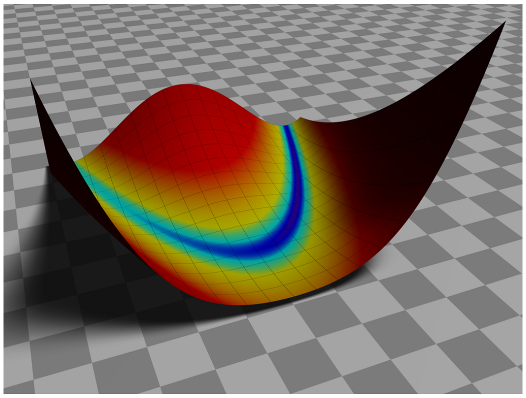
Gradient-Based Optimization Wrap Up

Lecture 21



ME EN 575
Andrew Ning
aning@byu.edu

Outline

Multiobjective Optimization

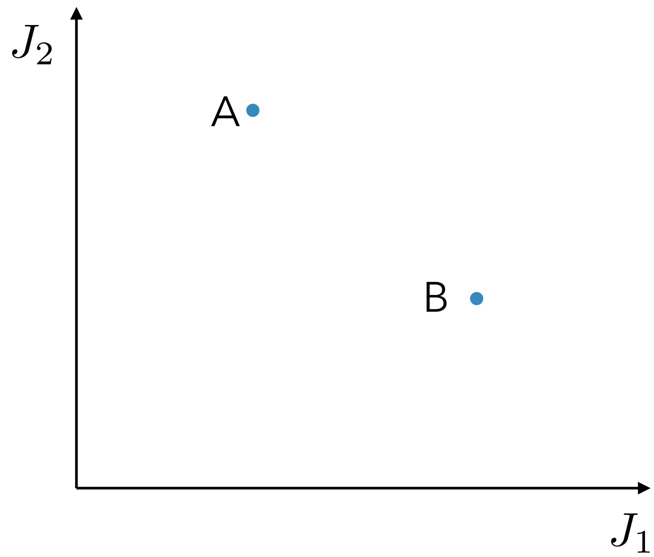
Multiobjective Optimization

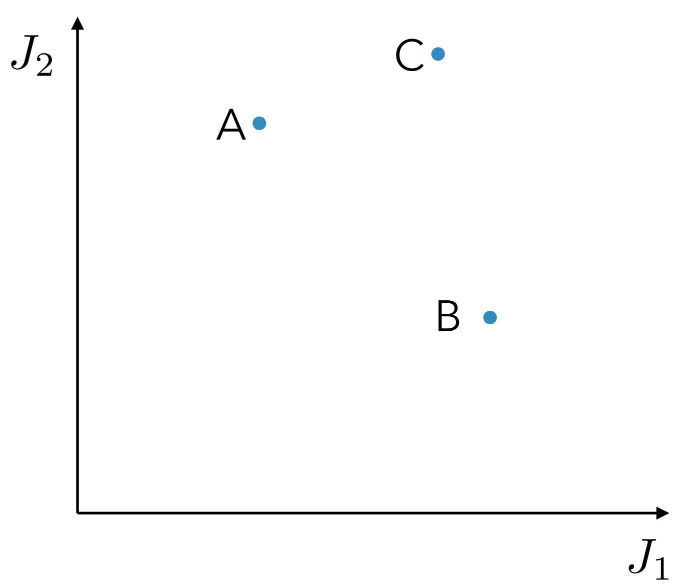
First, ask if it should be multiobjective:

