Aggregative optimization problems: relaxation and numerical resolution

Laurent Pfeiffer
Inria and CentraleSupélec, Université Paris-Saclay

Joint work with Frédéric Bonnans (Inria, L2S), Kang Liu (Polytechnique, L2S), Nadia Oudjane and Cheng Wan (EDF).

Journées-Ateliers FIME
EDF
September 13, 2023

Introduction

We investigate large scale **aggregative optimization problem**.

- Approximation by a convex mean-field optimization problem.
- Estimation of the relaxation gap.
- Numerical resolution with the **conditional gradient algorithm** (also called **Frank-Wolfe** algorithm).

Bonnans, Liu, Oudjane, Pfeiffer, Wan. Large-scale nonconvex optimization: randomization, gap estimation, and numerical resolution, *SIAM J. Optim.*, to appear.

Setting

Consider the N-agent problem

$$\inf_{x \in \mathcal{X}} J(x) = f\Big(\underbrace{\frac{1}{N} \sum_{i=1}^{N} g_i(x_i)}_{\text{aggregate}} \Big) + \frac{1}{N} \sum_{i=1}^{N} h_i(x_i), \qquad (\mathcal{P})$$

where $x = (x_1, \ldots, x_N) \in \mathcal{X} = \prod_{i=1}^{N} \mathcal{X}_i$.

Data:

- the feasible sets $\mathcal{X}_i$
- the individual costs $h_i \colon \mathcal{X}_i \to \mathbb{R}$
- the aggregate space $\mathcal{E}$, a Hilbert space
- the contribution functions $g_i \colon \mathcal{X}_i \to \mathcal{E}$
- the social cost $f \colon \mathcal{E} \to \mathbb{R}$.

Application

Applications in energy management problems:

- Set of agents: a (large) set of **small flexible consumptions units** (e.g. batteries, heating devices).
  Flexible: consumption can be shifted over time.
- Aggregate: the **total consumption**, at each time step of a given time interval.
- Social cost: **penalty function** for the difference between total consumption and a reference production level.

Wang. Vanishing Price of Decentralization in Large Coordinative Nonconvex Optimization, *SIAM J. Optimization*, 2017.

Séguret et al. Decomposition of convex high dimensional aggregative stochastic control problems, *Appl. Math Optim.*, 2023.

Applications

Our problem covers the case **training neural networks with a single hidden layer**.

- Social cost $\rightarrow$ fidelity function.
- Individual cost $\rightarrow$ regulizer.

We use the same kind of relaxation as in:

Chizat, Bach. On the Global Convergence of Gradient Descent for Over-parameterized Models using Optimal Transport, *Advances in Neural Information Processing Systems*, 2018.

Mei, Misiakiewicz, Montanari. Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit, *32nd Conf. on Learning Theory*, 2019.

Assumptions

Assumptions:

- f is convex
- ∇f is D-Lipschitz continuous
- for all $i = 1, \ldots, N$, $\text{diam}(g_i(\mathcal{X}_i)) \leq D$.

All constants appearing later on depend on D but not on N.
Another "numerical" assumption will be made later.

General difficulties:

- No convexity property of J.
- No regularity property for $\mathcal{X}_i$, g_i, h_i. In general, J is not differentiable.
- Large-scale (when N is large)... but N large actually helps!

Relaxation

General idea:

- Variable x_i replaced by a **probability distribution** $\mu_i \in \mathcal{P}(\mathcal{X}_i)$.
- The terms $g_i(x_i)$ and $h_i(x_i)$ are respectively replaced by

$$\mathbb{E}_{\mu_i}[g_i] := \int_{\mathcal{X}_i} g_i(x_i)\,\mathrm{d}\mu_i(x_i), \quad \mathbb{E}_{\mu_i}[h_i] := \int_{\mathcal{X}_i} h_i(x_i)\,\mathrm{d}\mu_i(x_i).$$

The relaxed problem:

$$\inf_{\mu} \ \tilde{J}(\mu) := f\Big(\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mu_i}[g_i]\Big) + \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mu_i}[h_i], \qquad (\tilde{\mathcal{P}})$$

where $\mu = (\mu_1, \ldots, \mu_N) \in \prod_{i=1}^{N} \mathcal{P}(\mathcal{X}_i)$.

