Welcome to ASEN 6519

Optimization: Applications and Algorithms

Why Optimization?

Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!

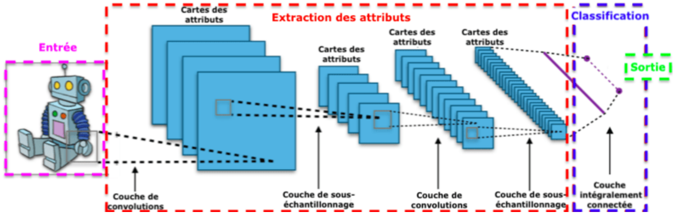
Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!
- Optimization is everywhere!

Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!

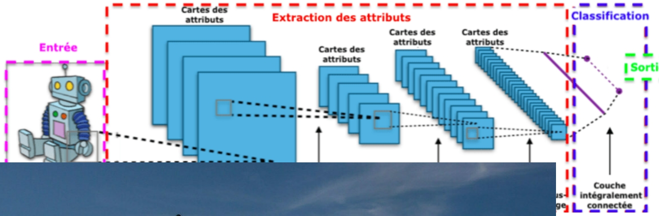
- Optimization is everywhere!



Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!

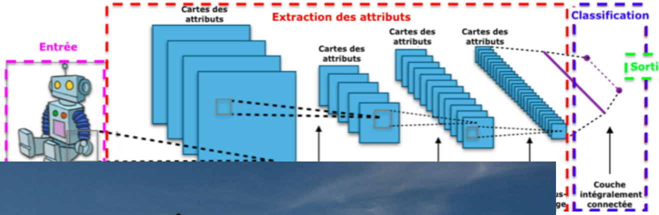
- Optimization is everywhere!



Why Optimization?

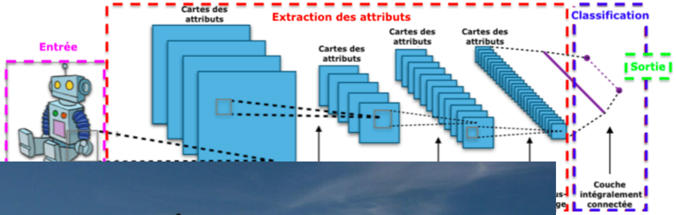
- At some point in your career you will have to optimize something. I want you to be an optimization hero!

- Optimization is everywhere!



Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!
- Optimization is everywhere!
- Optimization problems are a precise way to communicate about questions before knowing the answer. This means that you can work with other people and computers.



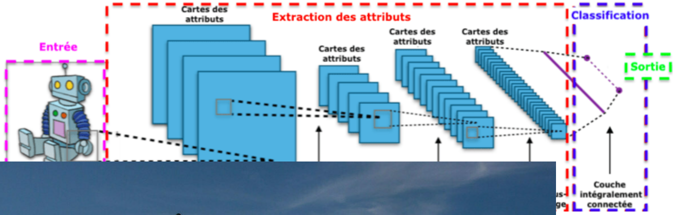
Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!

- Optimization is everywhere!

- Optimization problems are a precise way to communicate about questions before knowing the answer. This means that you can work with other people and computers.

- Someday, you might design something that I use, and I want it to be optimized! ●



Why Optimization?

- At some point in your career you will have to optimize something. I want you to be an optimization hero!
- Optimization is everywhere!
- Optimization problems are a precise way to communicate about questions before knowing the answer. This means that you can work with other people and computers.
- Someday, you might design something that I use, and I want it to be optimized! ●

https://imgur.com/gallery/musical-cars-sequel-to-15-point-turn-ADKFKpa

Optimization Problems and Solutions

Optimization Problem
Mathematical Program

objective

$$\text{minimize } f(\vec{x})$$

decision variables
"design point"

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

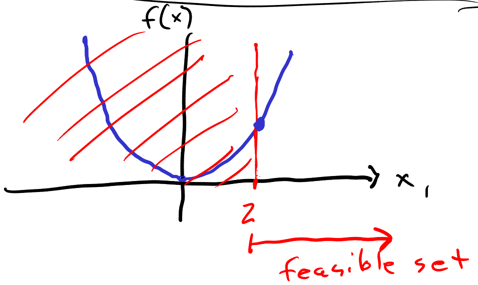
constraint

$$\text{subject to } \vec{x} \in X$$

feasible set

minimize $f(x)$ is equivalent to maximize $-f(x)$

$$\underset{x_1}{\text{minimize}}\ x_1^2$$
$$\text{subject to } x_1 \geq 2$$



$x^* = 2 \quad f(x^*) = 4$

Solution

if $X = \emptyset$ (empty) then problem is infeasible

$x_1 \leq -2$
$x_1 \geq 1$

A globally optimal solution is a design point $\vec{x}^* \in X$ where

$$f(\vec{x}^*) \leq f(\vec{x}) \quad \forall \vec{x} \in X$$

"solution"
"minimizer"

