Recent results on nonconvex Frank-Wolfe splitting algorithms

Frank-Wolfe, Moreau, and more

Zev Woodstock
woodstzc[at]jmu.edu
James Madison University (JMU)
Joint work with Nicholas Harsell, Antonio Silveti-Falls, Cesare Molinari, Sebastian Pokutta, and Victor Wang

Supported by NSF DMS-2532423

LANS Seminar
Argonne National Lab
Illinois, USA
September 2026

Frank-Wolfe Splitting Algorithms

1. Background
2. History & Motivation
3. Approach
4. Analysis
5. Conclusion

Optimization intro

Optimization in a nutshell ($\mathcal{H} = \mathbb{R}^n$ or any real Hilbert space)

- **Objective function** $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$.
  e.g., data fidelity in ML, energy efficiency, profit, statistical error, ...

- An "optimal" $x \in \mathcal{H}$ makes $f(x)$ the smallest or largest
  e.g., minimize error, maximize efficiency

$$\underset{x \in \mathcal{H}}{\text{minimize}} \; f(x)$$

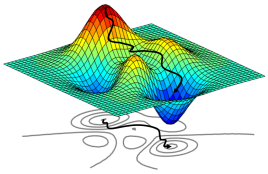


image: towardsdatascience.com

Optimization intro

Optimization in a nutshell ($\mathcal{H} = \mathbb{R}^n$ or any real Hilbert space)

- **Objective function** $f : \mathcal{H} \to \mathbb{R} \cup \{+\infty\}$.
  e.g., data fidelity in ML, energy efficiency, profit, statistical error, ...
- **Constraint** set(s) $C \subset \mathcal{H}$
  e.g., $\mathbb{R}^N_+$, $\mathbb{S}^N_+$, hypercube, ℓ_p ball, solution set of an inverse problem, ...
- An "optimal" $x \in C$ makes $f(x)$ the smallest or largest
  e.g., minimize error, maximize efficiency

$$\underset{x \in C}{\text{minimize}} \; f(x)$$

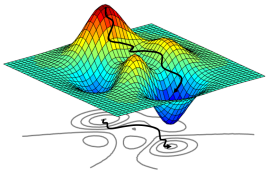
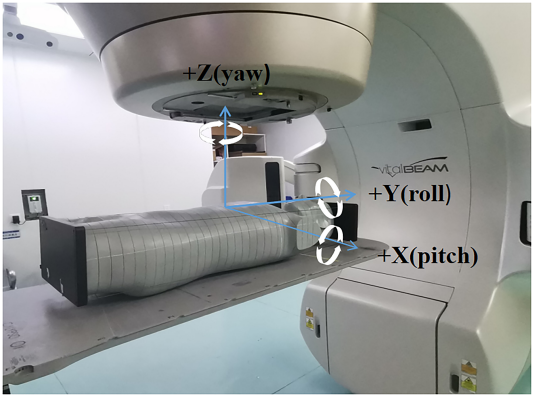


image: towardsdatascience.com

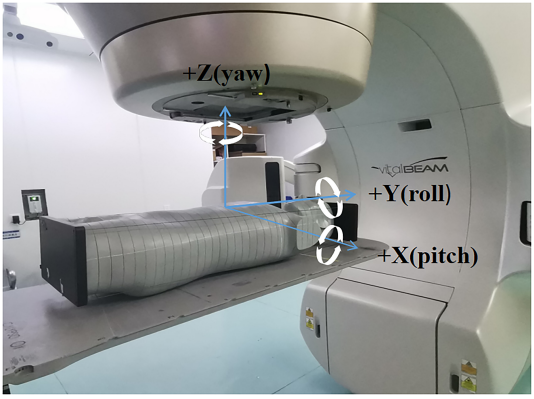
Modeling via optimization



[Torelli et al., *Med. Phys.*, 2023]

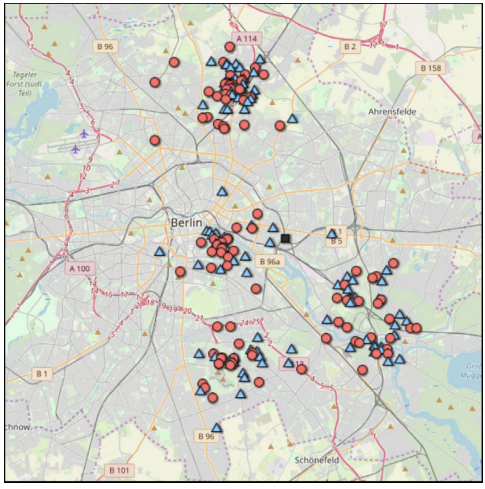
image: [Fu et al., *Tech. Cancer Res. Treatment*, 2023]

Modeling via optimization



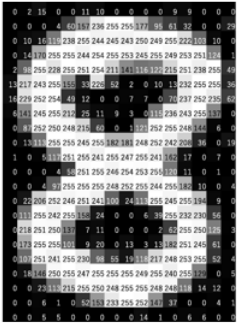
[Torelli et al., *Med. Phys.*, 2023]

image: [Fu et al., *Tech. Cancer Res. Treatment*, 2023]



[Sartori & Buriol, *Comput. Oper. Res.*, 2020]

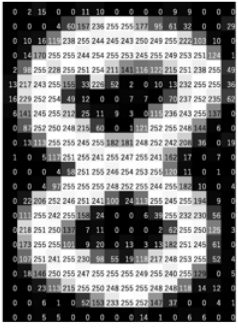
Modeling via optimization



0	2	15	0	0	11	10	0	0	0	0	0	9	9	0	0	0														
0	0	0	4	60	157	236	255	255	177	95	61	32	0	0	29															
0	10	16	119	238	255	244	245	243	250	249	255	222	103	10	0															
0	14	170	255	255	244	254	255	253	245	255	249	253	251	124	1															
2	98	255	228	255	251	254	211	141	116	122	215	251	238	255	49															
13	217	243	255	155	33	226	52	2	0	10	13	232	255	255	36															
16	229	252	254	49	12	0	0	7	7	0	70	237	252	235	62															
6	141	245	255	212	25	11	9	3	0	115	236	243	255	137	0															
0	87	252	250	248	215	60	0	1	121	252	255	248	144	6	0															
0	13	113	255	255	245	255	182	181	248	252	242	208	36	0	19															
1	0	5	117	251	255	241	255	247	255	241	162	17	0	7	0															
0	0	0	4	58	251	255	246	254	253	255	120	11	0	1	0															
0	0	4	97	255	255	255	248	252	255	244	255	182	10	0	4															
0	22	206	252	246	251	241	100	24	113	255	245	255	194	9	0															
0	111	255	242	255	158	24	0	0	6	39	255	232	230	56	0															
0	218	251	250	137	7	11	0	0	0	2	62	255	250	125	3															
0	173	255	255	101	9	20	0	13	3	13	182	251	245	61	0															
0	107	251	241	255	230	98	55	19	117	217	248	253	255	52	4															
0	18	146	250	255	247	255	255	255	249	255	240	255	129	0	5															
0	0	23	113	215	255	250	248	255	255	248	248	118	14	12	0															
0	0	6	1	0	52	153	233	255	252	147	37	0	0	4	1															
0	0	5	5	0	0	0	0	0	14	1	0	6	6	0	0															

image: towardsdatascience.com

Modeling via optimization



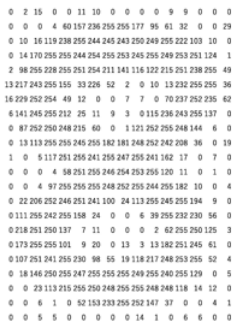
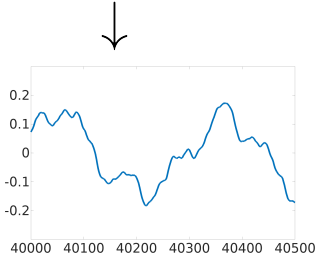
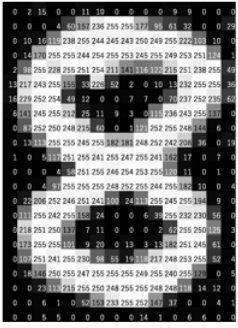
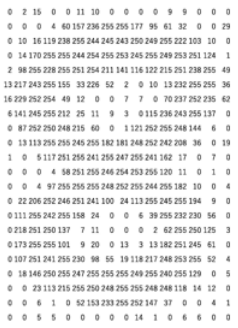


image: towardsdatascience.com



Modeling via optimization





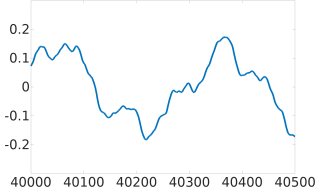


image: towardsdatascience.com

High-dimensional problems (computing $\nabla^2 f$ is costly)

$\Rightarrow$ use first-order methods (use evaluations of ∇f, prox_f, and/or f)

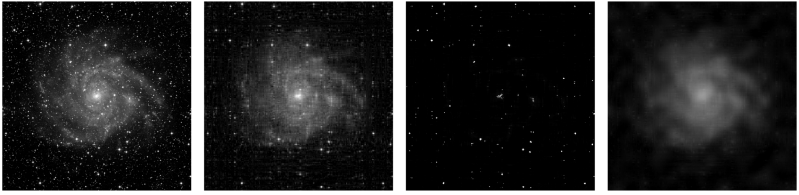
Modeling via optimization

Original Observation Recovery

[Combettes & ZW., *SIAM J. Imaging Sci.*, 2022]

Modeling via optimization

Original Observation Recovery



Original Observation Recovered stars Recovered galaxy

[Combettes & ZW., *SIAM J. Imaging Sci.*, 2022]

A few common constraints

Set name $C \subset \mathcal{H}$ and space $\mathcal{H}$	Constraint(s) on $x \in C$	Application Examples
ℓ_p ball ($p \geqslant 1$) $\mathcal{H} = \mathbb{R}^n$	$\sqrt[p]{\sum_{i \in I} x_i^p} \leqslant \gamma$ ($\gamma \geqslant 0$)	Signal processing (e.g., sparse recovery, matrix completion), graph analysis, relaxed combinatorial problems
Hypercube $\mathcal{H} = \mathbb{R}^n$	$\alpha_i \leqslant x_i \leqslant \beta_i$ ($\alpha_i, \beta_i \in \mathbb{R}$)	Image and signal processing, recommender system training, graph refinement
Birkhoff Polytope $\mathcal{H} = \mathbb{R}^{n \times n}$	$x_{i,j} \geqslant 0$, $\sum_{i \in [n]} x_{i,j} = 1$, $\sum_{j \in [n]} x_{i,j} = 1$	Graph automorphisms, communication theory, optimal transport, cluster analysis

