

Constrained optimization

V. Leclère (ENPC)

May 16th, 2025

Why should I bother to learn this stuff?

- Most real problems have constraints that you have to deal with.
- This course give a snapshot of the tools available to you.
- \implies useful for
 - ▶ having an idea of what can be done when you have constraints

Constrained optimization problem

- In the previous courses we have developed algorithms for **unconstrained** optimization problem.
- We now want to sketch some methods to deal with the constrained problem

$$\begin{array}{ll} \text{Min}_{x \in \mathbb{R}^n} & f(x) \\ \text{s.t.} & x \in X \end{array}$$

- We are going to discuss multiple types of constraints set X :
 - ▶ X is a ball : $\{x \mid \|x - x_0\|_2 \leq r\}$
 - ▶ X is a box : $\{x \mid \underline{x}_i \leq x_i \leq \bar{x}_i \quad \forall i \in [n]\}$
 - ▶ X is a polyhedron: $\{x \mid Ax \leq b\}$
 - ▶ X is given through explicit constraints $\{x \mid g(x) = 0, \quad h(x) \leq 0\}$

How to deal with constraints: a method map



We want to solve

$$\min_{x \in X} f(x),$$

where X may be simple, polyhedral, or defined by general constraints.

Constraint structure	Main idea	Typical methods
X simple (ball, box, easy cone)	Project after each step	Projected gradient, proximal methods
X polyhedral / simplex	Minimize linearization over X	Conditional gradient (Frank–Wolfe)
General $g(x) = 0, h(x) \leq 0$	Move constraints into the cost	Quadratic / L^1 penalization, augmented Lagrangian
General, separable structure	Dualize coupling constraints	Dual ascent, Uzawa, decomposition

How to deal with constraints: a method map



We want to solve

$$\min_{x \in X} f(x),$$

where X may be simple, polyhedral, or defined by general constraints.

Constraint structure	Main idea	Typical methods
X simple (ball, box, easy cone)	Project after each step	Projected gradient, proximal methods
X polyhedral / simplex	Minimize linearization over X	Conditional gradient (Frank-Wolfe)
General $g(x) = 0, h(x) \leq 0$	Move constraints into the cost	Quadratic / L^1 penalization, augmented Lagrangian
General, separable structure	Dualize coupling constraints	Dual ascent, Uzawa, decomposition

How to deal with constraints: a method map



We want to solve

$$\min_{x \in X} f(x),$$

where X may be simple, polyhedral, or defined by general constraints.

Constraint structure	Main idea	Typical methods
X simple (ball, box, easy cone)	Project after each step	Projected gradient, proximal methods
X polyhedral / simplex	Minimize linearization over X	Conditional gradient (Frank-Wolfe)
General $g(x) = 0, h(x) \leq 0$	Move constraints into the cost	Quadratic / L^1 penalization, augmented Lagrangian
General, separable structure	Dualize coupling constraints	Dual ascent, Uzawa, decomposition

Contents

1 Constructing an admissible trajectory

- Admissible direction
- Projected direction

2 From constraints to cost

- Penalization
- Dualization

Contents

1 Constructing an admissible trajectory

- Admissible direction
- Projected direction

2 From constraints to cost

- Penalization
- Dualization

Admissible descent direction

- Recall that a descent direction d at point $x^{(k)} \in \mathbb{R}^n$ is a vector such that $\nabla f(x^{(k)})^\top d < 0$.
- An **admissible descent direction** at point $x^{(k)} \in X$ is a descent direction $d \in \mathbb{R}^n$ such that,

$$\exists \varepsilon > 0, \quad \forall t \in [0, \varepsilon], \quad x^{(k)} + td \in X.$$

- In other words, an admissible descent direction, is a direction that locally decreases the objective while staying in the constraint set.
- An admissible descent direction algorithm is naturally defined by:
 - ▶ A choice of admissible descent direction $d^{(k)}$
 - ▶ A choice of (sufficiently small) step $t^{(k)}$
 - ▶ $x^{(k+1)} = x^{(k)} + t^{(k)}d^{(k)} \in X$
- Warning: this does not necessarily converge. We can construct examples where the step size gets increasingly small because of the constraints.

A warning: a naive “masked gradient” can stall



Consider a feasible set $X = \mathbb{R}_+^n$ and a differentiable convex f . A tempting heuristic is to use the **masked gradient** direction

$$d_i^{(k)} := -\nabla_i f(x^{(k)}) \mathbb{1}_{x_i^{(k)} > 0},$$

i.e. we freeze coordinates that are currently on the boundary.

- This direction is *feasible* for the orthant (for small $t > 0$).
- **But** it is *not* a principled way to enforce first-order stationarity on X .
- The correct necessary condition is the **normal cone condition**

$$0 \in \nabla f(x^*) + N_X(x^*) \iff \begin{cases} x^* \geq 0, \\ \nabla f(x^*) \geq 0, \\ x_i^* \nabla_i f(x^*) = 0 \quad \forall i, \end{cases}$$

which the masking rule does not target.

