AARHUS
UNIVERSITET

Optimization
in
ICT and Physical Systems

Henrik Karstoft
@
Aarhus University, Department of Engineering

Optimization

Course outline, formal stuff

- Prerequisite
- Lectures
- Homework
- Textbook, Homepage and CampusNet, http://kurser.iha.dk/ee-ict-master/tiopti/

Course goal

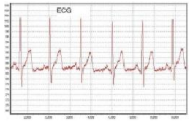
- increased mathematical skills,
- a broad knowledge of the classical and modern mathematical optimization techniques

Optimization

Practical optimization is the art and science of allocating scarce resources to the best possible effect.

Optimization problems

- How should the transistors and other devices be laid out in a new computer chip so that the layout takes up the least area?
- What is the smallest number of warehouses, and where should they be located so that the maximum travel time from any retail sales outlet to the closest warehouse is less than 6 hours?
- How should telephone calls be routed between two cities to permit the maximum number of simultaneous calls?
- How should the computational problem be solved must efficiently, when more than 1 processor core is available?
- How do we produce the best a fit of ecg complex using Gaussians?



Optimization

Many other fields of applications:

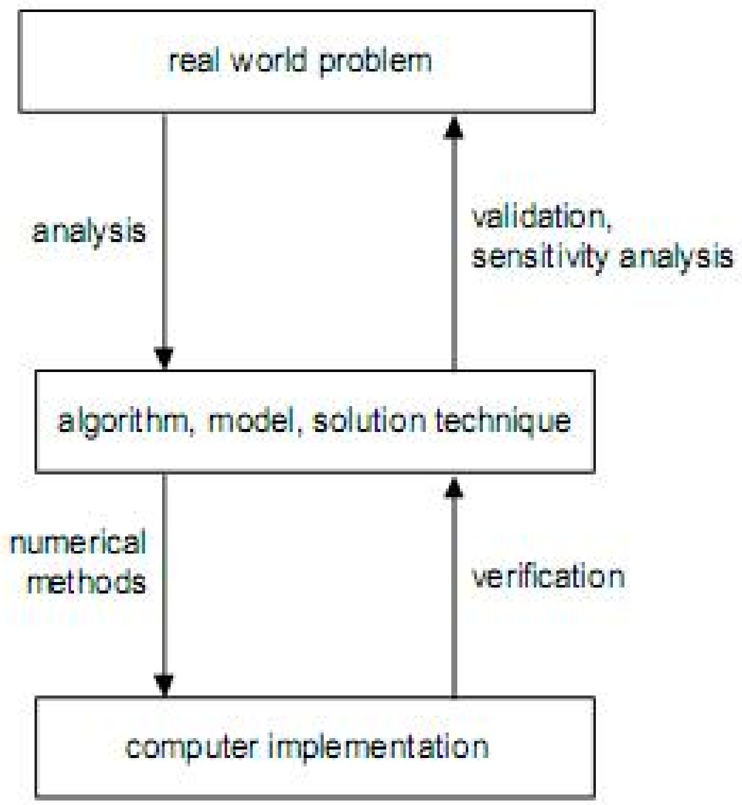
- Computer Science
- Economics
- Circuit design
- Control
- Signal processing
- Communications
- Design (automotive, aerospace, biomechanical
- …

Henrik Karstoft
Department of Engineering

Optimization

The optimization cycle

1. Real world problem
2. Algorithm, model, solution technique
3. Computer implementation



Optimization

Mathematical Optimization

$$\text{minimize } f_0(x)$$
$$\text{subject to } f_i(x) \leq b_i, \text{ where } i = 1, \dots, m$$

- $x = (x_1, \dots, x_n)$, optimization variables, or decision variables
- $f_0\colon\ R^n \to R$, objective function
- $f_i(x)\ R^n \to R$, where $i = 1, \dots, m$: constraint functions
- x^* **optimal solution**, smallest value of f_0 among $x = (x_1, \dots, x_n)$, satisfying the constraint

Optimization

Solving optimization problems in general

- Very difficult to solve
- Methods involve compromise, e.g. very long computational time, or non optimal solution

Solving optimization problems, special cases – exceptions

- least-square problems
- linear programming problems
- unconstrained problems
- convex optimization

Optimization

Least Square

$$\text{minimize } |Ax - b|^2$$

- analytic solution : $x^* = (A^T A)^{-1} A^T b$
- reliable and efficient algorithms and software
- computational time proportional to $n^2 m$, where $A \in R^{m \times n}$, in some case less than this
- mature technology

Using least-square

- datafitting
- ...