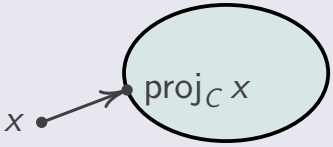
A few common constraints

Set name $C \subset \mathcal{H}$ and space $\mathcal{H}$	Constraint(s) on $x \in C$	Application Examples
Polyhedron $\mathcal{H} = \mathbb{R}^n$	$(Ax)_i \leqslant b_i$ $(A \in \mathbb{R}^{n \times m}, b \in \mathbb{R}^m)$	Optimal transport, production scheduling, inventory control, network flow, computational geometry
Spectrahedron $\mathcal{H} = \mathbb{S}^{n \times n}$	$x \succeq 0$, $\mathrm{trace}(x) = 1$	Covariance estimation, relaxed combinatorial problems, low-rank recovery
Nuclear norm ball $\mathcal{H} = \mathbb{R}^{n \times m}$	$\sum_{i=1}^{\min\{n,m\}} S_{i,i} \leqslant \gamma$ $(x = USV^\top, \gamma \geqslant 0)$	Graph denoising, recommender system training, cluster analysis, protein interaction prediction, low-rank recovery

A tale of two subroutines

Projection operators

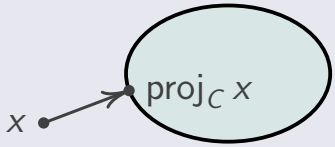
$$\text{proj}_C : x \mapsto \operatorname*{Argmin}_{c \in C} \|x - c\|^2$$



A tale of two subroutines

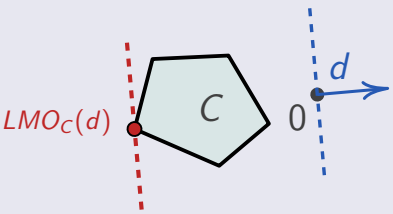
Projection operators

$$\text{proj}_C : x \mapsto \underset{c \in C}{\text{Argmin}} \|x - c\|^2$$



Linear Minimization Oracles (used in Frank-Wolfe algorithms, AKA "Conditional Gradient" algs.)

$$\text{LMO}_C : d \mapsto y \in \underset{c \in C}{\text{Argmin}} \langle c \mid d \rangle$$



A tale of two subroutines

Projection operators

$$\mathrm{proj}_C : x \mapsto \operatorname*{Argmin}_{c \in C} \|x - c\|^2$$

Open source repositories:

`proximity-operator.net`

`ProximalOperators.jl`

Linear Minimization Oracles (used in Frank-Wolfe algorithms, AKA "Conditional Gradient" algs.)

$$\mathrm{LMO}_C : d \mapsto y \in \operatorname*{Argmin}_{c \in C} \langle c \mid d \rangle$$

Open source repository:

`FrankWolfe.jl`

Why use LMOs?

[C. Combettes, Pokutta, '21] + many others:
Particularly in high-dimensions, LMO is currently *faster* than projection on a many C_i.

Why use LMOs?

[C. Combettes, Pokutta, '21] + many others:
Particularly in high-dimensions, LMO is currently *faster* than projection on a many C_i.

[W., '25]:
If C_i is compact/convex, approximate LMO is no slower than projection.
If C_i is also polyhedral, LMO is no slower than projection .

[Hu, '26]:
If C_i is compact/convex, approximate LMO is no slower than approximate projection.

Why use LMOs?

[C. Combettes, Pokutta, '21] + many others:
Particularly in high-dimensions, LMO is currently *faster* than projection on a many C_i.

[W., '25]:
If C_i is compact/convex, approximate LMO is no slower than projection.
If C_i is also polyhedral, LMO is no slower than projection .

[Hu, '26]:
If C_i is compact/convex, approximate LMO is no slower than approximate projection.

Question:
Is there a compact convex set C such that proj_C requires fewer arithmetic operations than LMO_C?

Comparison with projections

Comparison with projections

Example: Nuclear norm ball

For $x \in \mathbb{R}^{n \times n}$

$$\|x\|_{nuc} = \sum_{1 \leqslant i \leqslant n} \sigma_i(x).$$

For $\beta \geqslant 0$, $C = \{x \in \mathbb{R}^{n \times n} \mid \|x\|_{nuc} \leqslant \beta\}$,

Comparison with projections

Example: Nuclear norm ball

For $x \in \mathbb{R}^{n \times n}$

$$\|x\|_{nuc} = \sum_{1 \leqslant i \leqslant n} \sigma_i(x).$$

For $\beta \geqslant 0$, $C = \{x \in \mathbb{R}^{n \times n} \mid \|x\|_{nuc} \leqslant \beta\}$,

$\text{proj}_C(x)$: requires a full SVD!
$(\sigma_1, \ldots, \sigma_n, U, V)$, where $x = U\text{diag}(\sigma_1, \ldots, \sigma_n)V^\top$

Full SVD

$n = 500$: `0.11 sec`
$n = 1000$: `0.47 sec`
$n = 2000$: `4.87 sec`

Comparison with projections

Example: Nuclear norm ball

For $x \in \mathbb{R}^{n \times n}$

$$\|x\|_{nuc} = \sum_{1 \leqslant i \leqslant n} \sigma_i(x).$$

For $\beta \geqslant 0$, $C = \{x \in \mathbb{R}^{n \times n} \mid \|x\|_{nuc} \leqslant \beta\}$,

$\text{proj}_C(x)$: requires a full SVD!
$(\sigma_1, \ldots, \sigma_n, U, V)$, where $x = U\text{diag}(\sigma_1, \ldots, \sigma_n)V^\top$

$\text{LMO}_C(x)$: requires only first singular value/vectors
$(\sigma_1, U_1, V_1^\top)$

Full SVD

$n = 500$: 0.11 sec
$n = 1000$: 0.47 sec
$n = 2000$: 4.87 sec

Just $(\sigma_1, U_1, V_1^\top)$

$n = 500$: 0.0081 sec
$n = 1000$: 0.056 sec
$n = 2000$: 0.638 sec

Comparison with projections

Challenge: While Projections are analytically "well-behaved"

- Lipschitz continuous ($L = 1$)
- Firmly nonexpansive
- Fixed-point theory
- $\vdots$

Comparison with projections

Challenge: While Projections are analytically "well-behaved"

- Lipschitz continuous ($L = 1$)
- Firmly nonexpansive
- Fixed-point theory
- ⋮

LMOs are *discontinuous*.

Operations are not interchangeable.

New theory and algorithms are often needed.

Frank-Wolfe Splitting Algorithms

1. Background
2. History & Motivation
3. Approach
4. Analysis
5. Conclusion

A problem with multiple constraints

This line of work started with

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

- ∇f is L_f-Lipschitz continuous (nonconvex)
- C_i are nonempty, compact, convex subsets of a real Hilbert space $\mathcal{H}$.

Classical projection-based approaches

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

For appropriately-chosen stepsizes $(\gamma_k)_{k\in\mathbb{N}}$, $(*)$ can be solved via Forward-Backward, AKA projected gradient descent (PGD) using ∇f and $\text{proj}_{\bigcap_{i=1}^{m} C_i}$:

$$x_{k+1} = \text{proj}_{\bigcap_{i=1}^{m} C_i}(x_k - \gamma_k \nabla f(x_k))$$

Classical projection-based approaches

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

For appropriately-chosen stepsizes $(\gamma_k)_{k\in\mathbb{N}}$, $(*)$ can be solved via Forward-Backward, AKA projected gradient descent (PGD) using ∇f and $\operatorname{proj}_{\bigcap_{i=1}^{m} C_i}$:

$$x_{k+1} = \operatorname{proj}_{\bigcap_{i=1}^{m} C_i}(x_k - \gamma_k \nabla f(x_k))$$

Problem:
$\sum_{i=1}^{m}$ [Cost of $\operatorname{proj}_{C_i}$] $\ll$ [Cost of $\operatorname{proj}_{\bigcap_{i=1}^{m} C_i}$].

Classical projection-based approaches

$$\underset{x \in \bigcap_{i=1}^m C_i}{\text{minimize}} \; f(x) \qquad (*)$$

For appropriately-chosen stepsizes $(\gamma_k)_{k\in\mathbb{N}}$, $(*)$ can be solved via Forward-Backward, AKA projected gradient descent (PGD) using ∇f and $\text{proj}_{\bigcap_{i=1}^m C_i}$:

$$x_{k+1} = \text{proj}_{\bigcap_{i=1}^m C_i}(x_k - \gamma_k \nabla f(x_k))$$

Problem:
$\sum_{i=1}^m$ [Cost of proj_{C_i}] $\ll$ [Cost of $\text{proj}_{\bigcap_{i=1}^m C_i}$].

Algorithmic "splitting" idea:

$\text{proj}_{\bigcap_{i=1}^m C_i}$

$\text{proj}_{C_1}, \ldots \text{proj}_{C_m}$

Classical projection-based approaches

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

Projection-based splitting algorithms (dating back to the 1970s) e.g., Douglas-Rachford, Forward-Backward, ADMM, and many others solve $(*)$ using:

- First-order information on f (∇f or prox_f)
- $\text{proj}_{C_1}, \ldots \text{proj}_{C_m}$

Algorithmic "splitting" idea:

$\text{proj}_{\bigcap_{i=1}^{m} C_i}$

$\text{proj}_{C_1}, \ldots \text{proj}_{C_m}$

Classical projection-based approaches

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

Projection-based splitting algorithms (dating back to the 1970s) e.g., Douglas-Rachford, Forward-Backward, ADMM, and many others solve $(*)$ using:

- First-order information on f (∇f or prox_f)
- $\text{proj}_{C_1}, \ldots \text{proj}_{C_m}$

Algorithmic "splitting" idea:

$\text{proj}_{\bigcap_{i=1}^{m} C_i}$

$\text{proj}_{C_1}, \ldots \text{proj}_{C_m}$

A Frank-Wolfe setup for the splitting problem

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

- ∇f is L_f-Lipschitz continuous (nonconvex)
- C_i are nonempty, compact, convex subsets of a real Hilbert space $\mathcal{H}$.

A Frank-Wolfe setup for the splitting problem

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

- ∇f is L_f-Lipschitz continuous (nonconvex)
- C_i are nonempty, compact, convex subsets of a real Hilbert space $\mathcal{H}$.