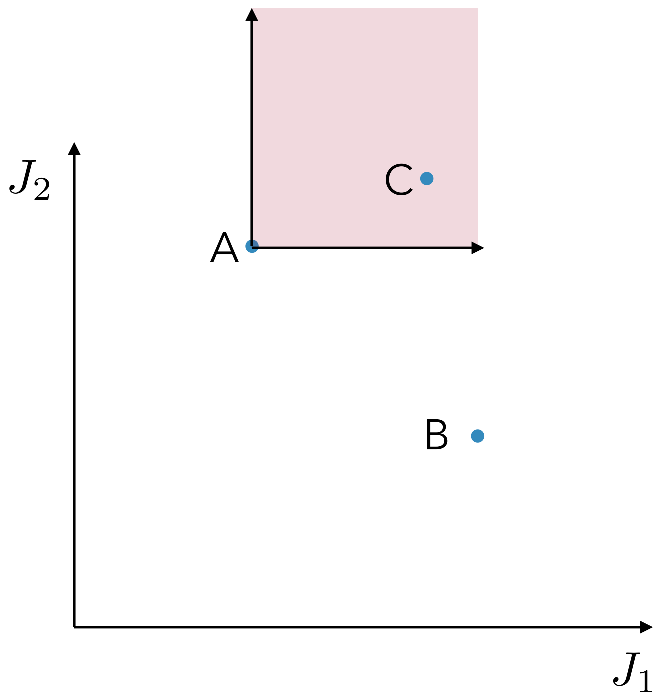
- Is there a higher-level objective that you are actually after?
- Are some of your objectives actually constraints?

Good reasons to pursue multiobjective:

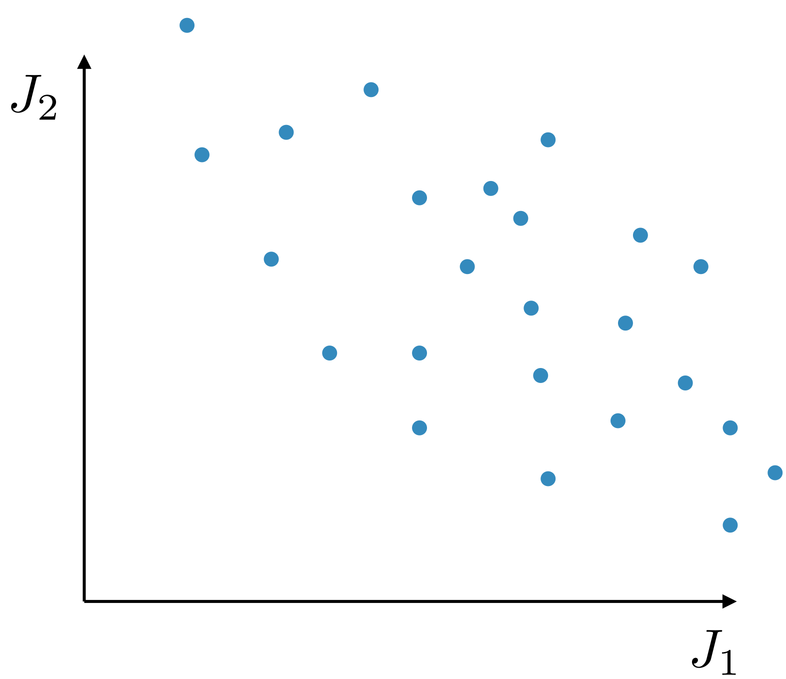
- Explore tradeoffs between potential objectives
- You aren't making the decision and want to present options

