Continuous optimization
ENT 305

Elise Grosjean

Ensta-Paris
Institut Polytechnique de Paris

November 9, 2021

And to sum up the courses ...

	Necessary conditions	Sufficient conditions
Abstract formulation (exist.)		if K compact, $f \in C^0(K)$ then at least one solution
		if K closed, $f \in C^0(K)$, coercive then at least one solution

	Necessary conditions	Sufficient conditions
No constraints $K = \mathbb{R}^d$ (opt.)	if $\overline{x}$ local sol., $f \in C^2(K)$ then, $D^2 f(\overline{x})$ is positive semi-def.	if $f \in C^2(K)$, $\nabla f(\overline{x}) = 0$, $D^2 f(\overline{x})$ positive def. then $\overline{x}$ local sol.
Affine constraints	$\overline{x}$ local sol. then KKT	f convex, then KKT=global sol.
Non-linear constraints	$\overline{x}$ local sol., LICQ then KKT	f convex, h affine, g convex, then KKT=global sol.

And to sum up the courses ...

	Necessary conditions	Sufficient conditions
Abstract formulation (exist.)		if K compact, $f \in C^0(K)$ then at least one solution
		if K closed, $f \in C^0(K)$, coercive then at least one solution

	Find a local solution
No constraints	Gradient Descent
Affine constraints	Penalty methods
Non-linear constraints	

Introduction

Aim of the lecture: a general presentation of one numerical methods for constrained optimization.

- **Penalty methods** $\rightsquigarrow$ equality constraints
- **Projected gradient methods** $\rightsquigarrow$ inequality constraints

well suited if constraints projection is possible and easy to compute.

Reference:

Nocedal and Wright. Numerical optimization. Springer Science and Business Media, 2006.

Boyd and Vandenberghe. Convex Optimization. Cambridge University Press, 2004.

1 Penalty methods for constrained optimization
- Quadratic penalization
- Augmented Lagrangian
- Lagrangian decomposition

2 Projected gradient method
- Projection
- Method
- Combination with penalty methods

Quadratic penalization

We consider in this section

$$\inf_{x \in \mathbb{R}^n} f(x), \quad \text{subject to: } h(x) = 0, \tag{P}$$

where $f\colon \mathbb{R}^n \to \mathbb{R}$ and $h\colon \mathbb{R}^n \to \mathbb{R}^m$ are given and "smooth".

A general difficulty: we need to cope with **two general goals**:

- Minimizing f
- Ensuring the feasibility of x.

When designing a numerical method, the question arises:
Given an iterate x_k, should we look for x_{k+1} so that

$$f(x_{k+1}) < f(x_k) \quad \text{or} \quad \|h(x_{k+1})\| < \|h(x_k)\| \quad ?$$

Quadratic penalization

Main idea: combining the two objectives into a single one.
Given a real number $c \geq 0$, consider the **penalty problem:**

$$\inf_{x \in \mathbb{R}^n} Q_c(x) := f(x) + \frac{c}{2} \|h(x)\|^2. \tag{P_c}$$

A rough statement: if c is large, (P) and (P_c) are "almost" equivalent.

Big advantage of the approach: numerical **methods of unconstrained optimization** can be employed for solving (P_c).

Quadratic penalization

Exercise.

Consider the problem:

$$\inf_{x \in \mathbb{R}} x, \quad \text{subject to: } x = 0.$$

1. What is the solution $\bar{x}$ to the problem?
2. Calculate the solution x_c to the corresponding penalized problem P_c.
3. Verify that $x_c \underset{c \to +\infty}{\longrightarrow} \bar{x}$.

Quadratic penalization

Solution.

1. Obviously $\bar{x} = 0$, since 0 is the unique feasible point of the problem.

2. Let $c > 0$. We have $Q_c(x) = x + \frac{c}{2}x^2$ and $\nabla Q_c(x) = 1 + cx$. Therefore,
$$\nabla Q_c(x) = 0 \Longleftrightarrow x = -\frac{1}{c}.$$
Since Q_c is convex, $x_c := -1/c$ is the unique solution of (P_c).

3. Obviously
$$x_c = -1/c \underset{c \to \infty}{\longrightarrow} 0 = \bar{x}.$$

Quadratic penalization

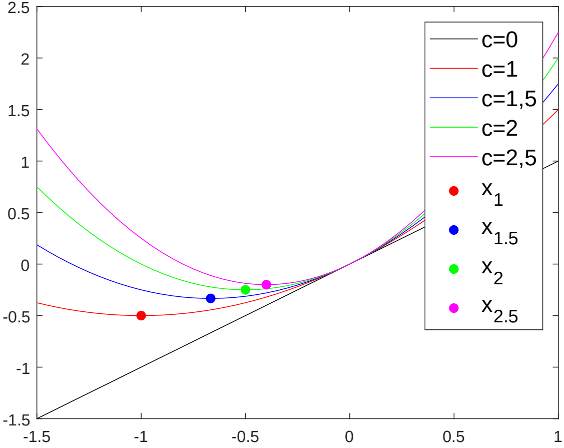


Figure: Graph of Q_c, for various values of c

Quadratic penalization

Lemma 1

Let $c_k \to \infty$. Let $(x_k)_{k\in\mathbb{N}}$ be a sequence in $\mathbb{R}^n$. Assume that

- *For all $k \in \mathbb{N}$, x_k is the* **solution** *to (P_{c_k}).*
- *The sequence $(x_k)_{k\in\mathbb{N}}$* **converges**, *let $\bar{x}$ denote the limit.*
- *There exists $\tilde{x}$ such that $h(\tilde{x}) = 0$.*

Then, $\bar{x}$ is a **solution** *to the original constrained problem (P).*

Proof. Step 1. Let x be a feasible point (that is, $h(x) = 0$). Then,

$$Q_{c_k}(x) = f(x) + \frac{c_k}{2}\|h(x)\|^2 = f(x).$$

In particular, $Q_{c_k}(\tilde{x}) = f(\tilde{x})$.