Remark: The cost function $\tilde{J}$ is **convex**.

Mean field relaxation

Remark: In the **homonegeous** case where $\mathcal{X} = \mathcal{X}_i$, $g = g_i$, $h = h_i$, for all $i = 1, ..., N$, the original problem is equivalent to

$$\inf_{\mu \in \mathcal{P}_N(\mathcal{X})} f\left(\mathbb{E}_\mu[g]\right) + \mathbb{E}_\mu[h],$$

where $\mathcal{P}_N(\mathcal{X}) = \left\{ \frac{1}{N} \sum_{i=1}^N \delta_{x_i} \,\middle|\, x_i \in \mathcal{X},\ \forall i = 1, \ldots, N \right\}$.

The relaxed problem is equivalent to:

$$\inf_{\mu \in \mathcal{P}(\mathcal{X})} f\left(\mathbb{E}_\mu[g]\right) + \mathbb{E}_\mu[h],$$

in which μ models the **distribution of the decisions** of a continuum of agents.

Gap estimation

Theorem

There exists $C > 0$ (depending on D only) such that

$$\mathrm{Val}(\tilde{\mathcal{P}}) \leq \mathrm{Val}(\mathcal{P}) \leq \mathrm{Val}(\tilde{\mathcal{P}}) + \frac{C}{N}.$$

Proof. **Lower bound** of $\mathrm{Val}(\mathcal{P})$.
Let $x \in \mathcal{X}$. Let $\mu = (\delta_{x_1}, ..., \delta_{x_N})$. Then,

$$\mathrm{Val}(\tilde{\mathcal{P}}) \leq \tilde{J}(\mu) = J(x).$$

Minimizing with respect to x yields the result.

Gap estimation

Upper bound of $\mathrm{Val}(\mathcal{P})$. Let $\varepsilon > 0$. Let $\mu \in \prod_{i=1}^{N} \mathcal{P}(\mathcal{X}_i)$ be ε-optimal for the relaxed problem.
Let X_1,...,X_N be N independent random variables such that

$$\mathrm{Law}(X_i) = \mu_i, \quad i = 1, ..., N.$$

Then, setting $Y = \frac{1}{N} \sum_{i=1}^{N} g_i(X_i)$,

$$\begin{aligned} \tilde{J}(\mu) &= f\Big(\frac{1}{N} \sum_{i=1}^{N} \mathbb{E}[g_i(X_i)] \Big) + \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}[h_i(X_i)], \\ &= f(\mathbb{E}[Y]) + \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}[h_i(X_i)]. \end{aligned}$$

Therefore, $\mathbb{E}[J(X)] - \tilde{J}(\mu) = \mathbb{E}[f(Y)] - f(\mathbb{E}[Y])$.

Gap estimation

Using the Lipschitz continuity of ∇f, it is easy to show that:

$$\mathbb{E}[f(Y)] - f(\mathbb{E}[Y]) \leq \frac{D}{2}\mathbb{E}\left[\|Y - \mathbb{E}[Y]\|^2\right]$$

Since $Y = \frac{1}{N}\sum_{i=1}^N g_i(X_i)$ and since the X_i are independent,

$$\mathbb{E}\left[\|Y - \mathbb{E}[Y]\|^2\right] = \frac{1}{N^2}\sum_{i=1}^N \mathbb{E}\left[\|g_i(X_i) - \mathbb{E}[g_i(X_i)]\|^2\right] \leq \frac{D^2}{N}.$$