$f(x^*)$ is the value of the problem

$\underset{x}{\min} f(x)$ returns value

$\underset{x}{\operatorname{argmin}} f(x)$ returns x^*

Optimization in my career

1. Gradient descent for space domain awareness

Optimization in my career

1. Gradient descent for space domain awareness

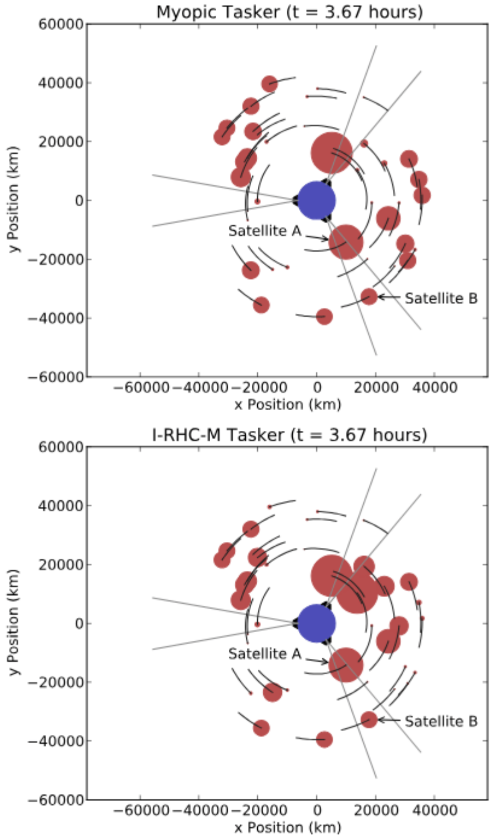


Fig. 3. Sensor network representation early in simulation 18.

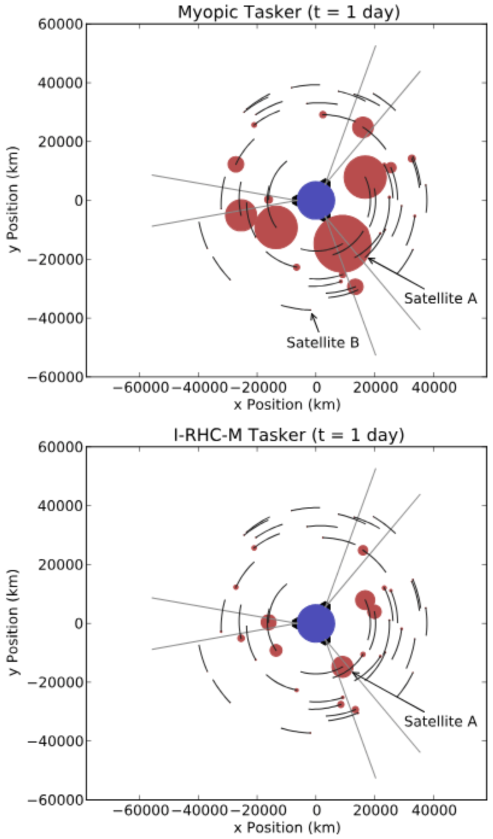


Fig. 4. Sensor network representation at end of simulation 18.

Optimization in my career

1. Gradient descent for space domain awareness

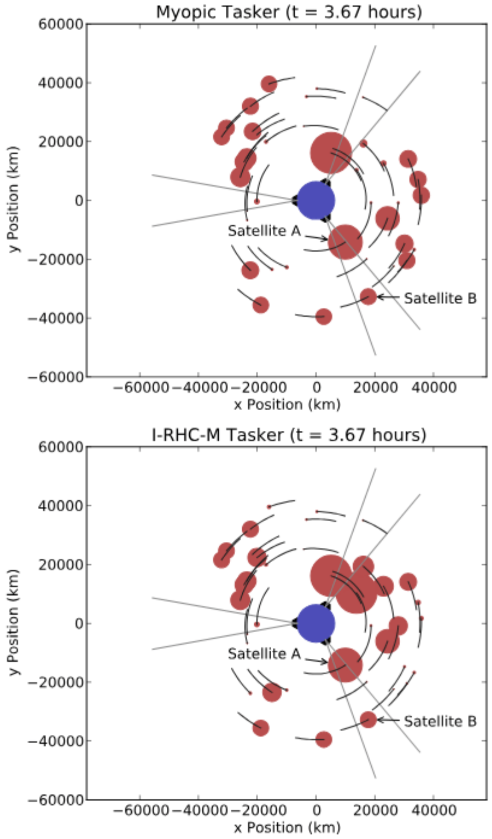


Fig. 3. Sensor network representation early in simulation 18.

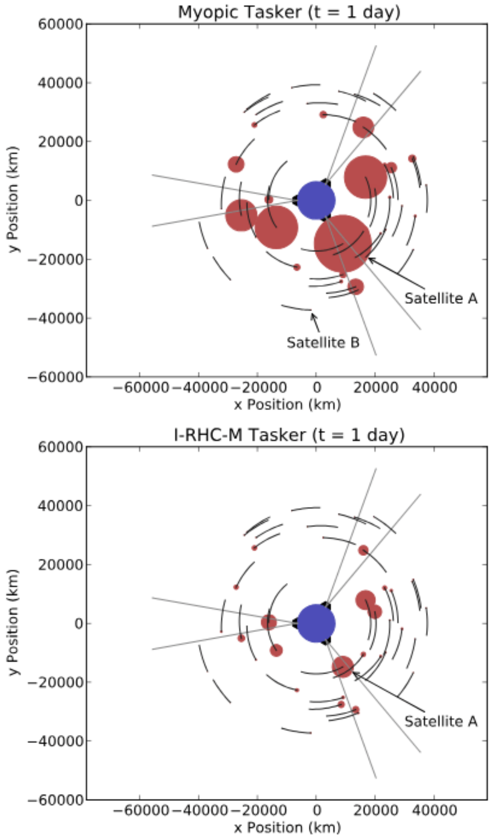


Fig. 4. Sensor network representation at end of simulation 18.

