Optimization Methods (CS1.404)
Spring 2023

Naresh Manwani

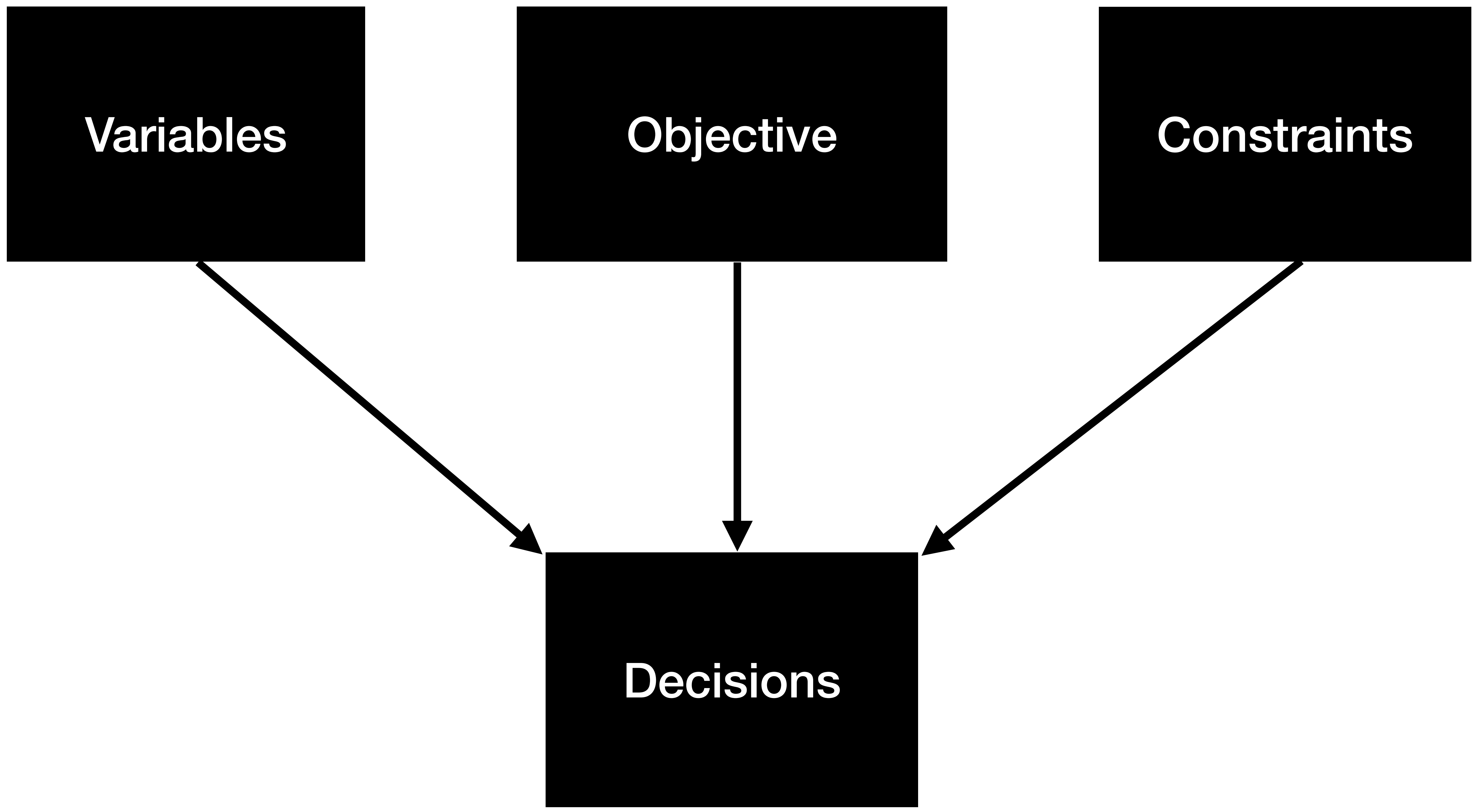
Machine Learning Lab, IIIT-H

January 4th, 2024

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
HYDERABAD

What is this course about?

The mathematics behind making optimal decisions



Finance

Variables
Amounts invested in each asset

Constraints
Budget, investment per asset,
minimum return, etc.

Objective
Maximize profit, minus risk

Optimal control

Variables
Inputs: thrust, flaps, etc.

Constraints
System limitations, obstacles, etc.

