Introduction to Optimisation

M. Monaci

S.I.D.R.A. PhD Summer School 2023
July 6, 2023

Outline

Introduction to Optimization

What is Operations Research

- Operations Research is a science somewhere between Applied Mathematics and Computer Science
- used to optimize the performances of complex systems
  - growth in the size and complexity of organizations since the advent of the industrial revolution
  - nowadays: applications in logistics, transportation systems, telecommunications, energy management . . .
  - these systems must be handled both from a tactical and from an operational viewpoint
- take decisions $\rightarrow$ decision science
- sometimes referred to as Management Science

History of Operations Research

- The birth of modern OR is dated to the military services early in World War II
- war effort imposed the need to allocate scarce resources in an effective manner
- British and U.S. military created a team of scientists to deal with strategic and tactical problems and do *research on (military) operations*
    - effective methods of using radars, instrumental in winning the Air Battle of Britain
    - maojr role in winning the Battle of the North Atlantic

History of Operations Research

- After the war: interest in applying OR outside the military
- industrial boom caused an increasing complexity and specialization in organizations
- two key factors in the success of OR
  - a substantial progress in improving the techniques of OR; e.g., the *simplex method* (Dantzig, 1947)
  - the computer revolution, allowing arithmetic calculations thousands or even millions of times faster than a human being
- further boost in the 1980s: development of increasingly powerful personal computers and availability of good software packages for doing OR
- Today: millions of individuals have ready access to OR software to routinely solve optimization problems

The nature of Operations Research

- operations research involves “research on operations”
- how to perform a set of operations (activities) into an organization
- “Research” means that the approach should follow the scientific standard
    1. data collection: obtain all relevant information on the problem
    2. identification of the problem: fully understand the problem and the objectives
    3. modellization of the problem: reformulate the problem in a form that is convenient for the analysis
    4. solution of the model: solve the mathematical model
    5. validation of the model: check if the approximation induced by the model is satisfactory

Mathematical models

- Typically, OR specialists describe the system using a *mathematical model*
  - decisions to be taken are modelled using *decision variables*
  - the system is described by means of mathematical relations among the variables
- a model is not the system, but it only represents the system with some approximation
- different models can be used to describe the same system
  - different degree of approximation
  - possibly, some decisions are fixed in advance
  - possibly, some constraints are removed/relaxed
- find the right compromise between
  - the possibility to *solve* the model
  - the *applicability* of the solutions resulting from the model to the real system

Mathematical models

Main elements of a mathematical model

- variables, that correspond to the decisions to be taken; the number of variables will be denoted by n
- the *feasible set* $F \subseteq \Re^n$, that is the set of all possible combinations of the variable values that can be implemented in the real system
- the *objective function* $f : F \rightarrow \Re$, that is used to determine the best solution among all possible ones

The definition of the feasible set and of the objective function includes some constants (coefficients) that are called the *parameters* of the model

Example 1: production planning

- An industry produces n types of products using m different machines
- Each product of each type
  - requires some working time on each machine, and
  - gives a certain reward
- Each machine has a maximum workload

Problem: determine the optimal amount of each product so that the total reward is a maximum

Variables: number of products of each type to be produced
Constraints: maximum workload for each machine

Production planning: numerical example

Parameters

$n = 3$ types of products (A, B, and C) with rewards 4, 5, and 3
$m = 2$ machines, max workoads 240 and 320
working times

	A	B	C
M_1	10	15	7
M_2	20	10	18

Mathematical model

$$\begin{aligned} \max\ & 4x_A + 5x_B + 3x_C \\ \text{subject to}\ & 10x_A + 15x_B + 7x_C \leq 240 \\ & 20x_A + 10x_B + 18x_C \leq 320 \\ & x_A, x_B, x_c \geq 0 \end{aligned}$$

$$\Rightarrow x_A = 12, x_B = 8, x_C = 0$$

Example 2: The Assignment Problem

- n activities to be assigned to n persons
- each person can perform each activity in a certain (known) working time

Problem: find the assignment of activities to persons so that

- each activity is assigned to a person
- each person is assigned one activity
- the total working time is a minimum

Variables: activity assigned to each person
Constraints: each activity must be assigned
each person must be assigned an activity

How hard is the assignment problem?

When $n = 2$ there are only two possible solutions

	A_1	A_2
P_1	20	40
P_2	30	25

- solution 1:
  $P_1 \rightarrow A_1$ and $P_2 \rightarrow A_2$
  cost = 20 + 25 = 45
- solution 2:
  $P_1 \rightarrow A_2$ and $P_2 \rightarrow A_1$
  cost = 40 + 30 = 70

$\Rightarrow$ optimal solution by inspection

How hard is the assignment problem?

When $n = 3$:

	A_1	A_2	A_3
P_1	20	40	30
P_2	30	25	90
P_3	50	70	90

For any assignment of an activity to a person, the residual problem is a 2×2 assignment problem
$\Rightarrow$ number of solutions $3 \times 2 = 6$

For larger n: number of feasible solutions is n!