If we could evaluate $\mathrm{LMO}_{\bigcap_{i=1}^{m} C_i}$, then we could use existing Frank-Wolfe algorithms.

A Frank-Wolfe setup for the splitting problem

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

- ∇f is L_f-Lipschitz continuous (nonconvex)
- C_i are nonempty, compact, convex subsets of a real Hilbert space $\mathcal{H}$.

If we could evaluate $\mathrm{LMO}_{\bigcap_{i=1}^{m} C_i}$, then we could use existing Frank-Wolfe algorithms.

However, just like projections,

$$\textstyle\sum_{i=1}^{m} [\text{Cost of } \mathrm{LMO}_{C_i}] \ll [\text{Cost of } \mathrm{LMO}_{\bigcap_{i=1}^{m} C_i}].$$

A Frank-Wolfe setup for the splitting problem

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

- ∇f is L_f-Lipschitz continuous (nonconvex)
- C_i are nonempty, compact, convex subsets of a real Hilbert space $\mathcal{H}$.

Goal: Solve $(*)$ using basic algebraic operations and

- ∇f
- *Linear minimization oracles* for C_i:

$$\mathrm{LMO}_{C_i} : x \mapsto y \in \underset{v \in C_i}{\operatorname{argmin}} \langle x \mid v \rangle$$

This is splitting, but with LMOs instead of projection operators.

A Frank-Wolfe setup for the splitting problem

$$\underset{x \in \bigcap_{i=1}^{m} C_i}{\text{minimize}} \; f(x) \qquad (*)$$

- ∇f is L_f-Lipschitz continuous (nonconvex)
- C_i are nonempty, compact, convex subsets of a real Hilbert space $\mathcal{H}$.

Goal: Solve $(*)$ using basic algebraic operations and

- ∇f
- *Linear minimization oracles* for C_i:

$$\text{LMO}_{C_i} : x \mapsto y \in \underset{v \in C_i}{\text{argmin}} \langle x \mid v \rangle$$

This is splitting, but with LMOs instead of projection operators.
Open-source LMO repository: `FrankWolfe.jl` (Github)

Literature

Frank-Wolfe splitting algorithms when f is **convex**: [Gidel et al., '18], [Yurtsever et al., '19], [Silveti-Falls et al., '20], [Lan et al., '21], ...
Best convergence rate on $\max\{|f(x_k) - f^*|, \text{dist}_{\bigcap_{i=1}^m C_i}(x_k)\}$ is $\mathcal{O}(1/\sqrt{k})$

Literature

Frank-Wolfe splitting algorithms when f is **convex**: [Gidel et al., '18], [Yurtsever et al., '19], [Silveti-Falls et al., '20], [Lan et al., '21], ...
Best convergence rate on $\max\{|f(x_k) - f^*|, \text{dist}_{\bigcap_{i=1}^m C_i}(x_k)\}$ is $\mathcal{O}(1/\sqrt{k})$

Frank-Wolfe splitting algorithms when f is **convex** with **convex inequality constraints** $g_i(x) \leqslant 0$: [Yurtsever et al., '19] [Lan et al., '21], [Asgari, Neely, '24], [Wang, Pong, W., '26]
Worst-case rate on $\max\{|f(x_k) - f^*|, [g_i(x_k)]_+\}$ is $\mathcal{O}(1/\sqrt{k})$.

Literature

Frank-Wolfe splitting algorithms when f is **convex**: [Gidel et al., '18], [Yurtsever et al., '19], [Silveti-Falls et al., '20], [Lan et al., '21], ...
Best convergence rate on $\max\{|f(x_k) - f^*|, \mathrm{dist}_{\bigcap_{i=1}^m C_i}(x_k)\}$ is $\mathcal{O}(1/\sqrt{k})$

Frank-Wolfe splitting algorithms when f is **convex** with **convex inequality constraints** $g_i(x) \leqslant 0$: [Yurtsever et al., '19] [Lan et al., '21], [Asgari, Neely, '24], [Wang, Pong, W., '26]
Worst-case rate on $\max\{|f(x_k) - f^*|, [g_i(x_k)]_+\}$ is $\mathcal{O}(1/\sqrt{k})$.
With uniform convexity assumption: rate arbitrarily close to $\mathcal{O}(1/k)$.

Literature

Frank-Wolfe splitting algorithms when f is **convex**: [Gidel et al., '18], [Yurtsever et al., '19], [Silveti-Falls et al., '20], [Lan et al., '21], ...
Best convergence rate on $\max\{|f(x_k) - f^*|, \mathrm{dist}_{\bigcap_{i=1}^m C_i}(x_k)\}$ is $\mathcal{O}(1/\sqrt{k})$

Frank-Wolfe splitting algorithms when f is **convex** with **convex inequality constraints** $g_i(x) \leqslant 0$: [Yurtsever et al., '19] [Lan et al., '21], [Asgari, Neely, '24], [Wang, Pong, W., '26]
Worst-case rate on $\max\{|f(x_k) - f^*|, [g_i(x_k)]_+\}$ is $\mathcal{O}(1/\sqrt{k})$.
With uniform convexity assumption: rate arbitrarily close to $\mathcal{O}(1/k)$.

Today's talk: Frank-Wolfe splitting when f is **nonconvex**: [W., Pokutta, '25], [Halbey et al., '25], [Silveti-Falls, Molinari, W., '26]

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \quad f(x). \qquad (1)$$

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \quad f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0, 1]$:

1. $v_k = \text{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1 - \gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \quad f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0, 1]$:

1. $v_k = \text{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1 - \gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.
Requires LMO_C and ∇f.

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \; f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0, 1]$:

1. $v_k = \mathrm{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1 - \gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.
Requires LMO_C and ∇f.

Stationarity criterion: *Frank-Wolfe gap* of (1) at x:

$$G(x) = \max_{v \in C} \langle \nabla f(x) \mid x - v \rangle.$$

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \; f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0, 1]$:

1. $v_k = \mathrm{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1 - \gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.
Requires LMO_C and ∇f.

Stationarity criterion: *Frank-Wolfe gap* of (1) at x:

$$G(x) = \max_{v \in C} \langle \nabla f(x) \mid x - v \rangle.$$

Facts:

- If $x \in C$, $G(x) \geqslant 0$.

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \; f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0,1]$:

1. $v_k = \mathrm{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1-\gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.
Requires LMO_C and ∇f.

Stationarity criterion: *Frank-Wolfe gap* of (1) at x:

$$G(x) = \max_{v \in C} \langle \nabla f(x) \mid x - v \rangle.$$

Facts:

- If $x \in C$, $G(x) \geqslant 0$.
- $[x \in C \text{ and } G(x) = 0] \Leftrightarrow x$ is stationary for (1).

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x \in C}{\text{minimize}} \; f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0,1]$:

1. $v_k = \mathrm{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1-\gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.
Requires LMO_C and ∇f.

Stationarity criterion: *Frank-Wolfe gap* of (1) at x:

$$G(x) = \max_{v \in C} \langle \nabla f(x) \mid x - v \rangle.$$

Facts:

- If $x \in C$, $G(x) \geqslant 0$.
- $[x \in C \text{ and } G(x) = 0] \Leftrightarrow x$ is stationary for (1).

Under a variety of schedules γ_t:

- $0 \leqslant \min_{0 \leqslant j \leqslant k} G(x_j) \leqslant Ck^{-1/2} \to 0$

Frank-Wolfe recap

Classical Frank-Wolfe algorithm is for

$$\underset{x\in C}{\text{minimize}}\ f(x). \qquad (1)$$

For $k \in \mathbb{N}$ and $\gamma_k \in [0,1]$:

1. $v_k = \mathrm{LMO}_C(\nabla f(x_k))$
2. $x_{k+1} = (1-\gamma_k)x_k + \gamma_k v_k$

Guarantees $x_k \in C$.
Requires LMO_C and ∇f.

Stationarity criterion: *Frank-Wolfe gap* of (1) at x:

$$G(x) = \max_{v\in C} \langle \nabla f(x) \mid x - v \rangle.$$

Facts:

- If $x \in C$, $G(x) \geqslant 0$.
- $[x \in C \text{ and } G(x) = 0] \Leftrightarrow x$ is stationary for (1).

Under a variety of schedules γ_t:

- $0 \leqslant \min_{0\leqslant j\leqslant k} G(x_j) \leqslant Ck^{-1/2} \to 0$
- Along the subsequence minimizing G, all cluster points are stationary points.
- [Bolte, C. Combettes, Pauwels, '23]: $(x_k)_{k\in\mathbb{N}}$ may not converge

Frank-Wolfe Splitting Algorithms

1. Background
2. History & Motivation
3. Approach
4. Analysis
5. Conclusion

Product space reformulation

[Pierra, '84]: Reformulation of $(*)$ on $\mathcal{H}^m$: Let $\{\omega_i\}_{i=1}^m \subset]0,1]$ s.t. $\sum_{i=1}^m \omega_i = 1$, let $A\colon \mathcal{H}^m \to \mathcal{H}\colon \boldsymbol{x} \mapsto \sum_{i=1}^m \omega_i x_i$, and set $D = \{(x_1,\ldots,x_m) \,|\, (\forall i,j \in \{1,\ldots,m\} \;\; x_i = x_j\}$.

Product space reformulation

[Pierra, '84]: Reformulation of $(*)$ on $\mathcal{H}^m$: Let $\{\omega_i\}_{i=1}^m \subset]0,1]$ s.t. $\sum_{i=1}^m \omega_i = 1$, let $A\colon \mathcal{H}^m \to \mathcal{H}\colon \boldsymbol{x} \mapsto \sum_{i=1}^m \omega_i x_i$, and set $D = \{(x_1,\dots,x_m) \,|\, (\forall i,j \in \{1,\dots,m\}\ \ x_i = x_j\}$.

$$x \in \operatorname*{Argmin}_{z \in \bigcap_{i=1}^m C_i} f(z) \quad \Leftrightarrow \quad (x,\dots,x) \in \operatorname*{Argmin}_{\boldsymbol{z}=(z_1,\dots,z_m) \in D \cap \bigtimes_{i=1}^m C_i} f(A\boldsymbol{z})$$

Product space reformulation

[Pierra, '84]: Reformulation of $(*)$ on $\mathcal{H}^m$: Let $\{\omega_i\}_{i=1}^m \subset]0,1]$ s.t. $\sum_{i=1}^m \omega_i = 1$, let $A\colon \mathcal{H}^m \to \mathcal{H}\colon \boldsymbol{x} \mapsto \sum_{i=1}^m \omega_i x_i$, and set $D = \{(x_1,\dots,x_m) \,|\, (\forall i,j \in \{1,\dots,m\}\ \ x_i = x_j\}$.

$$x \in \operatorname*{Argmin}_{z \in \bigcap_{i=1}^m C_i} f(z) \qquad \Leftrightarrow \qquad (x,\dots,x) \in \operatorname*{Argmin}_{\boldsymbol{z}=(z_1,\dots,z_m) \in D \cap \bigtimes_{i=1}^m C_i} f\,(A\boldsymbol{z})$$

Reformulated Frank-Wolfe gaps are the same:

$$\max_{\boldsymbol{v} \in D \cap \bigtimes_{i=1}^m C_i} \langle \nabla(f \circ A)\boldsymbol{x} \,|\, \boldsymbol{x} - \boldsymbol{v}\rangle = \max_{\substack{(\forall i \in [m])\ v_i \in C_i \\ (\forall i,j \in [m])\ v_i = v_j}} \langle \nabla f(Ax) \,|\, Ax - Av\rangle$$

$$= \max_{v \in \bigcap_{i=1}^m C_i} \langle \nabla f(A\boldsymbol{x}) \,|\, A\boldsymbol{x} - v\rangle.$$

Caution: This Frank-Wolfe gap can be negative!