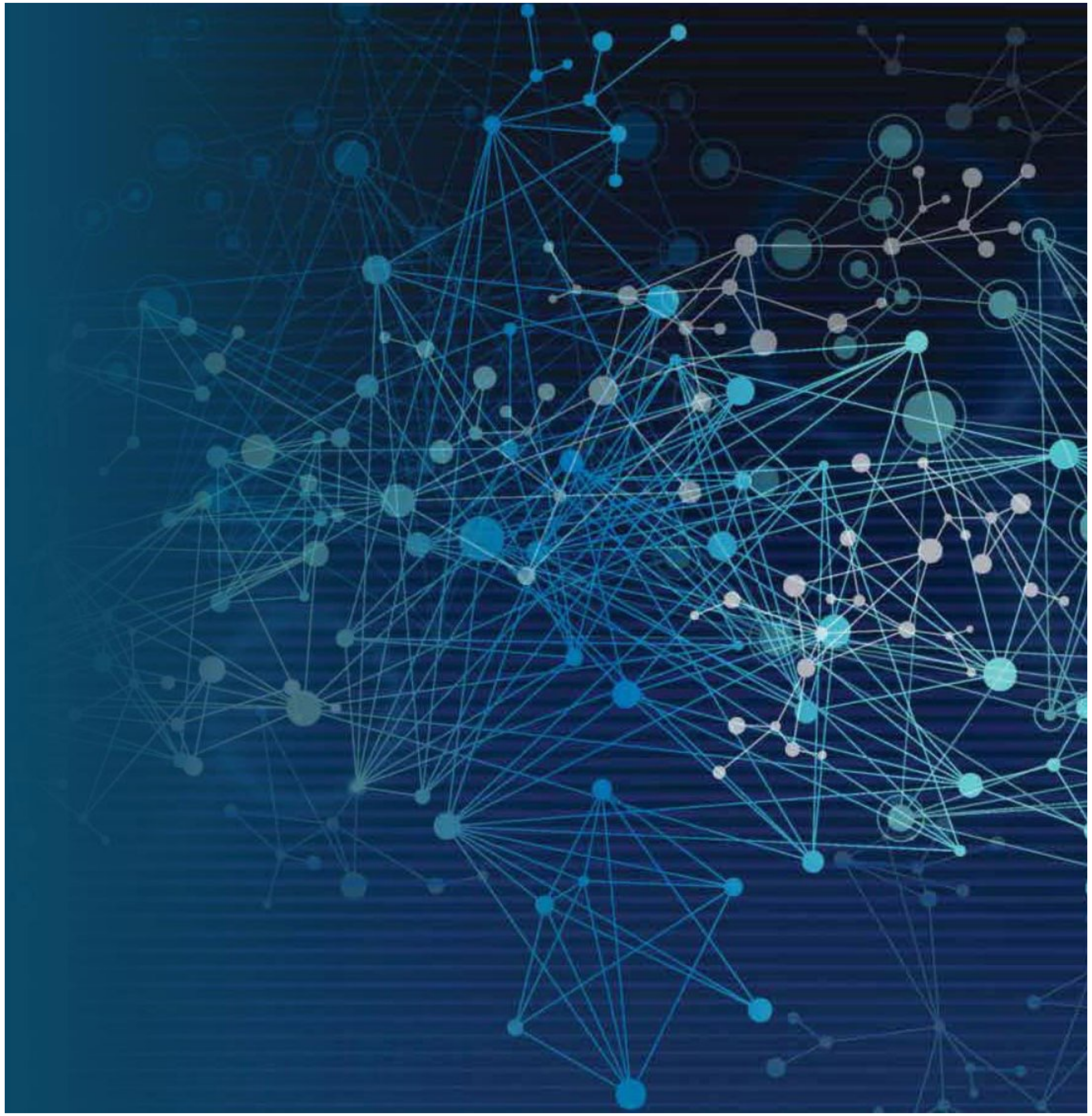
Objective
Minimize distance to target and fuel consumption

Machine learning

Variables
Model parameters

Constraints
Prior information, parameter limits

Objective
Minimize prediction error, plus regularization



Mathematical optimization

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \ldots, m \end{array}$$

$x = (x_1, \ldots, x_n)$ Variables

$f : \mathbf{R}^n \to \mathbf{R}$ Objective function

$g_i : \mathbf{R}^n \to \mathbf{R}$ Constraint functions

$x^\star$ Solution/Optimal point

$f(x^\star)$ Optimal value

Solving optimization problems

General case → **Very hard!**

Compromises

- Long computation times
- Not finding the solution
  (in practice it may not matter)

Solving optimization problems

General case → **Very hard!**

Compromises

- Long computation times
- Not finding the solution
  (in practice it may not matter)

Exceptions

- Linear optimization
- Convex optimization

→ **Can be solved very efficiently and reliably**

Linear optimization

$$
\begin{array}{ll}
\text{minimize} & c^T x \\
\text{subject to} & a_i^T x \leq b_i, \quad i = 1. \dots, m
\end{array}
$$

No analytical formula (99% of the time there will be none in this course!)

