

# ~~Breaking the cycle:~~ Flexible block-iterative analysis for the Frank-Wolfe algorithm

ISMP 2024, Montréal, QC

**Zev Woodstock\***, Gábor Braun, and Sebastian Pokutta

Zuse Institute Berlin (ZIB) & Technische Universität Berlin  
Interactive Optimization and Learning (IOL) Lab

July 2024

\*- also James Madison University starting Aug. 2024



# Flexible Block-Coordinate Frank-Wolfe Algorithm

1. Motivation
2. Our approach
3. Analysis
4. Numerical experiments

## Problem setting

Given  $m$  nonempty closed convex sets  $C_i \subset \mathbb{R}^{n_i}$  with  $i \in \{1, \dots, m\} =: I$  and a smooth function  $f: \mathbb{R}^N \rightarrow \mathbb{R}$  with  $N = \sum_{i \in I} n_i$ , solve

$$\underset{\mathbf{x} \in C_1 \times \dots \times C_m}{\text{minimize}} \quad f(\mathbf{x}). \quad (1)$$

Applications: matrix factorization, SVM training, sequence labeling, splitting, ...

## Problem setting

Given  $m$  nonempty closed convex sets  $C_i \subset \mathbb{R}^{n_i}$  with  $i \in \{1, \dots, m\} =: I$  and a smooth function  $f: \mathbb{R}^N \rightarrow \mathbb{R}$  with  $N = \sum_{i \in I} n_i$ , solve

$$\underset{\mathbf{x} \in C_1 \times \dots \times C_m}{\text{minimize}} \quad f(\mathbf{x}). \quad (1)$$

Applications: matrix factorization, SVM training, sequence labeling, splitting, ...

Two families of first-order methods to solve (1): **projection** methods and Frank-Wolfe AKA “CG” methods, which use **linear minimization oracles**.

$$\text{proj}_C(\mathbf{x}) = \underset{\mathbf{v} \in C}{\text{Argmin}} \|\mathbf{x} - \mathbf{v}\|^2 \quad \text{LMO}_C(\mathbf{x}) \in \underset{\mathbf{v} \in C}{\text{Argmin}} \langle \mathbf{x} \mid \mathbf{v} \rangle \quad (2)$$

## Problem setting

Given  $m$  nonempty closed convex sets  $C_i \subset \mathbb{R}^{n_i}$  with  $i \in \{1, \dots, m\} =: I$  and a smooth function  $f: \mathbb{R}^N \rightarrow \mathbb{R}$  with  $N = \sum_{i \in I} n_i$ , solve

$$\underset{\mathbf{x} \in C_1 \times \dots \times C_m}{\text{minimize}} \quad f(\mathbf{x}). \quad (1)$$

Applications: matrix factorization, SVM training, sequence labeling, splitting, ...

Two families of first-order methods to solve (1): **projection** methods and Frank-Wolfe AKA “CG” methods, which use **linear minimization oracles**.

$$\text{proj}_C(\mathbf{x}) = \underset{\mathbf{v} \in C}{\text{Argmin}} \|\mathbf{x} - \mathbf{v}\|^2 \quad \text{LMO}_C(\mathbf{x}) \in \underset{\mathbf{v} \in C}{\text{Argmin}} \langle \mathbf{x} \mid \mathbf{v} \rangle \quad (2)$$

[Combettes/Pokutta, '21]: For many constraints,  $C$ ,  $\text{proj}_C$  is **more expensive** than  $\text{LMO}_C$ .  
(e.g., nuclear norm ball,  $\ell_1$  ball, probability simplex, Birkhoff polytope, general LP, ...)

## Problem setting

Given  $m$  nonempty closed convex sets  $C_i \subset \mathbb{R}^{n_i}$  with  $i \in \{1, \dots, m\} =: I$  and a smooth function  $f: \mathbb{R}^N \rightarrow \mathbb{R}$  with  $N = \sum_{i \in I} n_i$ , solve

$$\underset{\mathbf{x} \in C_1 \times \dots \times C_m}{\text{minimize}} \quad f(\mathbf{x}). \quad (1)$$

Applications: matrix factorization, SVM training, sequence labeling, splitting, ...