$$\frac{\partial J_{\log}}{\partial \pi^{j}_{t,k}} = E\left[\sum_{i \in T^{j}(t)} \log(J(\chi_i, \bar{U})) \middle| u^{j}_{t} = k \right]$$

Optimization in my career

1. Gradient descent for space domain awareness

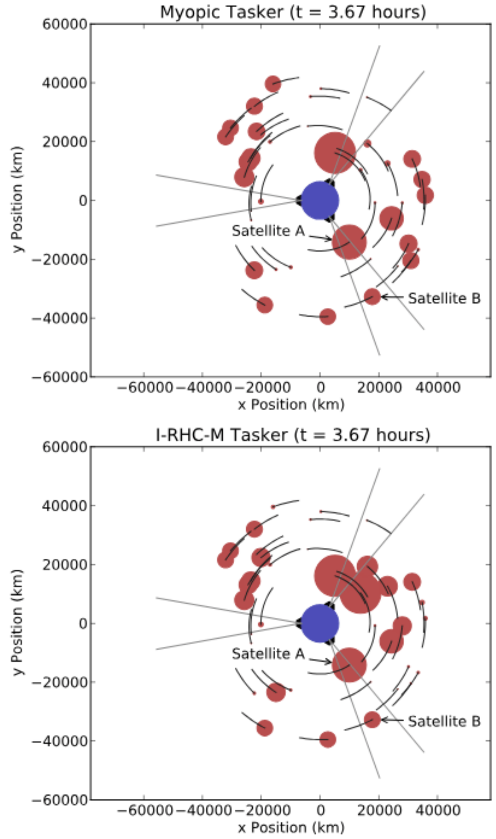


Fig. 3. Sensor network representation early in simulation 18.

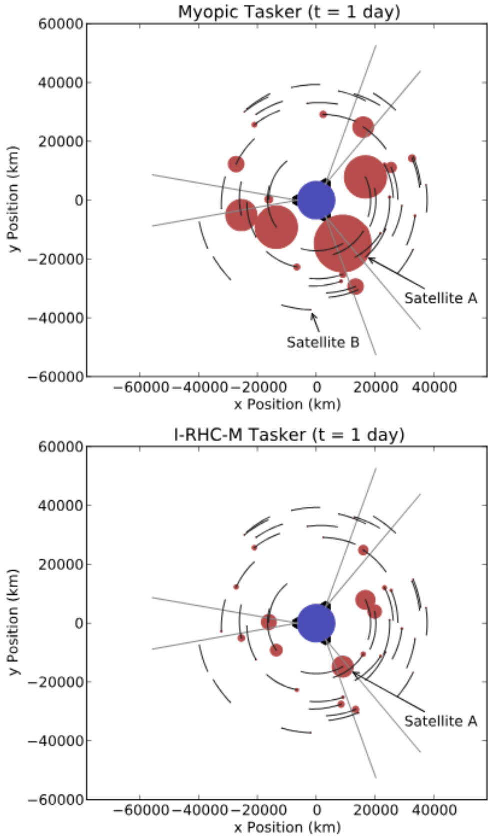


Fig. 4. Sensor network representation at end of simulation 18.

$$\frac{\partial J_{\log}}{\partial \pi^{j}_{t,k}} = E\left[\sum_{i\in T^{j}(t)} \log(J(\chi_i, \bar{U}))\bigg|\, u^{j}_{t} = k\right]$$

$$\Pi^{j}_{t} = \mathcal{P}_P\left[\Pi^{j}_{t} + \gamma \frac{\widehat{\partial J}}{\partial \Pi^{j}_{t}}\right]$$

Optimization in my career

1. Gradient descent for space domain awareness

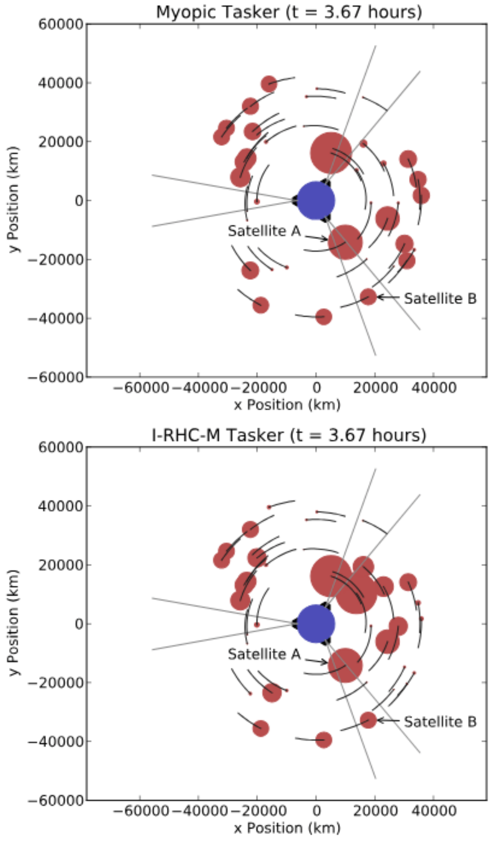


Fig. 3. Sensor network representation early in simulation 18.