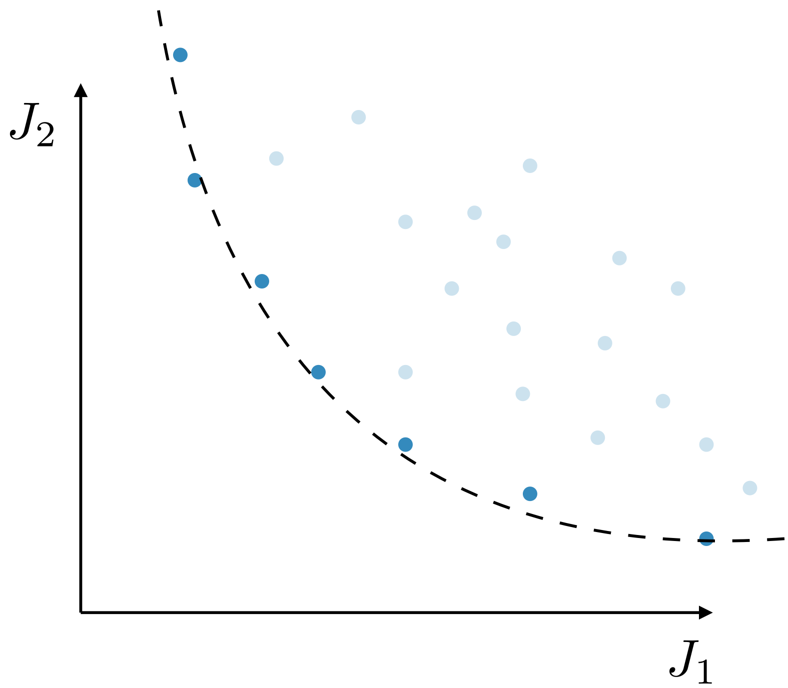

J_2
A
B
J_1


J_2
C
A
B
J_1



Point C is dominated by Point A





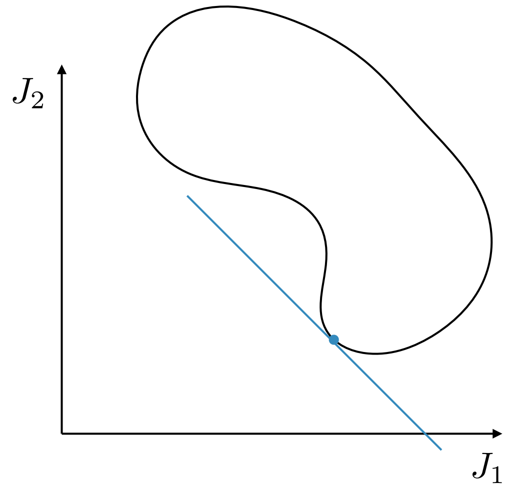
The dashed line represents the Pareto front and points on the front are called Pareto optimal.

Weighted Sum

Combine the two objectives as a weighted sum:

$$J(x) = J_1(x) + kJ_2(x)$$

vary k to define the Pareto front.



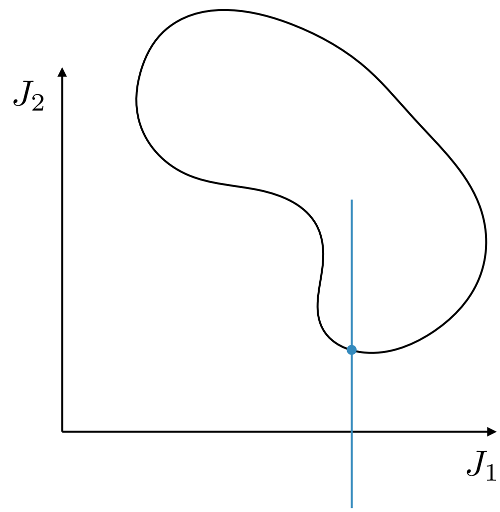
Pros: Simple

Cons: Difficult to determine appropriate weightings, nonuniform spacing, yields only the convex portion of the Pareto front.

Constraint Epsilon

Set the other objectives as constraints, but vary their limits:

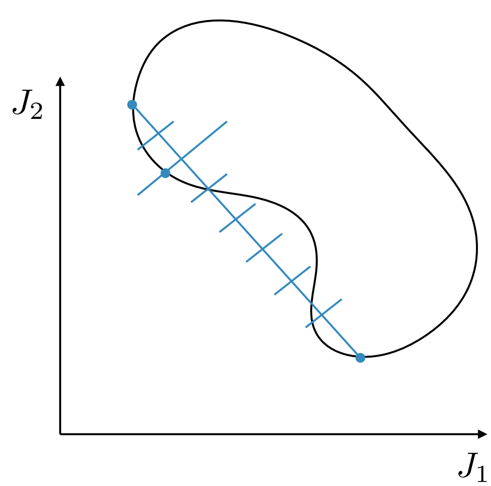
$$c(x) \leq c_{max}$$



Pros: Simple, constraint limits are more intuitive.