For  $\mathbf{x} \in \mathbb{R}^N$  with components  $\mathbf{x} = (\mathbf{x}^1, \dots, \mathbf{x}^m)$  ( $\mathbf{x}_i \in \mathbb{R}^{n_i}$ ),

$$\text{LMO}_{C_1 \times \dots \times C_m}(\mathbf{x}^1, \dots, \mathbf{x}^m) = (\text{LMO}_{C_1} \mathbf{x}^1, \dots, \text{LMO}_{C_m} \mathbf{x}^m) \quad (\text{\$}\text{\$}\text{\$})$$

## Problem setting

Given  $m$  nonempty closed convex sets  $C_i \subset \mathbb{R}^{n_i}$  with  $i \in \{1, \dots, m\} =: I$  and a smooth function  $f: \mathbb{R}^N \rightarrow \mathbb{R}$  with  $N = \sum_{i \in I} n_i$ , solve

$$\underset{\mathbf{x} \in C_1 \times \dots \times C_m}{\text{minimize}} \quad f(\mathbf{x}). \quad (1)$$

Applications: matrix factorization, SVM training, sequence labeling, splitting, ...

For  $\mathbf{x} \in \mathbb{R}^N$  with components  $\mathbf{x} = (\mathbf{x}^1, \dots, \mathbf{x}^m)$  ( $\mathbf{x}_i \in \mathbb{R}^{n_i}$ ),

$$\text{LMO}_{C_1 \times \dots \times C_m}(\mathbf{x}^1, \dots, \mathbf{x}^m) = (\text{LMO}_{C_1} \mathbf{x}^1, \dots, \text{LMO}_{C_m} \mathbf{x}^m) \quad (\text{\$ \$ \$})$$

“Let’s avoid computing so many LMOs per iteration!” (paraphrased)

– [Patriksson, '98], [Lacoste-Julien et al., 2013], [Beck et al., 2015], [Wang et al., 2016], [Osokin et al., 2016], [Bomze et al., 2024], ...

# (Generic) BCFW Algorithm

Known modes of convergence:

```
1: for  $t = 0, 1$  to ... do
2:   Select  $l_t \subset \{1, \dots, m\}$ 
3:    $\mathbf{g}_t \leftarrow \nabla f(\mathbf{x}_t)$ 
4:   for  $i = 1$  to  $m$  do
5:     if  $i \in l_t$  then
6:        $\mathbf{v}_t^i \leftarrow \text{LMO}_i(\mathbf{g}_t)$ 
7:        $\gamma_t^i \leftarrow$  Step size
8:        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i + \gamma_t^i(\mathbf{v}_t^i - \mathbf{x}_t^i)$ 
9:     else
10:       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$ 
11:    end if
12:  end for
13: end for
```



# (Generic) BCFW Algorithm

Known modes of convergence:

```
1: for  $t = 0, 1$  to ... do
2:   Select  $I_t \subset \{1, \dots, m\}$ 
3:    $\mathbf{g}_t \leftarrow \nabla f(\mathbf{x}_t)$ 
4:   for  $i = 1$  to  $m$  do
5:     if  $i \in I_t$  then
6:        $\mathbf{v}_t^i \leftarrow \text{LMO}_i(\mathbf{g}_t^i)$ 
7:        $\gamma_t^i \leftarrow$  Step size
8:        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i + \gamma_t^i(\mathbf{v}_t^i - \mathbf{x}_t^i)$ 
9:     else
10:       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$ 
11:    end if
12:  end for
13: end for
```

- [Patriksson, 1998]:

- Asymptotic convergence if  $f$  convex
- Exact and Armijo linesearches fixed across all components  $\gamma_t^i = \gamma_t$
- Full update ( $I_t = \{1, \dots, m\}$ )
- Deterministic essentially cyclic ( $\exists K > 0$ ):

$$I_t = \{i_t\}, \text{ with } \{i_t, \dots, i_{t+K}\} = \{1, \dots, m\}$$

# (Generic) BCFW Algorithm

Known modes of convergence:

```
1: for  $t = 0, 1$  to ... do
2:   Select  $I_t \subset \{1, \dots, m\}$ 
3:    $\mathbf{g}_t \leftarrow \nabla f(\mathbf{x}_t)$ 
4:   for  $i = 1$  to  $m$  do
5:     if  $i \in I_t$  then
6:        $\mathbf{v}_t^i \leftarrow \text{LMO}_i(\mathbf{g}_t^i)$ 
7:        $\gamma_t^i \leftarrow$  Step size
8:        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i + \gamma_t^i(\mathbf{v}_t^i - \mathbf{x}_t^i)$ 
9:     else
10:       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$ 
11:    end if
12:  end for
13: end for
```

- [Patriksson, 1998]:

