TTK4135 Optimization and control

To learn:

- Optimization – important concepts and theory
- Formulating an engineering problem into a mathematical optimization problem (modeling for optimization)
- Solving an optimization problem – choosing the right algorithm, using algorithms, some implementation of algorithms
- Applications in control engineering – model predictive control

Instructors:

- Lecturer: Lars Imsland
- Teaching Assistant: Joakim R. Andersen
- 6 Student Assistants (assignments and labs)

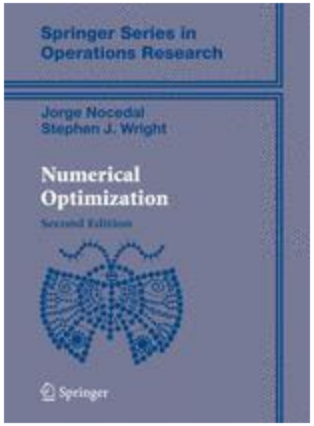
Course information

- All course information is provided through Blackboard
- Course description: http://www.itk.ntnu.no/emner/ttk4135

- Assignment and lab report deadlines are absolute, no extensions except for very special cases
- Helicopter lab must be approved
- At least 7 of 10 assignments must be approved (no extra assignment will be given)
- Pay attention and make sure your assignments get approved
- Please do not copy ("koke") assignments – it will be reported if noticed

- Video lectures from 2014:
  – http://video.adm.ntnu.no/openVideo/serier/52b1a0ef0c035

Course information, cont'd



- I will not cover the complete curriculum in my lectures;
  I will focus on the most important and difficult parts,
  as well as trying to build your understanding and intuition

- You will need to read the book:
  - *Numerical Optimization*, J. Nocedal and S. J Wright, 2nd ed., Springer (ISBN-10: 0-387-30303-0 or ISBN-13: 987-0387-30303-1). Download at https://link.springer.com/book/10.1007%2F978-0-387-40065-5 from campus or through VPN.
  - Errata at Blackboard.
  - I encourage you to buy the book!

- In addition the note *Merging Optimization and Control*, B. Foss and T. A. Heirung, will be used. Will be provided electronically.

- Note on matrix calculus

Course information, cont'd

- Grading
  - Final exam: 80%
  - Lab report (helicopter): 20% (group work).

- Timetable
  - Tuesday lecture: 10:15 – 12:00 in S3
  - Thursday lecture: 12:15 – 14:00 in S6
  - Assignment help sessions: Tuesdays 18:15 – 19:00 in EL6

- TA office hours: TBA

- Exam: June 3, 09:00 – 13:00

- Reference group!

Tentative lecture schedule

	TTK4135 Plan for Spring 2019					
Week no.	**Lectures Tuesday 10:15-12:00 S3**	**Lectures Thursday 12:15-14:00 S6**	**Helicopter project**	**Exercise out (Mon 15:00)**	**Help session Tuesday 18:15-19:00 EL6**	**Exercise in (Wed 23:59)**
2	Lecture 1 Introduction on optimization - N&W Ch.1	Lecture 2 Optimality conditions - N&W Ch. 12.1-12.2		0: Matrix Calculus, 1: KKT		
3	Lecture 3 Optimality conditons and linear algebra - N&W Ch.12.3, 12.5 (12.8, 12.9)	Lecture 4 Linear Programming - N&W Ch.13.1-13.5		2: LP	0, 1, 2	
4	Lecture 5 Linear Programming - N&W Ch.13.1-13.5	Lecture 6 Quadratic programming - N&W Ch.15.3-15.5, 16.1-2,4-5		3: LPQP	2, 3	0, 1
5	Lecture 7 Quadratic Programming - N&W Ch.15.3-15.5, 16.1-2,4-5	Lecture 8 Open loop dynamic optimization - MPC note Ch.3-3.2	Helicopter Lab week	4: QP	3, 4	2
6	Lecture 9 Model predictive control - MPC note Ch.3.3-4.2.1	Lecture 10 Model predictive control - MPC note Ch.4.2.2-4.3.1	Helicopter Lab week	5: OLMPC	4, 5	3
7	Lecture 11 Linear quadratic control - MPC note Ch.4.3.2-4.4	Lecture 12 Linear quadratic control - MPC note repetition and 4.6	Helicopter Lab week	6: MPCLQR	5	4
8	No lecture	No lecture	Helicopter Lab week		5, 6	
9	Lecture 13 Unconstrained optimization - N&W Ch.2.1-2.2	Lecture 14 Line search methods - N&W Ch.3-3.1, 3.4, 3.5	Helicopter Lab week	7: RICATTI	6, 7	5
10	Lecture 15 Quasi Newton methods - N&W Ch.6-6.1, 8-8.1	Lecture 16 Derivative free optimization - Ch.9, 9.5	Helicopter Lab week	8: UNCON	7, 8	6
11	Lecture 17 Sequential quadratic programming (SQP) - N&W Ch.18-18.2, 11-11.1	Lecture 18 Sequential quadratic programming (SQP) - Ch.18.3, 15.4	Helicopter Lab week	9: OPTALG	8, 9	7
12	Lecture 19 Sequential quadratic programming (SQP) - Ch.18.4, 18.8, 15.5	Lecture 20 TBD		10: SQP	9, 10	8
13					10	9
14	No lecture	No lecture				10
15	Excursion week					
16	Easter vacation					
17	Easter vacation					