Efficient algorithms and software we can solve problems with several thousands of variables and constraints

Extensive theory (duality, degeneracy, sensitivity)

- A chemical company has 2 factories F_1 and F_2 and a dozen retail outlets $R_1, R_2, \ldots, R_{12}$.
- Each factory F_i can produce a_i tons of a certain chemical product each week.
- Each retail outlet R_j has a known weekly demand of b_j tons of the product.
- Cost of shipping one ton of the product from factory F_i to retail outlet R_j is c_{ij}.
- **Goal:** Determine how much of the product to ship from each factory to each outlet so as to satisfy all the requirements and minimize cost.

- Let x_{ij} be the number of tons of the product shipped from factory F_i to retail outlet R_j.
- The variables of the problem are $x_{ij},\ i = 1, 2;\ j = 1, \ldots, 12$.
- We can write the problem as

$$\min \sum_{i=1}^{2} \sum_{j=1}^{12} c_{ij} x_{ij}$$

$$s.t. \begin{cases} \sum_{j=1}^{12} x_{ij} \leq a_i, & i = 1, 2 \\ \sum_{i=1}^{2} x_{ij} \geq b_j, & j = 1, \ldots, 12 \\ x_{ij} \geq 0, & i = 1, 2;\ j = 1, \ldots, 12 \end{cases}$$

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY HYDERABAD

Nonlinear optimization

$$\begin{array}{ll}\text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \dots, m\end{array}$$

Hard to solve in general

- multiple local minima
- discrete variables $x \in \mathbf{Z}^n$
- hard to certify optimality

Convex optimization

Convex functions

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & g_i(x) \leq 0, \quad i = 1, \ldots, m \end{array}$$

All local minima are global!

Efficient algorithms and software

Extensive theory (convex analysis and conic optimization) [ORF523]

Used to solve non convex problems

Prehistory of optimization

Calculus of variations

Fermat/Newton

minimize $\quad f(x),\ x \in \mathbf{R}$

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = 0$$

1670

Euler

minimize $\quad f(x),\ x \in \mathbf{R}^n$

$$\nabla f(x) = 0$$

1755

Lagrange

minimize $\quad f(x)$

subject to $\quad g(x) = 0$

1797

Time

History of optimization
Algorithms

Origin of linear optimization (Kantorovich, Koopmans)	Simplex algorithm (Dantzig)	Ellipsoid method (Khachyan)	Interior-point method for linear optimization (Karmarkar)	Interior-point methods for convex optimization (Nesterov, Nemirovski)	Large-scale optimization
1930s	1947	1970s	1984	1990s	2000s

History of optimization

Algorithms

Origin of linear optimization (Kantorovich, Koopmans)	Simplex algorithm (Dantzig)	Ellipsoid method (Khachyan)	Interior-point method for linear optimization (Karmarkar)	Interior-point methods for convex optimization (Nesterov, Nemirovski)	Large-scale optimization
1930s	1947	1970s	1984	1990s	2000s

Linear vs Nonlinear

History of optimization

Algorithms

Origin of linear optimization (Kantorovich, Koopmans)	Simplex algorithm (Dantzig)	Ellipsoid method (Khachyan)	Interior-point method for linear optimization (Karmarkar)	Interior-point methods for convex optimization (Nesterov, Nemirovski)	Large-scale optimization
1930s	1947	1970s	1984	1990s	2000s

Linear vs Nonlinear

Convex vs Nonconvex

History of optimization

Applications

Operations Research
Economics

Engineering
(control, signal processing, communications, …)

Machine learning
Statistics

1990s

2000s

Applications of Optimization Algorithms

- Almost all engineering fields
- **Mechanics:** Rigid body problems, differential equations
- **Economics:** Pricing, profit maximization, cost minimization, trade theory, market portfolio optimization
- **Electrical and Communication Engineering:** space mapping structures of microwave structures, antenna design, circuit design for chip so as to save space, power systems managements
- **Computer Science:** Machine learning, social network optimization, graph theory (min-cut/max flow, matching algorithms, independent set) etc.