- Asymptotic convergence if  $f$  convex
- Exact and Armijo linesearches fixed across all components  $\gamma_t^i = \gamma_t$
- Full update ( $I_t = \{1, \dots, m\}$ )
- Deterministic essentially cyclic ( $\exists K > 0$ ):

$$I_t = \{i_t\}, \text{ with } \{i_t, \dots, i_{t+K}\} = \{1, \dots, m\}$$

- [Beck et al., 2015]:

- $\mathcal{O}(1/t)$  convergence ( $f$  convex)
- open-loop, short-step, and backtracking  $\gamma_t^i$
- Deterministic cyclic updates

$$I_t = \{i_t\}, \text{ with } \{i_t, \dots, i_{t+m}\} = \{1, \dots, m\}$$

# (Generic) BCFW Algorithm

Known modes of convergence:

```
1: for  $t = 0, 1$  to  $\dots$  do
2:   Select  $l_t \subset \{1, \dots, m\}$ 
3:    $\mathbf{g}_t \leftarrow \nabla f(\mathbf{x}_t)$ 
4:   for  $i = 1$  to  $m$  do
5:     if  $i \in l_t$  then
6:        $\mathbf{v}_t^i \leftarrow \text{LMO}_i(\mathbf{g}_t^i)$ 
7:        $\gamma_t^i \leftarrow$  Step size
8:        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i + \gamma_t^i(\mathbf{v}_t^i - \mathbf{x}_t^i)$ 
9:     else
10:       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$ 
11:    end if
12:  end for
13: end for
```

- Stochastic variants:
  - $\mathcal{O}(1/t)$  primal convergence rate ( $f$  convex)
  - Uniform singleton selection [Lacoste-Julien et al., 2013]
  - Non-uniform singleton selection (based on suboptimality criterion) [Osokin et al., 2016]
  - Uniform parallel selection with fixed block-sizes  $|l_t| = p$  [Wang et al., 2016]

# (Generic) BCFW Algorithm

Known modes of convergence:

```
1: for  $t = 0, 1$  to  $\dots$  do
2:   Select  $l_t \subset \{1, \dots, m\}$ 
3:    $\mathbf{g}_t \leftarrow \nabla f(\mathbf{x}_t)$ 
4:   for  $i = 1$  to  $m$  do
5:     if  $i \in l_t$  then
6:        $\mathbf{v}_t^i \leftarrow \text{LMO}_i(\mathbf{g}_t^i)$ 
7:        $\gamma_t^i \leftarrow$  Step size
8:        $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i + \gamma_t^i(\mathbf{v}_t^i - \mathbf{x}_t^i)$ 
9:     else
10:       $\mathbf{x}_{t+1}^i \leftarrow \mathbf{x}_t^i$ 
11:    end if
12:  end for
13: end for
```

- Stochastic variants:
  - $\mathcal{O}(1/t)$  primal convergence rate ( $f$  convex)
  - Uniform singleton selection [Lacoste-Julien et al., 2013]
  - Non-uniform singleton selection (based on suboptimality criterion) [Osokin et al., 2016]
  - Uniform parallel selection with fixed block-sizes  $|l_t| = p$  [Wang et al., 2016]
- [Bomze et al., 2024]:
  - Linear convergence (KL condition +  $\dots$ )
  - Short-Step Chain (SSC) procedure:  $\gamma_t^i, \mathbf{v}_t^i$
  - Full updates ( $l_t = \{1, \dots, m\}$ )
  - Uniform singleton selection ( $l_t = \{i_t\}$ )
  - Gauss-Southwell “greedy” singleton updates (based on suboptimality criterion).

## Let's recap...

- **Singleton updates:**  
→ cyclic, essentially cyclic, Gauss-Southwell, (uniform or non-uniform) random
- **Parallel updates:**  
→ Full ( $I_t = \{1, \dots, m\}$ ), or uniformly-random blocks of fixed size  $|I_t| = p$

What if my LMOs have very different costs? What if I only have 4 processor cores?

## Let's recap...

- **Singleton updates:**  
→ cyclic, essentially cyclic, Gauss-Southwell, (uniform or non-uniform) random
- **Parallel updates:**  
→ Full ( $I_t = \{1, \dots, m\}$ ), or uniformly-random blocks of fixed size  $|I_t| = p$

What if my LMOs have very different costs? What if I only have 4 processor cores?

What about...

- **deterministic** parallel updates?