It finally follows that

$$\begin{aligned}\mathrm{Val}(\mathcal{P}) - \mathrm{Val}(\tilde{\mathcal{P}}) &\leq \mathbb{E}[J(X)] - \tilde{J}(\mu) + \varepsilon \\ &\leq \frac{L}{2}\mathbb{E}\left[\|Y - \mathbb{E}[Y]\|^2\right] + \varepsilon \leq \frac{D^2 L}{2N} + \varepsilon.\end{aligned}$$

Gap estimation

Theorem

Assume that $q := \dim \mathcal{E} + 1 \leq N$. There exists $C > 0$ (depending on D only) such that

$$\mathrm{Val}(\tilde{\mathcal{P}}) \leq \mathrm{Val}(\mathcal{P}) \leq \mathrm{Val}(\tilde{\mathcal{P}}) + \frac{Cq}{N^2}.$$

Proof. Let μ be as before. Using **Shapley-Folkman's** theorem, we can construct independent r.v. $\tilde{X}_i$, valued in $\mathcal{X}_i$ and such that

- $\tilde{J}(\mu) = f(\mathbb{E}[\tilde{Y}]) + \frac{1}{N} \sum_i \mathbb{E}[h_i(\tilde{X}_i)]$, where $\tilde{Y} = \frac{1}{N} \sum_{i=1}^N g_i(\tilde{X}_i)$,
- All r.v. $\tilde{X}_i$ are deterministic, except at most q of them.

Then $\mathbb{E}\big[\| \tilde{Y} - \mathbb{E}[\tilde{Y}]\|^2\big] \leq Cq/N^2$.

Frank-Wolfe algorithm

Consider the following problem:

$$\inf_{x \in \mathbb{R}^n} F(x), \quad \text{subject to: } x \in K. \qquad (\mathcal{P})$$

Assumptions:

- $F \colon \mathbb{R}^n \to \mathbb{R}$ is convex, continuously differentiable, with Lipschitz-continuous gradient.
- $K \subseteq \mathbb{R}^n$ is convex and compact.

The **linearized problem** at $\tilde{x}$ is defined by

$$\inf_{x \in \mathbb{R}^n} \langle \nabla F(\tilde{x}), x \rangle, \quad \text{subject to: } x \in K. \qquad (\mathcal{P}_{\text{lin}}(\tilde{x}))$$

We assume that it is easy to solve numerically, for any $\tilde{x}$.

Frank-Wolfe algorithm

Algorithm 1: Frank-Wolfe algorithm

Input: $\bar{x}_0 \in K$;
for $k = 0, 1, \ldots$ **do**
  Find a solution x_k to $\mathcal{P}_{\text{lin}}(\bar{x}_k)$;
  Set $\omega_k = 2/(k+2)$;
  Set $\bar{x}_{k+1} = (1-\omega_k)\bar{x}_k + \omega_k x_k$;
end

Lemma

There exists a constant C such that

$$f(\bar{x}_k) \leq f(\bar{x}) + \frac{C}{k}, \quad \forall k > 0,$$

where $\bar{x}$ denotes a solution of $(\mathcal{P})$.

The subproblem

We call any map $\mathbb{S}\colon \lambda \in \mathcal{E} \mapsto \big(\mathbb{S}_1(\lambda), \ldots, \mathbb{S}_N(\lambda)\big) \in \mathcal{X}$ a **best-response** function if for any $\lambda \in \mathcal{E}$,

$$\mathbb{S}_i(\lambda) \in \operatorname*{argmin}_{x_i \in \mathcal{X}_i} \langle \lambda, g_i(x_i) \rangle + h_i(x_i), \quad \text{for } i = 1, \ldots, N.$$

The variable λ can be here interpreted as a **price** for the contribution to the aggregate.

Numerical assumption. We assume that such a function can be easily constructed numerically. The evaluation of $\mathbb{S}$ relies on the resolution of N **independent** optimization problems.