Expected background

- Linear algebra and real analysis
  - quick recap next Tuesday
- Some numerical analysis (Newton's method)
- Basic control theory:
  - TTK4105 Control engineering
  - Advantage: TTK4115 Linear system theory

Lecture 1:

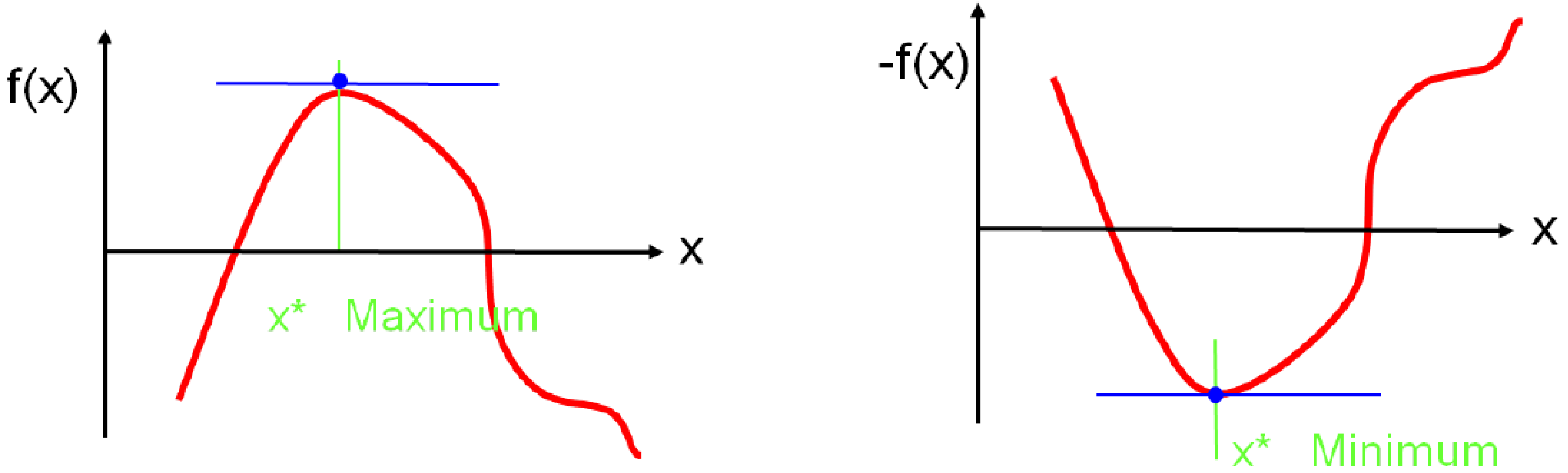
- About optimization, what is it and why do we need it?
- Formulating an optimization problem: From an "engineering problem" to a mathematical description
- Definition of important terms
  - Convexity and non-convexity
  - Global vs. local solution
  - Constrained vs. unconstrained problems
  - Feasible set
- Reference: Chapter 1 in Nocedal and Wright (N&W)