## Let's recap...

- **Singleton updates:**  
→ cyclic, essentially cyclic, Gauss-Southwell, (uniform or non-uniform) random
- **Parallel updates:**  
→ Full ( $I_t = \{1, \dots, m\}$ ), or uniformly-random blocks of fixed size  $|I_t| = p$

What if my LMOs have very different costs? What if I only have 4 processor cores?

What about...

- **deterministic** parallel updates?
- blocks with **different sizes**?

## Let's recap...

- **Singleton updates:**  
→ cyclic, essentially cyclic, Gauss-Southwell, (uniform or non-uniform) random
- **Parallel updates:**  
→ Full ( $I_t = \{1, \dots, m\}$ ), or uniformly-random blocks of fixed size  $|I_t| = p$

What if my LMOs have very different costs? What if I only have 4 processor cores?

What about...

- **deterministic** parallel updates?
- blocks with **different sizes**?
- **cost-aware** methodologies? (e.g., if some LMOs are numerically expensive, and others are cheap)



# Flexible Block-Coordinate Frank-Wolfe Algorithm

1. Motivation
2. Our approach
3. Analysis
4. Numerical experiments

## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (\star)$$

## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (*)$$

Allows for:

- Deterministic, variable-size, parallel updates

## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (*)$$

Allows for:

- Deterministic, variable-size, parallel updates
- Already known to converge: Full, cyclic, essentially cyclic, ...

## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (*)$$

Allows for:

- Deterministic, variable-size, parallel updates
- Already known to converge: Full, cyclic, essentially cyclic, ...
- **“Lazy” updates**: Over  $K$  iterations, update expensive LMO(s) once, and update cheap LMOs multiple times.

## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (\star)$$

Allows for:

- Deterministic, variable-size, parallel updates
- Already known to converge: Full, cyclic, essentially cyclic, ...
- **“Lazy” updates**: Over  $K$  iterations, update expensive LMO(s) once, and update cheap LMOs multiple times.

→ We can set the ratio of  $\frac{(\text{expensive LMO evals})}{(\text{cheap LMO evals})} = \frac{1}{K}$  arbitrarily small.

## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (*)$$

To my knowledge, first appears in [Ottavy, 1988].



## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (*)$$

To my knowledge, first appears in [Ottavy, 1988].

Related to lazily updating Hessians in Newton's method [Shamanskii, 1967]



**1967:**  
Canada  
turns 100!



## A bit of history

### Assumption

There exists a positive integer  $K$  such that, for every iteration  $t$ ,

$$(\forall 1 \leq i \leq m) \quad i \in \bigcup_{n=t}^{t+K-1} I_n. \quad (*)$$

To my knowledge, first appears in [Ottavy, 1988].

Related to lazily updating Hessians in Newton's method [Shamanskii, 1967]

Apparently **never considered for F-W algorithms** before!?



1967:

Canada  
turns 100!

# Goals

Under Assumption  $(\star)$ , establish competitive convergence rates.

What we did:

- $f$  convex:  $\mathcal{O}(K/t)$  rate (for primal gap) using:
  - Short-step  $\gamma_t^i$
  - An adaptive stepsize scheme  $\gamma_t^i$
- $f$  nonconvex:  $\mathcal{O}(K/\sqrt{t})$  rate (for F-W optimality gap) using short-step  $\gamma_t^i$
- Some conjectures and interesting analysis along the way...

# Flexible Block-Coordinate Frank-Wolfe Algorithm

1. Motivation
2. Our approach
- 3. Analysis**
4. Numerical experiments

# Notation and Background

Frank Wolfe gaps

Recall  $I = \{1, \dots, m\}$ . The **Frank-Wolfe gap** at  $\mathbf{x} \in \mathbb{R}^N$  is

$$G_I(\mathbf{x}) = \langle \nabla f(\mathbf{x}) \mid \mathbf{x} - \text{LMO}_{\mathbf{x} \in I} c_i(\nabla f(\mathbf{x})) \rangle$$

# Notation and Background

Frank Wolfe gaps

Recall  $I = \{1, \dots, m\}$ . The **Frank-Wolfe gap** at  $\mathbf{x} \in \mathbb{R}^N$  is

$$G_I(\mathbf{x}) = \langle \nabla f(\mathbf{x}) \mid \mathbf{x} - \text{LMO}_{\mathbf{x}_{i \in I} \mathbf{c}_i}(\nabla f(\mathbf{x})) \rangle = \sum_{i \in I} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{\mathbf{c}_i}(\nabla^i f(\mathbf{x})) \rangle.$$