Quadratic penalization

Step 2: $\bar{x}$ is feasible. For all $k \in \mathbb{N}$, we have

$$\begin{aligned} c_k \|h(x_k)\|^2 &= Q_{c_k}(x_k) - f(x_k) \\ &\leq Q_{c_k}(\tilde{x}) - f(x_k) && [\textit{Optimality of } x_k] \\ &= f(\tilde{x}) - f(x_k). && [\textit{Equality of Step 1}] \end{aligned}$$

Since $f(x_k) \to f(\bar{x})$, the sequence $(f(x_k))_{k\in\mathbb{N}}$ is bounded. Therefore, there exist $M > 0$ such that $c_k \|h(x_k)\|^2 \leq M$. Thus

$$\|h(x_k)\| \leq \sqrt{M/c_k}, \quad \forall k \in \mathbb{N}.$$

Passing to the limit, we get $\|h(\bar{x})\| \leq 0$. Thus $\bar{x}$ is **feasible**.

Quadratic penalization

Step 3. Optimality of $\bar{x}$. Let x be feasible. We have

$$\begin{aligned} f(x_k) &\le f(x_k) + c_k \|h(x_k)\|^2 \\ &= Q_{c_k}(x_k) \\ &\le Q_{c_k}(x) && [\textit{Optimality of } x_k] \\ &= f(x). && [\textit{Equality of Step 1}] \end{aligned}$$

Passing to the limit, we get

$$f(\bar{x}) \le f(x).$$

Thus $\bar{x}$ is optimal.

Quadratic penalization

The result of the lemma must be seen as an “ideal” situation.

Difficulties in practice:

- The problem (P_c) **may not have a solution**, even if (P) has a solution. Example:

$$\inf_{x \in \mathbb{R}} x^3, \quad \text{subject to: } x = 0.$$

- The sequence $(x_k)_{k \in \mathbb{N}}$ may not converge.
- The problem (P_c) is **hard to solve** when c is large, it is likely to be ill-conditioned (see next example).

Quadratic penalization

Example. Consider:

$$\inf_{(x,y)\in\mathbb{R}^2} \frac{1}{2}\big(x^2+(y-1)^2\big), \quad \text{subject to: } x=y.$$

Projection problem of the point $(0,1)$ on the line $\{(x,y)\,|\,y=x\}$.

Exercise. Verify the following statements.

- Solution: $x^* = (0.5, 0.5)$.
- Solution of P_c, the penalty function, is:
  $\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \frac{1}{1+2c}\begin{pmatrix} c \\ 1+c \end{pmatrix}.$
- There exists a constant M such that for all $c \geq 0$,

$$\|(x_c,y_c)-(\bar{x},\bar{y})\| \leq M/c.$$

Quadratic penalization

Solution.

1. $\nabla f(x,y) = \begin{pmatrix} x \\ y-1 \end{pmatrix}$. The function f is convex and thus, the global solution of the unconstrained version is $(0,1)$. With the constraints, we aim at minimizing $\frac{1}{2}\left(2x^2 - 2x + 1\right)$, and the unique solution is obviously $x = 0.5$.

2. $Q_c(x) = \frac{1}{2}\left(x^2 + (y-1)^2\right) + \frac{c}{2}(y-x)^2$ and $\nabla Q_c(x,y) = \begin{pmatrix} x - c(y-x) \\ y - 1 + c(y-x) \end{pmatrix}$, and since Q_c is convex, the unique solution of P_c is: $\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \dfrac{1}{1+2c}\begin{pmatrix} c \\ 1+c \end{pmatrix}$.

3. $\lim\limits_{c\to\infty} \begin{pmatrix} x_c \\ y_c \end{pmatrix} = \lim\limits_{c\to\infty} \frac{c}{c(1/c+2)} \begin{pmatrix} 1 \\ 1/c+1 \end{pmatrix} = \frac{1}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix}$
$\|(x_c,y_c) - (0.5,0.5)\|^2 = \frac{0.5}{(1+2c)^2} \Rightarrow \|(x_c,y_c) - (0.5,0.5)\| = \frac{\sqrt{0.5}}{1+2c} \leq \frac{M}{c}$.
Yet, $\nabla^2 Q(x,y) = \begin{pmatrix} 1+c & -c \\ -c & 1+c \end{pmatrix}$ which is ill-conditioned for large c. It yields difficulties with e.g. Newton algorithm $(\nabla^2 Q \cdot p = -\nabla Q)$ with abrupt function changes.