How hard is the assignment problem?

n	$n!$
5	120
10	3,628,800
20	$2.4\ 10^{18}$

$n = 20$

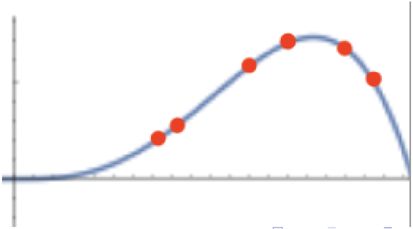
- PC running at 1 GHz (optimistic: 10^9 solutions per second)
- time 2.410^9 seconds $\sim$ 28158 days $\sim$ 77 years

Blue Gene: Supercomputer @IBM

- 182k processors running at 2.3 GHz
- can evaluate all solutions for $n = 20$ in ten hours
- for $n = 24$ it takes 200 years
- for $n = 30$ the exstimated time is 84 billions of years (=5 times the age of the universe)

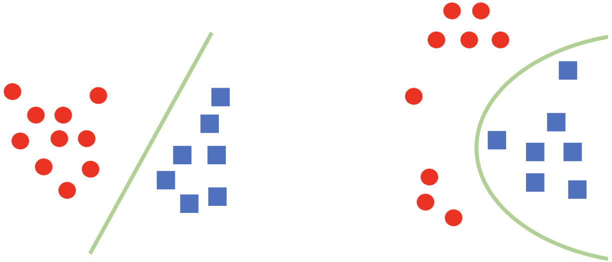
Example 3: Fitting function

- A phenomenon has been observed and measured at a set M of time instants
- y_i is the value measured at time instant t_i
- what is the the analytic expression of a function $f(t)$ such that $f(t_i) = y_i$ (for each sample $i \in M$)?
- if no such function exists, how can we approximate the samples?
- assume function f be defined by a polynomial function depending on some parameters; e.g. $f(x; t) = x_0 + x_1 t + x_2 t^2 + x_3 t^3 + x_4 t^4$
- what is the value for coefficients $x_0, x_1, \ldots, x_4$ so that the resulting function approximates at the best the given samples (t_i, y_i)?



Example 4: Classification

Classification in supervised learning: Given two sets of points, each with a target class, find the hyperplane/function that separates the two sets.



In an n-dimensional space a separating hyperplane is defined by parameters $w_1, w_2, \ldots, w_n, b$ (to be determined).

Definition of a model

Without loss of generality we assume that the objective function has to be minimized

$$z^* = \min_{x \in F} f(x),$$

where z^* denotes the optimal solution value

For solving a maximization problem, the following transformation can be applied

$$\max_{x \in F} f(x) = -\min_{x \in F} g(x),$$

where $g : F \rightarrow \Re, x \rightarrow -f(x)$

Solution of a model

Assuming min form, a model can be described by a pair (F, f)

- *feasible* solution of the model: vector $x \in F$
- *optimal* solution (global minimum, global optimum) of the model: vector $x \in F$ such that

$$f(y) \geq f(x) \quad \forall y \in F$$

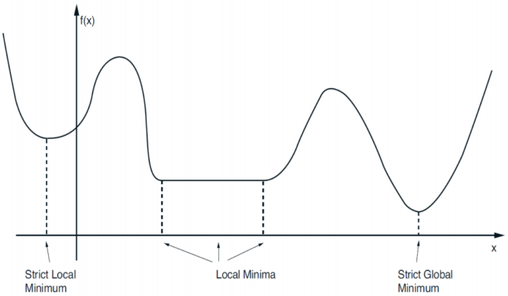
Solution of a model

In some cases, the determination of an optimal solution may be very challenging in practice

- *local optimal* solution: vector $x' \in F$ such that

$$f(y) \geq f(x') \quad \forall y \in F \cap B(x', \rho)$$

  where $B(x', \rho) = \{y \in \Re^n : ||y - x'|| \leq \rho\}$ is a ball centered in x' with some positive radius ρ.
- i.e., x' is a local optimum if $\exists \rho > 0$ s.t. x' is a global optimum over $B(x', \rho) \cap F$



Typical assumptions

- There is a single objective function to be optimized
  - When multiple conflicting objectives are given
    hierarchical definition of the objectives
    multi-objective optimization
- The exact value of all parameters is known in advance
  - uncertain data in real applications
    stochastic optimization
    robust optimization
- All decisions have to be taken at the same time
  - some strategic decisions to be taken immediately,
  - while some other recourse decisions that can be postponed at a second time, e.g., when uncertainty materializes

Classification of the models

NLP: Nonlinear Programming

- most general form

$$\min f(x) \quad x \in F$$

- constraints may be imposed as equations and inequalities

$$
\begin{aligned}
(P) \qquad \min f(x) & \\
x \in \Re^n & \\
g_i(x) \leq 0 & \qquad i \in I \\
h_j(x) = 0 & \qquad j \in E
\end{aligned}
$$

- extremely hard from a practical point of view
- only approximate solutions are required (and can be computed)

Classification of the models

CP: Convex Programming

- problems of the form

$$\min f(x) \quad x \in F$$

  where

  - the feasible region F is a convex set
  - the objective function f is a convex function

- specific exact algorithms have been proposed in the literature

Classification of the models

LP: Linear Programming

- special case of CPs in which
  - the feasible region F is defined by linear equations and inequalities
  - the objective function f is a linear function
- Matricial form

$$\begin{aligned} \min c^T x \\ A x = b \\ x \geq 0 \end{aligned}$$

- Efficient solution using the simplex algorithm

Classification of the models

ILP: Integer Linear Programming

- an ILP is an LP with the additional constraint imposiing integrality of the variables
- $$\begin{aligned} \min c^T x & \\ Ax = b & \\ x \geq 0 & \\ x & \quad \text{integer} \end{aligned}$$
- integrality is a non linear constraint

$$x_j \text{ integer} \quad \leftrightarrow \quad \sin(\pi x_j) = 0$$

- however: nonlinearity only in integrality constraints $\rightarrow$ specific approaches for solving this class of problems