Take-away. “Feasible direction” is not enough. You need a direction rule that is consistent with stationarity (e.g., projection/prox, FW, or a proper active-set / SQP logic).

A warning: a naive “masked gradient” can stall



Consider a feasible set $X = \mathbb{R}_+^n$ and a differentiable convex f . A tempting heuristic is to use the **masked gradient** direction

$$d_i^{(k)} := -\nabla_i f(x^{(k)}) \mathbb{1}_{x_i^{(k)} > 0},$$

i.e. we freeze coordinates that are currently on the boundary.

- This direction is *feasible* for the orthant (for small $t > 0$).
- **But** it is *not* a principled way to enforce first-order stationarity on X .
- The correct necessary condition is the **normal cone condition**

$$0 \in \nabla f(x^*) + N_X(x^*) \iff \begin{cases} x^* \geq 0, \\ \nabla f(x^*) \geq 0, \\ x_i^* \nabla_i f(x^*) = 0 \quad \forall i, \end{cases}$$

which the masking rule does not target.

Take-away. “Feasible direction” is not enough. You need a direction rule that is consistent with stationarity (e.g., projection/prox, FW, or a proper active-set / SQP logic).

A warning: a naive “masked gradient” can stall



Consider a feasible set $X = \mathbb{R}_+^n$ and a differentiable convex f . A tempting heuristic is to use the **masked gradient** direction

$$d_i^{(k)} := -\nabla_i f(x^{(k)}) \mathbb{1}_{x_i^{(k)} > 0},$$

i.e. we freeze coordinates that are currently on the boundary.

- This direction is *feasible* for the orthant (for small $t > 0$).
- **But** it is *not* a principled way to enforce first-order stationarity on X .
- The correct necessary condition is the **normal cone condition**

$$0 \in \nabla f(x^*) + N_X(x^*) \iff \begin{cases} x^* \geq 0, \\ \nabla f(x^*) \geq 0, \\ x_i^* \nabla_i f(x^*) = 0 \quad \forall i, \end{cases}$$

which the masking rule does not target.

Take-away. “Feasible direction” is not enough. You need a direction rule that is consistent with stationarity (e.g., projection/prox, FW, or a proper active-set / SQP logic).

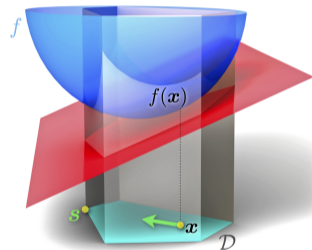
Conditional gradient algorithm

We address an optimization problem with a convex objective function f and compact polyhedral constraint set X , i.e.

$$\min_{x \in X \subset \mathbb{R}^n} f(x)$$

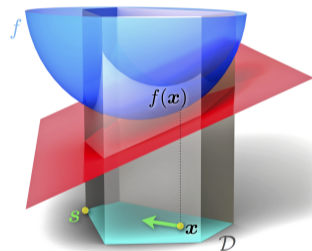
where

$$X = \{x \in \mathbb{R}^n \mid Ax \leq b, \quad \tilde{A}x = \tilde{b}\}$$



Conditional gradient algorithm

It is a descent algorithm, where we first look for an admissible descent direction $d^{(k)}$, and then look for the optimal step.

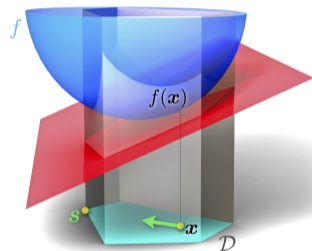


Conditional gradient algorithm

It is a descent algorithm, where we first look for an admissible descent direction $d^{(k)}$, and then look for the optimal step.

As f is convex, we know that for any point $x^{(k)}$,

$$f(y) \geq f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)})$$



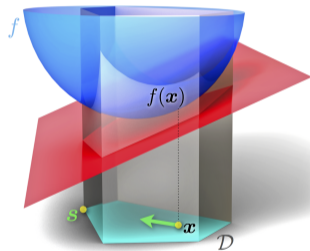
Conditional gradient algorithm

It is a descent algorithm, where we first look for an admissible descent direction $d^{(k)}$, and then look for the optimal step.

As f is convex, we know that for any point $x^{(k)}$,

$$f(y) \geq f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)})$$

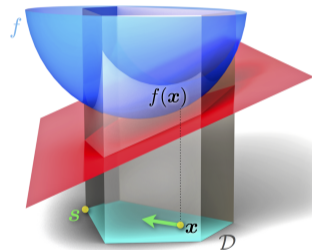
The conditional gradient method consists in choosing the descent direction that minimizes the linearization of f over X .