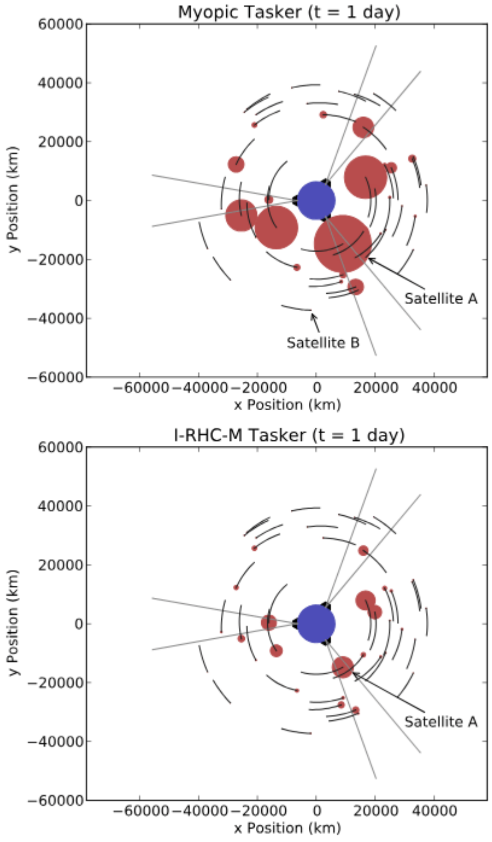


Fig. 4. Sensor network representation at end of simulation 18.

$$\frac{\partial J_{\log}}{\partial \pi^{j}_{t,k}} = E\left[\sum_{i \in T^{j}(t)} \log(J(\chi_i, \bar{U}))\,\middle|\, u^{j}_{t} = k\right]$$

$$\Pi^{j}_{t} = \mathcal{P}_{P}\left[\Pi^{j}_{t} + \gamma \widehat{\frac{\partial J}{\partial \Pi^{j}_{t}}}\right]$$

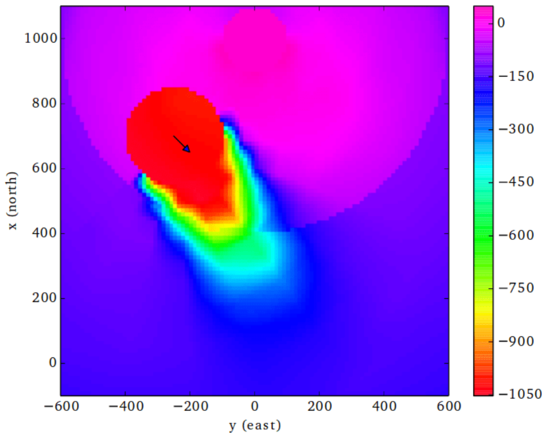
$$\widehat{\frac{\partial J}{\partial \pi^{j}_{t,k}}} = \begin{cases} \frac{1}{\pi^{j}_{t,k}} \sum_{i \in T^{j}(t)} \log(J(\chi_i, \bar{U}, \omega)) & \text{if } u^{j}_{t} = k \\ 0 & \text{o.w.} \end{cases}$$