# Notation and Background

## Frank Wolfe gaps

Recall  $I = \{1, \dots, m\}$ . The **Frank-Wolfe gap** at  $\mathbf{x} \in \mathbb{R}^N$  is

$$G_I(\mathbf{x}) = \langle \nabla f(\mathbf{x}) \mid \mathbf{x} - \text{LMO}_{\mathbf{x}_{i \in I} \mathbf{c}_i}(\nabla f(\mathbf{x})) \rangle = \sum_{i \in I} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{\mathbf{c}_i}(\nabla^i f(\mathbf{x})) \rangle.$$

A **partial Frank-Wolfe gap** is given by

$$(\forall J \subset I) \quad G_J(\mathbf{x}) = \sum_{i \in J} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{\mathbf{c}_i}(\nabla^i f(\mathbf{x})) \rangle$$

# Notation and Background

Frank Wolfe gaps

Recall  $I = \{1, \dots, m\}$ . The **Frank-Wolfe gap** at  $\mathbf{x} \in \mathbb{R}^N$  is

$$G_I(\mathbf{x}) = \langle \nabla f(\mathbf{x}) \mid \mathbf{x} - \text{LMO}_{\times_{i \in I} C_i}(\nabla f(\mathbf{x})) \rangle = \sum_{i \in I} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{C_i}(\nabla^i f(\mathbf{x})) \rangle.$$

A **partial Frank-Wolfe gap** is given by

$$(\forall J \subset I) \quad G_J(\mathbf{x}) = \sum_{i \in J} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{C_i}(\nabla^i f(\mathbf{x})) \rangle$$

## Fact

(A) If  $\mathbf{x} \in \times_{i \in I} C_i$ , then  $(\forall J \subset I) \quad G_J(\mathbf{x}) \geq 0$ .

(B)  $\mathbf{x}$  is a stationary point of (1) if and only if  $\mathbf{x} \in \times_{i \in I} C_i$  and  $G_I(\mathbf{x}) = 0$ .

# Notation and Background

Frank Wolfe gaps

Recall  $I = \{1, \dots, m\}$ . The **Frank-Wolfe gap** at  $\mathbf{x} \in \mathbb{R}^N$  is

$$G_I(\mathbf{x}) = \langle \nabla f(\mathbf{x}) \mid \mathbf{x} - \text{LMO}_{\times_{i \in I} C_i}(\nabla f(\mathbf{x})) \rangle = \sum_{i \in I} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{C_i}(\nabla^i f(\mathbf{x})) \rangle.$$

A **partial Frank-Wolfe gap** is given by

$$(\forall J \subset I) \quad G_J(\mathbf{x}) = \sum_{i \in J} \langle \nabla^i f(\mathbf{x}) \mid \mathbf{x}^i - \text{LMO}_{C_i}(\nabla^i f(\mathbf{x})) \rangle$$

Fact

(A) If  $\mathbf{x} \in \times_{i \in I} C_i$ , then  $(\forall J \subset I) \quad G_J(\mathbf{x}) \geq 0$ .

(B)  $\mathbf{x}$  is a stationary point of (1) if and only if  $\mathbf{x} \in \times_{i \in I} C_i$  and  $G_I(\mathbf{x}) = 0$ .

$\Rightarrow$  nonconvex convergence results typically show **first order criticality**:  $G_I(\mathbf{x}_t) \rightarrow 0$ .



# Notation and Background

## Smoothness and short-steps

For  $L_f > 0$ , the function  $f$  is  $L_f$ -**smooth** on a convex set  $C$  if

$$(\forall \mathbf{x}, \mathbf{y} \in C) \quad f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f(\mathbf{x}) \mid \mathbf{y} - \mathbf{x} \rangle + \frac{L_f}{2} \|\mathbf{y} - \mathbf{x}\|^2.$$

# Notation and Background

## Smoothness and short-steps

For  $L_f > 0$ , the function  $f$  is  $L_f$ -**smooth** on a convex set  $C$  if

$$(\forall \mathbf{x}, \mathbf{y} \in C) \quad f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f(\mathbf{x}) \mid \mathbf{y} - \mathbf{x} \rangle + \frac{L_f}{2} \|\mathbf{y} - \mathbf{x}\|^2.$$

For BCFW, this means

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t) \leq \sum_{i \in I_t} \gamma_t^i \underbrace{\langle \nabla^i f(\mathbf{x}_t) \mid \mathbf{v}_t^i - \mathbf{x}_t^i \rangle}_{-G_i(\mathbf{x}_t)} + \frac{L_f}{2} (\gamma_t^i)^2 \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2.$$