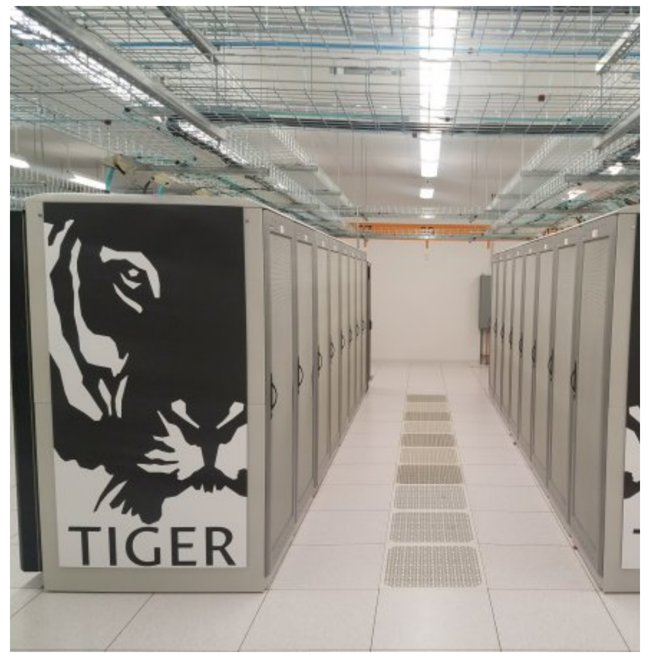
What is happening today?

Huge scale optimization

Massive data

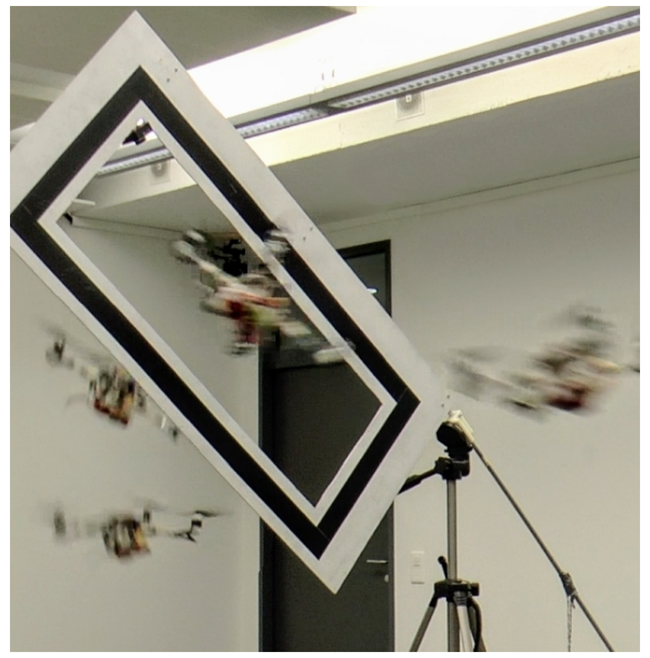
+

Massive computations



Real-time optimization

Fast real-time requirements



+

Low-cost computing platforms

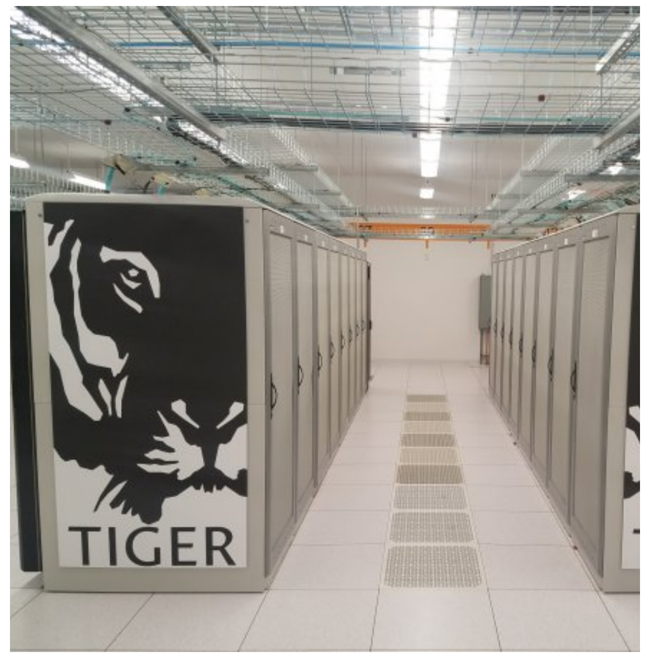
What is happening today?