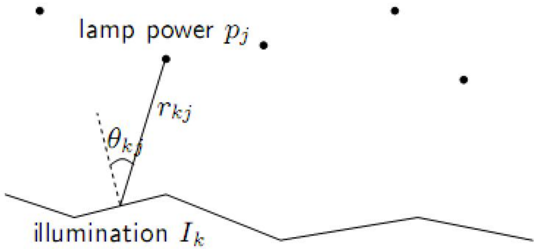
Example using least square

- observe the following data (x_i, y_i), where $i = 1, \ldots, m$.
- Find the best fit of a straight line $y = ax + b$

Optimization

Example using Least Square

m lamps illuminating n (small, flat) patches



intensity I_k at patch k depends linearly on lamp powers p_j:

$$I_k = \sum_{j=1}^{m} a_{kj} p_j, \qquad a_{kj} = r_{kj}^{-2} \max\{\cos\theta_{kj}, 0\}$$

problem: achieve desired illumination I_{des} with bounded lamp powers

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |\log I_k - \log I_{\text{des}}| \\ \text{subject to} & 0 \le p_j \le p_{\max}, \quad j = 1, \ldots, m \end{array}$$

Henrik Karstoft
Department of Engineering

Optimization

how to solve?

1. use uniform power: $p_j = p$, vary p
2. use least-squares:

$$\text{minimize} \quad \sum_{k=1}^{n} (I_k - I_{\mathrm{des}})^2$$

   round p_j if $p_j > p_{\mathrm{max}}$ or $p_j < 0$

3. use weighted least-squares:

$$\text{minimize} \quad \sum_{k=1}^{n} (I_k - I_{\mathrm{des}})^2 + \sum_{j=1}^{m} w_j (p_j - p_{\mathrm{max}}/2)^2$$

   iteratively adjust weights w_j until $0 \leq p_j \leq p_{\mathrm{max}}$

4. use linear programming:

$$\begin{array}{ll} \text{minimize} & \max_{k=1,\ldots,n} |I_k - I_{\mathrm{des}}| \\ \text{subject to} & 0 \leq p_j \leq p_{\mathrm{max}}, \quad j = 1, \ldots, m \end{array}$$

   which can be solved via linear programming

of course these are approximate (suboptimal) 'solutions'

Optimization

Linear programming

$$\text{minimize } c^T x$$
$$\text{subject to } {a_i}^T x \leq b_i \; i = 1, \ldots, , m$$

- *no* analytic solution
- reliable and efficient algorithms and software
- computational time proportional to $n^2 m$, where $A \in R^{m \times n}$, in some case less than this
- mature technology

Using linear programming

- Computer science
- Economics

Henrik Karstoft
Department of Engineering

Optimization

Example linear programming

7. You are a radiologist planning a program of radiation therapy for a cancer patient. Your beam treatment machine can pass two beams of ionizing radiation through the patient's body. Your mission is to select the radiation dose from the two beams so as to minimize damage to healthy tissue, while subjecting the tumor to a large enough dose to kill cancer cells, and without overdosing sensitive tissues. Here are the data you need to make your decision

- The radiation dose received by various tissues as a result of a 1-sec exposure from each beam is listed in the table below

Area	Beam 1 dose /krad	Beam 2 dose /krad
Healthy tissue	0.7	0.3
Sensitive tissues	0.5	0.2
Tumor center	3.0	0.6
Tumor edge	1.0	2.0

- You goal is to minimize the dose received by healthy tissue
- The tumor center must be exposed to at least 6 kilorads of radiation
- The entire tumor must be exposed to at least 4 kilorads of radiation
- Sensitive tissues cannot be exposed to more than 1.5 kilorads of radiation

Optimization

Matrix games

EXAMPLE 1 Each player has a supply of pennies, nickels, and dimes. At a given signal, both players display (or "play") one coin. If the displayed coins are not the same, then the player showing the higher-valued coin gets to keep both. If they are both pennies or both nickels, then player C keeps both; but if they are both dimes, then player R keeps them. Construct a payoff matrix, using p for display of a penny, n for a nickel, and d for a dime.

R plays safe:

Examine each row for the minimal value, now choose row where this value is max

C plays safe:

Examine each column for the maximal value, now choose column where this value is min

Optimization

Matrix games

DEFINITION

The number v_R, defined by

$$v_R = \max_{\mathbf{x}\in X} v(\mathbf{x}) = \max_{\mathbf{x}\in X} \min_{\mathbf{y}\in Y} E(\mathbf{x},\mathbf{y}) = \max_{\mathbf{x}\in X} \min_{j} \mathbf{x}\cdot\mathbf{a}_j$$

with the notation as described above, is called the **value of the game to row player** R. A strategy $\hat{\mathbf{x}}$ for R is called **optimal** if $v(\hat{\mathbf{x}}) = v_R$.