Cons: Nonuniform spacing.

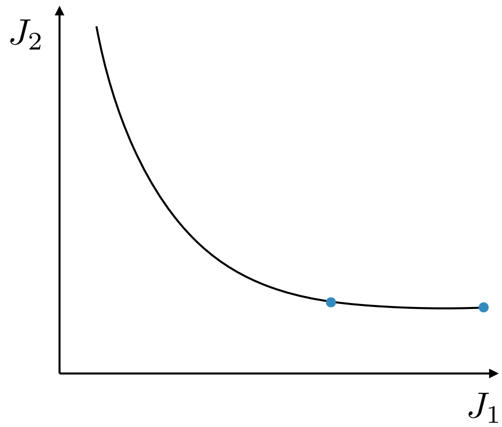
Normal Boundary Interface or Normal Constraint Method



Pros: Uniform spacing

Cons: Can waste time resolving portions of the

Smart Pareto Method



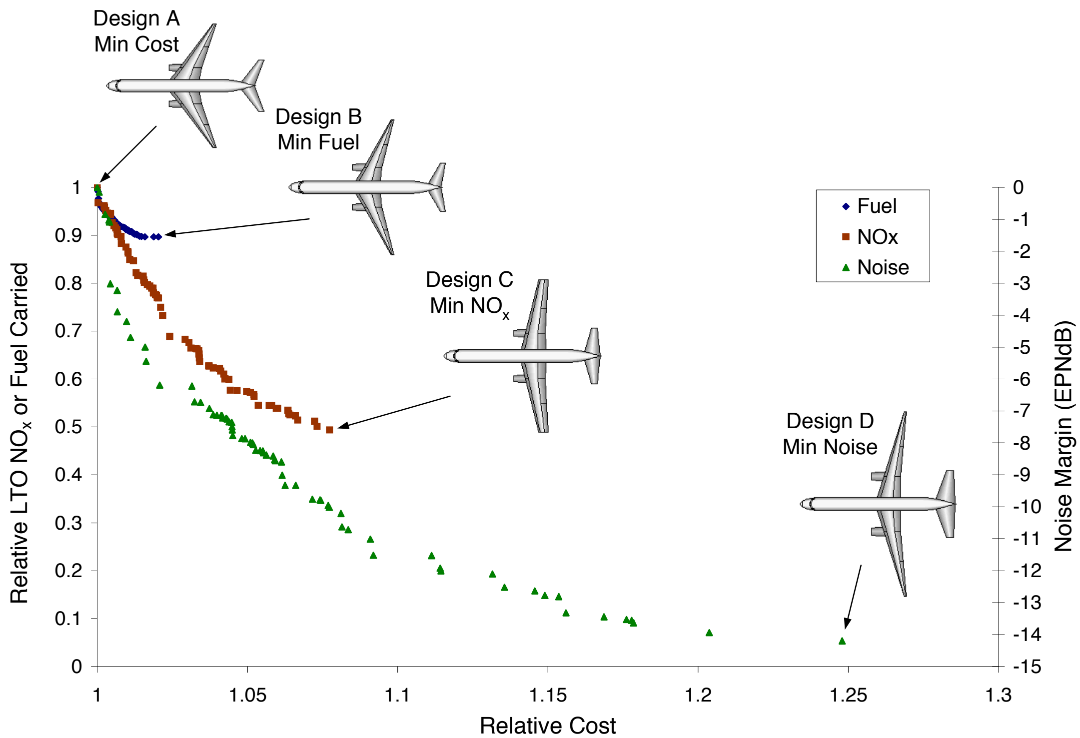
Pros: Uniform spacing, avoids resolving unimportant regions.

Hancock and Mattson, 2014

Gradient-free Approach

Use a multiobjective gradient-free method like an evolutionary algorithm.