Quadratic penalization

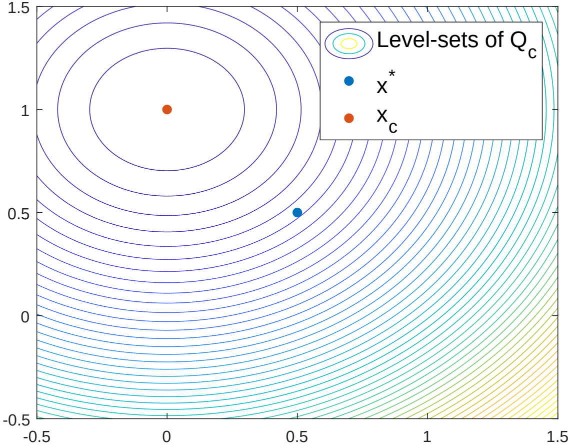


Figure: Graph of Q_c, for $c = 0$.

Quadratic penalization

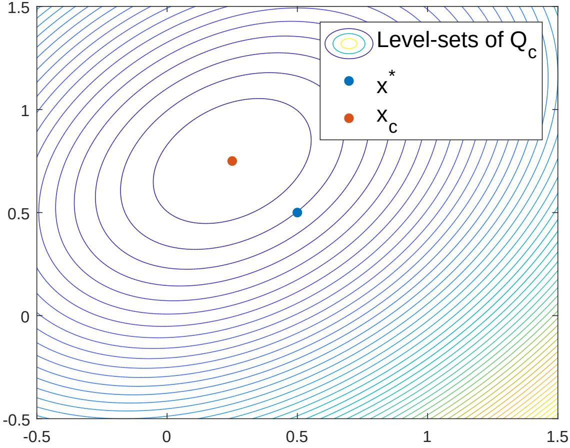


Figure: Graph of Q_c, for $c = 0.5$.

Quadratic penalization

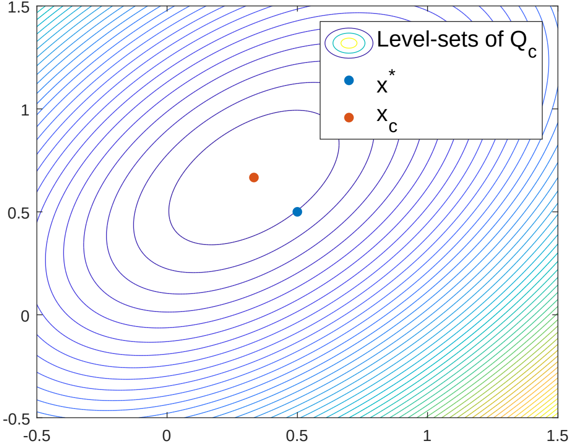


Figure: Graph of Q_c, for $c = 1$.

Quadratic penalization

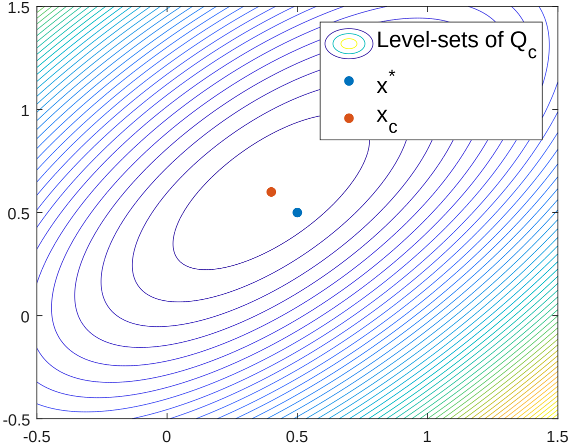


Figure: Graph of Q_c, for $c = 2$.

Quadratic penalization

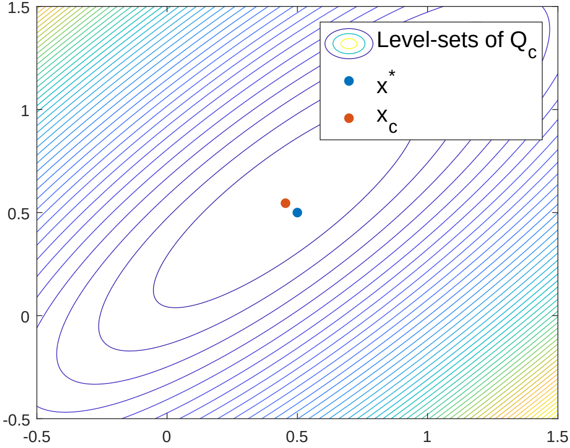


Figure: Graph of Q_c, for $c = 5$.

Penalty algorithm

General idea: increase the value of c progressively, to mitigate the difficulty of minimizing Q_c.

Algorithm:

1. Input: Choose $c_0 > 0$, starting point $x_0 \in \mathbb{R}^n$.
2. For $k = 1, ..., K - 1$, do
   - Solve (P_{c_k}) (e.g. with a gradient descent algorithm starting from x_{k-1}) and set x_k the solution.
   - If x_k is such that $h(x_k) = 0$, stop.
   - Otherwise choose $c_{k+1} > c_k$.

   End for.
3. Output: x_K.

Penalty algorithm

$$Q_c(x) = f(x) + \frac{c}{2}\|h(x)\|^2$$

$$\begin{aligned}\nabla Q_c(x) &= \nabla f(x) + c\langle h(x), \nabla h(x)\rangle \\ &= \nabla L(x, ch(x))\end{aligned}$$

$$c_k h(x_k) \simeq \overline{\mu}$$

Augmented Lagrangian

Unlike the penalty method, with the **augmented Lagrangian method** is not necessary to take $c \to \infty$ in order to solve the original constrained problem, avoiding ill-conditioning.