Huge scale optimization

Massive
data

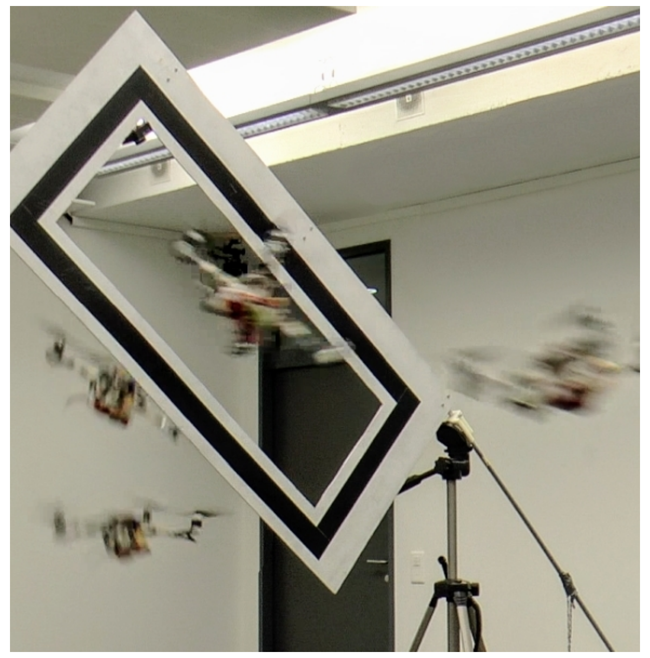
+

Massive
computations



Real-time optimization

Fast real-time
requirements



+

Low-cost computing
platforms

**Renewed interest in
old methods (70s)**

- Subgradient methods
- Proximal algorithms

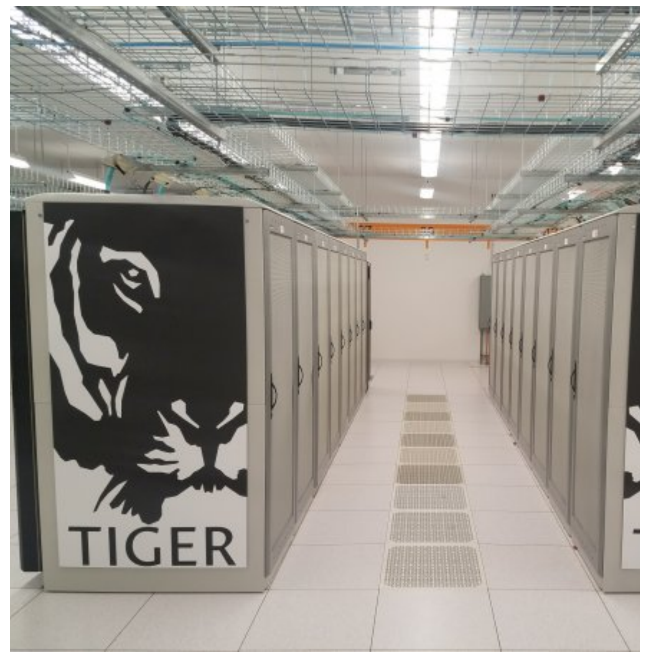
What is happening today?

Huge scale optimization

Massive data

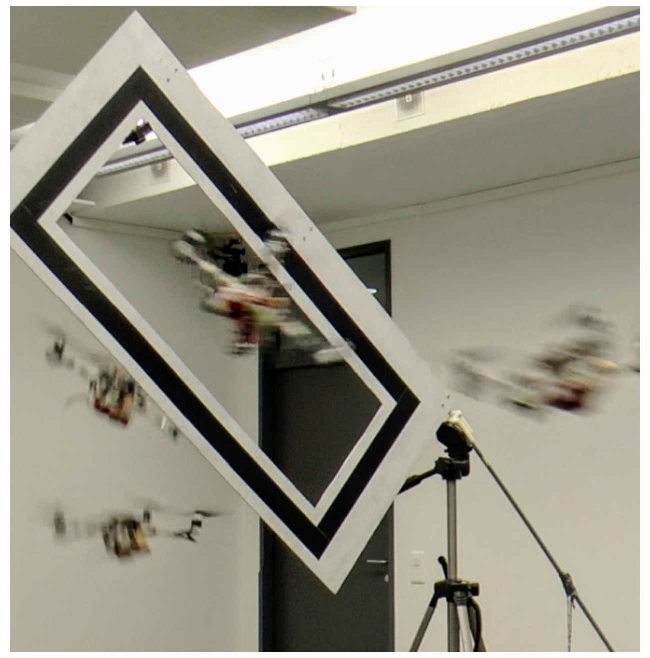
+

Massive computations



Real-time optimization

Fast real-time requirements



+

Low-cost computing platforms

Renewed interest in old methods (70s)

- Subgradient methods
- Proximal algorithms

$\longrightarrow$

- Cheap iterations
- Simple implementation

Course Content

1. **CO-1:** Linear Programming, Geometric Interpretation, Simplex Method, Duality, primal dual method, Interior point methods, Ellipsoidal methods, Computational Issues.
2. **CO-2:** Integer programming, LP relaxation, Examples from combinatorial optimization. Shortest paths, network flows and matchings.
3. **CO-3:** Convex sets and functions. Need for constrained methods in solving constrained problems.
4. **CO-4:** Unconstrained optimization, Optimality conditions, Gradient Descent, Newton Method, Quasi- Newton Methods, Trust Region Methods. Conjugate Gradient Methods. Least Squares Problems.
5. **CO-5:** Constrained Optimization, Optimality Conditions and Duality. Convex Programming Problem. Quadratic Programming. Dual Methods, Penalty and Barrier Methods, Interior Point Methods.