The subproblem

Lemma

Let $\tilde{\mu} \in \prod_{i=1}^{N} \mathcal{P}(\mathcal{X}_i)$. *Let* $\lambda = \nabla f\left(\frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\tilde{\mu}_i}[g_i]\right)$. *Define*

$$\hat{\mu} = \left(\delta_{\mathbb{S}_1(\lambda)}, \ldots, \delta_{\mathbb{S}_N(\lambda)}\right).$$

Then $\hat{\mu}$ *is a solution to*

$$\inf_{\mu \in \prod_{i=1}^{N} \mathcal{P}(\mathcal{X}_i)} D\tilde{J}(\tilde{\mu}).\mu. \qquad (\tilde{\mathcal{P}}_{\text{lin}}(\tilde{\mu}))$$

Proof. Straightforward calculations yield:

$$D\tilde{J}(\tilde{\mu}).\mu = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\mu_i}\Big[\langle \lambda, g_i(\cdot)\rangle + h_i(\cdot)\Big].$$

Frank-Wolfe algorithm

Algorithm 2: Frank-Wolfe algorithm

Input: $\bar{\mu}^0$;

for $k = 0, 1, \dots$ **do**

Find a solution μ^k to $\tilde{\mathcal{P}}_{\text{lin}}(\bar{\mu}^k)$;

Set $\omega_k = \frac{2}{k+2}$;

Set $\bar{\mu}^{k+1} = (1 - \omega_k)\bar{\mu}^k + \omega_k \mu^k$;

end

Difficulties:

- How to deduce an **approximate solution** to $(\mathcal{P})$ from $\bar{\mu}^k$?
- The support of $\bar{\mu}_i^k$ possibly is of cardinality k.

Selection

Selection: A simple **stochastic method** for constructing $x \in \mathcal{X}$ out of $\mu \in \prod_{i=1}^{N} \mathcal{P}(\mathcal{X}_i)$.

Algorithm 3: Selection algorithm

Input: μ, $n \in \mathbb{N}$;
Construct a random variable $X = (X_1, ..., X_N)$ such that

$$X_1,...,X_N \text{ are independent}, \qquad \mathrm{Law}(X_i) = \mu_i.$$

for $j = 1, ..., n$ **do**
  Draw samples $\hat{x}^j = (x_1^j, ..., x_N^j)$ of $(X_1, ..., X_N)$.
end
Output: $\hat{x} \in \underset{x \in \{\hat{x}^1,...,\hat{x}^n\}}{\operatorname{argmin}} J(x)$.

Selection

Lemma

Let $\mu \in \prod_{i=1}^{N} \mathcal{P}(\mathcal{X}_i)$ *and let* $n \in \mathbb{N}$. *There exists a constant* $C > 0$ *such that for any* $\varepsilon > 0$,

$$\mathbb{P}\Big[J(\hat{x}) \geq \tilde{J}(\mu) + \frac{C}{N} + \varepsilon\Big] \leq \exp\Big(-\frac{nN\varepsilon^2}{C}\Big).$$

Proof. Let X be as in the selection algorithm. We know that

$$\tilde{J}(\mu) - \mathbb{E}[J(X)] \leq \frac{C}{N}.$$

Concentration inequality: by McDiarmid's inequality, there exists $C > 0$ such that for any $\varepsilon > 0$,

$$\mathbb{P}\Big[J(X) \geq \mathbb{E}[J(X)] + \varepsilon\Big] \leq \exp\Big(-\frac{N\varepsilon^2}{C}\Big).$$