Product space reformulation

[Pierra, '84]: Reformulation of $(*)$ on $\mathcal{H}^m$: Let $\{\omega_i\}_{i=1}^m \subset]0,1]$ s.t. $\sum_{i=1}^m \omega_i = 1$, let $A\colon \mathcal{H}^m \to \mathcal{H}\colon \boldsymbol{x} \mapsto \sum_{i=1}^m \omega_i x_i$, and set $D = \{(x_1,\dots,x_m) \,|\, (\forall i,j \in \{1,\dots,m\}\ \ x_i = x_j\}$.

$$x \in \operatorname*{Argmin}_{z \in \bigcap_{i=1}^m C_i} f(z) \qquad \Leftrightarrow \qquad (x,\dots,x) \in \operatorname*{Argmin}_{\mathbf{z}=(z_1,\dots,z_m) \in \bigtimes_{i=1}^m C_i} f(A\mathbf{z}) + \iota_D(\mathbf{z})$$

Reformulated Frank-Wolfe gaps are the same:

$$\max_{\boldsymbol{v} \in D \cap \bigtimes_{i=1}^m C_i} \langle \nabla(f \circ A)\boldsymbol{x} \,|\, \boldsymbol{x} - \boldsymbol{v}\rangle = \max_{\substack{(\forall i \in [m])\ v_i \in C_i \\ (\forall i,j \in [m])\ v_i = v_j}} \langle \nabla f(Ax) \,|\, Ax - Av\rangle$$

$$= \max_{v \in \bigcap_{i=1}^m C_i} \langle \nabla f(A\boldsymbol{x}) \,|\, A\boldsymbol{x} - v\rangle.$$

Caution: This Frank-Wolfe gap can be negative!

Setting of [S-F, M, W, '26+]

$$\underset{x \in \mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \qquad (\star)$$

1. $\mathcal{C} \subset \mathbb{R}^n$ is nonempty, convex, and compact
2. $f : \mathbb{R}^n \to \mathbb{R}$ and ∇f is $L_{\nabla f}$-Lipschitz continuous (nonconvex) on $\mathcal{C}$

Setting of [S-F, M, W, '26+]

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \qquad (\star)$$

1. $\mathcal{C} \subset \mathbb{R}^n$ is nonempty, convex, and compact
2. $f\colon \mathbb{R}^n \to \mathbb{R}$ and ∇f is $L_{\nabla f}$-Lipschitz continuous (nonconvex) on $\mathcal{C}$
3. $T\colon \mathbb{R}^n \to \mathbb{R}^m$ is linear.
4. $g\colon \mathbb{R}^m \to]-\infty, +\infty]$ is ϱ-weakly convex and one of the following holds.
   - 4.1 $\mathcal{D} \subset \mathbb{R}^m$ is a nonempty closed convex set and $g(x) = \iota_{\mathcal{D}}(x) = \begin{cases} 0 & \text{if } x \in \mathcal{D} \\ +\infty & \text{if } x \notin \mathcal{D}, \end{cases}$
     where $\rho = 0$, and we use the convention "$\rho^{-1} = +\infty$".
   - 4.2 g is L_g-Lipschitz continuous

Setting of [S-F, M, W, '26+]

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x)+g(Tx) \tag{$\star$}$$

1. $\mathcal{C}\subset\mathbb{R}^n$ is nonempty, convex, and compact
2. $f\colon\mathbb{R}^n\to\mathbb{R}$ and ∇f is $L_{\nabla f}$-Lipschitz continuous (nonconvex) on $\mathcal{C}$
3. $T\colon\mathbb{R}^n\to\mathbb{R}^m$ is linear.
4. $g\colon\mathbb{R}^m\to\left]-\infty,+\infty\right]$ is ϱ-weakly convex and one of the following holds.
   - 4.1 $\mathcal{D}\subset\mathbb{R}^m$ is a nonempty closed convex set and $g(x)=\iota_{\mathcal{D}}(x)=\begin{cases}0 & \text{if } x\in\mathcal{D}\\ +\infty & \text{if } x\notin\mathcal{D},\end{cases}$ where $\rho=0$, and we use the convention "$\rho^{-1}=+\infty$".
   - 4.2 g is L_g-Lipschitz continuous

The splitting problem $(*)$ is a special case of $(\star)$ under Assumption 4.1 for a linear subspace $\mathcal{D}=D$ and Cartesian product $\mathcal{C}=\bigtimes_{i=1}^{m}C_i$.

Setting of [S-F, M, W, '26+]

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \tag{$\star$}$$

1. $\mathcal{C} \subset \mathbb{R}^n$ is nonempty, convex, and compact
2. $f\colon \mathbb{R}^n \to \mathbb{R}$ and ∇f is $L_{\nabla f}$-Lipschitz continuous (nonconvex) on $\mathcal{C}$
3. $T\colon \mathbb{R}^n \to \mathbb{R}^m$ is linear.
4. $g\colon \mathbb{R}^m \to]-\infty, +\infty]$ is ϱ-weakly convex and one of the following holds.
   - 4.1 $\mathcal{D} \subset \mathbb{R}^m$ is a nonempty closed convex set and $g(x) = \iota_{\mathcal{D}}(x) = \begin{cases} 0 & \text{if } x \in \mathcal{D} \\ +\infty & \text{if } x \notin \mathcal{D}, \end{cases}$
     where $\rho = 0$, and we use the convention "$\rho^{-1} = +\infty$".
   - 4.2 g is L_g-Lipschitz continuous

The splitting problem $(*)$ is a special case of $(\star)$ under Assumption 4.1 for a linear subspace $\mathcal{D} = D$ and Cartesian product $\mathcal{C} = \bigtimes_{i=1}^{m} C_i$.

Today, we'll mostly focus on Assumption 4.1 instead of 4.2.

Setting of [S-F, M, W, '26+]

$$\underset{x \in \mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \qquad (\star)$$

We assume we can compute:

- ∇f
- $\mathrm{LMO}_{\mathcal{C}}$
- $\mathrm{prox}_{\beta g}$ for $\beta \in \left]0, \rho^{-1}\right[$. (i.e., under Assumption 4.1, $\mathrm{proj}_{\mathcal{D}}$;

Setting of [S-F, M, W, '26+]

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \qquad (\star)$$

We assume we can compute:

- ∇f note: $\nabla(f \circ A) = A^* \circ \nabla f \circ A$
- $\mathrm{LMO}_{\mathcal{C}}$ note: $\mathrm{LMO}_{\times_{i=1}^m C_i} = (\mathrm{LMO}_{C_1} x_1, \ldots, \mathrm{LMO}_{C_m} x_m)$.
- $\mathrm{prox}_{\beta g}$ for $\beta \in \left]0, \rho^{-1}\right[$. (i.e., under Assumption 4.1, $\mathrm{proj}_{\mathcal{D}}$; for the splitting problem $(*)$, proj_D is just averaging)

Setting of [S-F, M, W, '26+]

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \qquad (\star)$$

We assume we can compute:

- ∇f note: $\nabla(f \circ A) = A^* \circ \nabla f \circ A$
- $\mathrm{LMO}_{\mathcal{C}}$ note: $\mathrm{LMO}_{\times_{i=1}^m C_i} = (\mathrm{LMO}_{C_1} x_1, \ldots, \mathrm{LMO}_{C_m} x_m)$.
- $\mathrm{prox}_{\beta g}$ for $\beta \in \left]0, \rho^{-1}\right[$. (i.e., under Assumption 4.1, $\mathrm{proj}_{\mathcal{D}}$; for the splitting problem $(*)$, proj_D is just averaging)

We'll be using the *Moreau envelope* of g for $\beta \in \left]0, \rho^{-1}\right[$:

$$g^\beta(x) = \min_{z\in\mathbb{R}^n} g(z) + \frac{1}{2\beta}\|x - z\|^2 \qquad \Rightarrow \qquad \nabla g^\beta(x) = \beta^{-1}(x - \mathrm{prox}_{\beta g} x)$$

This avoids smoothing $g \circ T$.

Approach: a sequence of smoothed subproblems

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x)+g(Tx) \tag{$\star$}$$

For $\beta \in \left]0, \rho^{-1}\right[$, consider the smoothed problem:

$$\underset{x\in\mathcal{C}}{\text{minimize}}\;\; \Phi_\beta(x) := f(x)+g^\beta(Tx). \tag{S}$$

Approach: a sequence of smoothed subproblems

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x)+g(Tx) \tag{$\star$}$$

For $\beta \in \left]0, \rho^{-1}\right[$, consider the smoothed problem:

$$\underset{x\in\mathcal{C}}{\text{minimize}}\;\; \Phi_\beta(x) := f(x)+g^\beta(Tx). \tag{S}$$

For the splitting problem $(*)$, the smoothed problem (S) becomes the relaxation,

$$\underset{\mathbf{x}=(x_1,\ldots,x_m)\in \bigtimes_{i=1}^{m} C_i}{\text{minimize}} \quad f(A\mathbf{x})+\frac{1}{2\beta}\,\mathrm{dist}^2_D(\mathbf{x}). \tag{R}$$

Approach: a sequence of smoothed subproblems

$$\underset{x \in \mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \tag{$\star$}$$

For $\beta \in \left]0, \rho^{-1}\right[$, consider the smoothed problem:

$$\underset{x \in \mathcal{C}}{\text{minimize}}\;\; \Phi_\beta(x) := f(x) + g^\beta(Tx). \tag{S}$$

The smoothed problem (S) can be solved with Frank-Wolfe:

Approach: a sequence of smoothed subproblems

$$\underset{x \in \mathcal{C}}{\text{minimize}}\; f(x) + g(Tx) \tag{$\star$}$$

For $\beta \in \left]0, \rho^{-1}\right[$, consider the smoothed problem:

$$\underset{x \in \mathcal{C}}{\text{minimize}}\;\; \Phi_\beta(x) := f(x) + g^\beta(Tx). \tag{S}$$

The smoothed problem (S) can be solved with Frank-Wolfe:
$\nabla\Phi$ depends on ∇f and $\nabla g^\beta(\boldsymbol{x}) = \beta^{-1}(\boldsymbol{x} - \text{prox}_{\beta g})$.