THEOREM 1

In any matrix game, $v_R \leq v_C$.

THEOREM 2

Minimax Theorem

In any matrix game, $v_R = v_C$. That is,

$$\max_{\mathbf{x}\in X} \min_{\mathbf{y}\in Y} E(\mathbf{x},\mathbf{y}) = \min_{\mathbf{y}\in Y} \max_{\mathbf{x}\in X} E(\mathbf{x},\mathbf{y})$$

Optimization

Matrix games

DEFINITION

The common value $v = v_R = v_C$ is called the **value of the game**. Any pair of optimal strategies $(\hat{\mathbf{x}}, \hat{\mathbf{y}})$ is called a **solution** to the game.

THEOREM 3

Fundamental Theorem for Matrix Games

In any matrix game, there are always optimal strategies. That is, every matrix game has a solution.

Optimization

Matrix games

EXAMPLE 4 Consider the game whose payoff matrix is

$$A = \begin{bmatrix} 1 & 5 & 3 & 6 \\ 4 & 0 & 1 & 2 \end{bmatrix}$$

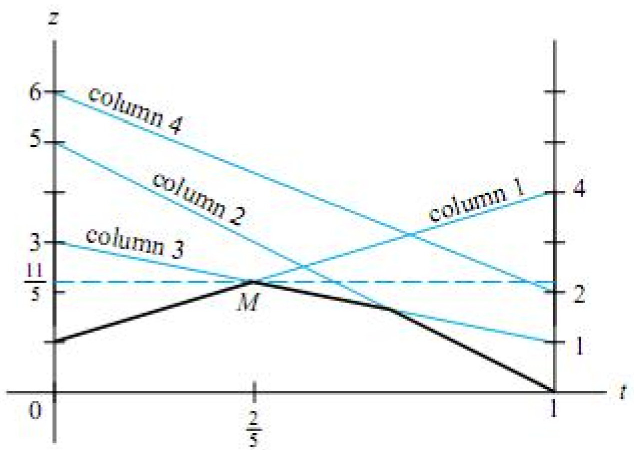


FIGURE 2

Optimization

Matrix games

THEOREM 4 Let $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ be optimal strategies for an $m \times n$ matrix game whose value is v, and suppose that

$$\hat{\mathbf{x}} = \hat{x}_1\mathbf{e}_1 + \cdots + \hat{x}_m\mathbf{e}_m \quad \text{in } \mathbb{R}^m \tag{5}$$

Then $\hat{\mathbf{y}}$ is a convex combination of the pure strategies $\mathbf{e}_j$ in $\mathbb{R}^n$ for which $E(\hat{\mathbf{x}}, \mathbf{e}_j) = v$. In addition, $\hat{\mathbf{y}}$ satisfies the equation

$$E(\mathbf{e}_i, \hat{\mathbf{y}}) = v \tag{6}$$

for each i such that $\hat{x}_i \neq 0$.

Optimization

Matrix games (Extra)

When two animals play hawk, the cost of losing is given by the cost, C, since one of the opponents is going to be injured. Either contestant has a 1/2 probability of losing so the average cost is C/2. Likewise, each hawk gains an average resource value of V/2. The net payoff for each individual hawk would be V/2-C/2 or (V-C)/2. When two animals play dove, there is no cost to the doves. Each dove has a 1/2 probability of winning so they divide the resource and the net payoff is V/2. When a hawk meets a dove, the hawk always wins at no cost, so the net payoff for the hawk is V. Conversely, the dove that engages the hawk gains nothing, but experiences no cost so the net payoff is 0.

Table 8.1 Payoff matrix for Hawk-Dove.		Common type	
		Hawk	**Dove**
Rare type	**Hawk**	$\frac{V-C}{2}$	V
	Dove	0	$\frac{V}{2}$

Optimization

Matrix games (Extra)

A Minimax Optimal strategy for a player is a (possibly randomized) strategy with the best guarantee on its expected gain over strategies the opponent could play in response — i.e., it is the strategy you would want to play if you imagine that your opponent knows you well.

Here is another game: Suppose a kicker is shooting a penalty kick against a goalie who is a bit weaker on one side. Let's say the kicker can kick left or right, the goalie can dive left or right, and the payoff matrix for the kicker (the chance of getting a goal) looks as follows:

Goalie

Kicker	*left*	*right*
left	0	1
right	1	0.5