About optimization

You have met optimization in earlier courses

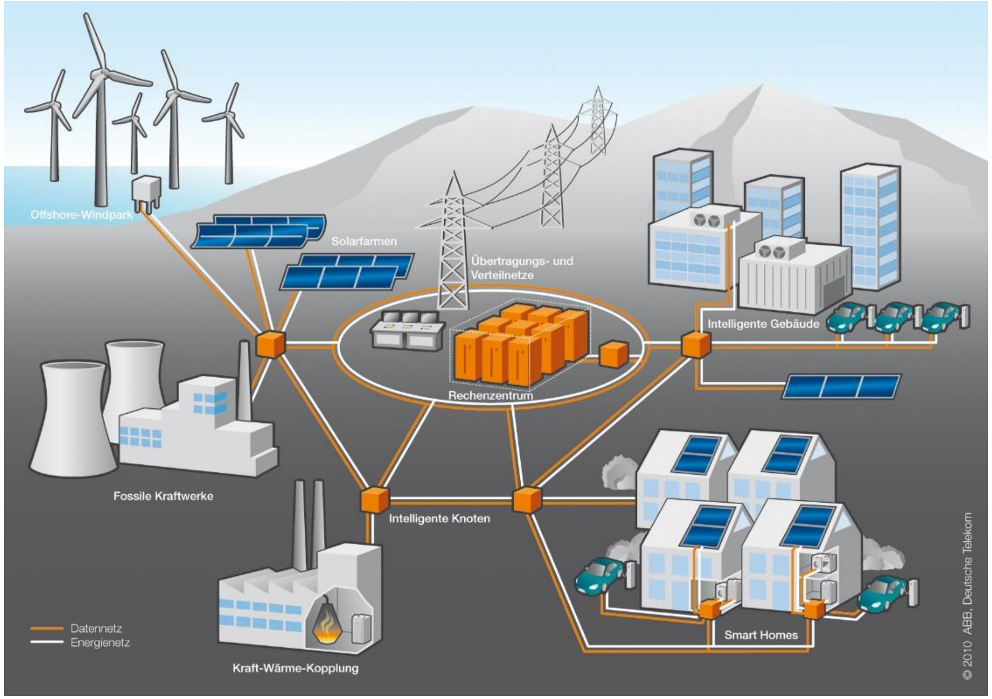
- Finding max and min of a function (Calc 1)
- The Lagrange Method (Calc 2)
- Algorithms course (shortest path, dynamic programming, max flow, traveling salesman etc.)
- Least squares (data fitting)

Minimization or maximization?



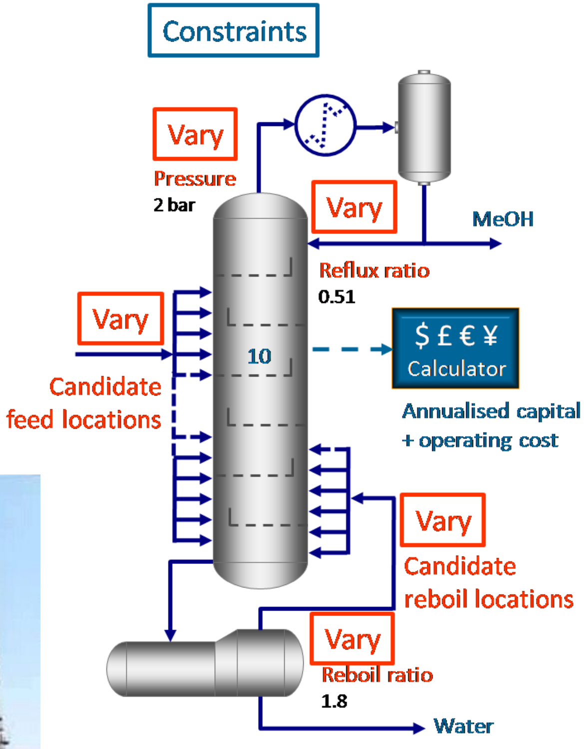
- Convention this course: minimization!