Classification of the models

MILP: Mixed Integer Linear Programming

- Generalization of ILPs in which only a subset J of the variables are required to be integer
- $$\begin{aligned} \min c^T x & \\ Ax = b & \\ x \geq 0 & \\ x_j & \quad \text{integer} \quad j \in J \end{aligned}$$
- if $J = \emptyset \Rightarrow$ LP
- if $J = \{1, \ldots, n\} \Rightarrow$ ILP

Nonlinear optimization: optimality conditions

Unconstrained Optimization

General optimization problem $\min f(x), x \in F$

Unconstrained Optimization

- special case arising when $F = \Re^n$
- $(P) \qquad \min f(x), x \in \Re^n$
- assumption: function f is smooth, i.e., its gradient can be computed in every point

Unconstrained Optimization

Main idea

- Check if a given point $\overline{x} \in \Re^n$ is a local minimum
- local optimality requires evaluating function f in a neighborhood of $\overline{x}$
- for points x that are "close" to $\overline{x}$, one can replace function f with its first-order Taylor approximation

$$f(x) = f(\overline{x}) + \nabla f(\overline{x})^T (x - \overline{x}) + R_1(\overline{x}, |x - \overline{x}|)$$

where $\lim_{x \to \overline{x}} \frac{R_1(\overline{x}, |x-\overline{x}|)}{|x-\overline{x}|} \to 0$

Necessary condition

Descendant directions

Definition A vector $d \in \Re^n$ is a *descendant direction* for function f in $\overline{x}$ if $\exists \delta > 0 : f(\overline{x} + \alpha d) < f(\overline{x}) \quad \forall \alpha \in (0, \delta)$.

d is a descendant direction in $\overline{x}$ only if $\nabla f(\overline{x})^T d < 0$

First-Order Necessary Condition

Theorem *Let $f \in C^1$. If $\overline{x} \in \Re^n$ is a local minimum for problem (P), then $\nabla f(\overline{x}) = 0$*

- if $\overline{x}$ does not satisfy the required condition, it cannot be a local minimum
- The first order necessary condition is not a sufficient condition

  Example: $f : \Re \to \Re$, $f(x) = -x^2$, and $\overline{x} = 0$
  $\nabla f(\overline{x}) = 0$ and though $\overline{x} = 0$ is not a local minimum

Necessary condition

Second-Order Necessary Condition

Theorem *Let $f \in C^2$. If $\bar{x} \in \Re^n$ is a local minimum for problem (P), then*

(i) $\nabla f(\bar{x}) = 0$

(ii) $d^T \nabla^2 f(\bar{x})\, d \geq 0 \quad \forall d \in \Re^n$

- the second order necessary condition is stronger than the first order condition
- however, it requires the function to be in class C^2
- and it is not a sufficient condition

  Example: $f : \Re \to \Re$, $f(x) = x^3$, and $\bar{x} = 0$
  $\nabla f(\bar{x}) = 0$ and $\nabla^2 f(\bar{x}) = 0$ though $\bar{x} = 0$ is not a local minimum

Sufficient condition

Second-Order Sufficient Condition

Theorem *Let $f \in C^2$. A solution $\overline{x} \in \Re^n$ that satisfies these conditions:*

(i) $\nabla f(\overline{x}) = 0$

(ii) $\nabla^2 f(\overline{x})$ *is positive definite*

is a (strict) local minimum for problem (P)

- the second order sufficient condition is aimed at indentifying *strict* local minimum
- it is not a necessary condition

  Example: $f : \Re \to \Re, f(x) = x^4$.
  The solution $\overline{x} = 0$ is a strict local minimum for the function but $\nabla^2 f(\overline{x})$ is not positive definite

Constrained Optimization

General optimization problem $\min f(x), x \in F$

Constrained Optimization

- feasible region $F = \{x \in \Re^n : \quad g_i(x) \leq 0 \quad i \in I$
  $\quad h_j(x) = 0 \quad j \in E\}$
- assumption: f, g_i and h_j are in class C^1
- idea: a point $\overline{x} \in F$ is a local minimum if there is no descendant direction that preserves feasibility

Constrained Optimization

Simple case: inequalities only ($E = \emptyset$)

- let $I_a(\overline{x}) = \{i \in I : g_i(\overline{x}) = 0\}$ be the set of constraints that are tight in $\overline{x}$
- to preserve feasibility one must consider only constraints in $I_a(\overline{x})$
- by continuity of $g_i(\cdot)$ functions, in a sufficiently small neighborhood of $\overline{x}$, all the remaining constraints are satisfied
- by linearization of active constraints, if $\overline{x}$ is a local minimum then

$$\nexists d \in \Re^n \text{ such that } \nabla f(\overline{x})^T d < 0 \text{ and } \nabla g_i(\overline{x})^T d < 0 \;\; \forall i \in I_a(\overline{x})$$

Constrained Optimization

Simple case: inequalities only ($E = \emptyset$)

By linear algebra, the condition above yields

Theorem *(Fritz-John conditions:) Let $f \in C^1$ and $g_i \in C^1$ $\forall i \in I$. If $\overline{x} \in F$ is a local minimum for f over F, then there exist scalar numbers λ_0 and λ_i $(i \in I)$ such that*