Approach: a sequence of smoothed subproblems

$$\underset{x\in\mathcal{C}}{\text{minimize}}\; f(x)+g(Tx) \tag{$\star$}$$

For $\beta \in \left]0, \rho^{-1}\right[$, consider the smoothed problem:

$$\underset{x\in\mathcal{C}}{\text{minimize}}\;\; \Phi_\beta(x) := f(x)+g^\beta(Tx). \tag{S}$$

The smoothed problem (S) can be solved with Frank-Wolfe:

$\nabla\Phi$ depends on ∇f and $\nabla g^\beta(\boldsymbol{x}) = \beta^{-1}(\boldsymbol{x} - \text{prox}_{\beta g})$.

Pseudocode: For a positive sequence $\beta_k \searrow 0$, consider

(A) Perform *one* Frank-Wolfe step for minimizing $\Phi_{\beta_k}(x) = f(x)+g^{\beta_k}(Tx)$ over $\mathcal{C}$.

(B) Decrease β_k, go to (A).

Algorithm

Algorithm: **Fra**nk-Wolfe with **M**oreau **E**nvelope **S**moothing (**Frames**)

Input: $x_0 \in \mathcal{C}$, step sizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing schedule $\{\beta_k\}_{k\in\mathbb{N}} \subset]0, \rho^{-1}[$

For $k = 0, 1, 2, \ldots$ **do**

1. Compute the gradient of the smoothed objective

$$\nabla \Phi_{\beta_k}(x_k) = \nabla f(x_k) + \frac{1}{\beta_k} T^* \Big(Tx_k - \operatorname{prox}_{\beta_k g}(Tx_k) \Big).$$

2. Compute the LMO for this direction $s_k \in \arg\min_{s\in\mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k), s \rangle$.
3. Update the iterate $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

Algorithm

Algorithm: **Fra**nk-Wolfe with **M**oreau **E**nvelope **S**moothing (**Frames**)

Input: $x_0 \in \mathcal{C}$, step sizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing schedule $\{\beta_k\}_{k\in\mathbb{N}} \subset]0, \rho^{-1}[$

For $k = 0, 1, 2, \ldots$ **do**

1. Compute the gradient of the smoothed objective

$$\nabla \Phi_{\beta_k}(x_k) = \nabla f(x_k) + \frac{1}{\beta_k} T^* \Big(Tx_k - \operatorname{prox}_{\beta_k g}(Tx_k) \Big).$$

2. Compute the LMO for this direction $s_k \in \arg\min_{s\in\mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k), s \rangle$.
3. Update the iterate $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

Usually: For $0 < q < p < 1$, $p + q < 1$, $\gamma_k = \Theta((k+1)^{-p})$ and $\beta_k = \Theta((k+1)^{-q})$

Algorithm

Algorithm: **Fra**nk-Wolfe with **M**oreau **E**nvelope **S**moothing (**Frames**)

Input: $x_0 \in \mathcal{C}$, step sizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing schedule $\{\beta_k\}_{k\in\mathbb{N}} \subset]0, \rho^{-1}[$

For $k = 0, 1, 2, \ldots$ **do**

1. Compute the gradient of the smoothed objective

$$\nabla \Phi_{\beta_k}(x_k) = \nabla f(x_k) + \frac{1}{\beta_k} T^* \Big(Tx_k - \operatorname{prox}_{\beta_k g}(Tx_k) \Big).$$

2. Compute the LMO for this direction $s_k \in \arg\min_{s\in\mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k), s \rangle$.
3. Update the iterate $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

Guarantees feasibility in $\mathcal{C}$: $(\forall k \in \mathbb{N}) \quad x_k \in \mathcal{C}$, but under 4.1, $Tx_k \in \mathcal{D}$ not guaranteed.

Base Splitting Algorithm

Input: $x_0 \in \bigtimes_{i=1}^{m} C_i$, stepsizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing $\{\beta_k\}_{k\in\mathbb{N}} \subset]0,+\infty[$

For $k = 0, 1, 2, \ldots$ **do**

1. $(\nabla\Phi_{\beta_k}(x_k))^i = \nabla f(Ax_k) + \frac{1}{\beta_k}((x_k)^i - A(x_k))$
2. **For** $i = 1, 2, \ldots, m$ **do**
   $s_k^i = \mathrm{LMO}_{C_i}((\nabla\Phi_{\beta_k}(x_k))^i, s\rangle)$.
   end for
3. $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

Frank-Wolfe Splitting Algorithms

1. Background
2. History & Motivation
3. Approach
4. Analysis
5. Conclusion

Stationarity criterion: Frank-Wolfe gaps

Smoothed Frank-Wolfe gap

$$G^{\beta_k}(x_k) = \max_{s \in \mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k) \mid x - s \rangle$$

$x \in \mathcal{C}$ & $G^{\beta_k}(x) = 0 \Leftrightarrow$ x is a stationary point of $\underset{z \in C}{\text{minimize}}\ \Phi_k(z)$

Stationarity criterion: Frank-Wolfe gaps

Smoothed Frank-Wolfe gap

$$G^{\beta_k}(x_k) = \max_{s \in \mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k) \mid x - s \rangle$$

$x \in \mathcal{C}$ & $G^{\beta_k}(x) = 0 \;\Leftrightarrow\;$ x is a stationary point of $\underset{z \in C}{\text{minimize}}\ \Phi_k(z)$

Nonsmooth Frank-Wolfe gaps

- Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), set $\widetilde{\mathcal{C}} = \mathcal{C} \cap T^{-1}(\mathcal{D})$

$$\widetilde{G}(x_k) = \max_{s \in \widetilde{\mathcal{C}} = \mathcal{C} \cap T^{-1}(\mathcal{D})} \langle \nabla f(x_k) \mid x_k - s \rangle$$

$x \in \widetilde{\mathcal{C}}$ & $\widetilde{G}(x) = 0 \;\Leftrightarrow\;$ x is a stationary point of $(\star)$.

Stationarity criterion: Frank-Wolfe gaps

Smoothed Frank-Wolfe gap

$$G^{\beta_k}(x_k) = \max_{s \in \mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k) \mid x - s \rangle$$

$x \in \mathcal{C}$ & $G^{\beta_k}(x) = 0 \;\Leftrightarrow\;$ x is a stationary point of $\underset{z \in C}{\text{minimize}}\ \Phi_k(z)$

Nonsmooth Frank-Wolfe gaps

- Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), set $\widetilde{\mathcal{C}} = \mathcal{C} \cap T^{-1}(\mathcal{D})$

$$\widetilde{G}(x_k) = \max_{s \in \widetilde{\mathcal{C}} = \mathcal{C} \cap T^{-1}(\mathcal{D})} \langle \nabla f(x_k) \mid x_k - s \rangle$$

$x \in \widetilde{\mathcal{C}}$ & $\widetilde{G}(x) = 0 \;\Leftrightarrow\;$ x is a stationary point of $(\star)$.
Since $Tx_k \in \mathcal{D}$ (and $x \in \widetilde{\mathcal{C}}$) not guaranteed, $\widetilde{G}(x_k)$ could be negative!

Stationarity criterion: Frank-Wolfe gaps

Smoothed Frank-Wolfe gap

$$G^{\beta_k}(x_k) = \max_{s \in \mathcal{C}} \langle \nabla \Phi_{\beta_k}(x_k) \mid x - s \rangle$$

$x \in \mathcal{C}$ & $G^{\beta_k}(x) = 0 \;\Leftrightarrow\;$ x is a stationary point of $\underset{z \in C}{\text{minimize}}\ \Phi_k(z)$

Nonsmooth Frank-Wolfe gaps

- Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), set $\widetilde{\mathcal{C}} = \mathcal{C} \cap T^{-1}(\mathcal{D})$

$$\widetilde{G}(x_k) = \max_{s \in \widetilde{\mathcal{C}} = \mathcal{C} \cap T^{-1}(\mathcal{D})} \langle \nabla f(x_k) \mid x_k - s \rangle$$

  $x \in \widetilde{\mathcal{C}}$ & $\widetilde{G}(x) = 0 \;\Leftrightarrow\;$ x is a stationary point of $(\star)$.

- Not this talk's focus: Under Assumption 4.2 (Lipschitz ρ-weakly convex g)
  $G(x; \xi) = \max_{s \in \mathcal{C}} \langle \nabla f(x) + T^* \xi \mid x - s \rangle$ for $\xi \in \partial g(Tx)$.
  $x \in \mathcal{C}$, $\xi \in \partial g(Tx)$, & $G(x; \xi) = 0 \;\Leftrightarrow\;$ x is a stationary point of $(\star)$.

The base algorithm

If $\mathcal{H} = \mathbb{R}^n$, the algorithm [W., Pokutta '25] is a special case of "FraMes"

Split Conditional Gradient Algorithm [W., Pokutta, '25]

$T = \mathrm{Id}$, $\mathcal{C} = \bigtimes_{i \in I} C_i$ and $\mathcal{D} = D$. Let $\beta_t = \mathcal{O}(1/\log(k+1))$ and $\gamma_k = \mathcal{O}(1/\sqrt{k+1})$. Set $G^{\beta_k}(x_k) = \max_{\boldsymbol{v} \in \bigtimes_{i \in I} C_i} \langle \nabla \Phi_{\beta_k}(\boldsymbol{x}_k) \mid \boldsymbol{x}_k - \boldsymbol{v} \rangle$.