Electric power production

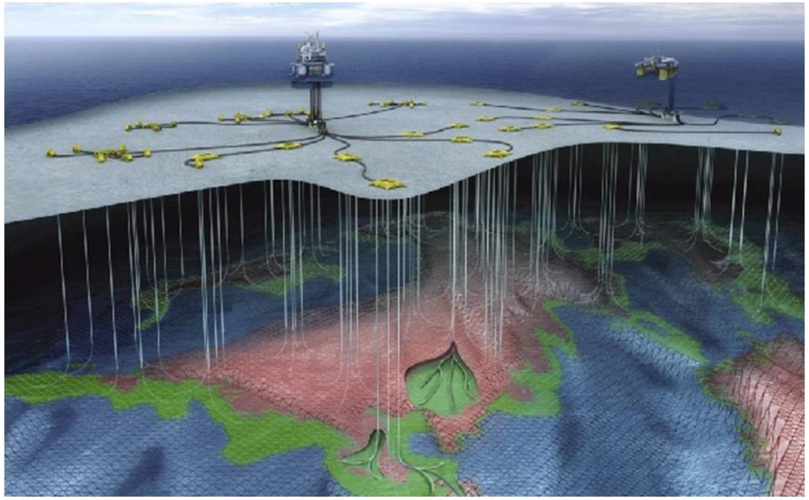


Planning and production of electric power

Process control

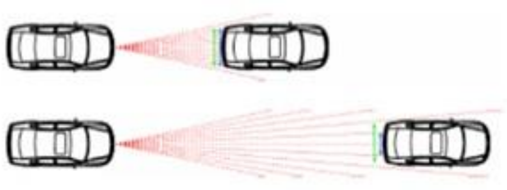


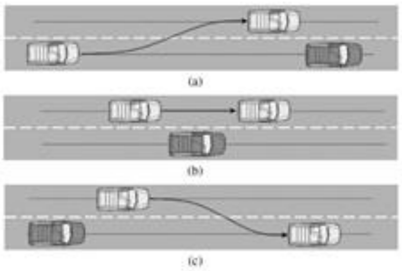
Optimizing operations of chemical processes



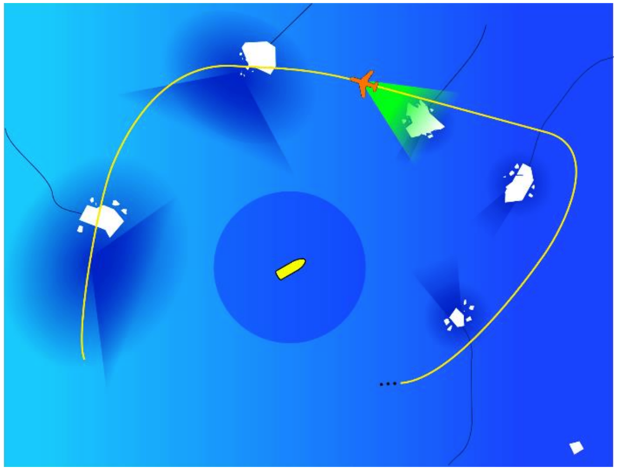
Production optimization in oil and gas systems

Motion control





Collision avoidance



Tracking ice-field movements

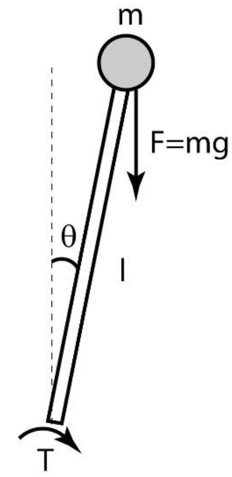
Scheduling and operations research

Train scheduling

(and scheduling of lecture halls)