Conditional gradient algorithm

The conditional gradient method consists in choosing the descent direction that minimizes the linearization of f over X . More precisely, at step k we solve

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$



Conditional gradient (Frank–Wolfe): update + certificate



Given $x^{(k)} \in X$, compute a **linear minimization oracle**

$$y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)})^\top y, \quad d^{(k)} := y^{(k)} - x^{(k)}.$$

- **Update (always feasible):**

$$x^{(k+1)} = (1 - \gamma_k)x^{(k)} + \gamma_k y^{(k)}, \quad \gamma_k \in [0, 1]$$

(via line search or a fixed rule, e.g. $\gamma_k = \frac{2}{k+2}$ for smooth convex).

- **Frank–Wolfe gap (optimality certificate):**

$$g_{\text{FW}}(x^{(k)}) := \nabla f(x^{(k)})^\top (x^{(k)} - y^{(k)}) \geq 0.$$

If f is convex, then $f(x^{(k)}) - f^* \leq g_{\text{FW}}(x^{(k)})$ and $g_{\text{FW}}(x^{(k)}) = 0 \Leftrightarrow x^{(k)}$ optimal.

Conditional gradient (Frank–Wolfe): update + certificate



Given $x^{(k)} \in X$, compute a **linear minimization oracle**

$$y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)})^\top y, \quad d^{(k)} := y^{(k)} - x^{(k)}.$$

- **Update (always feasible):**

$$x^{(k+1)} = (1 - \gamma_k)x^{(k)} + \gamma_k y^{(k)}, \quad \gamma_k \in [0, 1]$$

(via line search or a fixed rule, e.g. $\gamma_k = \frac{2}{k+2}$ for smooth convex).

- **Frank–Wolfe gap (optimality certificate):**

$$g_{\text{FW}}(x^{(k)}) := \nabla f(x^{(k)})^\top (x^{(k)} - y^{(k)}) \geq 0.$$

If f is convex, then $f(x^{(k)}) - f^* \leq g_{\text{FW}}(x^{(k)})$ and $g_{\text{FW}}(x^{(k)}) = 0 \Leftrightarrow x^{(k)}$ optimal.

Remarks on conditional gradient

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$

- This problem is linear, hence easy to solve.
- By the convexity inequality, the value of the linearized Problem is a lower bound to the true problem.
- As $y^{(k)} \in X$, $d^{(k)} = y^{(k)} - x^{(k)}$ is a *feasible direction*, in the sense that for all $t \in [0, 1]$, $x^{(k)} + td^{(k)} \in X$.
- If $y^{(k)}$ is obtained through the simplex method it is an extreme point of X , which means that, for $t > 1$, $x^{(k)} + td^{(k)} \notin X$.
- If $y^{(k)} = x^{(k)}$ then we have found an optimal solution.
- We also have $y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)}) \cdot y$, the lower-bound being obtained easily.

Remarks on conditional gradient

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$

- This problem is linear, hence easy to solve.
- By the convexity inequality, the value of the linearized Problem is a lower bound to the true problem.
- As $y^{(k)} \in X$, $d^{(k)} = y^{(k)} - x^{(k)}$ is a *feasible direction*, in the sense that for all $t \in [0, 1]$, $x^{(k)} + td^{(k)} \in X$.
- If $y^{(k)}$ is obtained through the simplex method it is an extreme point of X , which means that, for $t > 1$, $x^{(k)} + td^{(k)} \notin X$.
- If $y^{(k)} = x^{(k)}$ then we have found an optimal solution.
- We also have $y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)}) \cdot y$, the lower-bound being obtained easily.

Remarks on conditional gradient

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$

- This problem is linear, hence easy to solve.
- By the convexity inequality, the value of the linearized Problem is a lower bound to the true problem.
- As $y^{(k)} \in X$, $d^{(k)} = y^{(k)} - x^{(k)}$ is a *feasible direction*, in the sense that for all $t \in [0, 1]$, $x^{(k)} + td^{(k)} \in X$.
- If $y^{(k)}$ is obtained through the simplex method it is an extreme point of X , which means that, for $t > 1$, $x^{(k)} + td^{(k)} \notin X$.
- If $y^{(k)} = x^{(k)}$ then we have found an optimal solution.
- We also have $y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)}) \cdot y$, the lower-bound being obtained easily.

Remarks on conditional gradient

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$

- This problem is linear, hence easy to solve.
- By the convexity inequality, the value of the linearized Problem is a lower bound to the true problem.
- As $y^{(k)} \in X$, $d^{(k)} = y^{(k)} - x^{(k)}$ is a *feasible direction*, in the sense that for all $t \in [0, 1]$, $x^{(k)} + t d^{(k)} \in X$.
- If $y^{(k)}$ is obtained through the simplex method it is an extreme point of X , which means that, for $t > 1$, $x^{(k)} + t d^{(k)} \notin X$.
- If $y^{(k)} = x^{(k)}$ then we have found an optimal solution.
- We also have $y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)}) \cdot y$, the lower-bound being obtained easily.