The base algorithm

If $\mathcal{H} = \mathbb{R}^n$, the algorithm [W., Pokutta '25] is a special case of "FraMes"

Split Conditional Gradient Algorithm [W., Pokutta, '25]

$T = \mathrm{Id}$, $\mathcal{C} = \bigtimes_{i \in I} C_i$ and $\mathcal{D} = D$. Let $\beta_t = \mathcal{O}(1/\log(k+1))$ and $\gamma_k = \mathcal{O}(1/\sqrt{k+1})$. Set $G^{\beta_k}(x_k) = \max_{\boldsymbol{v} \in \bigtimes_{i \in I} C_i} \langle \nabla \Phi_{\beta_k}(\boldsymbol{x}_k) \mid \boldsymbol{x}_k - \boldsymbol{v} \rangle$. Then

$$0 \leqslant \min_{0 \leqslant j \leqslant k} G^{\beta_j}(\boldsymbol{x}_j) \leqslant \mathcal{O}\left(\frac{\ln k}{\sqrt{k+1}}\right)$$

Also, $\mathrm{dist}_D(x_k) \to 0$.

The base algorithm

If $\mathcal{H} = \mathbb{R}^n$, the algorithm [W., Pokutta '25] is a special case of "FraMes"

Split Conditional Gradient Algorithm [W., Pokutta, '25]

$T = \mathrm{Id}$, $\mathcal{C} = \bigtimes_{i \in I} C_i$ and $\mathcal{D} = D$. Let $\beta_t = \mathcal{O}(1/\log(k+1))$ and $\gamma_k = \mathcal{O}(1/\sqrt{k+1})$. Set $G^{\beta_k}(x_k) = \max_{\boldsymbol{v} \in \bigtimes_{i \in I} C_i} \langle \nabla \Phi_{\beta_k}(\boldsymbol{x}_k) \mid \boldsymbol{x}_k - \boldsymbol{v} \rangle$. Then

$$0 \leqslant \min_{0 \leqslant j \leqslant k} G^{\beta_j}(\boldsymbol{x}_j) \leqslant \mathcal{O}\left(\frac{\ln k}{\sqrt{k+1}}\right)$$

Also, $\mathrm{dist}_D(x_k) \to 0$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point yields a stationary point of $(*)$.

The base algorithm

If $\mathcal{H} = \mathbb{R}^n$, the algorithm [W., Pokutta '25] is a special case of "FraMes"

Split Conditional Gradient Algorithm [W., Pokutta, '25]

$T = \mathrm{Id}$, $\mathcal{C} = \bigtimes_{i \in I} C_i$ and $\mathcal{D} = D$. Let $\beta_t = \mathcal{O}(1/\log(k+1))$ and $\gamma_k = \mathcal{O}(1/\sqrt{k+1})$. Set $G^{\beta_k}(x_k) = \max_{\boldsymbol{v} \in \bigtimes_{i \in I} C_i} \langle \nabla \Phi_{\beta_k}(\boldsymbol{x}_k) \mid \boldsymbol{x}_k - \boldsymbol{v} \rangle$. Then

$$0 \leqslant \min_{0 \leqslant j \leqslant k} G^{\beta_j}(\boldsymbol{x}_j) \leqslant \mathcal{O}\left(\frac{\ln k}{\sqrt{k+1}}\right)$$

Also, $\mathrm{dist}_D(x_k) \to 0$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point yields a stationary point of $(*)$.

[Halbey et al., '25] Could remove the $\ln k$ factor with similar schedules β_k, γ_k.

The base algorithm

If $\mathcal{H} = \mathbb{R}^n$, the algorithm [W., Pokutta '25] is a special case of "FraMes"

Split Conditional Gradient Algorithm [W., Pokutta, '25]

$T = \mathrm{Id}$, $\mathcal{C} = \bigtimes_{i\in I} C_i$ and $\mathcal{D} = D$. Let $\beta_t = \mathcal{O}(1/\log(k+1))$ and $\gamma_k = \mathcal{O}(1/\sqrt{k+1})$. Set $G^{\beta_k}(x_k) = \max_{\boldsymbol{v}\in\bigtimes_{i\in I} C_i} \langle \nabla\Phi_{\beta_k}(\boldsymbol{x}_k) \mid \boldsymbol{x}_k - \boldsymbol{v}\rangle$. Then

$$0 \leqslant \min_{0\leqslant j\leqslant k} G^{\beta_j}(\boldsymbol{x}_j) \leqslant \mathcal{O}\left(\frac{\ln k}{\sqrt{k+1}}\right)$$

Also, $\mathrm{dist}_D(x_k) \to 0$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point yields a stationary point of $(*)$.

[Halbey et al., '25] Could remove the $\ln k$ factor with similar schedules β_k, γ_k.
Q:Will this iteratively-smoothed F-W scheme work for $(\star)$?

The base algorithm

If $\mathcal{H} = \mathbb{R}^n$, the algorithm [W., Pokutta '25] is a special case of "FraMes"

Split Conditional Gradient Algorithm [W., Pokutta, '25]

$T = \mathrm{Id}$, $\mathcal{C} = \bigtimes_{i\in I} C_i$ and $\mathcal{D} = D$. Let $\beta_t = \mathcal{O}(1/\log(k+1))$ and $\gamma_k = \mathcal{O}(1/\sqrt{k+1})$. Set $G^{\beta_k}(x_k) = \max_{\boldsymbol{v}\in\bigtimes_{i\in I} C_i} \langle \nabla\Phi_{\beta_k}(\boldsymbol{x}_k) \mid \boldsymbol{x}_k - \boldsymbol{v}\rangle$. Then

$$0 \leqslant \min_{0\leqslant j\leqslant k} G^{\beta_j}(\boldsymbol{x}_j) \leqslant \mathcal{O}\left(\frac{\ln k}{\sqrt{k+1}}\right)$$

Also, $\mathrm{dist}_D(x_k) \to 0$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point yields a stationary point of $(*)$.

[Halbey et al., '25] Could remove the $\ln k$ factor with similar schedules β_k, γ_k.
Q:Will this iteratively-smoothed F-W scheme work for $(\star)$? This analysis provides qualitative stationarity. What about a quantitative convergence rate on $\widetilde{G}$?

Smoothed Frank-Wolfe gap convergence

Theorem (Silveti-Falls, Molinari, W.)

Let $p, q \in]0,1[$ with $q < p$ and $p + q < 1$. Set step sizes $\gamma_k = (k+1)^{-p}$ and smoothing schedule $\beta_k = (k+1)^{-q}$. Then there exists $C_{p,q}$ such that

$$0 \leqslant \frac{1}{N} \sum_{k=0}^{N-1} G^{\beta_k}(x_k) \leqslant C_{p,q} N^{-\min\{p-q,\ 1-p-q\}}$$

Consequently, $0 \leqslant \min_{0 \leqslant k \leqslant N-1} G^{\beta_k}(x_k) \leqslant C_{p,q} N^{-\min\{p-q,\ 1-p-q\}}$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point is a stationary point of $(\star)$.

Smoothed Frank-Wolfe gap convergence

Theorem (Silveti-Falls, Molinari, W.)

Let $p, q \in]0,1[$ with $q < p$ and $p + q < 1$. Set step sizes $\gamma_k = (k+1)^{-p}$ and smoothing schedule $\beta_k = (k+1)^{-q}$. Then there exists $C_{p,q}$ such that

$$0 \leqslant \frac{1}{N} \sum_{k=0}^{N-1} G^{\beta_k}(x_k) \leqslant C_{p,q} N^{-\min\{p-q,\ 1-p-q\}}$$

Consequently, $0 \leqslant \min_{0 \leqslant k \leqslant N-1} G^{\beta_k}(x_k) \leqslant C_{p,q} N^{-\min\{p-q,\ 1-p-q\}}$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point is a stationary point of $(\star)$.

$p = \frac{1}{2}$, $q \to 0$ recovers rates arbitrarily close to $\mathcal{O}(N^{-1/2})$ akin to [Halbey et al., '25] and [W., Pokutta, '25].

Smoothed Frank-Wolfe gap convergence

Theorem (Silveti-Falls, Molinari, W.)

Let $p, q \in]0,1[$ with $q < p$ and $p + q < 1$. Set step sizes $\gamma_k = (k+1)^{-p}$ and smoothing schedule $\beta_k = (k+1)^{-q}$. Then there exists $C_{p,q}$ such that

$$0 \leqslant \frac{1}{N} \sum_{k=0}^{N-1} G^{\beta_k}(x_k) \leqslant C_{p,q} N^{-\min\{p-q,\ 1-p-q\}}$$

Consequently, $0 \leqslant \min_{0 \leqslant k \leqslant N-1} G^{\beta_k}(x_k) \leqslant C_{p,q} N^{-\min\{p-q,\ 1-p-q\}}$. Along the subsequence with $G^{\beta_{k_j}}(\boldsymbol{x}_{k_j}) \to 0$, every cluster point is a stationary point of $(\star)$.

$p = \frac{1}{2}$, $q \to 0$ recovers rates arbitrarily close to $\mathcal{O}(N^{-1/2})$ akin to [Halbey et al., '25] and [W., Pokutta, '25]. To balance optimality and feasibility, we advise $p = \frac{1}{2}$, $q = \frac{1}{4}$.

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -(\overbrace{\sup_{z \in \mathcal{C}} \|\nabla f(z)\|}^{L_f}) \overbrace{\sup_{(z,s) \in \mathcal{C}^2} \|z - s\|}^{\operatorname{diam}_{\mathcal{C}}}$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -(\overbrace{\sup_{z \in \mathcal{C}} \|\nabla f(z)\|}^{L_f}) \overbrace{\sup_{(z,s) \in \mathcal{C}^2} \|z - s\|}^{\operatorname{diam}_{\mathcal{C}}} > -\infty.$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -(\overbrace{\sup_{z \in \mathcal{C}} \|\nabla f(z)\|}^{L_f}) \overbrace{\sup_{(z,s) \in \mathcal{C}^2} \|z - s\|}^{\operatorname{diam}_{\mathcal{C}}} > -\infty.$$

So, along the subsequence $G^{\beta_k}(x_k) \to 0$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -\overbrace{\left(\sup_{z \in \mathcal{C}} \|\nabla f(z)\|\right)}^{L_f} \overbrace{\sup_{(z,s) \in \mathcal{C}^2} \|z - s\|}^{\operatorname{diam}_{\mathcal{C}}} > -\infty.$$

So, along the subsequence $G^{\beta_k}(x_k) \to 0$ (using $\beta_k \to 0$),

$$0 \leqslant \operatorname{dist}^2_{\mathcal{D}}(Tx_k) \leqslant \beta_k G^{\beta_k}(x_k) + \beta_k L_f \operatorname{diam}_{\mathcal{C}} \to 0$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -\overbrace{\left(\sup_{z \in \mathcal{C}} \|\nabla f(z)\|\right)}^{L_f} \overbrace{\sup_{(z,s) \in \mathcal{C}^2} \|z - s\|}^{\operatorname{diam}_{\mathcal{C}}} > -\infty.$$

So, along the subsequence $G^{\beta_k}(x_k) \to 0$ (using $\beta_k \to 0$),

$$0 \leqslant \operatorname{dist}^2_{\mathcal{D}}(Tx_k) \leqslant \beta_k G^{\beta_k}(x_k) + \beta_k L_f \operatorname{diam}_{\mathcal{C}} \to 0$$

hence every cluster point $\overline{x}$ satisfies $T\overline{x} \in \mathcal{D}$.