Stock exchange

Example: Optimization in control

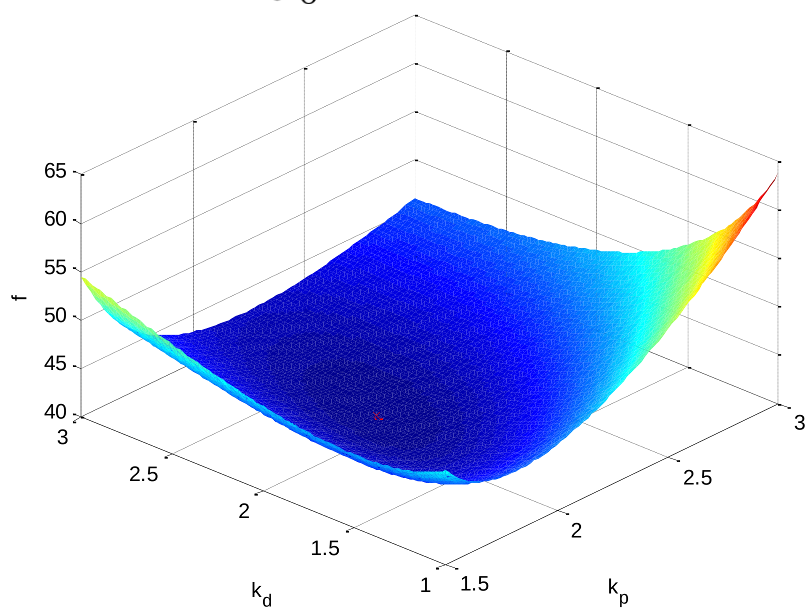


- Model of inverted pendulum: $\ddot{\theta} - \frac{g}{l}\sin\theta = \tau$
- Control problem: Find gains in “PD”-controller

$$\tau = -k_p\theta - k_d\dot{\theta}$$

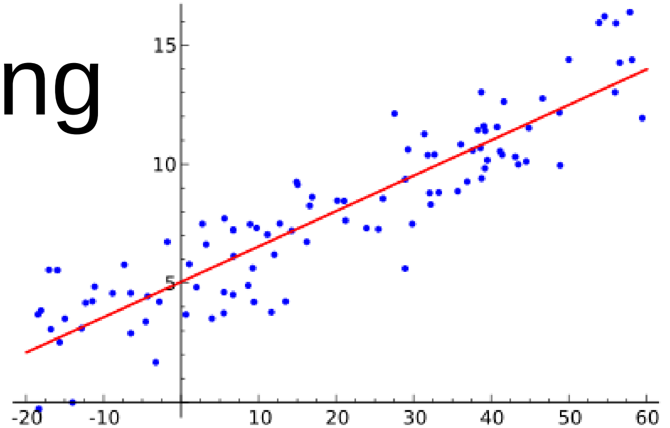
- that gives best “control performance:

$$f(k_p, k_d) = \int_0^\infty \theta^2 + \dot{\theta}^2 + \alpha\tau^2 \,\mathrm{d}t$$



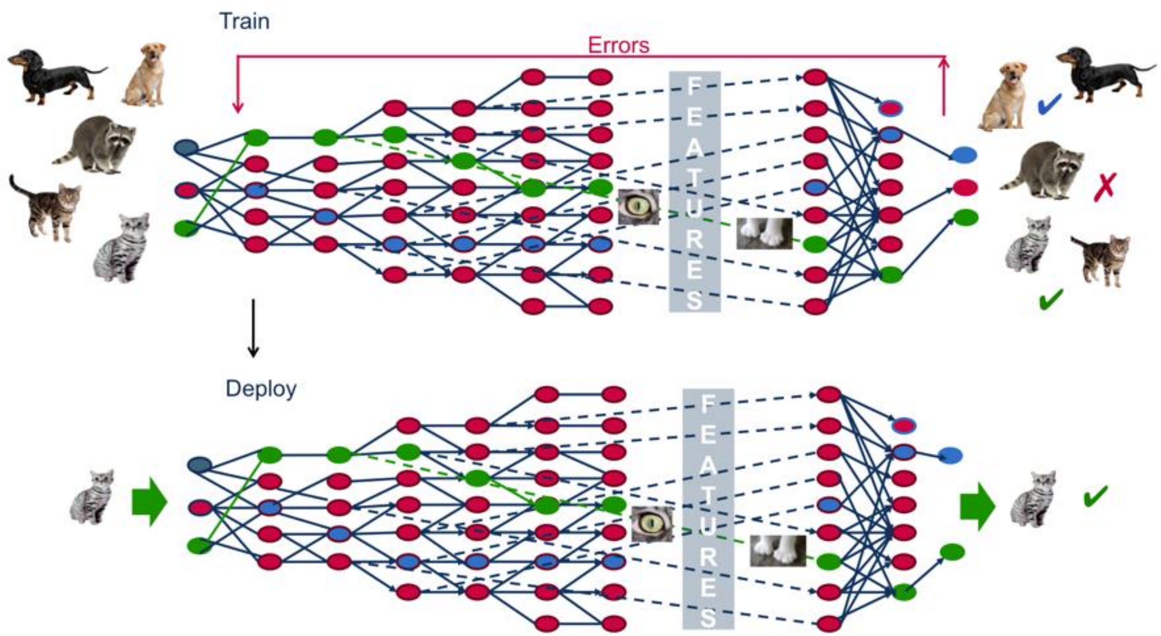
More on these types of optimization problems (“optimal control”) later in the course.

