

(pro) Not having H-C = more exploration

(con) Might leave a good solution before you find it.

MH: Iterated Local Search
Variable Neighborhood Search

Topic 16 - Neighborhoods in Continuous Space

In our MHs in cns space that needed a tweak, we've used a very simple one:

Start with $x = (x_1, x_2, \ldots, x_d)$

$s = \text{tweak}(x) = x + (r_1\delta_1, r_2\delta_2, \ldots, r_d\delta_d)$

where r_i is a uniform random # between -1 and 1 and δ_i is a pre-determined parameter that specifies max. change allowed.

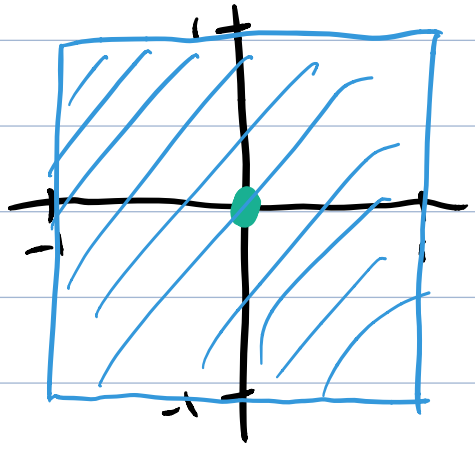
* Many of our examples: the δ_i were all the same, $\delta_1 = \delta_2 = 0.1$
  $\delta_1 = \delta_2 = 0.01$

* Spring example: $\delta_1 = \delta_2 = 0.01$
  $\delta_3 = 0.1$

For now, assume $\delta_i = 1$ and $x = \vec{0}$

so, $s = \text{tweak}(x) = (r_1, r_2, \ldots, r_d)$.

The new point s is somewhere in the d-dimensional cube centered at the origin



with side length 2.

The furthest a point could move $\sqrt{d}$.

$(0,0\ldots,0) \to (1,1,\ldots 1)$

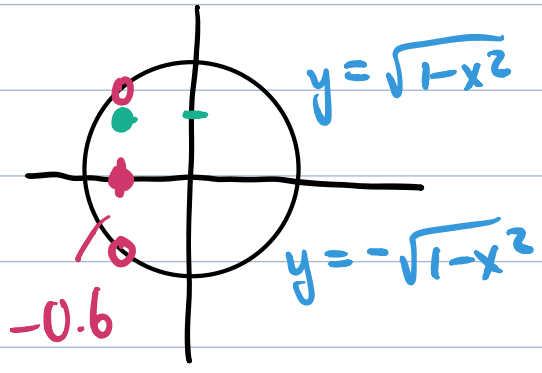
If d is large, your points might move a lot.

Alternatives:

* Scale down by a factor of $\frac{1}{\sqrt{d}}$
* Instead of a square, pick random points in a circle/sphere of radius 1.

How?

Bad ex 1: Pick $x \in [-1,1]$ uniformly at random.



Pick $y \in [-\sqrt{1-x^2}, \sqrt{1-x^2}]$ uniformly at random

Bad ex 2: Pick $\theta \in [0, 2\pi)$ uniformly at random, then a radius $r \in [0,1]$

uniformly at random.