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Stationarity argument: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -(\overbrace{\sup_{z \in \mathcal{C}} \|\nabla f(z)\|}^{L_f}) \overbrace{\sup_{(z,s) \in \mathcal{C}^2} \|z - s\|}^{\operatorname{diam}_{\mathcal{C}}} > -\infty.$$

So, along the subsequence $G^{\beta_k}(x_k) \to 0$ (using $\beta_k \to 0$),

$$0 \leqslant \operatorname{dist}^2_{\mathcal{D}}(Tx_k) \leqslant \beta_k G^{\beta_k}(x_k) + \beta_k L_f \operatorname{diam}_{\mathcal{C}} \to 0$$

hence every cluster point $\overline{x}$ satisfies $T\overline{x} \in \mathcal{D}$. So, $0 \leqslant \widetilde{G}(\overline{x}) \leqslant \lim_{k \to \infty} G^{\beta_k}(x_k) = 0$.

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Different observation: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}_{\mathcal{D}}^2(Tx) \leqslant G^{\beta}(x).$$

Different observation: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -\overbrace{\left(\sup_{z \in \mathcal{C}} \|\nabla f(z)\|\right)}^{L_f} \overbrace{\|x - \operatorname{proj}_{\widetilde{\mathcal{C}}}(x)\|}^{\operatorname{dist}_{\widetilde{\mathcal{C}}}(x)}$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x).$$

Different observation: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -\overbrace{(\sup_{z \in \mathcal{C}} \|\nabla f(z)\|)}^{L_f} \overbrace{\|x - \operatorname{proj}_{\widetilde{\mathcal{C}}}(x)\|}^{\operatorname{dist}_{\widetilde{\mathcal{C}}}(x)}$$

So, if $\operatorname{dist}_{\widetilde{\mathcal{C}}}(x) \leqslant \kappa \operatorname{dist}_{\mathcal{D}}(Tx)$,

$$\widetilde{G}(x) \geqslant -L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx) \qquad \text{and} \qquad -L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx) + \frac{1}{\beta}\operatorname{dist}^2_{\mathcal{D}}(Tx) \leqslant G^{\beta}(x)$$

Stationarity arguments for $(\star)$

Lemma (Silveti-Falls, Molinari, W.)

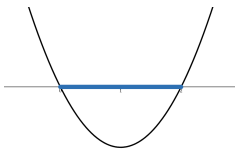
Under Assumption 4.1 ($g = \iota_{\mathcal{D}}$), for every $\beta > 0$,

$$\widetilde{G}(x) + \frac{1}{\beta}\operatorname{dist}_{\mathcal{D}}^2(Tx) \leqslant G^{\beta}(x).$$

Different observation: For $x \in \mathcal{C}$,

$$\widetilde{G}(x) = \max_{s \in \widetilde{\mathcal{C}}} \langle \nabla f(x) \mid x - s \rangle \geqslant -\overbrace{\left(\sup_{z \in \mathcal{C}} \|\nabla f(z)\|\right)}^{L_f} \overbrace{\|x - \operatorname{proj}_{\widetilde{\mathcal{C}}}(x)\|}^{\operatorname{dist}_{\widetilde{\mathcal{C}}}(x)}$$

So, if $\operatorname{dist}_{\widetilde{\mathcal{C}}}(x) \leqslant \kappa \operatorname{dist}_{\mathcal{D}}(Tx)$,



$$\widetilde{G}(x) \geqslant -L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx) \qquad \text{and} \qquad \frac{1}{\beta}\left(\operatorname{dist}_{\mathcal{D}}(Tx)\right)^2 - L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx) - G^{\beta}(x) \leqslant 0$$

What about direct convergence rates ?

What about direct convergence rates ?

Under some standard C.Q.'s (e.g., T surjective and $\operatorname{relint}(\mathcal{C}) \cap \operatorname{relint}(\mathcal{D}) \neq \varnothing$),

$$(\exists \kappa > 0) \qquad (\forall x \in \mathcal{C}) \qquad \operatorname{dist}_{\mathcal{C} \cap T^{-1}(\mathcal{D})}(x) \leqslant \kappa \operatorname{dist}_{\mathcal{D}}(Tx). \tag{A}$$

For the splitting problem, $\bigcap_{i=1}^{m} \operatorname{relint}(C_i) \neq \varnothing$ is sufficient for (A) [Bauschke et al., '99].

What about direct convergence rates ?

Under some standard C.Q.'s (e.g., T surjective and $\operatorname{relint}(\mathcal{C}) \cap \operatorname{relint}(\mathcal{D}) \neq \varnothing$),

$$(\exists \kappa > 0) \qquad (\forall x \in \mathcal{C}) \qquad \operatorname{dist}_{\mathcal{C} \cap T^{-1}(\mathcal{D})}(x) \leqslant \kappa \operatorname{dist}_{\mathcal{D}}(Tx). \tag{A}$$

For the splitting problem, $\bigcap_{i=1}^{m} \operatorname{relint}(C_i) \neq \varnothing$ is sufficient for (A) [Bauschke et al., '99].

Lemma (Silveti-Falls, Molinari, W.)

Let $x \in \mathcal{C}$ and $\beta > 0$. Suppose Assumption 4.1 and (A) hold. Then

$$\operatorname{dist}_{\mathcal{D}}(Tx) \leqslant \frac{L_f \kappa \beta}{2} + \frac{1}{2}\sqrt{L_f \kappa^2 \beta^2 + 4\beta G^{\beta}(x)} = O\Big(\max\{\beta, \sqrt{\beta G^{\beta}(x)}\}\Big)$$

and

$$\left|\widetilde{G}(x)\right| \leqslant \max\{G^{\beta}(x), L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx)\} = \mathcal{O}\left(\max\left\{\beta, G^{\beta}(x)\right\}\right).$$

What about direct convergence rates ?

Lemma (Silveti-Falls, Molinari, W.)

Let $x \in \mathcal{C}$ and $\beta > 0$. Suppose Assumption 4.1 and (A) hold. Then

$$\operatorname{dist}_{\mathcal{D}}(Tx) \leqslant \frac{L_f \kappa \beta}{2} + \frac{1}{2}\sqrt{L_f \kappa^2 \beta^2 + 4\beta G^\beta(x)} = \mathcal{O}\Big(\max\big\{\beta, \sqrt{\beta G^\beta(x)}\big\}\Big)$$

and

$$\left|\widetilde{G}(x)\right| \leqslant \max\{G^\beta(x), L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx)\} = \mathcal{O}\left(\max\left\{\beta, G^\beta(x)\right\}\right).$$

- Characterizes nonsmooth stationarity in terms of smoothing parameter β and smoothed F-W gap G^β

What about direct convergence rates ?

Lemma (Silveti-Falls, Molinari, W.)

Let $x \in \mathcal{C}$ and $\beta > 0$. Suppose Assumption 4.1 and (A) hold. Then

$$\operatorname{dist}_{\mathcal{D}}(Tx) \leqslant \frac{L_f \kappa \beta}{2} + \frac{1}{2}\sqrt{L_f \kappa^2 \beta^2 + 4\beta G^\beta(x)} = \mathcal{O}\Big(\max\{\beta, \sqrt{\beta G^\beta(x)}\}\Big)$$

and

$$\left|\widetilde{G}(x)\right| \leqslant \max\{G^\beta(x), L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx)\} = \mathcal{O}\left(\max\left\{\beta, G^\beta(x)\right\}\right).$$

- Characterizes nonsmooth stationarity in terms of smoothing parameter β and smoothed F-W gap G^β
- Algorithm independent bounds!

What about direct convergence rates ?

Lemma (Silveti-Falls, Molinari, W.)

Let $x \in \mathcal{C}$ and $\beta > 0$. Suppose Assumption 4.1 and (A) hold. Then

$$\operatorname{dist}_{\mathcal{D}}(Tx) \leqslant \frac{L_f \kappa \beta}{2} + \frac{1}{2}\sqrt{L_f \kappa^2 \beta^2 + 4\beta G^\beta(x)} = \mathcal{O}\Big(\max\{\beta, \sqrt{\beta G^\beta(x)}\}\Big)$$

and

$$\left|\widetilde{G}(x)\right| \leqslant \max\{G^\beta(x), L_f \kappa \operatorname{dist}_{\mathcal{D}}(Tx)\} = \mathcal{O}\left(\max\left\{\beta, G^\beta(x)\right\}\right).$$

- Characterizes nonsmooth stationarity in terms of smoothing parameter β and smoothed F-W gap G^β
- Algorithm independent bounds!
- Shows that for previous works [W., Pokutta, '25], [Halbey et al., '25], both feasibility $\operatorname{dist}_{\mathcal{D}}(Tx_k)$ and stationarity $|\widetilde{G}(x_k)|$ converge like $\mathcal{O}(1/\log k)$.

What this means for the algorithm

Theorem (Silveti-Falls, Molinari, W.)