Remarks on conditional gradient

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$

- This problem is linear, hence easy to solve.
- By the convexity inequality, the value of the linearized Problem is a lower bound to the true problem.
- As $y^{(k)} \in X$, $d^{(k)} = y^{(k)} - x^{(k)}$ is a *feasible direction*, in the sense that for all $t \in [0, 1]$, $x^{(k)} + t d^{(k)} \in X$.
- If $y^{(k)}$ is obtained through the simplex method it is an extreme point of X , which means that, for $t > 1$, $x^{(k)} + t d^{(k)} \notin X$.
- If $y^{(k)} = x^{(k)}$ then we have found an optimal solution.
- We also have $y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)}) \cdot y$, the lower-bound being obtained easily.

Remarks on conditional gradient

$$y^{(k)} \in \arg \min_{y \in X} f(x^{(k)}) + \nabla f(x^{(k)}) \cdot (y - x^{(k)}).$$

- This problem is linear, hence easy to solve.
- By the convexity inequality, the value of the linearized Problem is a lower bound to the true problem.
- As $y^{(k)} \in X$, $d^{(k)} = y^{(k)} - x^{(k)}$ is a *feasible direction*, in the sense that for all $t \in [0, 1]$, $x^{(k)} + t d^{(k)} \in X$.
- If $y^{(k)}$ is obtained through the simplex method it is an extreme point of X , which means that, for $t > 1$, $x^{(k)} + t d^{(k)} \notin X$.
- If $y^{(k)} = x^{(k)}$ then we have found an optimal solution.
- We also have $y^{(k)} \in \arg \min_{y \in X} \nabla f(x^{(k)}) \cdot y$, the lower-bound being obtained easily.

Contents

1 Constructing an admissible trajectory

- Admissible direction
- Projected direction

2 From constraints to cost

- Penalization
- Dualization



Let $X \subset \mathbb{R}^n$ be a nonempty closed convex set. We call $P_X : \mathbb{R}^n \rightarrow \mathbb{R}^n$ the **projection on X** the function such that

$$P_X(x) = \arg \min_{x' \in X} \|x' - x\|_2^2$$

We have

- $\bar{x} = P_X(x)$ iff $(x - \bar{x}) \in N_X(\bar{x})$ (i.e. $\langle x - \bar{x}, x' - \bar{x} \rangle \leq 0, \quad \forall x' \in X$)
- $\langle P_X(y) - P_X(x), y - x \rangle \geq 0$ (P_X is *non-decreasing*)
- $\|P_X(y) - P_X(x)\|_2 \leq \|y - x\|$ (P_X is a *contraction*)

♠ Exercise: Prove these results



Let $X \subset \mathbb{R}^n$ be a nonempty closed convex set. We call $P_X : \mathbb{R}^n \rightarrow \mathbb{R}^n$ the **projection on X** the function such that

$$P_X(x) = \arg \min_{x' \in X} \|x' - x\|_2^2$$

We have

- $\bar{x} = P_X(x)$ iff $(x - \bar{x}) \in N_X(\bar{x})$ (i.e. $\langle x - \bar{x}, x' - \bar{x} \rangle \leq 0, \quad \forall x' \in X$)
- $\langle P_X(y) - P_X(x), y - x \rangle \geq 0$ (P_X is *non-decreasing*)
- $\|P_X(y) - P_X(x)\|_2 \leq \|y - x\|$ (P_X is a *contraction*)

♠ Exercise: Prove these results

Consider

$$\begin{array}{ll} \text{Min}_{x \in \mathbb{R}^n} & f(x) \\ \text{s.t.} & x \in X \end{array}$$

where f is differentiable and X convex.

The projected gradient algorithm generates the following sequence

$$x^{(k+1)} = P_X[x^{(k)} - t^{(k)} g^{(k)}]$$

Projected gradient: stationarity + convergence statement



Let $X \subset \mathbb{R}^n$ be nonempty closed convex and f differentiable.

Stationarity on X

$x^\sharp \in X$ is (first-order) stationary iff

$$0 \in \nabla f(x^\sharp) + N_X(x^\sharp) \iff x^\sharp = P_X(x^\sharp - t \nabla f(x^\sharp)) \text{ for some (equiv. all) } t > 0.$$

Projected gradient mapping (certificate)

$$G_t(x) := \frac{1}{t} \left(x - P_X(x - t \nabla f(x)) \right), \quad G_t(x) = 0 \iff x \text{ stationary.}$$

If f has L -Lipschitz gradient and $t \in (0, 2/L)$ is fixed, projected gradient satisfies $f(x^{(k)})$ decreases and $\|G_t(x^{(k)})\| \rightarrow 0$. If f is convex, every cluster point is optimal; if the minimizer is unique (e.g. f strongly convex), then $x^{(k)} \rightarrow x^*$.

Projected gradient: stationarity + convergence statement



Let $X \subset \mathbb{R}^n$ be nonempty closed convex and f differentiable.