Example: Machine Learning



Wikipedia.org

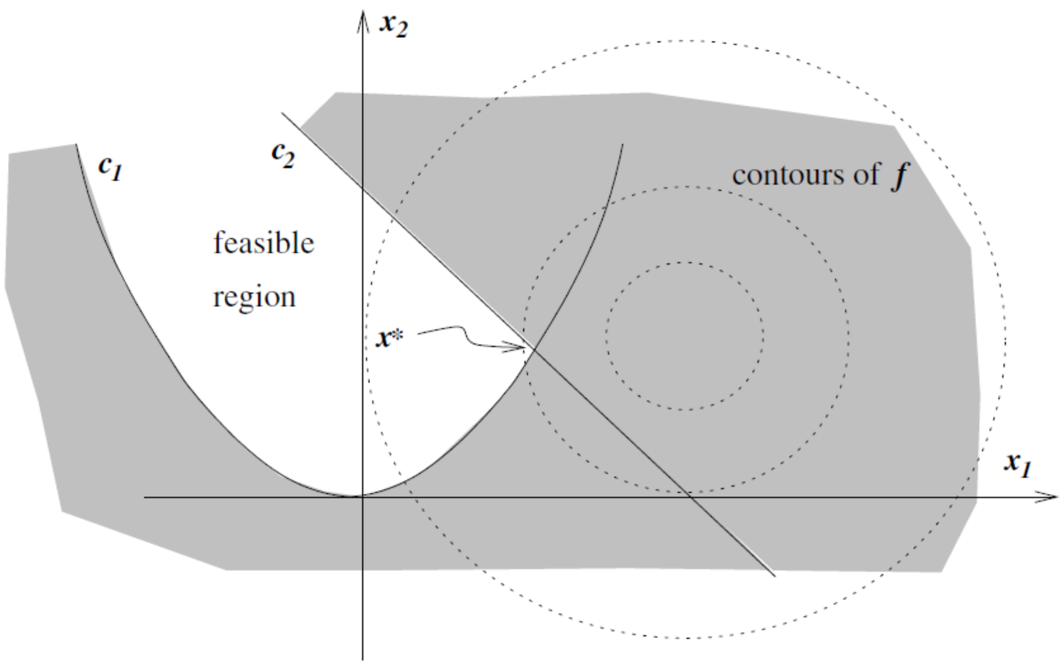
- Learn, and make predictions, from data
- Linear regression is the most basic ML algorithm, solved using optimization
  - “least squares”, Ch. 10, N&W
- In a similar fashion: ML, neural networks, deep learning etc. are “trained” (in general) using “gradient descent” algorithms
  - topic of Ch. 2-10, N&W