Optimization in my career

2. Airborne collision avoidance

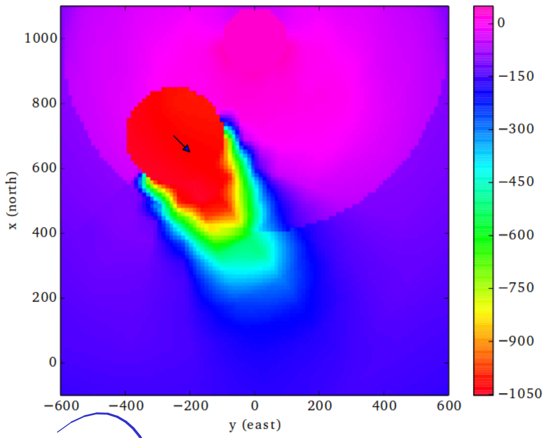
Optimization in my career

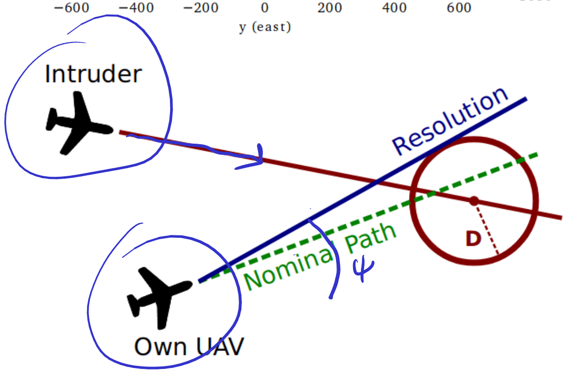
2. Airborne collision avoidance



Optimization in my career

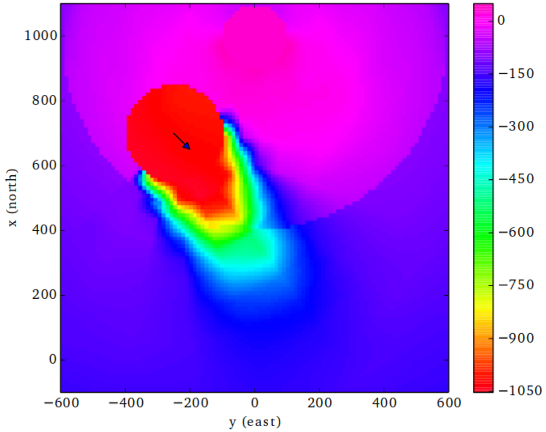
2. Airborne collision avoidance

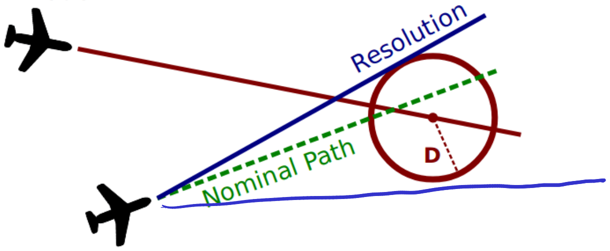




Optimization in my career

2. Airborne collision avoidance





$$\underset{\psi}{\text{minimize}} \quad |\psi - \psi_{goal}|$$

$$\text{subject to} \quad d_{min}(\psi) \geq D$$

also an optimization problem

$$d_{min} \equiv \min_{\tau} d(\psi, \tau)$$

$$d(\psi, \tau) = \sqrt{\Delta x(\tau)^2 + \Delta y(\tau)^2}$$

$$\frac{d\, d(\psi,\tau)}{d\tau} = 0$$

$$\tau_{\min}\left(s, \psi^{(o)}_{\text{cand}}\right) = \max\left\{\frac{a+b}{c-2d}, 0\right\}$$

$$a := -v^{(i)} x^{(i)} \cos(\psi^{(i)}) - v^{(i)} y^{(i)} \sin(\psi^{(i)})$$

$$b := v^{(o)} x^{(i)} \cos(\psi^{(o)}_{\text{cand}}) + v^{(o)} y^{(i)} \sin(\psi^{(o)}_{\text{cand}})$$

$$c := v^{(o)\,2} + v^{(i)\,2} \cos^2(\psi^{(i)}) + v^{(i)\,2} \sin^2(\psi^{(i)})$$

$$d := v^{(o)} v^{(i)} (\cos(\psi^{(i)}) \cos(\psi^{(o)}_{\text{cand}}) + \sin(\psi^{(i)}) \sin(\psi^{(o)}_{\text{cand}})).$$

Optimization in my career

3. Assigning teams based on preferences

Optimization in my career

3. Assigning teams based on preferences

Integer-Linear Program

$$\underset{x_{ij}}{\text{minimize}} \quad \sum_i \sum_j c_{ij} x_{ij} \quad (1)$$

Linear

$$\text{subject to} \quad x_{ij} \in \{0, 1\} \quad (2)$$

$$\sum_j x_{ij} = 1 \quad \forall i \quad (3)$$

$$\underline{s}_j \leq \sum_i s_i x_{ij} \leq \bar{s}_j \quad \forall j \quad (4)$$

$$\sum_i r_{ki} x_{ij} \geq \underline{r}_{kj} \quad \forall k, j \quad (5)$$

$$c_{ij} = 2^{\log_2(n)(p_{ij}-1)} s_i$$

Group: i ← not individuals

of friends

Project: j

Group size: s

Group skills: r

Group preference: p

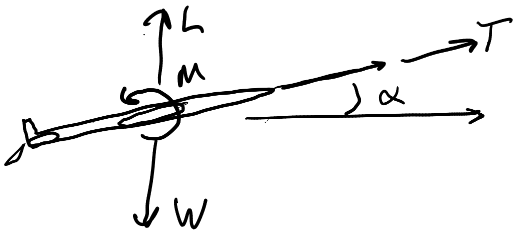
x_{ij}

$q_{ik} x_{ij} x_{kj}$

what didn't work: optimizing groups and preferences simultaneously

Optimization in my career

4. Finding trim points in aircraft dynamics



$\alpha, \delta_e, \delta_t \rightarrow L(\alpha, \delta_e)$, $M(\alpha, \delta_e)$

elevator throttle

$$C_{L_{trim}} = C_W = C_{L_\alpha}\alpha^* + C_{L_{\delta_e}}\delta_e^*$$

$$C_{M_{trim}} = 0 = C_{M_\alpha}\alpha^* + C_{M_{\delta_e}}\delta_e^*$$

Linear Equation

$$\underset{\alpha, \delta_e, \delta_t}{\text{minimize}} \quad |L + T\sin\alpha - W| + |M|$$

subject to $-10^\circ \leq \delta_e \leq 10^\circ$

$0 \leq \delta_t \leq 1$

Matlab : fmincon

Wisdom

Wisdom

- There is no single algorithm that is best for all problems

Wisdom

- There is no single algorithm that is best for all problems
- Structures/Properties of a problem determine which algorithms are appropriate and how easy it is to solve

Wisdom

- There is no single algorithm that is best for all problems
- Structures/Properties of a problem determine which algorithms are appropriate and how easy it is to solve
    - Art: Taking real-world problem and formulating it as an optimization problem with favorable properties

Wisdom

- There is no single algorithm that is best for all problems
- Structures/Properties of a problem determine which algorithms are appropriate and how easy it is to solve
  - Art: Taking real-world problem and formulating it as an optimization problem with favorable properties
- Ease of implementation/use matters!

Wisdom

- There is no single algorithm that is best for all problems
- Structures/Properties of a problem determine which algorithms are appropriate and how easy it is to solve
    - Art: Taking real-world problem and formulating it as an optimization problem with favorable properties
- Ease of implementation/use matters!
    - Can you either (1) put the problem into a form that an existing solver can use, or (2) code up the algorithm itself easily?