(i) $\lambda_0 \nabla f(\overline{x}) + \sum_{i \in I} \lambda_i \nabla g_i(\overline{x}) = 0$

(ii) $\lambda_i g_i(\overline{x}) = 0 \quad \forall i \in I$

(iii) $\lambda_0 \geq 0,\ \lambda_i \geq 0 \quad (\forall i \in I)$ *and not all λ's are zero.*

Constrained Optimization

Simple case: inequalities only ($E = \emptyset$)

- When $\lambda_0 = 0$ Fritz-John conditions reduce to $\sum_{i \in I} \lambda_i \nabla g_i(\overline{x}) = 0$
- define a subset of Fritz-John points by the additional requirement $\lambda_0 > 0$ (e.g., $\lambda_0 = 1$)

Definition *A point $\overline{x} \in F$ is a Karush-Kuhn-Tucker (KKT) point if there exist scalar numbers λ_i ($i \in I$) such that*

(i) $\nabla f(\overline{x}) + \sum_{i \in I} \lambda_i \nabla g_i(\overline{x}) = 0$

(ii) $\lambda_i g_i(\overline{x}) = 0 \quad \forall i \in I$

(iii) $\lambda_i \geq 0 \quad (\forall i \in I)$.

Constrained Optimization

KKT conditions

- KKT points represent a subset of FJ points
- if F is “regular enough” (constraint qualification conditions), a local minimum is a KKT point
- for the general case where F is defined also by equalities, KKT conditions are

  (i) $\lambda_0 \nabla f(\overline{x}) + \sum_{i \in I} \lambda_i \nabla g_i(\overline{x}) + \sum_{j \in E} \mu_j \nabla h_j(\overline{x}) = 0$

  (ii) $\lambda_i g_i(\overline{x}) = 0 \quad \forall i \in I$

  (iii) $\lambda_0 \geq 0, \quad \lambda_i \geq 0 \quad \forall i \in I$

Algorithms for unconstrained optimization

Most algorithms are iterative schemes that

- start from an initial solution (denoted by x^0),
- define a sequence $\{x^k\}$ of points
- until some stopping criterion is met

- at each iteration k, let x^k the current point; the next point is defined as $x^{k+1} = x^k + \alpha_k d^k$, where
  - $d^k \in \Re^n, ||d^k|| = 1$ is the *search direction*
  - $\alpha_k \in \Re_+$ is the *step size*

Two main classes of algorithms

- *line search* algorithms: determine the search direction, and then the step size;
- *trust region* algorithms: determine the step size, and then the search direction.

Algorithms for unconstrained optimization

Line search methods

At each iteration k

1. define a descendant direction d^k
   - gradient method, stochastic gradient descent
   - Newton's method, quasi-Newton's method
2. define a step size α_k
   - best step size
   - constant step size, Wolfe conditions

Algorithms for unconstrained optimization

Trust region methods

At each iteration k

1. define a trust region for x^k as $T = \{x : ||x - x^k|| \leq \Delta^k\}$
   - replace f by a function $\tilde{f}$
   - typically, $\tilde{f}$ is the Taylor series up to the second order (quadratic function)
2. optimize function $\tilde{f}$ over the trust region

- how to define Δ^k?

  too small Δ^k: the algorithm could miss an opportunity to take a substantial improvement

  too large Δ^k: $\tilde{f}$ can be a poor approximation of f

  $\rightarrow$ the size of the region is defined according to the performance during previous iterations

Example: stochastic gradient descent

Fitting function problem

- set M of samples; for each sample i: time t_i, value y_i
- function $f \in \mathcal{F}$ parametrized by a weight vector $x \in \Re^n$
- for each sample i, discrepancy $e_i(x) = f(x; t_i) - y_i$

Loss function

$$E(x) = \sum_{i \in M} E_i(x) = \frac{1}{2} \sum_{i \in M} ||e_i(x)||^2$$

unconstrained optimization problem

$$\min E(X) : x \in \Re^n$$

Example: stochastic gradient descent

What would gradient method do?

- At each iteration k the gradient method computes the gradient

$$\nabla E(x) = \sum_{i \in M} \nabla E_i(x)$$

- $|M|$ terms, each with n partial derivates
- unpractical if $|M|$ and/or n is large

Stochastic Gradient Descent

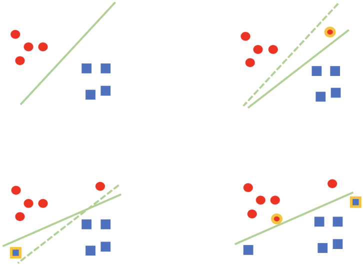
Stochastic gradient

- At each iteration k, replace $\nabla E(x)$ by an approximation
- Select a sample p_i ...and compute $\nabla E_{p_i}(x)$ only

- Computation is faster by a factor of $|M|$
- When data are redundant, the individual gradients are aligned and the approximation is good
- Convergence requires the step size to tend to zero
- Computationally faster for computing near-optimal solutions $\rightarrow$ very attractive in Machine Learning applications
  - optimality is not needed to avoid overfitting
  - very large number of samples
- this method can be used online
- possibly use a mini-batch of samples at each iteration

Stochastic Gradient Descent

Same idea for binary classification



- classification is carried out in an iterative fashion
- until all samples are correctly classified
- fast re-optimization of the classifier for a new sample