General optimization problem

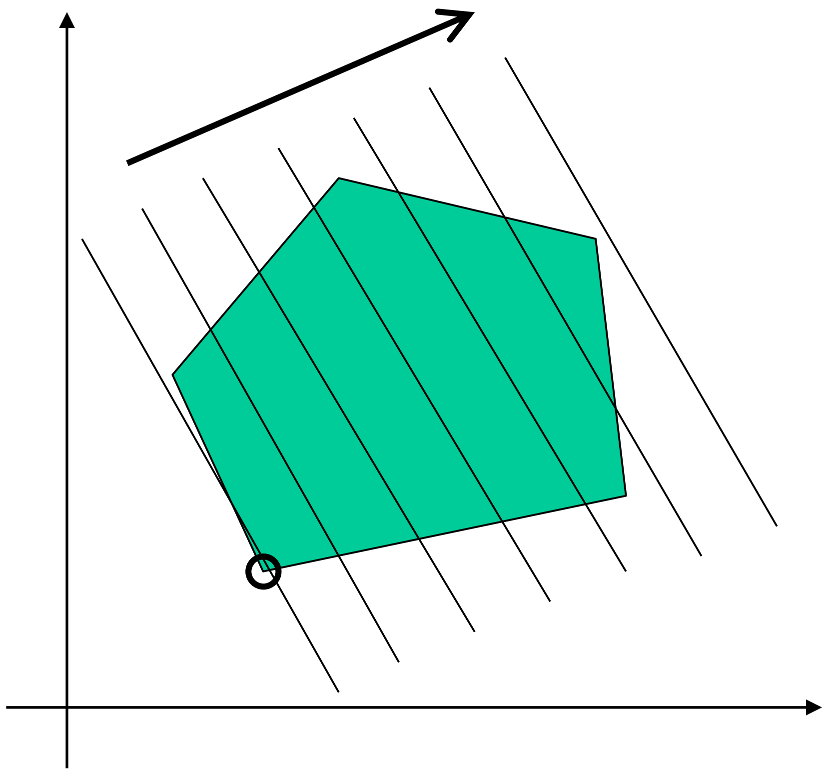
$$\min_{x \in \mathbb{R}^n} f(x) \qquad \text{subject to} \quad \begin{array}{ll} c_i(x) = 0, & i \in \mathcal{E}, \\ c_i(x) \geq 0, & i \in \mathcal{I}. \end{array}$$

- Example: $\min\ (x_1 - 2)^2 + (x_2 - 1)^2 \qquad \text{subject to} \quad \begin{array}{l} x_1^2 - x_2 \leq 0, \\ x_1 + x_2 \leq 2. \end{array}$

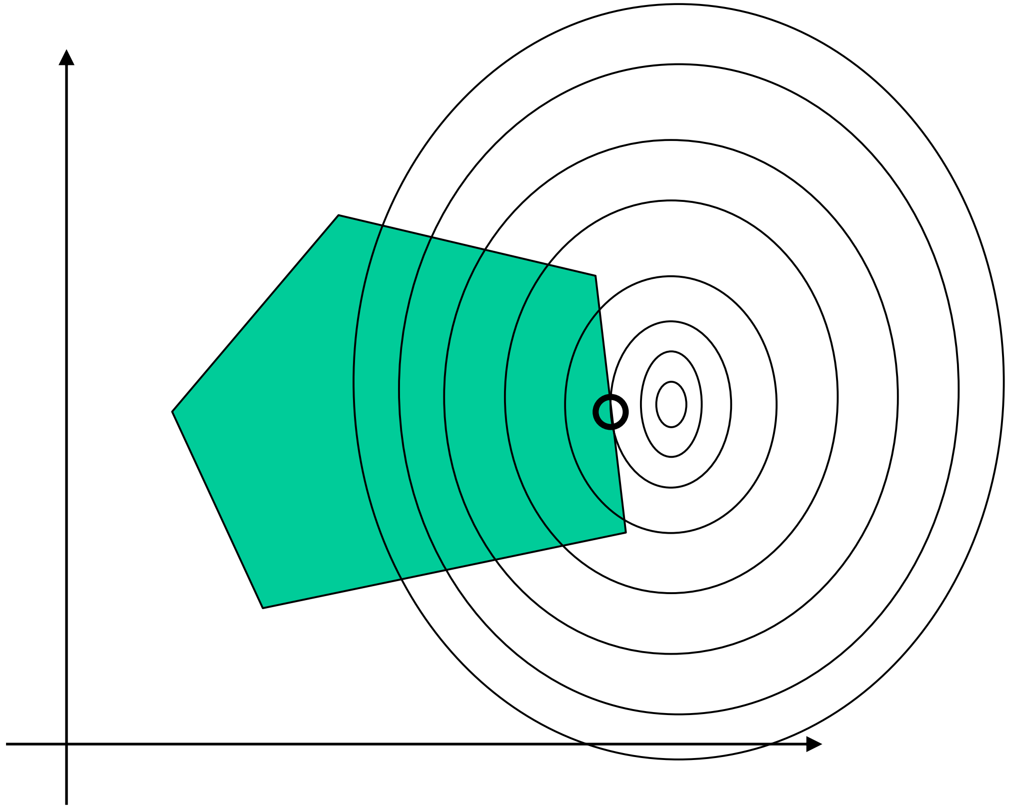


- What if we add equality-constraint $x_1 = 0$?