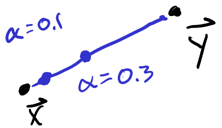
Wisdom

- There is no single algorithm that is best for all problems
- Structures/Properties of a problem determine which algorithms are appropriate and how easy it is to solve
  - Art: Taking real-world problem and formulating it as an optimization problem with favorable properties
- Ease of implementation/use matters!
  - Can you either (1) put the problem into a form that an existing solver can use, or (2) code up the algorithm itself easily?
  - Even if an algorithm performs better, it will be neglected if it is more difficult to use.

A very important property: Convexity

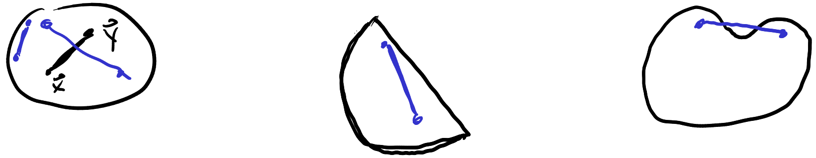
A convex combination of two points $\vec{x}$ and $\vec{y}$

$$\alpha\vec{x} + (1-\alpha)\vec{y} \quad \text{for } \alpha \in [0,1]$$



A set S is a convex set if and only if

$$\alpha\vec{x} + (1-\alpha)\vec{y} \in S \quad \forall \vec{x},\vec{y} \in S, \ \alpha \in [0,1]$$



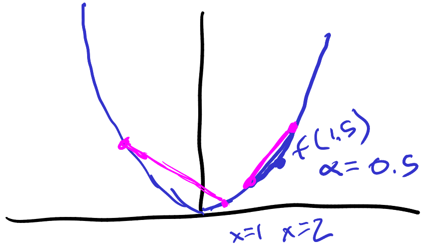
How many US states are convex?

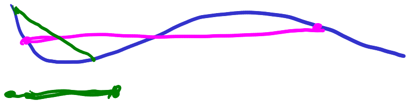
A very important property: Convexity

A function f is a convex function over convex set S iff

$$f(\alpha\vec{x}+(1-\alpha)\vec{y}) \le \alpha f(\vec{x}) + (1-\alpha) f(\vec{y}) \quad \forall \vec{x},\vec{y}\in S,\ \alpha\in[0,1]$$

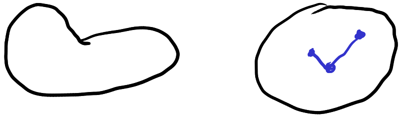
$f(x)=x^2$





An optimization problem is a convex optimization problem iff the objective function and feasible set are convex.

Convex optimization problems can be solved reliably in a (short) predictable amount of time.



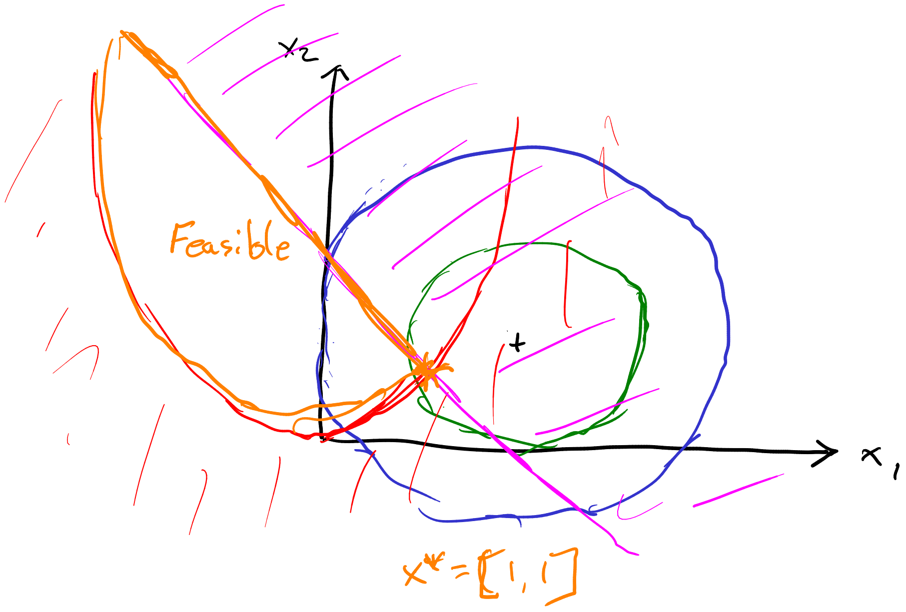
Simple Example

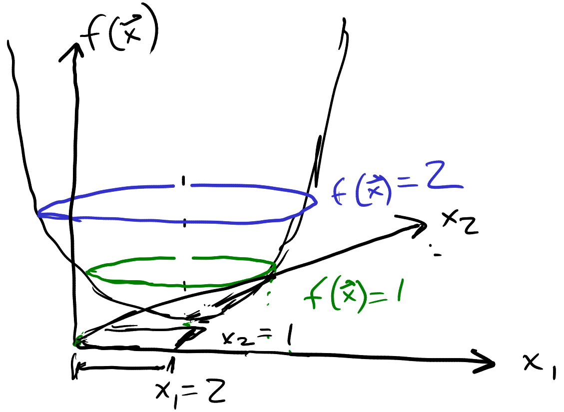
(N+W p. 3)