Stationarity on X

$x^\sharp \in X$ is (first-order) stationary iff

$$0 \in \nabla f(x^\sharp) + N_X(x^\sharp) \iff x^\sharp = P_X(x^\sharp - t \nabla f(x^\sharp)) \text{ for some (equiv. all) } t > 0.$$

Projected gradient mapping (certificate)

$$G_t(x) := \frac{1}{t} \left(x - P_X(x - t \nabla f(x)) \right), \quad G_t(x) = 0 \iff x \text{ stationary.}$$

If f has L -Lipschitz gradient and $t \in (0, 2/L)$ is fixed, projected gradient satisfies $f(x^{(k)})$ decreases and $\|G_t(x^{(k)})\| \rightarrow 0$. If f is convex, every cluster point is optimal; if the minimizer is unique (e.g. f strongly convex), then $x^{(k)} \rightarrow x^*$.

Projected gradient: stationarity + convergence statement



Let $X \subset \mathbb{R}^n$ be nonempty closed convex and f differentiable.

Stationarity on X

$x^\sharp \in X$ is (first-order) stationary iff

$$0 \in \nabla f(x^\sharp) + N_X(x^\sharp) \iff x^\sharp = P_X(x^\sharp - t \nabla f(x^\sharp)) \text{ for some (equiv. all) } t > 0.$$

Projected gradient mapping (certificate)

$$G_t(x) := \frac{1}{t} \left(x - P_X(x - t \nabla f(x)) \right), \quad G_t(x) = 0 \iff x \text{ stationary.}$$

If f has L -Lipschitz gradient and $t \in (0, 2/L)$ is fixed, projected gradient satisfies $f(x^{(k)})$ decreases and $\|G_t(x^{(k)})\| \rightarrow 0$. If f is convex, every cluster point is optimal; if the minimizer is unique (e.g. f strongly convex), then $x^{(k)} \rightarrow x^*$.



- Projected gradient is useful only if the projection is simple, as projecting over a convex set consists in solving a constrained optimization problem.
- Projection is simple for balls and boxes.
- Finding an admissible direction is doable if the constraint set is polyhedral, or more generally conic-representable.

Contents

1 Constructing an admissible trajectory

- Admissible direction
- Projected direction

2 From constraints to cost

- Penalization
- Dualization

Contents

1 Constructing an admissible trajectory

- Admissible direction
- Projected direction

2 From constraints to cost

- Penalization
- Dualization



We consider the constrained optimization problem

$$\begin{aligned} (\mathcal{P}) \quad & \underset{x \in \mathbb{R}^n}{\text{Min}} && f(x) \\ & \text{s.t.} && x \in X \end{aligned}$$

and the following **penalized** version

$$(\mathcal{P}_r) \quad \underset{x \in \mathbb{R}^n}{\text{Min}} \quad f(x) + r p(x)$$

Thus, a (constrained) problem is replaced by a sequence of (unconstrained) problems.

♣ Exercise: What is happening if $p = \mathbb{I}_X$?



We consider the constrained optimization problem

$$\begin{aligned} (\mathcal{P}) \quad & \underset{x \in \mathbb{R}^n}{\text{Min}} && f(x) \\ & \text{s.t.} && x \in X \end{aligned}$$

and the following **penalized** version

$$(\mathcal{P}_r) \quad \underset{x \in \mathbb{R}^n}{\text{Min}} \quad f(x) + r p(x)$$

Thus, a (constrained) problem is replaced by a sequence of (unconstrained) problems.

♣ Exercise: What is happening if $p = \mathbb{I}_X$?



$$(\mathcal{P}_r) \quad \underset{x \in \mathbb{R}^n}{\text{Min}} \quad f(x) + r p(x)$$

The idea is that, with higher r , the penalization has more impact on the problem. More precisely, let $0 < r_1 < r_2$, and x_{r_i} be an optimal solution of (\mathcal{P}_{r_i}) .

We have:

- $p(x_{r_1}) \geq p(x_{r_2})$
- $f(x_{r_1}) \leq f(x_{r_2})$

♣ Exercise: prove these results.

Outer penalization

A first idea for choosing a penalization function p consists in choosing a function p such that:

- $p(x) = 0$ for $x \in X$
- $p(x) > 0$ for $x \notin X$

intuitively the idea is that p is the fine to pay for not respecting the constraint. Heuristically, it should be increasing with the distance to X .



Assume that

- p is l.s.c on \mathbb{R}^n
- $p \geq 0$
- $p(x) = 0$ iff $x \in X$

Further assume that f is l.s.c and there exists $r_0 > 0$ such that $x \mapsto f(x) + r_0 p(x)$ is coercive (i.e. $\rightarrow \infty$ if $\|x\| \rightarrow \infty$).

Then,

- 1 for $r > r_0$, (\mathcal{P}_r) admit at least one optimal solution
- 2 $(x_r)_{r \rightarrow +\infty}$ is bounded
- 3 any adherence point of $(x_r)_{r \rightarrow +\infty}$ is an optimal solution of \mathcal{P} .