Augmented Lagrangian

The two ideas of the **augmented Lagrangian method**:

1. Solving a penalty problem (like (P_c)) also yields an approximation of the Lagrange multiplier.
2. We can "improve" the penalty function Q_c with the knowledge of that approximation.

Algorithm: at each iteration,

- the penalty parameter is increased
- the approximations x_k of the solution and λ_k of the Lagrange multiplier are improved.

Augmented Lagrangian

Let $c > 0$. The **augmented Lagrangian** $L_c \colon \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is defined by

$$L_c(x, \mu) = f(x) + \langle \mu, h(x) \rangle + \frac{c}{2} \|h(x)\|^2.$$

$$\begin{aligned} \nabla L_c(x, \mu) &= \nabla f(x) + \langle \mu, \nabla h(x) \rangle + \langle ch(x), \nabla h(x) \rangle \\ &= \nabla L(x, \mu + ch(x)) \end{aligned}$$

$$\mu_k + c_k h(x_k) \simeq \overline{\mu}$$

$$h(x_k) \simeq \frac{\overline{\mu} - \mu_k}{c_k}$$

$$\mu_{k+1} = \mu_k + c_k h(x_{k+1})$$

Augmented Lagrangian

$$L_c(x,\mu) = f(x) + \langle \mu, h(x)\rangle + \frac{c}{2}\|h(x)\|^2.$$

We have

$$\begin{aligned} L_c(x,\mu) &= L(x,\mu) + \frac{c}{2}\|h(x)\|^2 \\ &= Q_c(x) + \langle \mu, h(x)\rangle \\ &= f(x) + \frac{c}{2}\|h(x) + \frac{\mu}{c}\|^2 - \frac{\|\mu\|^2}{2c} \end{aligned}$$

For a fixed λ, $L_c(\cdot,\mu)$ still serves as a **penalty function**. If $x_{c,\mu}$ minimizes $L_c(x,\mu)$ and if c is very large, then

- $f(x_{c,\mu})$ is small
- $\frac{c}{2}\|h(x) + \frac{\mu}{c}\|^2$ is small $\rightarrow$ $\|h(x) + \frac{\mu}{c}\|$ is very small
  $\rightarrow$ $\|h(x)\|$ is very small.

Augmented Lagrangian

The new **penalty problem:**

$$\inf_{x \in \mathbb{R}^n} L_c(x, \mu). \qquad (P_{c,\mu})$$

Lemma 2

Let $\bar{x}$ be a local minimizer of (P). Under technical assumptions, there exists $\bar{\mu}$ and $\bar{c} \geq 0$ such that for all $c > \bar{c}$,

- *the* **KKT conditions** *hold true*
- *$\bar{x}$ is a* **local solution** *to $(P_{c,\bar{\mu}})$.*

Reminders

	Necessary conditions	Sufficient conditions
Abstract formulation (exist.)		if K compact, $f \in C^0(K)$ then at least one solution
		if K closed, $f \in C^0(K)$, coercive then at least one solution

	Necessary conditions	Sufficient conditions
No constraints $K = \mathbb{R}^d$ (opt.)	if $\overline{x}$ local sol., $f \in C^2(K)$ then, $D^2 f(\overline{x})$ is positive semi-def.	if $f \in C^2(K)$, $\nabla f(\overline{x}) = 0$, $D^2 f(\overline{x})$ positive def. then $\overline{x}$ local sol.
Affine constraints	$\overline{x}$ local sol. then KKT	f convex, then KKT=global sol.
Non-linear constraints	$\overline{x}$ local sol., LICQ then KKT	f convex, h affine, g convex, then KKT=global sol.

Augmented Lagrangian

The new **penalty problem:**

$$\inf_{x \in \mathbb{R}^n} L_c(x, \mu). \qquad (P_{c,\mu})$$

Lemma 3

Let $\bar{x}$ be a local minimizer of (P). Under technical assumptions, there exists $\bar{\mu}$ and $\bar{c} \geq 0$ such that for all $c > \bar{c}$,

- *the* **KKT conditions** *hold true*
- *$\bar{x}$ is a* **local solution** *to $(P_{c,\bar{\mu}})$.*

Idea of proof. We have

$$\nabla L_c(\bar{x}, \bar{\mu}) = \nabla L(\bar{x}, \bar{\mu} + ch(\bar{x})) = \nabla L(\bar{x}, \bar{\mu}) = 0.$$

$$\nabla^2 L_c(\bar{x}, \bar{\mu}) = \nabla^2 L(\bar{x}, \bar{\mu}) + c\langle \nabla h(\bar{x}), \nabla h(\bar{x}) \rangle$$

For c large enough, $\nabla^2 L_c(\bar{x}, \bar{\mu})$ is positive definite.
Therefore, $\bar{x}$ is a local solution.

Augmented Lagrangian

Example 1. Consider $\inf_{x \in \mathbb{R}} x - x^2$, subject to: $x = 0$.

Exercise.

- Write the Lagrangian formulation and find the Lagrangian multiplier.
- Does KKT holds for $\bar{x} = 0$?
- Write the augmented Lagrangian $(P_{c,\bar{\mu}})$ and show that $\bar{x}$ is a local solution to $(P_{c,\bar{\mu}})$ if $c > \bar{c}$.