minimize $(x_1-2)^2 + (x_2-1)^2$

subject to $x_1^2 - x_2 \le 0 \rightarrow x_2 \ge x_1^2$

$x_1 + x_2 \le 2 \rightarrow x_2 \le 2 - x_1$

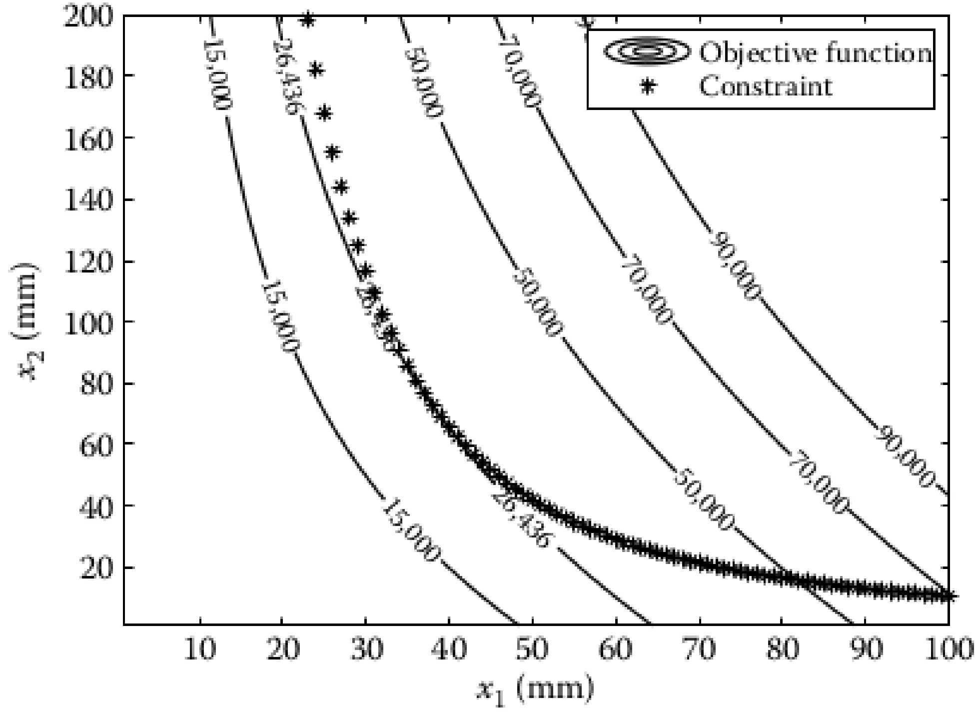




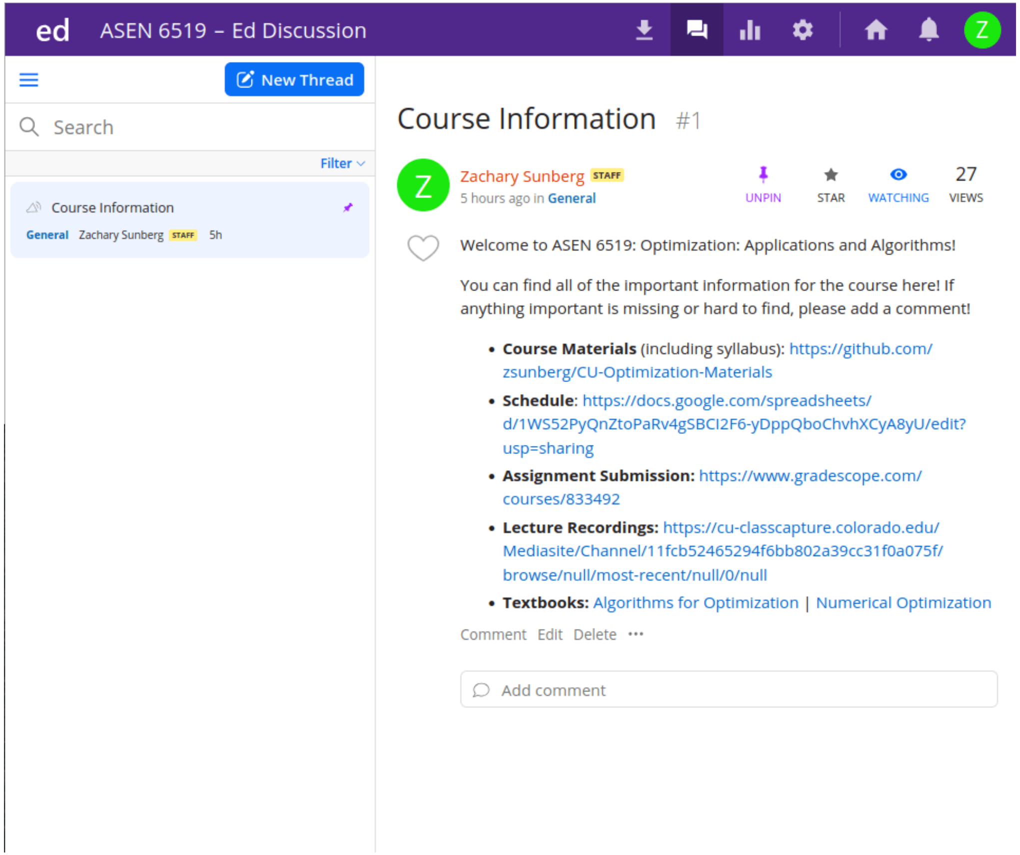
An Engineering Design Example: Problem Formulation