Algorithms for constrained optimization

Three main classes of algorithms

- adaptations of the algorithms for unconstrained optimization
- penalty algorithms
- algorithms based on Lagrangian relaxation

Penalty algorithms

Main idea

- Replace the constrained problem

$$\min f(x), x \in F$$

into an unconstrained optimization problem

$$\min P(x; c), x \in \Re^n$$

where

  - $P(x; c) = f(x) + c\phi(x)$
  - $c > 0$ is a parameter, and
  - $\phi(x)$ is a penalty function

Penalty Algorithms

Penalty function

- Ideally $\phi(x) = \begin{cases} 0 & \text{if } x \in F; \\ +\infty & \text{otherwise} \end{cases}$ (not continuous function)
- Approximation: require that
  - ϕ is continuous;
  - $\phi(x) = 0 \quad \forall x \in F$;
  - $\phi(x) > 0 \quad \forall x \notin F$;
- Typical choices
  - $\phi(x) = \sum_{i=1}^{m} \max(g_i(x), 0) + \sum_{j=1}^{p} |h_j(x)|$
  - $\phi(x) = \sum_{i=1}^{m} [\max(g_i(x), 0)]^2 + \sum_{j=1}^{p} |h_j(x)|^2$

Penalty Algorithms

Parameter c

Represents the weight of the constraint violation

- c “small”
  - $P(x; c) \simeq f(x)$
  - feasibility is not so relevant in the objective function
  - likely to produce an infeasible solution
- c “large”
  - $\rightarrow$ large penalty for infeasible solutions
  - likely to find a feasible solution
  - $P(x; c) \not\simeq f(x) \rightarrow$ likely to find a non-optimal solution

Example

$$\begin{aligned} \min \quad & 2x_1^2 + 4x_2^2 - 2x_1x_2 \\ & 2x_1 - x_2 - 1 \leq 0 \\ & 4x_1x_2 + x_2^2 - 1 = 0 \end{aligned}$$

Given a value $c > 0$ the problem to be solved is

$$\min P(x; c) = 2x_1^2 + 4x_2^2 - 2x_1x_2 + c\left[(2x_1 - x_2 - 1)_+^2 + (4x_1x_2 + x_2^2 - 1)^2\right]$$

where $(a)_+ = \max(a, 0)$

Penalty Algorithms

- Search for feasible solutions requires a large value of c
- However, optimizing with a large value of c may be computationally challenging (numerical instability)
- $\rightarrow$ penalty algorithms are executed with different (increasing) values of c
- at each iteration a value of c is selected, and a candidate solution is produced

Theorem *Let $\overline{x}$ be a local minimum for function $P(x; c)$ for some parameter c. If $\phi(\overline{x}) = 0$ then $\overline{x}$ is a local minimum for (P).*

Penalty Algorithms

If a sequence of increasing weights c_k is used, then:

- $P(x^k; c_k) \leq f(\overline{x})$

  at each iteration a lower bound is available
- $\phi(x^{k+1}) \leq \phi(x^k)$

  infeasibility decreases during the execution of the algorithm
- $f(x^{k+1}) \geq f(x^k)$

  solution value worsens during the execution of the algorithm
- $P(x^k; c_k) \leq P(x^{k+1}; c_{k+1})$

  lower bound value increases during the execution of the algorithm

The algorithm moves through a sequence of infeasible solutions (dual algorithm)

Barrier Algorithms

- similar idea: the penalty function is as follows

$$\phi(x) = \sum_{i \in I} -\log(-g_i(x))$$

- and is defined only for points x that have $g(x) < 0$
- points on the boundary of F are allowed in principle; however, for any $c > 0$, a barrier grows when x tends to the boundary of F
- idea: initilizing the algorithm with a point inside F, the next point is forced to remain inside F

Relaxations

Let $\mathcal{P}$ be an optimization problem defined as

$$(\mathcal{P}) \qquad z = \min \; f(x), \; x \in F(\mathcal{P})$$

Definition A *relaxation* is an optimization problem

$$(\mathcal{R}) \qquad z_r = \min \; \Phi(x), \; x \in F(\mathcal{R})$$

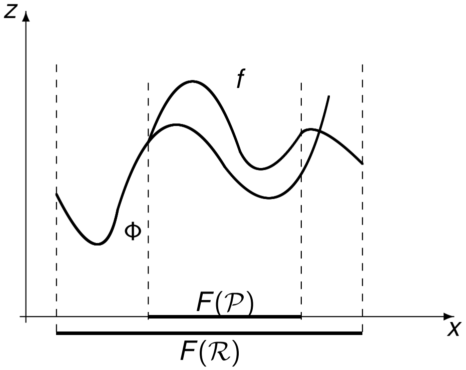
such that

(a) $F(\mathcal{P}) \subseteq F(\mathcal{R})$

(b) $\Phi(x) \leq f(x) \quad \forall x \in F(\mathcal{P})$

Relaxations

- the feasible set of $\mathcal{R}$ should contain the feasible set of $\mathcal{P}$
- the relaxed objective function Φ should be "not worse" than f for each point $x \in F(\mathcal{P})$



Relaxations

Theorem: *Let $(\mathcal{P})$ be an optimization problem with optimal value z. Let $(\mathcal{R})$ be a relaxation of $\mathcal{P}$ with optimal value z_r. Then, $z_r \leq z$.*