Augmented Lagrangian

Example 1. Consider $\inf_{x\in\mathbb{R}} x - x^2$, subject to: $x = 0$.

- Solution $\bar{x} = 0$.
- Lagrangian $L(x,\mu) = x - x^2 + \mu x$. We have

$$\nabla L(\bar{x},\mu) = 1 - 2\bar{x} + \mu = 1 + \mu \quad \Longrightarrow \quad \bar{\mu} = -1.$$

- Augmented lagrangian:

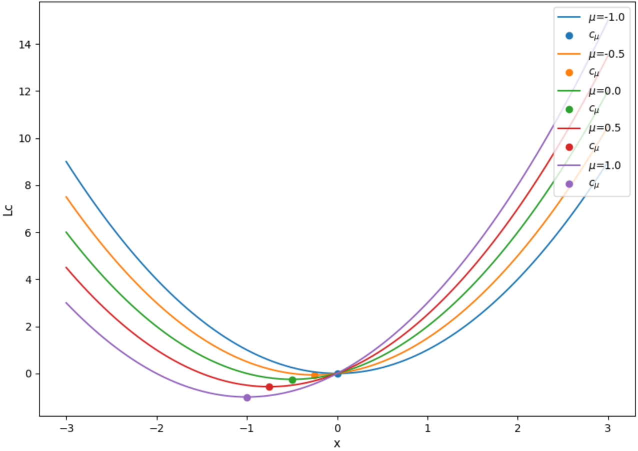
$$L_c(x,\mu) = x - x^2 + \mu x + \frac{c}{2}x^2 = (1+\mu)x + \left(\frac{c}{2} - 1\right)x^2.$$

If $c > \bar{c} := 2$, $L_c(\cdot,\mu)$ has a unique minimizer

$$x_{c,\mu} = \frac{\mu+1}{2-c} = \frac{\mu-\bar{\mu}}{2-c}.$$

In particular, $x_{c,\bar{\mu}} = \bar{x}$.

Augmented Lagrangian



Quadratic penalization

Example 2. Consider:

$$\inf_{(x,y)\in\mathbb{R}^2} \frac{1}{2}\left(x^2 + (y-1)^2\right), \quad \text{subject to: } x = y.$$

Projection problem of the point $(0,1)$ on the line $\{(x,y) \,|\, y = x\}$.

Exercise. Verify the following statements.

- Solution: $(\bar{x}, \bar{y}) = (0.5, 0.5)$, $\bar{\mu} = 0.5$.
- Solution of $(P_{c,\mu})$ (aug. lagrangian):
  $\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \frac{1}{1+2c} \begin{pmatrix} c-\mu \\ 1+c+\mu \end{pmatrix}.$
- There exists a constant M such that for all $c > 0$,

$$\|(x_c, y_c) - (\bar{x}, \bar{y})\| \leq M|\bar{\mu} - \mu|/c.$$

Quadratic penalization

Solution.

1. $\nabla f(x,y) = \begin{pmatrix} x \\ y-1 \end{pmatrix}$. The function f is convex and thus, the global solution of the unconstrainted version is $(0,1)$. With the constraints, we aim at minimizing $\frac{1}{2}(2x^2 - 2x + 1)$, and the unique solution is $\bar{x} = 0.5$. With $L(\bar{x},\bar{y},\bar{\mu}) = f(\bar{x}) + \bar{\mu}(\bar{y} - \bar{x})$, we find $\bar{\mu} = 0.5$ from stationarity cond.

2. $L_{c,\mu}(x,y) = \frac{1}{2}\left(x^2 + (y-1)^2\right) + \frac{c}{2}(y-x)^2 + \mu(y-x)$ and $\nabla L_{c,\mu}(x,y) = \begin{pmatrix} x - c(y-x) + \mu \\ y - 1 + c(y-x) - \mu \end{pmatrix}$, and since $L_{c,\mu}$ is convex, the unique solution of $P_{c,\mu}$ is: $\begin{pmatrix} x_c \\ y_c \end{pmatrix} = \dfrac{1}{1+2c}\begin{pmatrix} c+\mu \\ 1+c-\mu \end{pmatrix}$.

Quadratic penalization

Solution.

3.

$$\begin{aligned}\|(x_c, y_c) - (\bar{x}, \bar{y})\|^2 &= \frac{1}{(1+2c)^2}\|(c+\mu-0.5)^2 + (1+c-\mu-0.5)^2\| \\ &= \frac{2(\mu-0.5)^2}{(1+2c)^2} \\ &= \frac{2(\mu-\bar{\mu})^2}{(1+2c)^2}\end{aligned}$$

$$\|(x_c, y_c) - (0.5, 0.5)\| = \tfrac{\sqrt{2}}{1+2c}|\mu-\bar{\mu}| \leq \tfrac{M|\mu-\bar{\mu}|}{c}.$$

Augmented Lagrangian

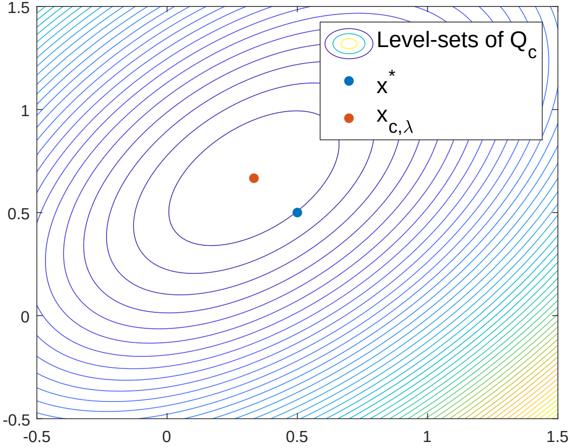


Figure: Level-sets $L_c(\cdot, \mu)$, for $c = 1$ and $\mu = 0$.