# Notation and Background

## Smoothness and short-steps

For  $L_f > 0$ , the function  $f$  is  $L_f$ -**smooth** on a convex set  $C$  if

$$(\forall \mathbf{x}, \mathbf{y} \in C) \quad f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f(\mathbf{x}) \mid \mathbf{y} - \mathbf{x} \rangle + \frac{L_f}{2} \|\mathbf{y} - \mathbf{x}\|^2.$$

For BCFW, this means

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t) \leq \sum_{i \in I_t} \gamma_t^i \underbrace{\langle \nabla^i f(\mathbf{x}_t) \mid \mathbf{v}_t^i - \mathbf{x}_t^i \rangle}_{-G_i(\mathbf{x}_t)} + \frac{L_f}{2} (\gamma_t^i)^2 \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2.$$

To tighten the inequality, the stepsize

$$\gamma_t^i = \underset{\gamma \in [0,1]}{\text{Argmin}} \left( -\gamma G_i(\mathbf{x}_t) + \gamma^2 \frac{L_f}{2} \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2 \right) = \min \left\{ \frac{G_i(\mathbf{x}_t)}{L_f \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2}, 1 \right\}, \quad (\text{short})$$

is known as the componentwise **short step**.

# Notation and Background

## Smoothness and short-steps

For  $L_f > 0$ , the function  $f$  is  $L_f$ -**smooth** on a convex set  $C$  if

$$(\forall \mathbf{x}, \mathbf{y} \in C) \quad f(\mathbf{y}) - f(\mathbf{x}) \leq \langle \nabla f(\mathbf{x}) \mid \mathbf{y} - \mathbf{x} \rangle + \frac{L_f}{2} \|\mathbf{y} - \mathbf{x}\|^2.$$

For BCFW, this means

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t) \leq \sum_{i \in I_t} \gamma_t^i \underbrace{\langle \nabla^i f(\mathbf{x}_t) \mid \mathbf{v}_t^i - \mathbf{x}_t^i \rangle}_{-G_i(\mathbf{x}_t)} + \frac{L_f}{2} (\gamma_t^i)^2 \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2.$$

To tighten the inequality, the stepsize

$$\gamma_t^i = \underset{\gamma \in [0,1]}{\text{Argmin}} \left( -\gamma G_i(\mathbf{x}_t) + \gamma^2 \frac{L_f}{2} \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2 \right) = \min \left\{ \frac{G_i(\mathbf{x}_t)}{\textcolor{red}{L}_f \|\mathbf{v}_t^i - \mathbf{x}_t^i\|^2}, 1 \right\}, \quad (\text{short})$$

is known as the componentwise **short step**. Downside: requires upper-estimate of  $\textcolor{red}{L}_f$ .

# Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

## Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

1. Update  $\gamma_t^i$  based on an estimated the smoothness constant  $\widetilde{M}$ .

## Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

1. Update  $\gamma_t^i$  based on an estimated the smoothness constant  $\widetilde{M}$ .
2. If a desired inequality holds between  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$ : done.

## Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

1. Update  $\gamma_t^i$  based on an estimated the smoothness constant  $\widetilde{M}$ .
2. If a desired inequality holds between  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$ : done.
3. Else, increase  $\widetilde{M} \leftarrow \tau \widetilde{M}$  by  $\tau > 1$  and recompute  $\mathbf{x}_{t+1}$  until the desired inequality holds.



## Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

1. Update  $\gamma_t^i$  based on an estimated the smoothness constant  $\widetilde{M}$ .
2. If a desired inequality holds between  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$ : done.
3. Else, increase  $\widetilde{M} \leftarrow \tau \widetilde{M}$  by  $\tau > 1$  and recompute  $\mathbf{x}_{t+1}$  until the desired inequality holds.

**Pros:** No a-priori knowledge of  $L_f$ ; sometimes we get larger steps.

**Cons:** Extra function and/or gradient evaluations.

## Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

1. Update  $\gamma_t^i$  based on an estimated the smoothness constant  $\widetilde{M}$ .
2. If a desired inequality holds between  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$ : done.
3. Else, increase  $\widetilde{M} \leftarrow \tau \widetilde{M}$  by  $\tau > 1$  and recompute  $\mathbf{x}_{t+1}$  until the desired inequality holds.