An Engineering Design Example: Graphical Solution

An Engineering Design Example: Graphical Solution



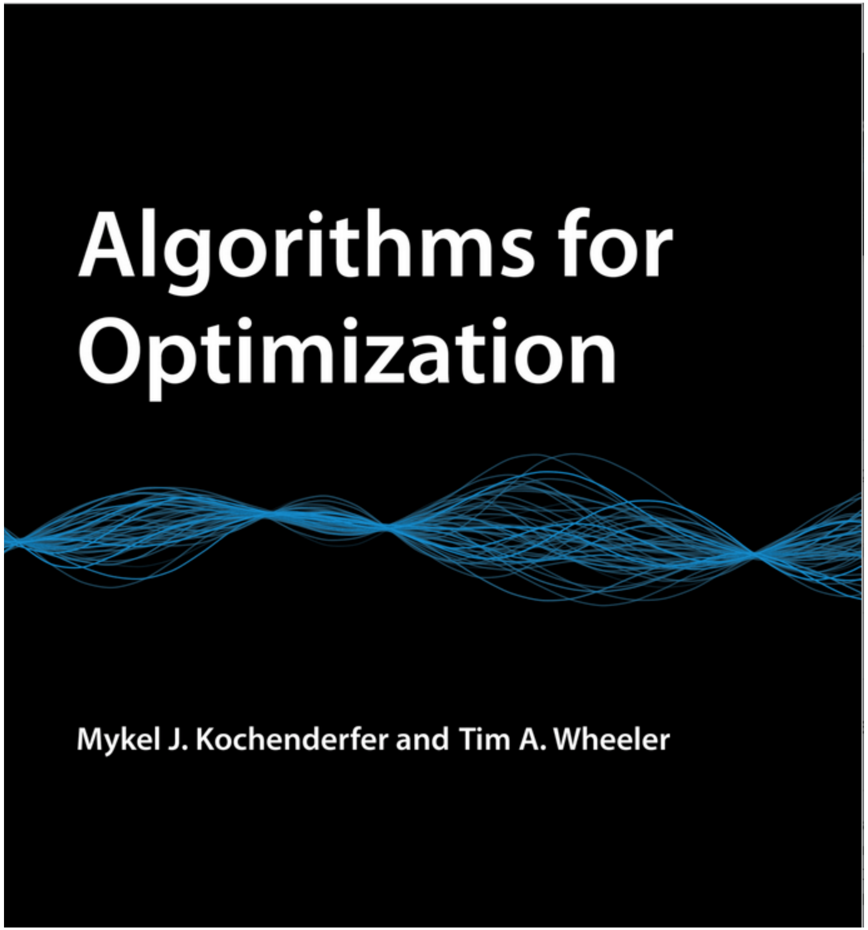
Course Hub: Edstem

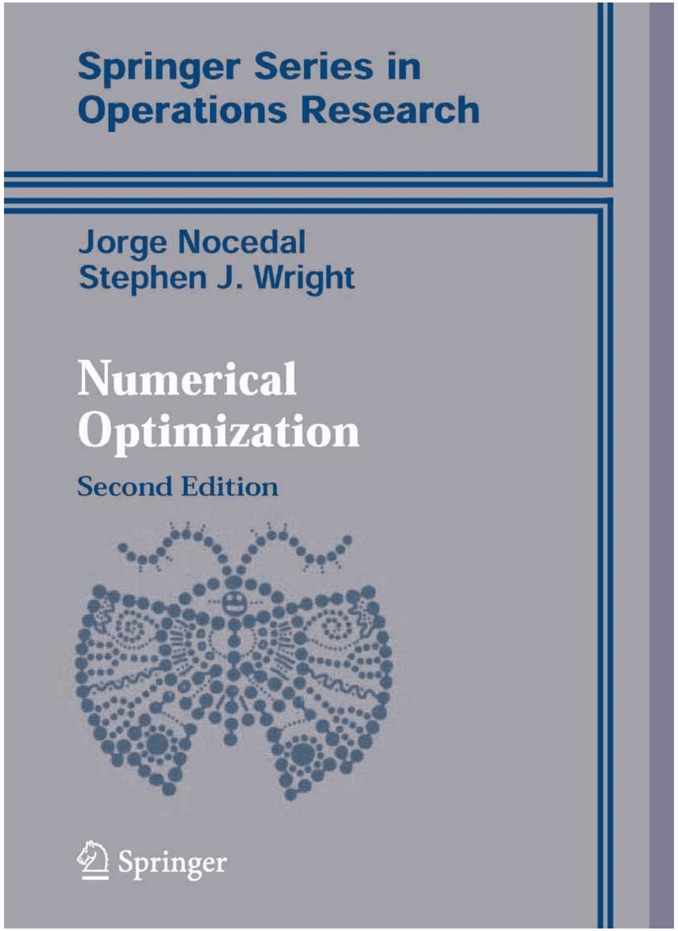


Syllabus:
https://github.com/zsunberg/CU-Optimization-Materials/blob/main/syllabus/syllabus.pdf

Join Ed:
https://edstem.org/us/join/txm78P

Books


Algorithms for Optimization
Mykel J. Kochenderfer and Tim A. Wheeler


Springer Series in Operations Research
Jorge Nocedal
Stephen J. Wright
Numerical Optimization
Second Edition
Springer