Books

Preferred Text Books

1. S. Boyd and L Vandenberghe. **Convex Optimization**, Cambridge University Press (Online Copy available at: hdp://www.stanford.edu/˜boyd/cvxbook/).
2. L Vandenberghe. **Lecture Notes for Applied Numerical Computing**, (Online available at: hdp://www.ee.ucla.edu/˜vandenbe/103/reader.pdf).
3. Edwin K. P. Chong, Stanislaw H. Żak. **Introduction to Optimization**, Fourth Edition, Wiley-Interscience Series in Discrete Mathematics and Optimization, John Wiley & Sons.
4. Amir Beck. **Introduction to Nonlinear Optimization**, MOS-SIAM Series on Optimization, 2014.

Reference Books

1. C H Papadimitriou and K Steiglitz. **Combinatorial Optimization: Algorithms and Complexity** (Most of First seven chapters), Dover.
2. D Bertisimas and J N Tsitsiklis. **Introduction to Linear Optimization**, Athena Scientific.
3. J Matousek and B. Gartner. **Understanding and Using Linear Programming**, Springer, 2007.
4. S. S. Rao. **Engineering Optimization: Theory and Practice**. New Age Publication, 2006.
5. Dimitri P Bertsekas. **Convex Optimization Theory**. Universities Press, 2010.

Online Video Lectures

1. Numerical Optimization by Professor Shirish Shevade (CSA, IISc Bangalore). (link: https://nptel.ac.in/courses/106108056)

Grading Plan

Evaluation	Weightage
Quiz 1	8%
Quiz 2	8%
Mid-Sem	16%
End-Sem Exam	32%
Assignments (3 Programming)	24% (8% each)
Scribe[1]	6%
Attendance in Tutorials[2]	6%

[1] Each lecture note will be scribed in latex by two separate teams. Each team will have only 2 students. These two teams will give one round of feedback to each other and then finally evaluate. These evaluation scores will be finally considered in overall rating of the scribe. Teams will be announced 1 day before the lecture. Latex template for scribing will be provided by the TAs.

[2] Tutorial Timing: 12.40 to 13.40 Hrs on Wednesdays.

Course TAs

1. **Tejas Kiran Chaudhari** (tejas.chaudhari@research.iiit.ac.in)
2. **M Elamparithy** (elamparithy.m@research.iiit.ac.in)
3. **Vikram A Rao** (vikram.rao@students.iiit.ac.in)
4. **Anurag Gupta** (anurag.g@students.iiit.ac.in)

Software (open-source)

CVXPY

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \end{array} \longrightarrow$$

```python
x = cp.Variable(n)
prob = cp.Problem(
    cp.Minimize(c.T@x),
    [A @ x <= b]
)
prob.solve()
print("The optimal value is", prob.value)
print("The solution x is", x.value)
```

Python

Numerical computations on numpy and scipy.

Learning goals

- **Model** your favorite decision-making problems as mathematical optimization problems.

- **Apply** the most appropriate optimization tools when faced with a concrete problem.

- **Implement** optimization algorithms and prove their convergence.

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Optimization problem

minimize $\quad \mathcal{L}(\theta) + \lambda r(\theta) = \sum_{i=1}^n \ell(\hat{y}^i, y^i) + \lambda r(\theta)$

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Optimization problem

Loss

minimize $\quad \boxed{\mathcal{L}(\theta)} + \lambda r(\theta) = \sum_{i=1}^n \ell(\hat{y}^i, y^i) + \lambda r(\theta)$

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Optimization problem

minimize $\underbrace{\mathcal{L}(\theta)}_{\text{Loss}} + \lambda \underbrace{r(\theta)}_{\text{Regularizer}} = \sum_{i=1}^n \ell(\hat{y}^i, y^i) + \lambda r(\theta)$

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Optimization problem

minimize $\quad \underbrace{\mathcal{L}(\theta)}_{\text{Loss}} + \lambda \underbrace{r(\theta)}_{\text{Regularizer}} = \sum_{i=1}^{n} \ell(\hat{y}^i, y^i) + \lambda r(\theta)$