Outer penalization - quadratic case

Assume that

$$X = \{x \in \mathbb{R}^n \mid g(x) = 0, \quad h(x) \leq 0\}$$

then the **quadratic penalization** consists in choosing

$$p : x \mapsto \|g(x)\|^2 + \|(h(x))^+\|^2$$

This choice is interesting as (for affinely lower-bounded f):

- if f, g, h are C^1 , then $x \mapsto f(x) + rp(x)$ is C^1
- as $r \rightarrow \infty$, any cluster point of (x_r) is feasible and (under standard assumptions) optimal for (\mathcal{P})

However, generally speaking, if the constraints are impactful (e.g. have non-zero optimal multipliers), then

$$x_r \notin X$$

Outer penalization - quadratic case

Assume that

$$X = \{x \in \mathbb{R}^n \mid g(x) = 0, \quad h(x) \leq 0\}$$

then the **quadratic penalization** consists in choosing

$$p : x \mapsto \|g(x)\|^2 + \|(h(x))^+\|^2$$

This choice is interesting as (for affinely lower-bounded f):

- if f, g, h are C^1 , then $x \mapsto f(x) + rp(x)$ is C^1
- as $r \rightarrow \infty$, any cluster point of (x_r) is feasible and (under standard assumptions) optimal for (\mathcal{P})

However, generally speaking, if the constraints are impactful (e.g. have non-zero optimal multipliers), then

$$x_r \notin X$$

Outer penalization - quadratic case

Assume that

$$X = \{x \in \mathbb{R}^n \mid g(x) = 0, \quad h(x) \leq 0\}$$

then the **quadratic penalization** consists in choosing

$$p : x \mapsto \|g(x)\|^2 + \|(h(x))^+\|^2$$

This choice is interesting as (for affinely lower-bounded f):

- if f, g, h are C^1 , then $x \mapsto f(x) + rp(x)$ is C^1
- as $r \rightarrow \infty$, any cluster point of (x_r) is feasible and (under standard assumptions) optimal for (\mathcal{P})

However, generally speaking, if the constraints are impactful (e.g. have non-zero optimal multipliers), then

$$x_r \notin X$$

Outer penalization - L^1 case

Assume that

$$X = \{x \in \mathbb{R}^n \mid g(x) = 0, \quad h(x) \leq 0\}$$

another natural penalization consists in choosing

$$p : x \mapsto \|g(x)\|_1 + \|(h(x))^+\|_1$$

The interest of this approach is that, if the problem is convex and the constraints are qualified at optimality, then, for r large enough, an optimal solution to the penalized problem (\mathcal{P}_r) is an optimal solution to the original problem (\mathcal{P}). Thus, we speak of **exact penalization**.

Unfortunately, this comes to the price of non-differentiability.

Outer penalization - L^1 case

Assume that

$$X = \{x \in \mathbb{R}^n \mid g(x) = 0, \quad h(x) \leq 0\}$$

another natural penalization consists in choosing

$$p : x \mapsto \|g(x)\|_1 + \|(h(x))^+\|_1$$

The interest of this approach is that, if the problem is convex and the constraints are qualified at optimality, then, for r large enough, an optimal solution to the penalized problem (\mathcal{P}_r) is an optimal solution to the original problem (\mathcal{P}). Thus, we speak of **exact penalization**.

Unfortunately, this comes to the price of non-differentiability.

Outer penalization - L^1 case

Assume that

$$X = \{x \in \mathbb{R}^n \mid g(x) = 0, \quad h(x) \leq 0\}$$

another natural penalization consists in choosing

$$p : x \mapsto \|g(x)\|_1 + \|(h(x))^+\|_1$$

The interest of this approach is that, if the problem is convex and the constraints are qualified at optimality, then, for r large enough, an optimal solution to the penalized problem (\mathcal{P}_r) is an optimal solution to the original problem (\mathcal{P}). Thus, we speak of **exact penalization**.

Unfortunately, this comes to the price of non-differentiability.

Inner penalization

Another approach consists in choosing a penalization function p that takes value $+\infty$ outside of X .

The idea here is to add a potential that keeps the *iterates* away from the boundary (while approaching optimality as the barrier vanishes).

This is typically done in a way to keep $f + \frac{1}{s}p$ smooth, and if possible convex.

Note that, for the inner penalization, we need the coefficient $\frac{1}{s} \rightarrow 0$, (hence $s \rightarrow +\infty$) for the penalized problem to converges toward the original one.

More on that in the next course.

Inner penalization

Another approach consists in choosing a penalization function p that takes value $+\infty$ outside of X .

The idea here is to add a potential that keeps the *iterates* away from the boundary (while approaching optimality as the barrier vanishes).

This is typically done in a way to keep $f + \frac{1}{s}p$ smooth, and if possible convex.