Pros: Easy to use Cons: Very slow and inefficient



Nick Antoine, Aircraft Optimization for Minimal Environmental Impact, PhD Dissertation, 2004.

Some line search techniques use a filter method. This is a multiobjective problem with two objectives:

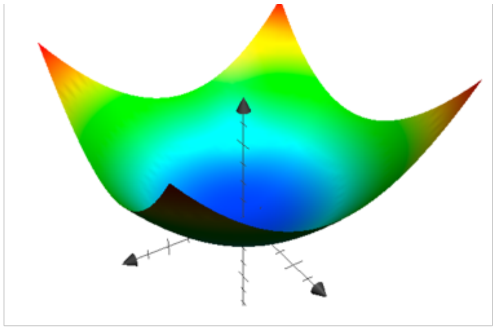
$$J_1(x) = f(x)$$

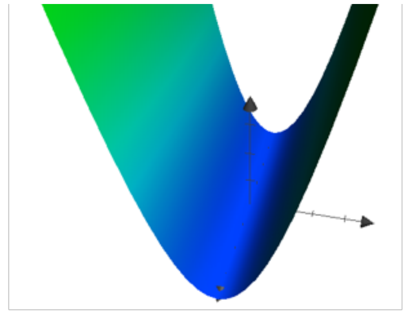
$$J_2(x) = \|c_{vio}(x)\|_1$$

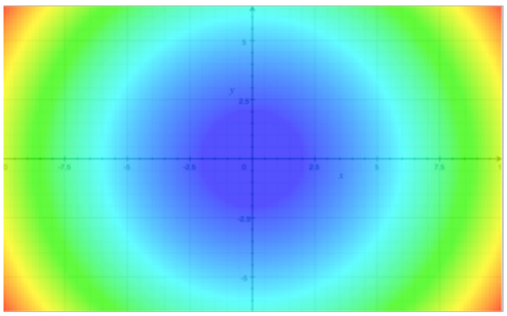
Other Techniques

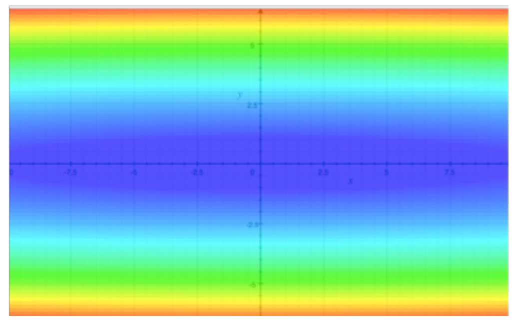
Scaling

Last time we discussed an example where scaling the design variables was important.









Scaling objectives and constraints is also important. Consider an infeasible point where ∇f is very large and ∇c is very small.

Rule of thumb:

Scale objectives, constraints, and design variables to all be of $\mathcal{O}(1)$.

Sometimes, intentionally skewed scaling is desirable.