Large-scale computing

- Parallel
- Distributed

Glance into modern optimization

Huge scale optimization

Dataset with
billions of datapoints (x^i, y^i) $\longrightarrow$ **Goal:** Design predictor $\hat{y}^i = g_\theta(x^i)$

Optimization problem

minimize $\underbrace{\mathcal{L}(\theta)}_{\text{Loss}} + \lambda \underbrace{r(\theta)}_{\text{Regularizer}} = \sum_{i=1}^n \ell(\hat{y}^i, y^i) + \lambda r(\theta)$

Large-scale computing

- Parallel
- Distributed

Many examples

- Support vector machines
- Regularized regression
- Neural networks

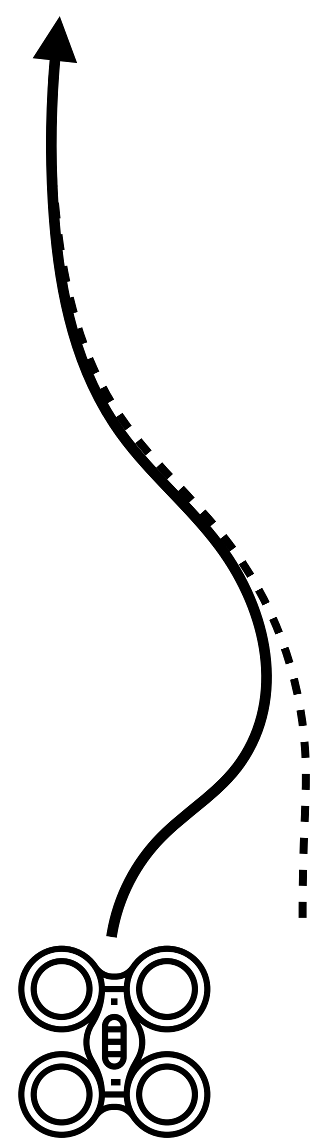
Glance into modern optimization

Real-time optimization

Dynamical system: $x_{t+1} = Ax_t + Bu_t$

$x_t \in \mathbf{R}^n$: state

$u_t \in \mathbf{R}^m$: input



Goal: track trajectory minimize $\sum_t \|x_t - x_t^{\mathrm{des}}\|$

Constraints: inputs $\|u\| \leq U$, states $a \leq x_t \leq b$

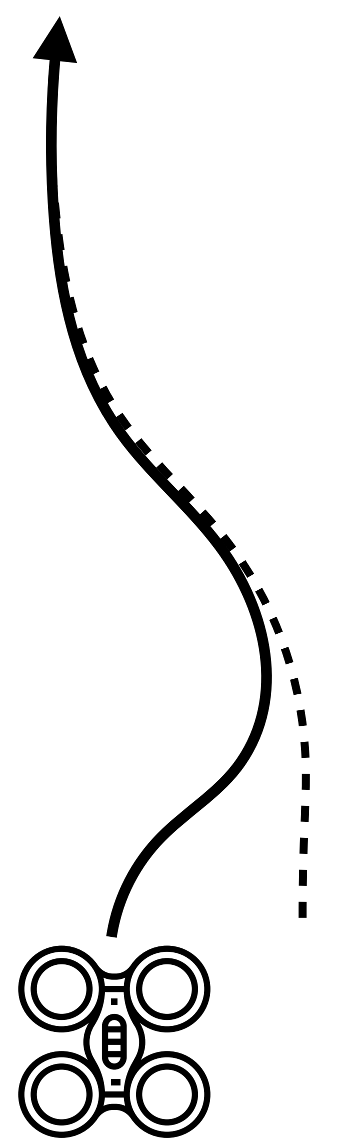
Glance into modern optimization

Real-time optimization

Dynamical system: $x_{t+1} = Ax_t + Bu_t$

$x_t \in \mathbf{R}^n$: state

$u_t \in \mathbf{R}^m$: input



Goal: track trajectory minimize $\sum_t \|x_t - x_t^{\text{des}}\|$

Constraints: inputs $\|u\| \leq U$, states $a \leq x_t \leq b$

1-norm $\longrightarrow$???