Stochastic Frank-Wolfe (SFW) algorithm

Algorithm 4: Stochastic Frank-Wolfe algorithm

Input: $\bar{\mu}^0$, a sequence $(n_k)_{k\in\mathbb{N}}$;
for $k = 0, 1, \dots$ **do**
  Find a solution μ^k to $\tilde{\mathcal{P}}_{\text{lin}}(\bar{\mu}^k)$;
  Set $\omega_k = \frac{2}{k+2}$;
  Set $\tilde{\mu}^{k+1} = (1-\omega_k)\bar{\mu}^k + \omega_k \mu^k$;
  Set $\bar{x}^{k+1} = \text{Selection}(\tilde{\mu}^{k+1}, n_k)$;
  Set $\bar{\mu}^{k+1} = \left(\delta_{\bar{x}_1^{k+1}}, \dots, \delta_{\bar{x}_N^{k+1}}\right)$.
end

The algorithm can be re-written as an **easy-to-implement** algorithm that does not involve probability distributions.

Stochastic Frank-Wolfe algorithm

Algorithm 5: SFW algorithm: practical version

Input: $\bar{x}^{(0)}$, a sequence $(n_k)_{k\in\mathbb{N}}$;

for $k = 0, 1, \ldots$ **do**

  Set $\lambda^k = \nabla f(\frac{1}{N}\sum_{i=1}^{N} g_i(\bar{x}_i^k))$;

  Compute $x^k = \mathbb{S}(\lambda^k)$;

  Set $\omega_k = 2/(k+2)$;

  for $j = 1, \ldots, n_k$ **do**

    for $i = 1, \ldots, N$ **do**

      Draw $Z_i^{k,j} \sim (1-\omega_k)\delta_0 + \omega_k\delta_1$;

      Set $x_i^{k,j} = (1 - Z_i^{k,j})\bar{x}_i^k + Z_i^{k,j} x_i^k$;

    end

    Set $x^{k,j} = (x_i^{k,j})_{i=1,\ldots,N}$;

  end

  Find $\bar{x}^{(k+1)} \in \underset{x\in\{x^{k,1},\ldots,x^{k,n_k}\}}{\operatorname{argmin}} J(x)$

end

Convergence result

Theorem

There exists a constant $C > 0$ such that for all $K \leq 2N$, for all $\varepsilon > 0$, it holds:

$$\mathbb{P}\Big[J(\bar{x}^K) \geq \mathrm{Val}(\tilde{P}) + \frac{C}{K} + \varepsilon\Big] \leq \exp\Big(-\frac{N\varepsilon^2}{C_1(K) + \varepsilon C_2(K)}\Big),$$

where

$$C_1(K) = C \sum_{k=1}^{K-1} \frac{k(k+1)^2}{n_k K^2 (K+1)^2},$$

$$C_2(K) = C \max_{k \leq K-1} \frac{(k+1)(k+2)}{n_k K(K+1)}.$$

Remark. We can find a C/N-optimal solution with arbitrarily small probability if $n_k \geq Ak^2/N$, with A large enough.

Numerical example

Let $A \in \mathbb{R}^{M\times N}$ and let $\bar{y} \in \mathbb{R}^M$. Consider:

$$\min_{x\in\{0,1\}^N} \frac{1}{N^2}\|Ax - \bar{y}\|^2 = \left\| \frac{1}{N}\sum_{i=1}^{N} \left(A_i x_i - \frac{\bar{y}_i}{N} \right) \right\|^2 . \qquad \text{(MIQP)}$$

Data: $M = N = 100$.

Remark: Problem (MIQP) is a discrete problem, over a set of cardinality 2^{100}.

Numerical example

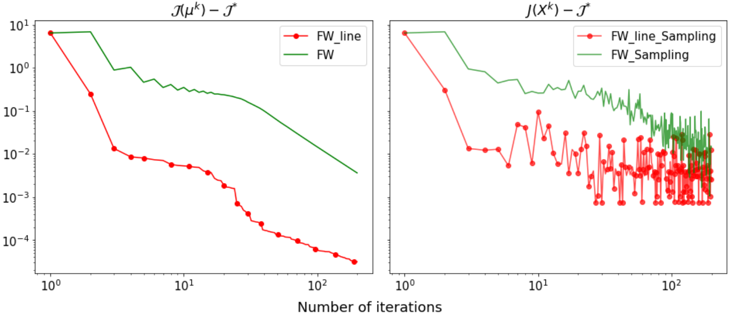

Figure: Convergence of the relaxed optimality gap.