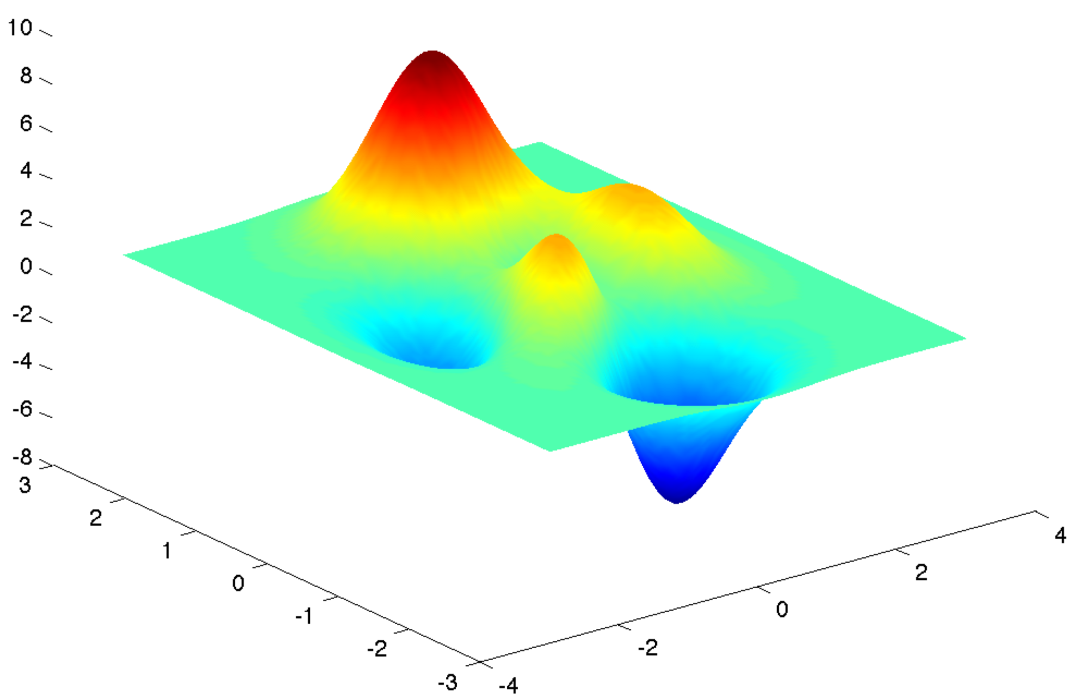
$$0 \leq \phi \leq 120^\circ$$

$$0 \leq \frac{\phi}{1000} \leq \frac{120^\circ}{1000}$$

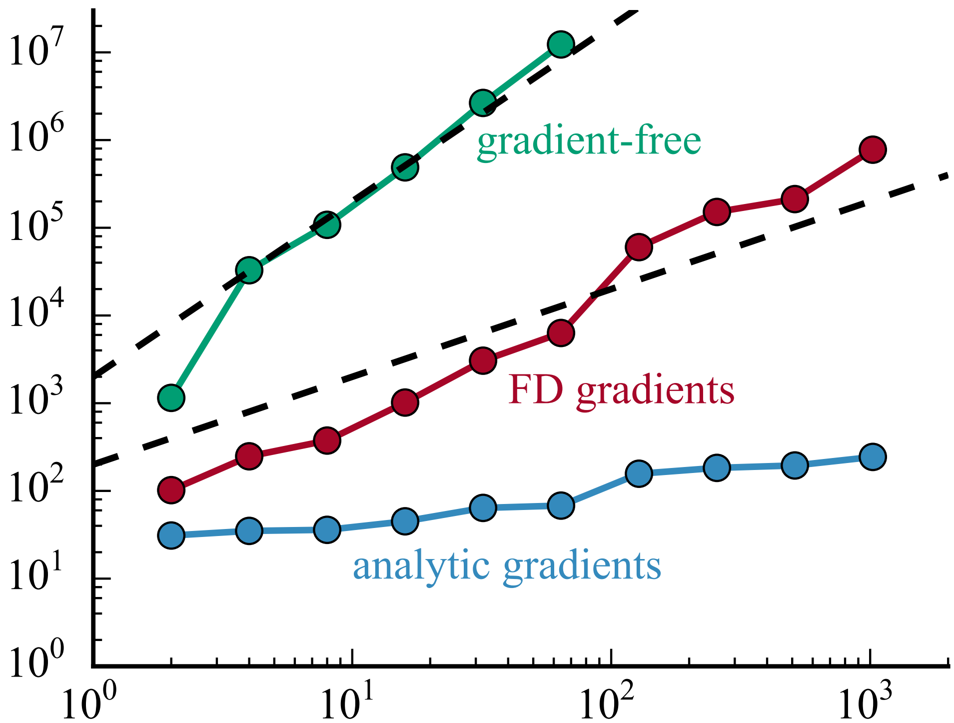
Reformulation

- absolute value, max/min
- mathematically equivalent formulations
- example: distance to barrier
- example: delegates per vote

Local Optima



Gradients



Parallelization

What opportunities have you noticed for parallelization?