Linear programming



Quadratic programming



LP example: Farming

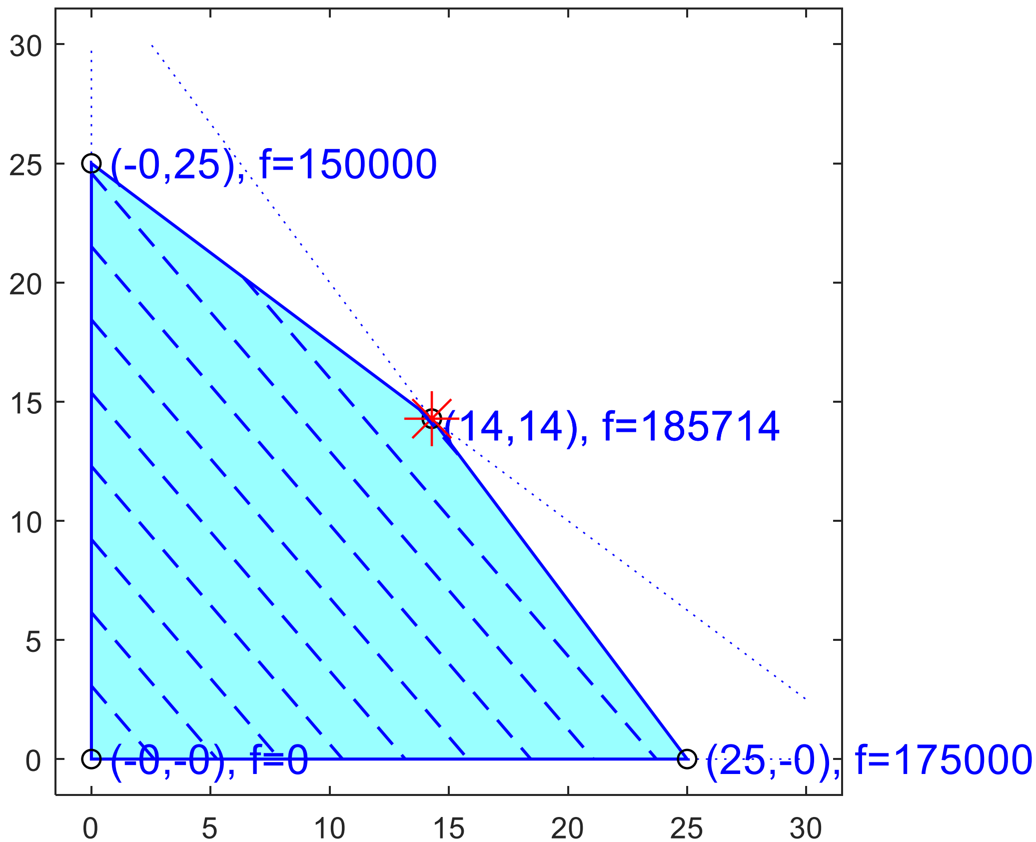
- A farmer wants to grow apples (A) and bananas (B)
- He has a field of size 100 000 m^2
- Growing 1 tonne of A requires an area of 4 000 m^2, growing 1 tonne of B requires an area of 3 000 m^2
- A requires 60 kg fertilizer per tonne grown, B requires 80 kg fertilizer per tonne grown
- The profit for A is 7000 per tonne (including fertilizer cost), the profit for B is 6000 per tonne (including fertilizer cost)
- The farmer can legally use up to 2000 kg of fertilizer

- He wants to maximize his profits

Farming example: Geometric interpretation and solution

$$\max_{x_1,\, x_2} \quad 7000x_1 + 6000x_2$$

$$\begin{aligned} \text{subject to:} \quad 4000x_1 + 3000x_2 &\leq 100000 \\ 60x_1 + 80x_2 &\leq 2000 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$



Constrained optimization

(Mathematical programming)

Convex		Non-convex	
Linear	Non-linear		Solvers
Linear programming			Simplex, interior point
	Quadratic programming		Active set, interior point
	Semi-definite programming (LMI, second-order cone p.)		Interior point
		Non-linear programming	SQP
		Mixed-integer programming	

This course