Left: Frank-Wolfe for the relaxed problem.
Right: Selection algorithm applied to the iterates.

Numerical example

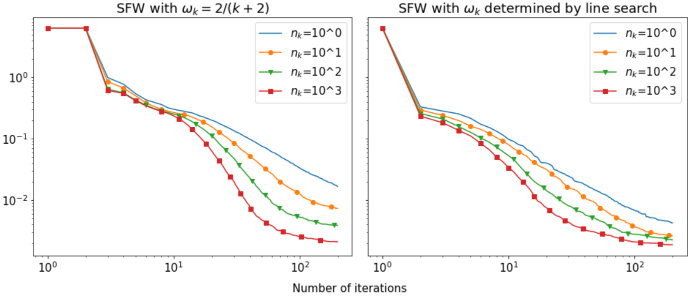

Figure: Relaxed optimality gap for Stochastic Frank-Wolfe algorithm.

Left: Stepsize $\delta_k = 2/(k+2)$.
Right: Stepsize determined by line-search.

Related works

1. **Two ideas for improvement:**
   - The convergence result for SFW is preserved if $\bar{x}^{k+1}$ is replaced by any other candidate x' such that $J(x') \leq J(\bar{x}^{k+1})$.
     $\rightarrow$ Motivates the design of **empirical** approaches.
   - In practical situations, the aggregative problem is "partially convex", i.e., is convex when some of the variables are fixed.
     $\rightarrow$ Motivates the **partial optimization** of the problem with the original Frank-Wolfe algorithm.

Related works

Numerical results (by Xinyu Huang, M2 student):

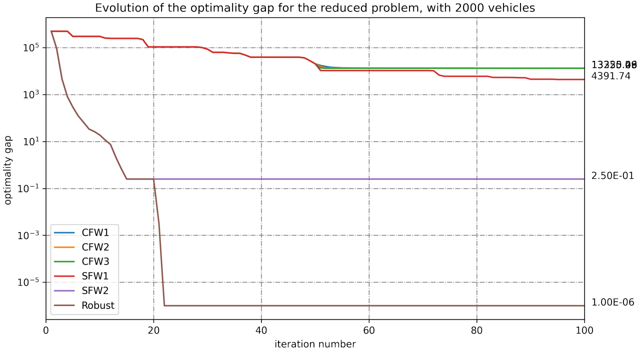

Figure: Red: SFW, Violet: SFW+ heuristic, Brown: SFW + heuristic + partial optimization.

Related works

2. The case of a non-smooth f.

- Concerning the **relaxation gap**, see:

Kerdreux, d'Aspremont, Colin: Stable Bounds on the Duality Gap of Separable Nonconvex Optimization Problems, *Maths Operations Research*, to appear.

- Ongoing work on non-smooth variants of the **Frank-Wolfe** algorithm (with Guilherme Mazanti and Thibault Moquet).

Silveti-Falls, Molinari, Fadili. Generalized conditional gradient with augmented lagrangian for composite minimization, *SIAM Journal on Optimization*, 2020.

Bach, Duality between subgradient and conditional gradient methods, *SIAM J. Optim.*, 2017.

Related works

3. **The case where x_i is a controlled dynamical system.**
   - The relaxed problem is a **mean-field optimal control problem** (an optimal control problem of the Fokker-Planck equation in continuous time).
   - **Frank-Wolfe** is applicable! Each sub-problem coincides with a standard stochastic optimal control problem.
   - In the case of second-order potential and convex MFG, **linear convergence** can be achieved.

Lavigne, Pfeiffer, Generalized conditional gradient and learning in potential mean field games, *Appl. Maths Optim.*, to appear.

Thank you for your attention!