- Let $\overline{x}$ be the optimal solution of problem $(\mathcal{P})$
- by definition $\overline{x} \in F(\mathcal{P})$
- requirement (i) of relaxation $\rightarrow \overline{x} \in F(\mathcal{R})$

  hence $z_r = \min_{x \in F(\mathcal{R})} \Phi(x) \leq \Phi(\overline{x})$
- requirement (ii) of relaxation $\rightarrow \Phi(\overline{x}) \leq f(\overline{x})$
- $\Rightarrow z_r \leq \Phi(\overline{x}) \leq f(\overline{x}) = z$

Lagrangian Relaxation

Let problem $(\mathcal{P})$ be defined as

$$(\mathcal{P}) \qquad z = \min f(x)$$
$$x \in X \subseteq \Re^n$$
$$g_i(x) \leq 0 \qquad i \in I$$
$$h_j(x) = 0 \qquad j \in E$$

Definition: given *Lagrangian multipliers* $u_i \geq 0 \quad (\forall i \in I)$ and $v_j \gtrless 0$ $(\forall j \in E)$, the *Lagrangian relaxation* of $\mathcal{P}$ is

$$(\mathcal{R}) \qquad \ell(u, v) = \min_{x \in X} \mathcal{L}(x; u, v) \tag{1}$$

where the *Lagrangian function* is

$$\mathcal{L} : X \to \Re, x \to \mathcal{L}(x; u, v) = f(x) + \sum_{i \in I} u_i g_i(x) + \sum_{j \in E} v_j h_j(x)$$

Example

$$\begin{aligned} \min \quad & 2x_1^2 + 4x_2^2 - 2x_1x_2 \\ & 2x_1 - x_2 - 1 \quad \leq 0 \\ & 4x_1x_2 + x_2^2 - 1 \quad = 0 \end{aligned}$$

Given multipliers $u \geq 0$ and $v \gtrless 0$, the Lagrangian function (to be minimized) is

$$\mathcal{L}(x; u, v) = 2x_1^2 + 4x_2^2 - 2x_1x_2 + u(2x_1 - x_2 - 1) + v(4x_1x_2 + x_2^2 - 1) =$$

$$2x_1^2 + (4 + v)x_2^2 + (4v - 2)x_1x_2 + 2ux_1 - ux_2 - (u + v)$$

Lagrangian Relaxation

- Lagrangian relaxation is an optimization problem which is easier to be solved (hard constraints have been moved to the objective function)
- similar to penalty algorithms but
  - continuous function
  - reward for constraints satisfaction
- in some cases the problem can be decomposed into a number of subproblems (that can be optimized indipendently)

Weak duality

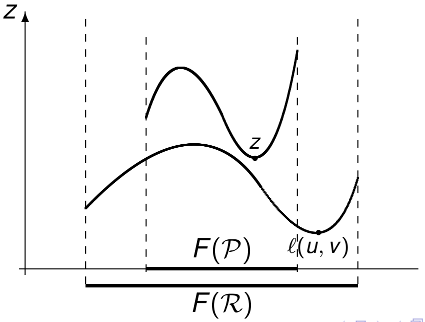
Theorem: *(weak duality) For any choice of the multipliers $u \in \Re^m, u \geq 0$ and $v \in \Re^p$, we have $\ell(u, v) \leq z$*

- $F(\mathcal{P}) \subseteq F(\mathcal{R})$ $\quad$ ($\leftarrow$ removed some constraints)
- for any $x \in F(\mathcal{P})$:
  - $\forall i \in I: \quad g_i(x) \leq 0$ and $u_i \geq 0 \quad \rightarrow \sum_{i \in I} u_i g_i(x) \leq 0$
  - $\forall j \in E: h_j(x) = 0 \quad \rightarrow \sum_{j \in E} v_j h_j(x) = 0$
- $\Rightarrow \mathcal{L}(x; u, v) \leq f(x)$
- the Lagrangian relaxation is a relaxation

Properties of the relaxed solution

General situation

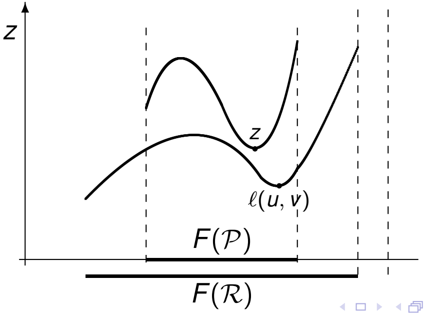
- an optimal solution $\tilde{x}$ for the Lagrangian relaxation is typically infeasible for $(\mathcal{P})$
- it provides a lower bound $\ell(u, v) = \mathcal{L}(\tilde{x}; u, v)$ on the optimal solution value



Properties of the relaxed solution

Special case 1: $\tilde{x}$ is feasible for $(\mathcal{P})$

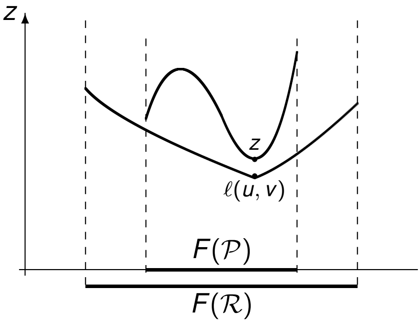
- in general $\tilde{x}$ is not optimal for $(\mathcal{P})$
- the relaxation provides a lower bound $\ell(u, v) = \mathcal{L}(\tilde{x}; u, v)$ on the optimal solution value
- and an upper bound $f(\tilde{x}) \geq z$