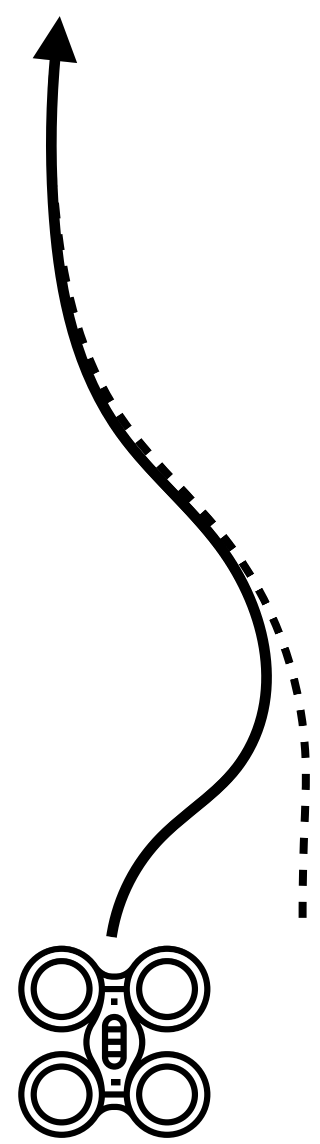
Glance into modern optimization

Real-time optimization

Dynamical system: $x_{t+1} = Ax_t + Bu_t$

$x_t \in \mathbf{R}^n$: state

$u_t \in \mathbf{R}^m$: input



Goal: track trajectory minimize $\sum_t \|x_t - x_t^{\mathrm{des}}\|$

Constraints: inputs $\|u\| \leq U$, states $a \leq x_t \leq b$

1-norm $\longrightarrow$???

∞-norm $\longrightarrow$???

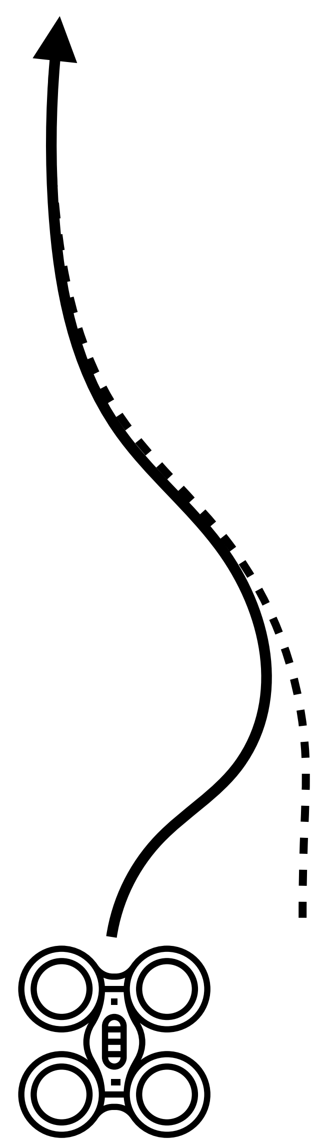
Glance into modern optimization

Real-time optimization

Dynamical system: $x_{t+1} = Ax_t + Bu_t$

$x_t \in \mathbf{R}^n$: state

$u_t \in \mathbf{R}^m$: input



Goal: track trajectory minimize $\sum_t \|x_t - x_t^{\text{des}}\|$

Constraints: inputs $\|u\| \leq U$, states $a \leq x_t \leq b$

1-norm
∞-norm $\longrightarrow$ Linear optimization
(more next lecture…)

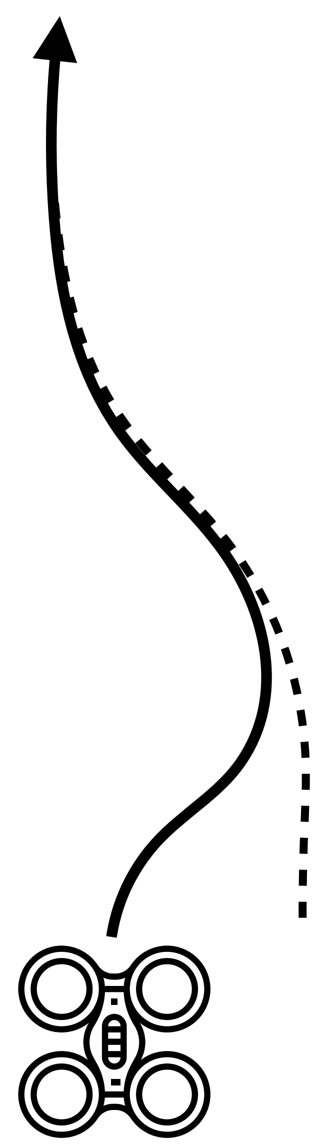
Glance into modern optimization

Real-time optimization

Dynamical system: $x_{t+1} = Ax_t + Bu_t$

$x_t \in \mathbf{R}^n$: state

$u_t \in \mathbf{R}^m$: input



Goal: track trajectory minimize $\sum_t \|x_t - x_t^{\mathrm{des}}\|$

Constraints: inputs $\|u\| \leq U$, states $a \leq x_t \leq b$

1-norm
∞-norm
$\longrightarrow$ Linear optimization
(more next lecture…)

Solve and repeat…..

How fast can we solve these problems?