**Pros:** No a-priori knowledge of  $L_f$ ; sometimes we get larger steps.

**Cons:** Extra function and/or gradient evaluations.

Fact (Hazan & Luo, 2016)

Let  $f$  be convex and  $L_f$ -smooth. Then,

$$(\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^N) \quad f(\mathbf{x}) - f(\mathbf{y}) - \langle \nabla f(\mathbf{y}) \mid \mathbf{x} - \mathbf{y} \rangle \geq \frac{\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2}{2L_f}.$$

## Adaptive step-size algorithm for convex functions

Typical adaptive setup [Pedregosa et al., 2020], [Pokutta, 2023]:

1. Update  $\gamma_t^i$  based on an estimated the smoothness constant  $\widetilde{M}$ .
2. If  $(2^*)$  holds between  $\mathbf{x}_t$  and  $\mathbf{x}_{t+1}$ : done.
3. Else, increase  $\widetilde{M} \leftarrow \tau \widetilde{M}$  by  $\tau > 1$  and recompute  $\mathbf{x}_{t+1}$  until  $(2^*)$  holds.

**Pros:** No a-priori knowledge of  $L_f$ ; sometimes we get larger steps.

**Cons:** Extra function and/or gradient evaluations.

Fact (Hazan & Luo, 2016)

Let  $f$  be convex and  $L_f$ -smooth. Then, for  $\widetilde{M}$  sufficiently large,

$$f(\mathbf{x}_t) - f(\mathbf{x}_{t+1}) - \langle \nabla f(\mathbf{x}_{t+1}) \mid \mathbf{x}_t - \mathbf{x}_{t+1} \rangle \geq \frac{\|\nabla f(\mathbf{x}_t) - \nabla f(\mathbf{x}_{t+1})\|^2}{2\widetilde{M}}. \quad (2^*)$$

## Progress lemma

Lemma (Progress bound via smoothness and convexity, short-step)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , and assume  $(\star)$ . Let  $\mathbf{x}^*$  solve (1), and set  $H_t = f(\mathbf{x}_t) - f(\mathbf{x}^*)$ . Then

$$H_t - H_{t+K} \geq \begin{cases} H_t + A_t - \frac{KL_f D^2}{2}, & \text{if } H_t + A_t \geq KL_f D^2; \\ \frac{(H_t + A_t)^2}{2KL_f D^2}, & \text{if } H_t + A_t \leq KL_f D^2, \text{ where} \end{cases}$$

$$A_t = \sum_{k=1}^{K-1} G_{\underbrace{I_{t+k-1} \cap (I_{t+k} \cup \dots \cup I_{t+K-1})}_{J_k}}(\mathbf{x}_{t+k}) \geq 0$$

$A_t$  describes partial F-W gaps for **all re-activated components**.

## Progress lemma

Lemma (Progress bound via smoothness and convexity, short-step)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , and assume  $(\star)$ . Let  $\mathbf{x}^*$  solve (1), and set  $H_t = f(\mathbf{x}_t) - f(\mathbf{x}^*)$ . Then

$$H_t - H_{t+K} \geq \begin{cases} H_t + A_t - \frac{KL_f D^2}{2}, & \text{if } H_t + A_t \geq KL_f D^2; \\ \frac{(H_t + A_t)^2}{2KL_f D^2}, & \text{if } H_t + A_t \leq KL_f D^2, \text{ where} \end{cases}$$

$$A_t = \sum_{k=1}^{K-1} \underbrace{G_{I_{t+k-1} \cap (I_{t+k} \cup \dots \cup I_{t+K-1})}}_{J_k}(\mathbf{x}_{t+k}) \geq \sum_{k=1}^{K-1} f(\mathbf{x}_{t+k}) - \min_{\substack{\mathbf{x} \in \times_{i \in I} C_i \\ \mathbf{x} \wedge J_k = \mathbf{x}_{t+k}^{\wedge J_k}}} f(\mathbf{x}) \geq 0.$$

$A_t$  describes partial F-W gaps for **all re-activated components**.

## Progress lemma

Lemma (Progress bound via smoothness and convexity, short-step)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , and assume  $(\star)$ . Let  $\mathbf{x}^*$  solve (1), and set  $H_t = f(\mathbf{x}_t) - f(\mathbf{x}^*)$ . Then

$$H_t - H_{t+K} \geq \begin{cases} H_t + A_t - \frac{KL_f D^2}{2}, & \text{if } H_t + A_t \geq KL_f D^2; \\ \frac{(H_t + A_t)^2}{2KL_f D^2}, & \text{if } H_t + A_t \leq KL_f D^2, \text{ where} \end{cases}$$

$$A_t = \sum_{k=1}^{K-1} \underbrace{G_{I_{t+k-1} \cap (I_{t+k} \cup \dots \cup I_{t+K-1})}}_{J_k}(\mathbf{x}_{t+k}) \geq \sum_{k=1}^{K-1} f(\mathbf{x}_{t+k}) - \min_{\substack{\mathbf{x} \in \times_{i \in I} C_i \\ \mathbf{x} \wedge J_k = \mathbf{x}_{t+k}^{\wedge J_k}}} f(\mathbf{x}) \geq 0.$$

$A_t$  may explain good behavior in experiments.