Augmented Lagrangian

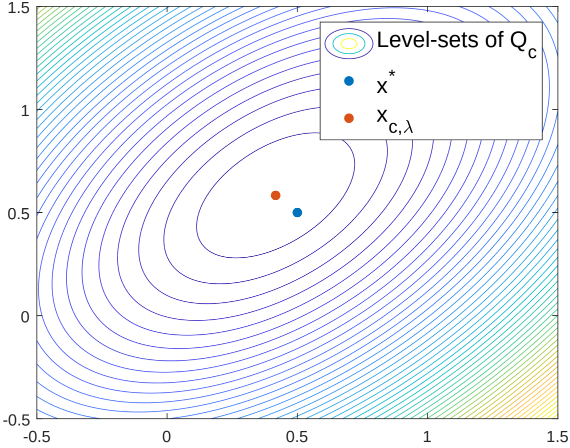


Figure: Level-sets $L_c(\cdot, \mu)$, for $c = 1$ and $\mu = 0, 25$.

Augmented Lagrangian

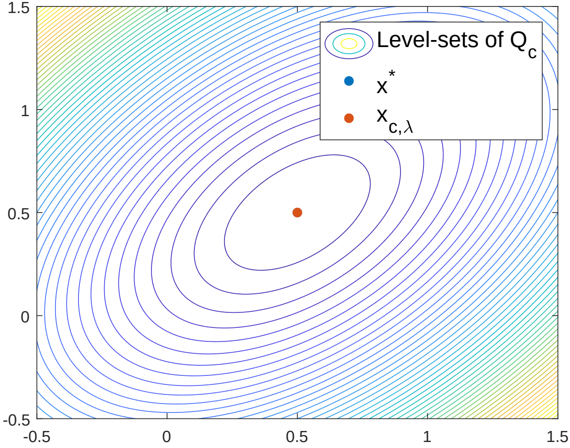


Figure: Level-sets $L_c(\cdot, \mu)$, for $c = 1$ and $\mu = 0,5$.

Augmented Lagrangian

Algorithm.

1. Input:
   - Initial point and multipliers $(x_0, \mu_0) \in \mathbb{R}^n \times \mathbb{R}^m$
   - Initial penalty parameter $c_0 > 0$, initial tolerance $\varepsilon_0 > 0$
   - Tolerance $\varepsilon > 0$.
2. Set $k = 0$.
3. While $\|D_x L(x_k, \mu_k)\| > \varepsilon$ and $\|h(x_k)\| > \varepsilon$,
   - Find x_{k+1} such that $\|D_x L_{c_k}(x_{k+1}, \mu_k)\| \leq \varepsilon_k$.
   - If $\|h(x_{k+1})\|$ is small, set $\mu_{k+1} = \mu_k + c_k h(x_{k+1})$. Reduce ε_k.
   - Otherwise, increase c_k.
   - Set $k = k + 1$.

   End while.
4. Output (x_k, λ_k).

Lagrangian decomposition

Main ideas of **Lagrangian decomposition** methods:

- We take $c = 0$ in the augmented Lagrangian. At iterate k, given an approximation μ_k of the Lagrange multiplier, we solve

$$\inf_{x \in \mathbb{R}^n} L(x, \mu_k). \tag{P_x}$$

  where μ_k is found with the following maximization

$$\sup_{\mu \in \mathbb{R}^m} L(x, \mu)$$

  Since $\nabla_\mu L(x, \mu) = h(x)$, this maximization is solved by iterating with an **ascent gradient step** to approximate the solution of $h(x) = 0$:
- Given a solution x_{k+1}, the Lagrange multiplier is updated by

$$\mu_{k+1} = \mu_k + \alpha h(x_{k+1}),$$

  where $\alpha > 0 \rightarrow$ **Uzawa's algorithm**.

Lagrangian decomposition

Remarks.

- Convergence of such methods can be established only under **convexity assumptions**.
- The stepsize $\alpha > 0$ must in general be small enough to ensure convergence. Instead of a fixed stepsize, one can use

$$\lambda_{k+1} = \lambda_k + \alpha_k g(x_{k+1}),$$

- One may consider instead of the primal problem (P) the dual problem

$$d^* := \sup_{\mu \in \mathbb{R}^m} \inf_{x \in \mathbb{R}^n} L(x, \mu) \qquad (P_{\mu_k})$$

and we have $p^* \geq d^*$.

Lagrangian decomposition

Main advantage of Lagrangian decomposition: very often the minimization of L can be "parallelized".

Standard case: additive constraints.

- Consider

$$\inf_{(x_1,x_2)\in X_1\times X_2} f_1(x_1)+f_2(x_2), \quad \text{subject to: } h_1(x_1)+h_2(x_2)=d,$$

  where f_1, f_2, X_1, X_2, h_1, h_2, and d are given.