Properties of the relaxed solution

Special case 2: $\widetilde{x}$ is feasible and (not provably) optimal for $(\mathcal{P})$

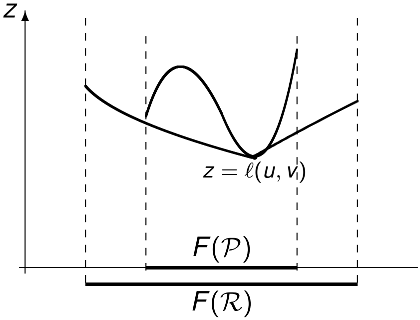
- *proving* optimality for $\widetilde{x}$ may be impossible
- in case $\mathcal{L}(\widetilde{x}; u, v) < f(\widetilde{x})$



Properties of the relaxed solution

Special case 3: $\tilde{x}$ is feasible and provably optimal for $(\mathcal{P})$

- *proving* optimality for $\tilde{x}$ is possible
- in case $\mathcal{L}(\tilde{x}; u, v) = f(\tilde{x})$



Lagrangian Dual problem

- the lower bound $\ell(u, v)$ depends on selected multipliers (u, v)
- which is the "best" lower bound that can be obtained using Lagrangian relaxation?

- Lagrangian dual

$$(D) \qquad \overline{\ell} = \max_{u \geq 0, v} \ell(u, v).$$

- it can be proved that $\ell(u, v)$ is convex with respect to (u, v)

Lagrangian Dual problem

(a) in general: $\overline{\ell} \leq z \rightarrow$ optimality gap $z - \overline{\ell}$

(b) if $\exists \overline{x} \in F(\mathcal{P})$ and $(\overline{u}, \overline{v}) \in \Re_+^m \times \Re^p$ such that $f(\overline{x}) = \ell(\overline{u}, \overline{v})$, then $\overline{x}$ and $(\overline{u}, \overline{v})$ are optimal solutions for the primal and dual problems, respectively;

(c) if $z = -\infty$ (unbounded primal), then
$\ell(u, v) = -\infty \quad \forall (u, v) \in \Re_+^m \times \Re^p$;

(d) if $\overline{\ell} = \infty$ (unbounded dual), then the primal is infeasible ($z = \infty$)

Lagrangian Dual problem

Optimality conditions

- let $\overline{x}$ and $(\overline{u}, \overline{v})$ be optimal solutions of problems $(\mathcal{P})$ and (D), respectively
- both optimal if $f(\overline{x}) = \ell(\overline{u}, \overline{v}) = \inf_x \mathcal{L}(x; \overline{u}, \overline{v}) \leq$
  $f(\overline{x}) + \sum_{i \in I} \overline{u}_i g_i(\overline{x}) + \sum_{j \in E} \overline{v}_j h_j(\overline{x})$
- i.e., if $\sum_{i \in I} \overline{u}_i g_i(\overline{x}) + \sum_{j \in E} \overline{v}_j h_j(\overline{x}) = 0$, meaning that
  $\overline{u}_i g_i(\overline{x}) = 0 \quad \forall i \in I,$ and $\overline{v}_j h_j(\overline{x}) = 0 \quad \forall j \in E$
- These *orthogonality conditions* impose that
  - $\overline{u}_i = 0$ for all all inequalities that are not tight: $g_i(\overline{x}) < 0 \Rightarrow \overline{u}_i = 0$
  - all inequalities associated to multipliers that are strictly positive must be tight: $\overline{u}_i > 0 \Rightarrow g_i(\overline{x}) = 0$

Lagrangian problem and KKT conditions

Assume that $X = \Re^n$

Definition A triplet $(\overline{x}, \overline{u}, \overline{v})$ with $\overline{x} \in \Re^n, \overline{u} \in \Re^m_+, \overline{v} \in \Re^p$ is a *saddle point* if, $\forall x \in \Re^n, u \in \Re^m_+, v \in \Re^p$ we have

$$\mathcal{L}(\overline{x}; u, v) \leq \mathcal{L}(\overline{x}; \overline{u}, \overline{v}) \leq \mathcal{L}(x; \overline{u}, \overline{v})$$

Theorem Let f, g_i $(i = 1, \ldots, m)$ and h_j $(j = 1, \ldots, p)$ continuous functions. Let $\overline{x} \in \Re^n$, $\overline{u} \in \Re^m$ and $\overline{v} \in \Re^p$. If $(\overline{x}, \overline{u}, \overline{v})$ is a saddle point, then

1. $g(\overline{x}) \leq 0$ and $h(\overline{x}) = 0$ (Primal feasibility)
2. $\overline{u} \geq 0$ (Dual feasibility)
3. $\mathcal{L}(\overline{x}; \overline{u}, \overline{v}) = \min_{x \in \Re^n} \mathcal{L}(x; \overline{u}, \overline{v})$ (Lagrangian optimality)
4. $\overline{u}_i \, g_i(\overline{x}) = 0 \quad i = 1, \ldots, m$ (Orthogonality)
5. $\overline{x}$ is a global minimum for $(\mathcal{P})$

Lagrangian problem and KKT conditions

- if a global minimum exists, it is a saddle point of the Lagrangian
  (note that the Lagrangian is an unconstrained optimization problem in the x variables)

