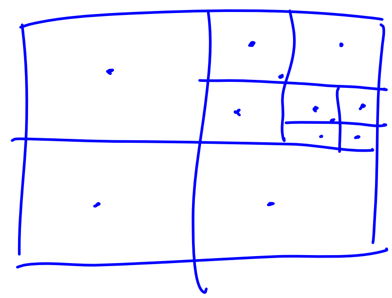
Global Optimization:

1. Pattern Search
2. Particle Swarm

Pattern Search:



- Grid search
- Use the patterns of the objective fn to figure out where to search
- Very small # of variable
- Functions smooth
- Use of surrogate function - replace the original function with ones that are easier to evaluate and to differentiate
- UOBQYA, DIRECT

Particle Swarm

Start with n particles

Particle i located at x_i, moving with velocity v_i (x_i and v_i are m-vectors)

for each iteration k

 Let $g = \underset{i}{\arg\min} f(x_i^k)$

 for each particle i

  r_1, r_2 = uniformly distributed random vectors

  $x_i^{k+1} = x_i^k + v_i^k$

  $v_i^{k+1} = c\, v_i^k + r_1 \circ (x_i^{k+1} - x_i^k) + r_2 \circ (x_i^{k+1} - g)$

end end

Mixed-Integer Nonlinear Programming

Bonmin → IPOPT

FilMINT → filterSQP

- Start with efficient NLP solver
- Convex NLPs
- Branch and bound : subproblems are NLPs
- Outer Approximation

$$\min_{x,y} \; f(x,y)$$
$$\text{s.t.} \quad g(x,y) \geq 0$$
$$x \in \mathbb{R}^n, \; y \in \mathbb{Z}^p$$

Iteration k: Fix y^k

Solve
$$\min_{x} \; f(x,y^k)$$
$$\text{s.t.} \quad g(x,y^k) \geq 0$$
$$x \in \mathbb{R}^n$$

Let the solution be x^k

Linearize the MINLP to update variables.

$$f(x,y) \approx f(x^k,y^k) + \nabla f(x^k,y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix}$$

$$g(x,y) \approx g(x^k,y^k) + \nabla g(x^k,y^k)^T \begin{bmatrix} x - x^k \\ y - y^k \end{bmatrix}$$

Solve

$$
\begin{aligned}
\min \quad & \alpha \\
\text{s.t.} \quad & f(x^l, y^l) + \nabla f(x^l, y^l)^T \begin{bmatrix} x - x^l \\ y - y^l \end{bmatrix} \leq \alpha \\
& g(x^l, y^l) + \nabla g(x^l, y^l)^T \begin{bmatrix} x - x^l \\ y - y^l \end{bmatrix} \geq 0 \\
& \text{for all } l = 1, \dots, k \\
& x \in \mathbb{R}^n, \; y \in \mathbb{Z}^p
\end{aligned}
$$

Mixed-integer linear programming problem