Note that, for the inner penalization, we need the coefficient $\frac{1}{s} \rightarrow 0$, (hence $s \rightarrow +\infty$) for the penalized problem to converges toward the original one.

More on that in the next course.

Inner penalization

Another approach consists in choosing a penalization function p that takes value $+\infty$ outside of X .

The idea here is to add a potential that keeps the *iterates* away from the boundary (while approaching optimality as the barrier vanishes).

This is typically done in a way to keep $f + \frac{1}{s}p$ smooth, and if possible convex.

Note that, for the inner penalization, we need the coefficient $\frac{1}{s} \rightarrow 0$, (hence $s \rightarrow +\infty$) for the penalized problem to converges toward the original one.

More on that in the next course.

Inner penalization

Another approach consists in choosing a penalization function p that takes value $+\infty$ outside of X .

The idea here is to add a potential that keeps the *iterates* away from the boundary (while approaching optimality as the barrier vanishes).

This is typically done in a way to keep $f + \frac{1}{s}p$ smooth, and if possible convex.

Note that, for the inner penalization, we need the coefficient $\frac{1}{s} \rightarrow 0$, (hence $s \rightarrow +\infty$) for the penalized problem to converges toward the original one.

More on that in the next course.

Contents

1 Constructing an admissible trajectory

- Admissible direction
- Projected direction

2 From constraints to cost

- Penalization
- Dualization

Duality, here we go again



Recall that to a primal problem

$$(\mathcal{P}) \quad \underset{x \in \mathbb{R}^n}{\text{Min}} \quad f(x) \quad (1)$$

$$\text{s.t.} \quad g(x) = 0 \quad (2)$$

$$h(x) \leq 0 \quad (3)$$

we associate the dual problem

$$(\mathcal{D}) \quad \underset{\lambda, \mu \geq 0}{\text{Max}} \quad \underbrace{\underset{x}{\text{Min}} \quad f(x) + \lambda^\top g(x) + \mu^\top h(x)}_{\Phi(\lambda, \mu)}$$

♣ Exercise: Under which sufficient conditions are these problems equivalent ?

Duality, here we go again



Recall that to a primal problem

$$(\mathcal{P}) \quad \underset{x \in \mathbb{R}^n}{\text{Min}} \quad f(x) \quad (1)$$

$$\text{s.t.} \quad g(x) = 0 \quad (2)$$

$$h(x) \leq 0 \quad (3)$$

we associate the dual problem

$$(\mathcal{D}) \quad \underset{\lambda, \mu \geq 0}{\text{Max}} \quad \underbrace{\underset{x}{\text{Min}} \quad f(x) + \lambda^\top g(x) + \mu^\top h(x)}_{\Phi(\lambda, \mu)}$$

♣ Exercise: Under which sufficient conditions are these problems equivalent ?

Duality as exact penalization (via a saddle point)



Consider the convex problem

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{s.t.} \quad g(\mathbf{x}) = 0, \quad h(\mathbf{x}) \leq 0, \quad \text{with Slater (so strong duality + multipliers).}$$

Let $(\mathbf{x}^*, \lambda^*, \mu^*)$ be a saddle point of the Lagrangian $L(\mathbf{x}, \lambda, \mu) = f(\mathbf{x}) + \lambda^\top g(\mathbf{x}) + \mu^\top h(\mathbf{x})$ with $\mu^* \geq 0$. Then

$$\mathbf{x}^* \in \arg \min_{\mathbf{x}} L(\mathbf{x}, \lambda^*, \mu^*),$$

and \mathbf{x}^* is primal optimal.

Interpretation. With the *right* multipliers, $L(\cdot, \lambda^*, \mu^*)$ acts like an exact penalization of the constraints.

Duality as exact penalization (via a saddle point)



Consider the convex problem

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad \text{s.t.} \quad g(\mathbf{x}) = 0, \quad h(\mathbf{x}) \leq 0, \quad \text{with Slater (so strong duality + multipliers).}$$

Let $(\mathbf{x}^*, \lambda^*, \mu^*)$ be a **saddle point** of the Lagrangian $L(\mathbf{x}, \lambda, \mu) = f(\mathbf{x}) + \lambda^\top g(\mathbf{x}) + \mu^\top h(\mathbf{x})$ with $\mu^* \geq 0$. Then

$$\mathbf{x}^* \in \arg \min_{\mathbf{x}} L(\mathbf{x}, \lambda^*, \mu^*),$$

and \mathbf{x}^* is primal optimal.

Interpretation. With the *right* multipliers, $L(\cdot, \lambda^*, \mu^*)$ acts like an exact penalization of the constraints.

Duality as exact penalization (via a saddle point)



Consider the convex problem

$$\min_x f(x) \quad \text{s.t.} \quad g(x) = 0, \quad h(x) \leq 0, \quad \text{with Slater (so strong duality + multipliers).}$$

Let (x^*, λ^*, μ^*) be a **saddle point** of the Lagrangian $L(x, \lambda, \mu) = f(x) + \lambda^\top g(x) + \mu^\top h(x)$ with $\mu^* \geq 0$. Then

$$x^* \in \arg \min_x L(x, \lambda^*, \mu^*),$$

and x^* is primal optimal.