- Lagrangian optimality for a saddle point $(\overline{x}, \overline{u}, \overline{v})$:
  $\nabla_x \mathcal{L}(x; \overline{u}, \overline{v})_{|x=\overline{x}} = 0 \Rightarrow$
  $\nabla f(\overline{x}) + \sum_{i \in I} \overline{u}_i \nabla g_i(\overline{x}) + \sum_{j \in E} \overline{v}_j \nabla h_j(\overline{x}) = 0.$
- by definition of saddle point and the orthogonality condition, we have $\forall i \in I: \quad \overline{u}_i \geq 0 \quad$ e $\quad \overline{u}_i \, g_i(\overline{x}) = 0$, i.e., KKT conditions
  (assuming constraint qualification conditions are satisfied)

- sufficient conditions for a certain feasible point $\overline{x} \in \Re^n$ to be a global minimum: there should exist Lagrangian multipliers $\overline{u}$ and $\overline{v}$ such that $(\overline{x}, \overline{u}, \overline{v})$ is a saddle point
- no similar necessary condition: for a given global minimum $\overline{x}$, the required multipliers may not exist

Example

$$\begin{aligned} \min \quad & \tfrac{1}{2}(x_1 - 1)^2 + \tfrac{1}{2}(x_2 - 2)^2 \\ & x_1 + x_2 - 1 \qquad = 0 \end{aligned}$$

- multiplier $v \to \mathcal{L}(x; v) = \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 - 2)^2 + v(x_1 + x_2 - 1)$
- necessary condition

$$\nabla_x \mathcal{L}(x; v) = \begin{bmatrix} x_1 - 1 + v \\ x_2 - 2 + v \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- feasibility condition: $x_1 + x_2 - 1 = 0$

- system with 3 conditions and 3 variables: solution $x_1 = 0, x_2 = 1$ and $v = 1$

Example

Is $(\overline{x}_1 = 0, \overline{x}_2 = 1, \overline{v} = 1)$ a saddle point?

- $\mathcal{L}(\overline{x}; v) = \frac{1}{2}(\overline{x}_1 - 1)^2 + \frac{1}{2}(\overline{x}_2 - 2)^2 + v(\overline{x}_1 + \overline{x}_2 - 1) = \frac{1}{2} + \frac{1}{2} = 1$, hence $\mathcal{L}(\overline{x}; v) \leq \mathcal{L}(\overline{x}; \overline{v})$ for all $v \in \Re$
- $\mathcal{L}(x; \overline{v}) = \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 - 2)^2 + \overline{v}(x_1 + x_2 - 1) = \frac{1}{2}(x_1 - 1)^2 + \frac{1}{2}(x_2 - 2)^2 + x_1 + x_2 - 1$
  - $\nabla_x \mathcal{L}(x; \overline{v}) = 0$ yields $x_1 = 0, x_2 = 1$
  - the Hessian matrix

$$\nabla^2 \mathcal{L}(x; \overline{v}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

is positive definite $\rightarrow x_1 = 0, x_2 = 1$ is a minimum for the lagrangian function $\rightarrow \mathcal{L}(\overline{x}; \overline{v}) \leq \mathcal{L}(x; \overline{v})$ for all $x \in \Re^2$.

Special cases

Optimization under bound constraints

- $\min f(x), \quad a_j \leq x_j \leq b_j \quad j = 1, \ldots, n \quad (a_j - x_j \leq 0, \text{ and } x_j - b_j \leq 0 \ \ \forall j)$
- multipliers $\lambda \in \Re^n_+$ and $\pi \in \Re^n_+$: Lagrangian relaxation
  $\min \mathcal{L}(x; \lambda) = \min_{x \in \Re^n} f(x) + \lambda^T(a - x) + \pi^T(x - b)$
- Necessary KKT conditions for a solution $\overline{x}$ to be a minimum:
  $\exists \lambda^*, \pi^* \in \Re^n_+$ such that
  (a) $\nabla_x \mathcal{L}(\overline{x}; \lambda^*, \pi^*) = \nabla f(\overline{x}) - \lambda^* + \pi^* = 0$
  (b) ${\lambda^*}^T(a - \overline{x}) = 0$
  (c) ${\pi^*}^T(\overline{x} - b) = 0$
  (d) $a \leq \overline{x} \leq b$
  (e) $\lambda^*, \pi^* \geq 0$

Special cases

Optimization under bound constraints

- for each j, at most one among λ_j^* and π_j^* can be positive
- for each j: $\overline{x}_j > a_j \rightarrow \lambda_j^* = 0$, hence $\frac{\partial f(\overline{x})}{x_j} = -\pi_j^* < 0$, i.e., if j is a decreasing direction for the objective function then x_j must attain its lower bound
- similarly: if $\overline{x}_j < b_j$ then $\frac{\partial f(\overline{x})}{x_j} = \lambda_j^* > 0$, i.e., variable x_j must be at the upper bound in case j is an increasing directon for the objective function
- if $a_j < x_j^* < b_j$ we have $\lambda_j^* = \pi_j^* = 0$, hence $\frac{\partial f(\overline{x})}{x_j} = 0$
- $\frac{\partial f(\overline{x})}{x_j} > 0$ implies $\overline{x}_j = a_j$, whereas $\frac{\partial f(\overline{x})}{x_j} < 0$ implies $\overline{x}_j = b_j$