- Lagrangian:

$$\begin{aligned} L(x_1,x_2,\mu) &= f_1(x_1)+f_2(x_2)+\langle \mu, h_1(x_1)+h_2(x_2)-d\rangle \\ &= \Big[\underbrace{f_1(x_1)+\langle \mu, h_1(x_1)\rangle}_{=:L_1(x_1,\mu)}\Big]+\Big[\underbrace{f_2(x_2)+\langle \mu, h_2 x_2\rangle}_{=:L_2(x_2,\mu)}\Big]-\langle \mu, d\rangle. \end{aligned}$$

Lagrangian decomposition

Given μ, the minimization of $L(\cdot, \lambda)$ is **decomposed** into two subproblems:

$$\inf_{x_1 \in \mathbb{R}^{n_1}} L_1(x_1, \lambda) \qquad \text{and} \qquad \inf_{x_2 \in \mathbb{R}^{n_2}} L_2(x_2, \lambda),$$

which can be solved independently. Very often the two subproblems are **much easier** to solve than the original problem.

Remark. Straightforward **generalization** to the case

$$\inf_{\substack{x_1, \dots, x_K \\ \in \mathbb{R}^{n_1} \times \dots \mathbb{R}^{n_K}}} f_1(x_1) + \ldots + f_K(x_K), \quad \text{s.t.: } h_1(x_1) + \ldots + h_K(x_K) = K.$$

$\rightarrow$ Decomposition in K subproblems (at each iteration).

Lagrangian decomposition

1. **Application 1:** time decomposition.
   - **Two production units**, with two independent production processes represented by the variables
     Problem:

$$\inf_{x\in\mathbb{R}^2} -\frac{x_1}{1+x_1} - \frac{x_2}{4+x_2}, \quad \text{s.t. } \left\{ x_1 + x_2 = d \right.$$

Lagrangian decomposition

$$L(x,\mu) = -\frac{x_1}{1+x_1} - \frac{x_2}{4+x_2} + \mu(x_1+x_2-d).$$

If $x_1 + x_2 > d$, the engine must be rented for a longer time: the cost associated to constraints is increased. The incentive μ_k is too small, it must be increased.

If $x_1 + x_2 < d$, the cost associated to constraints is decreased. The incentive μ_k is too big, it must be decreased.
This is consistent with the formula

$$\mu_{k+1} = \mu_k + \alpha_k(x_1 + x_2 - d) \tag{4.1}$$

Lagrangian decomposition

Application 2: stochastic decomposition.

- A production process is decomposed over two periods.
  A **random event** with two outcomes ω_1 and ω_2, with probabilities p and $(1-p)$, arises inbetween.
- Optimization variables:
  - x_1: decisions taken if outcome ω_1 arises
  - x_2: decisions taken if outcome ω_2 arises
  - y: decisions taken before the random event.

  Example: purchase of gas y on a day-ahead market (that is, on a given day for the next one).
  Random event: temperature, which impacts consumption.

Lagrangian decomposition

- Abstract problem:

$$\inf_{\substack{(x_1,x_2,y)\\(x_1,y)\in X\\(x_2,y)\in X}} pf(x_1,y,\omega_1)+(1-p)f(x_2,y,\omega_2).$$

- Equivalent problem (with non-anticipativity constraint):

$$\inf_{\substack{(x_1,x_2,y_1,y_2)\\(x_1,y_1)\in X\\(x_2,y_2)\in X}} pf(x_1,y_1,\omega_1)+(1-p)f(x_2,y_2,\omega_2), \quad \text{s.t. } y_2-y_1=0.$$

- Independent (w.r.t. randomness) sub-problems:

$$\inf_{(x_1,y_1)\in X_1} pf_1(x_1,y_1,\omega_1)+\mu_k y_1, \qquad \inf_{(x_2,y_2)\in X_2}(1-p)f_2(x_2,y_2,\omega_2)-\mu_k y_2.$$

Projection

Idea: Apply steepest descent method but project the path onto the constraints. The projected gradient method uses a mapping called **projection** defined below.

Lemma 4

Let $K \subset \mathbb{R}^n$ be a non-empty, convex, and closed set. For all $x_0 \in \mathbb{R}^n$, there exists a **unique solution** *to the problem*

$$\inf_{x \in \mathbb{R}^n} \|x - x_0\|^2, \quad \textit{subject to: } x \in K.$$

It is called **projection** *of x_0 on K, and denoted $Proj_K(x_0)$.*

Remark. The projection depends on the chosen norm $\|\cdot\|$. For simplicity, we consider the Euclidean norm.

Projection

Example 1: projection on a cuboid.
Let K be described by

$$K = \left\{ x \in \mathbb{R}^n \,|\, \ell_i \leq x_i \leq u_i \right\},$$

where the coefficients $\ell_1,\ldots,\ell_n \in \mathbb{R} \cup \{-\infty\}$ and $u_1,\ldots,u_n \in \mathbb{R} \cup \{+\infty\}$ are given.