Interpretation. With the *right* multipliers, $L(\cdot, \lambda^*, \mu^*)$ acts like an exact penalization of the constraints.

Dual ascent as projected (sub)gradient

Dual function:

$$\Phi(\lambda, \mu) := \inf_x (f(x) + \lambda^\top g(x) + \mu^\top h(x)), \quad \mu \geq 0,$$

is concave and typically **nonsmooth**.

If $x^\sharp(\lambda, \mu) \in \arg \min_x L(x, \lambda, \mu)$, then (Danskin)

$$\begin{pmatrix} g(x^\sharp(\lambda, \mu)) \\ h(x^\sharp(\lambda, \mu)) \end{pmatrix} \in \partial \Phi(\lambda, \mu).$$

Projected subgradient ascent with step $t > 0$:

$$\lambda^{(k+1)} = \lambda^{(k)} + t g(x^{(k+1)}), \quad \mu^{(k+1)} = [\mu^{(k)} + t h(x^{(k+1)})]^+.$$

Dual ascent as projected (sub)gradient

Dual function:

$$\Phi(\lambda, \mu) := \inf_x (f(x) + \lambda^\top g(x) + \mu^\top h(x)), \quad \mu \geq 0,$$

is concave and typically **nonsmooth**.

If $x^\sharp(\lambda, \mu) \in \arg \min_x L(x, \lambda, \mu)$, then (Danskin)

$$\begin{pmatrix} g(x^\sharp(\lambda, \mu)) \\ h(x^\sharp(\lambda, \mu)) \end{pmatrix} \in \partial \Phi(\lambda, \mu).$$

Projected subgradient ascent with step $t > 0$:

$$\lambda^{(k+1)} = \lambda^{(k)} + t g(x^{(k+1)}), \quad \mu^{(k+1)} = [\mu^{(k)} + t h(x^{(k+1)})]^+.$$

Dual ascent as projected (sub)gradient

Dual function:

$$\Phi(\lambda, \mu) := \inf_x (f(x) + \lambda^\top g(x) + \mu^\top h(x)), \quad \mu \geq 0,$$

is concave and typically **nonsmooth**.

If $x^\sharp(\lambda, \mu) \in \arg \min_x L(x, \lambda, \mu)$, then (Danskin)

$$\begin{pmatrix} g(x^\sharp(\lambda, \mu)) \\ h(x^\sharp(\lambda, \mu)) \end{pmatrix} \in \partial \Phi(\lambda, \mu).$$

Projected subgradient ascent with step $t > 0$:

$$\lambda^{(k+1)} = \lambda^{(k)} + t g(x^{(k+1)}), \quad \mu^{(k+1)} = [\mu^{(k)} + t h(x^{(k+1)})]^+.$$

Uzawa's algorithm

Data: Initial primal point $x^{(0)}$, Initial dual points $\lambda^{(0)}, \mu^{(0)}$, unconstrained optimization method, dual step $t > 0$.

while $\|g(x^{(k)})\|_2 + \|(h(x^{(k)}))^+\|_2 \geq \varepsilon$ **do**

Solve for $x^{(k+1)}$

$$\underset{x}{\text{Min}} \quad f(x) + \lambda^{(k)\top} g(x) + \mu^{(k)\top} h(x)$$

Update the multipliers

$$\lambda^{(k+1)} = \lambda^{(k)} + t g(x^{(k+1)})$$

$$\mu^{(k+1)} = [\mu^{(k)} + t h(x^{(k+1)})]^+$$

Algorithm 1: Uzawa algorithm

Convergence requires strong convexity and constraint qualifications.

Exercise: decomposition by prices

We consider the following energy problem:

- you are an energy producer with N production unit
 - you have to satisfy a given demand planning for the next 24h (i.e. the total output at time t should be equal to d_t)
 - the time step is the hour, and each unit has a production cost for each planning given as a convex quadratic function of the planning
- 1 Model this problem as an optimization problem. In which class does it belong? How many variables?
 - 2 Apply Uzawa's algorithm to this problem. Why could this be an interesting idea?
 - 3 Give an economic interpretation of this method.
 - 4 What would happen if each unit had production constraints?

What you have to know

- There is four main ways of dealing with constraints:
 - ▶ choosing an admissible direction
 - ▶ projection of the next iterate
 - ▶ penalizing the constraints
 - ▶ dualizing the constraints

What you really should know

- admissible direction methods are mainly useful for polyhedral constraint set
- projection is useful only if the admissible set is simple (ball or bound constraints)
- penalization can be inner or outer, differentiable or not.

What you have to be able to do

- Implement a penalization approach.

What you should be able to do

- Implement Uzawa's algorithm.