Suppose Assumption 4.1and (A) holds, and let $\{x_k\}_{n\in\mathbb{N}}$ be generated by the algorithm with $\gamma_k = (k+1)^{-1/2}$ and $\beta_k = \beta_0(k+1)^{-1/4}$.
For $N \geqslant 2$, let $k_N^\star \in \underset{\lfloor N/2\rfloor \leqslant k \leqslant N-1}{\operatorname{argmin}} G^{\beta_k}(x_k)$. Then

$$\operatorname{dist}_{\mathcal{D}}(Tx_{k_N^\star}) = \mathcal{O}(N^{-1/4}) \quad \text{and} \quad \left|\widetilde{G}(x_{k_N^\star})\right| = \mathcal{O}(N^{-1/4}).$$

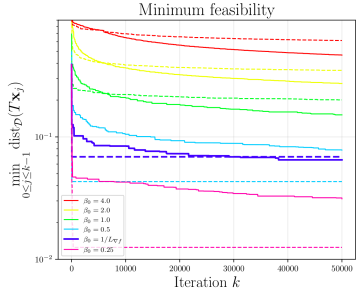
	Convex objective	Nonconvex objective
One constraint	$\mathcal{O}(1/k)$	$\mathcal{O}(1/\sqrt{k})^*$
Multiple constraints	$\mathcal{O}(1/\sqrt{k})^{**}$	[S-F, M, W '26]: $\mathcal{O}(1/k^{1/4})$

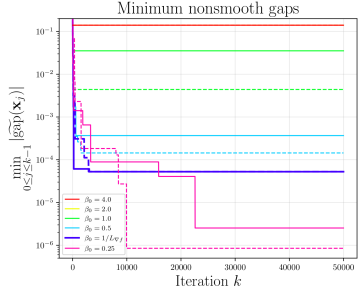
Extra assumptions can increase the rate, e.g.:

*: [Díaz Millán et al., '25];
**: [Wang, Pong, W., '26]

Other results

- Analogous results with g nonsmooth, weakly-convex, & Lipschitz continuous
- Inconsistent problems ($\mathcal{C} \cap T^{-1}\mathcal{D} = \varnothing$)
- Numerical experiments





sold lines – logarithmic smoothing;
Dashed lines – proposed smoothing schedules

Frank-Wolfe Splitting Algorithms

1. Background
2. History & Motivation
3. Approach
4. Analysis
5. Conclusion

Next steps

Input: $x_0 \in \bigtimes_{i=1}^m C_i$, stepsizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing $\{\beta_k\}_{k\in\mathbb{N}} \subset]0,+\infty[$

For $k = 0, 1, 2, \ldots$ **do**

1. $\nabla(\Phi_{\beta_k}(x_k))^i = \nabla f(Ax_k) + \frac{1}{\beta_k}((x_k)^i - A(x_k))$
2. **For** $i = 1, 2, \ldots, m$ **do**
   $s_k^i = \mathrm{LMO}_{C_i}((\nabla\Phi_{\beta_k}(x_k))^i, s\rangle)$.
   End for
3. $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

To be announced:

- Rate improvement to $\mathcal{O}(1/k^{1/3})$

Next steps

Input: $x_0 \in \bigtimes_{i=1}^m C_i$, stepsizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing $\{\beta_k\}_{k\in\mathbb{N}} \subset]0,+\infty[$

For $k = 0, 1, 2, \ldots$ **do**

1. $\nabla(\Phi_{\beta_k}(x_k))^i = \nabla f(Ax_k) + \frac{1}{\beta_k}((x_k)^i - A(x_k))$
2. **For** $i = 1, 2, \ldots, m$ **do**
   $s_k^i = \mathrm{LMO}_{C_i}((\nabla\Phi_{\beta_k}(x_k))^i, s\rangle)$.
   End for
3. $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

To be announced:

- Rate improvement to $\mathcal{O}(1/k^{1/3})$
- Stochastic estimation of ∇f (with V. Wang, A. Silveti-Falls)

Next steps

Input: $x_0 \in \bigtimes_{i=1}^m C_i$, stepsizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing $\{\beta_k\}_{k\in\mathbb{N}} \subset]0,+\infty[$

For $k = 0, 1, 2, \ldots$ **do**

1. $\nabla(\Phi_{\beta_k}(x_k))^i = \nabla f(Ax_k) + \frac{1}{\beta_k}((x_k)^i - A(x_k))$
2. **For** $i = 1, 2, \ldots, m$ **do**
   $s_k^i = \mathrm{LMO}_{C_i}((\nabla\Phi_{\beta_k}(x_k))^i, s\rangle)$.
   End for
3. $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

To be announced:

- Rate improvement to $\mathcal{O}(1/k^{1/3})$
- Stochastic estimation of ∇f (with V. Wang, A. Silveti-Falls)
- Block-iterative LMO activation (with N. Harsell)

Next steps

Input: $x_0 \in \bigtimes_{i=1}^{m} C_i$, stepsizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing $\{\beta_k\}_{k\in\mathbb{N}} \subset]0,+\infty[$

For $k = 0, 1, 2, \ldots$ **do**

1. $\nabla(\Phi_{\beta_k}(x_k))^i = \nabla f(Ax_k) + \frac{1}{\beta_k}((x_k)^i - A(x_k))$
2. **For** $i = 1, 2, \ldots, m$ **do**
   If $i \in I_t$
   $s_k^i = \mathrm{LMO}_{C_i}((\nabla\Phi_{\beta_k}(x_k))^i, s\rangle)$.
   Else $i \in \{1, \ldots, m\} \setminus I_t$
   $s_k^i = s_{k-1}^i$
   End for
3. $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

To be announced:

- Rate improvement to $\mathcal{O}(1/k^{1/3})$
- Stochastic estimation of ∇f (with V. Wang, A. Silveti-Falls)
- Block-iterative LMO activation (with N. Harsell) $(\exists K \in \mathbb{N})(\forall t \in \mathbb{N})$

$$\bigcup_{k=t}^{t+K-1} I_k = \{1, \ldots, m\}$$

Next steps

Input: $x_0 \in \bigtimes_{i=1}^{m} C_i$, stepsizes $\{\gamma_k\}_{k\in\mathbb{N}} \subset]0,1]$, smoothing $\{\beta_k\}_{k\in\mathbb{N}} \subset]0,+\infty[$

For $k = 0, 1, 2, \ldots$ **do**

1. $\nabla(\Phi_{\beta_k}(x_k))^i = \nabla f(Ax_k) + \frac{1}{\beta_k}((x_k)^i - A(x_k))$
2. **For** $i = 1, 2, \ldots, m$ **do**
   If $i \in I_t$
   $s_k^i = \mathrm{LMO}_{C_i}(\nabla\Phi_{\beta_k}(x_k), s\rangle)$.
   Else $i \in \{1, \ldots, m\} \setminus I_t$
   $s_k^i = s_{k-1}^i$
   End for
3. $x_{k+1} = x_k + \gamma_k(s_k - x_k)$.

End for

To be announced:

- Rate improvement to $\mathcal{O}(1/k^{1/3})$
- Stochastic estimation of ∇f (with V. Wang, A. Silveti-Falls)
- Block-iterative LMO activation (with N. Harsell) $(\exists K \in \mathbb{N})(\forall t \in \mathbb{N})$

$$\bigcup_{k=t}^{t+K-1} I_k = \{1, \ldots, m\}$$

- Adaptive smoothing

Conclusion

"Moreau smoothing + one F-W step" works in more settings than splitting.

Contact: woodstzc[at]jmu.edu

Preprint:
Frank-Wolfe with Moreau Envelope Smoothing for Nonsmooth Nonconvex Problems,
Antonio Silveti-Falls, Cesare Molinari, and Zev Woodstock,
arXiv arXiv:2606.00853

Thank you for your attention!

References


K. Asgari, M. J. Neely, Nonsmooth projection-free optimization with functional constraints
Comput. Optim. Appl., vol. 89 (3), pp. 927-–975, 2024.


H. H. Bauschke, J. M. Borwein, W. Li, Strong conical hull intersection property, bounded linear regularity, Jameson's property (G), and error bounds in convex optimization
Math. Program., Ser. A, vol. 86 (4) pp. 135-–160, 1999.


M. Besançon, M. Carderera, and S. Pokutta, FrankWolfe. jl: A high-performance and flexible toolbox for Frank–Wolfe algorithms and conditional gradients
INFORMS J. Comput., vol. 34 (5), pp. 2611–2620, 2022.


J. Bolte, C. W. Combettes, and E. Pauwels, The iterates of the Frank–Wolfe algorithm may not converge
Math. Oper. Res., vol. 49 (4), pp. 2565–2578, 2024.


C. W. Combettes and S. Pokutta, Complexity of linear minimization and projection on some sets
Oper. Res. Lett., vol. 49, no. 4, pp. 565–571, 2021


R. Díaz Millán, O. P. Ferreira, and J. Ugon, Frank–Wolfe algorithms for piecewise star-convex functions with a nonsmooth difference-of-convex structure
arXiv preprint 2308.16444, rev. 2026.

References

D. Garber, A. Kaplan, and S. Sabach, Improved complexities of conditional gradient-type methods with applications to robust matrix recovery problems,
Math. Program., vol. 186, pp. 185–208, 2021.

G. Gidel, F. Pedregosa, and S. Lacoste-Julien, Frank-Wolfe splitting via augmented Lagrangian Method,
Proc. AISTATS, pp. 1456–1465, 2018.

J. Halbey, S. Rakotomandimby, M. Besançon, S. Designolle, and S. Pokutta, Efficient quadratic corrections for Frank-Wolfe algorithms,
Proc. NeurIPS 38, 2025.

T. Hu, Robustness of the Frank-Wolfe method under inexact oracles and the cost of linear minimization
arXiv preprint 2601.11548, 2026.

M. Jaggi, Revisiting Frank-Wolfe: Projection-free sparse convex optimization,
Proc. 30th International Conference on Machine Learning, in PMLR, vol. 28 (1), pp. 427–435, 2013.

References

G. Lan, E. Romeijn, and Z. Zhou, Conditional gradient methods for convex optimization with general affine and nonlinear constraints,
SIAM J. Optim., vol. 31, no. 3, pp. 2307–2339, 2021.

G. Pierra, Decomposition through formalization in a product space,
Math. Program., vol. 28, pp. 96–115, 1984.

A. Silveti-Falls, C. Molinari and J. Fadili, Inexact and stochastic generalized conditional gradient with augmented Lagrangian and proximal step,
J. Nonsmooth Anal. Optim., vol. 2, 2021.

X. Wang, T. K. Pong, and Z. Woodstock, A conditional-gradient-based single-loop augmented Lagrangian method for inequality constrained optimization,
arXiv preprint 2605.22539, 2026.

Z. Woodstock, High-precision linear minimization is no slower than projection,
Optim. Lett., vol. 20, pp. 909–915, 2026.

Z. Woodstock and S. Pokutta, Splitting the conditional gradient algorithm,
SIAM J. Optim., vol. 35 (1), pp. 347–368, 2025.

References


A. Yurtsever, O. Fercoq, F. Locatello, and V. Cevher, A conditional gradient framework for composite convex minimization with applications to semidefinite programming
Proc. ICML, 2018.


A. Yurtsever, O. Fercoq, and V. Cevher, A conditional-gradient-based augmented Lagrangian framework
Proc. ICML, pp. 7272–7281, 2019.