Let $x \in \mathbb{R}^n$, let $y = \text{Proj}_K(x)$. Then

$$y_i = \min(\max(x_i, \ell_i), u_i), \quad \forall i = 1, \ldots, n.$$

Projection

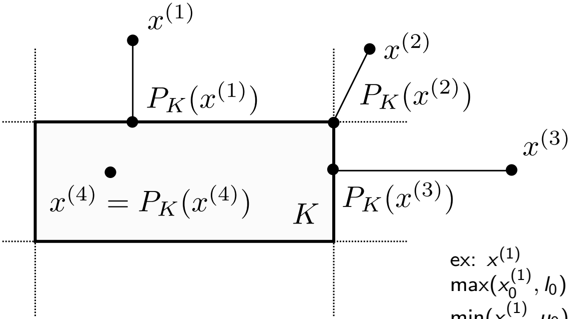


Figure: Projection on a cuboid.

ex: $x^{(1)}$

$\max(x_0^{(1)}, l_0) = x_0^{(1)}$

$\min(x_0^{(1)}, u_0) = x_0^{(1)}$

$\max(x_1^{(1)}, l_1) = x_1^{(1)}$

$\min(x_1^{(1)}, u_1) = u_1$

Projection

Example 2: projection on a ball.
Let K be described by

$$K = \left\{ x \in \mathbb{R}^n \mid \|x - x_C\| \leq R \right\},$$

where $x_C \in \mathbb{R}^n$ and $R \geq 0$ are given.

For all $x \in \mathbb{R}^n$,

$$\mathrm{Proj}_K(x) = x_C + \min\left(\|x - x_C\|, R\right) \frac{(x - x_C)}{\|x - x_C\|}.$$

Projection

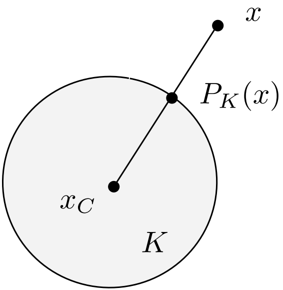


Figure: Projection on a ball.

Projection

Example 3: cartesian product.
Let K be given by

$$K = K_1 \times K_2,$$

where K_1 and K_2 are given non-empty closed and convex subsets of $\mathbb{R}^{n_1}$ and $\mathbb{R}^{n_2}$.

Then for all $x = (x_1, x_2) \in \mathbb{R}^{n_1+n_2}$,

$$\mathrm{Proj}_K(x) = \Big(\mathrm{Proj}_{K_1}(x_1), \mathrm{Proj}_{K_2}(x_2)\Big).$$

Method

Optimization problem. Consider

$$\inf_{x \in \mathbb{R}^n} f(x), \quad x \in K,$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is given and differentiable and K is a given non-empty **convex** and **closed** subset of $\mathbb{R}^n$.

Numerical assumption: $\mathrm{Proj}_K(\cdot)$ is **easy to compute**.

Gradient descent algorithm.

1. Input: $x_0 \in \mathbb{R}^n$, $\varepsilon > 0$. Set $k = 0$.
2. While $\|\nabla f(x_k)\| \geq \varepsilon$, do
   - Find a descent direction d_k.
   - Find $\alpha_k > 0$ such that $f(x_k + \alpha_k d_k) < f(x_k)$.
   - Set $x_{k+1} = x_k + \alpha_k d_k$.
   - Set $k = k + 1$.
3. Output: x_k.

Main idea:
at iteration k, replace the search on the half line $\{x_k + \alpha_k d_k \mid \alpha \geq 0\}$ used in unconstrained optimization by a **search** on

$$\{\underbrace{\mathrm{Proj}_K(x_k + \alpha_k d_k)}_{=:x_{k+1}(\alpha_k)} \mid \alpha_k \geq 0\}.$$

Combination with penalty methods

Consider the problem

$$\inf_{x\in\mathbb{R}^n} f(x), \quad \text{subject to: } \begin{cases} h_i(x) = 0 & \forall i \in \mathcal{E}, \\ g_i(x) \leq 0 & \forall i \in \mathcal{I}, \end{cases}$$

where $f\colon \mathbb{R}^n \to \mathbb{R}$, $h\colon \mathbb{R}^n \to \mathbb{R}^{m^1}$ and $g\colon \mathbb{R}^n \to \mathbb{R}^{m_2}$ are given.

Idea: Eliminate inequality constraints by slack variables. An **equivalent formulation** is

$$\inf_{\substack{x\in\mathbb{R}^n\\ y\in\mathbb{R}^m}} f(x), \quad \text{subject to: } \begin{cases} \Phi(x) - y = 0 \\ y \in K, \end{cases}$$

where: $\Phi_i(x) = \begin{cases} h_i(x), \ \forall i \in \mathcal{E}, \\ g_i(x), \ \forall i \in \mathcal{I}, \end{cases}$ and

$$K = \left\{ y \in \mathbb{R}^m \,\middle|\, \begin{cases} y_i = 0 & \forall i \in \mathcal{E} \\ y_i \leq 0 & \forall i \in \mathcal{I} \end{cases} \right\}.$$

Combination with penalty methods

Main idea: projection on K (a cuboid) is easy to compute.
Handle $y \in K$ with the projected gradient method.

Algorithm.

- At iteration k, the iterates $x_k \in \mathbb{R}^n$, $y_k \in \mathbb{R}^m$, $\mu_k \in \mathbb{R}^m$, and c_k are given.
- Solve (approximately) the penalty problem:

$$\inf_{\substack{x \in \mathbb{R}^n \\ y \in \mathbb{R}^m}} L_{c_k}(x, y, \mu_k) := f(x) + \langle \mu_k, \Phi(x) - y \rangle + \frac{c_k}{2} \|\Phi(x) - y\|^2,$$

subject to: $y \in K$,

with the projected gradient method.
Use (x_k, y_k) as a starting point.