## Progress lemma

Lemma (Progress bound via smoothness and convexity, short-step)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , and assume  $(\star)$ . Let  $\mathbf{x}^*$  solve (1), and set  $H_t = f(\mathbf{x}_t) - f(\mathbf{x}^*)$ . Then

$$H_t - H_{t+K} \geq \begin{cases} H_t + A_t - \frac{KL_f D^2}{2}, & \text{if } H_t + A_t \geq KL_f D^2; \\ \frac{(H_t + A_t)^2}{2KL_f D^2}, & \text{if } H_t + A_t \leq KL_f D^2, \text{ where} \end{cases}$$

$$A_t = \sum_{k=1}^{K-1} \underbrace{G_{I_{t+k-1} \cap (I_{t+k} \cup \dots \cup I_{t+K-1})}}_{J_k}(\mathbf{x}_{t+k}) \geq \sum_{k=1}^{K-1} f(\mathbf{x}_{t+k}) - \min_{\substack{\mathbf{x} \in \times_{i \in I} C_i \\ \mathbf{x} \wedge J_k = \mathbf{x}_{t+k}^{\wedge J_k}}} f(\mathbf{x}) \geq 0.$$

We don't know how to leverage  $A_t$ s for an improved rate!

## Progress lemma

Lemma (Progress bound via smoothness and convexity, adaptive step size strategy)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , let  $0 < \eta \leq 1 < \tau$  and  $M_0 > 0$ , and assume  $(\star)$ . Let  $\mathbf{x}^*$  solve (1), and set  $H_t = f(\mathbf{x}_t) - f(\mathbf{x}^*)$ . Then

$$H_t - H_{t+K} \geq \begin{cases} H_t + A_t - \frac{K \max\{\eta^t M_0, \tau L_f\} D^2}{2}, & \text{if } H_t + A_t \geq K \max\{\eta^t M_0, \tau L_f\} D^2; \\ \frac{(H_t + A_t)^2}{2K \max\{\eta^t M_0, \tau L_f\} D^2}, & \text{if } H_t + A_t \leq K \max\{\eta^t M_0, \tau L_f\} D^2, \end{cases}$$

$$A_t = \sum_{k=1}^{K-1} G_{\underbrace{I_{t+k-1} \cap (I_{t+k} \cup \dots \cup I_{t+K-1})}_{J_k}}(\mathbf{x}_{t+k}) \geq \sum_{k=1}^{K-1} f(\mathbf{x}_{t+k}) - \min_{\substack{\mathbf{x} \in \times_{i \in I} C_i \\ \mathbf{x} \wedge J_k = \mathbf{x}_{t+k}^{\wedge J_k}}} f(\mathbf{x}) \geq 0.$$

$A_t$  describes partial F-W gaps for **all re-activated components**.



## Convex setting: flexible stepsizes

### Theorem

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $\tau > 1 \geq \eta$  and  $M_0 > 0$  be approximation parameters, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , let  $\mathbf{x}_0 \in \mathbb{R}^N$ , let  $\mathbf{x}^*$  solve (1), and assume  $(\star)$ . Set  $n_0 := \max\{\lceil \log(\tau L_f / (\eta M_0)) / (K \log \eta) \rceil, 0\}$ . Then,

$$f(\mathbf{x}_{nK}) - f(\mathbf{x}^*) \leq \begin{cases} \min_{0 \leq p \leq n-1} \left\{ \frac{K\eta^{pK} M_0 D^2}{2} - A_{pK} \right\} & \text{if } 1 \leq n \leq n_0 + 1 \\ \frac{2K\tau L_f D^2}{n - n_0 + \sum_{p=n_0}^n \frac{2A_{pK}}{f(\mathbf{x}_{n_0}) - f(\mathbf{x}^*)} + \left( \frac{A_{pK}}{f(\mathbf{x}_{n_0}) - f(\mathbf{x}^*)} \right)^2} & \text{if } n > n_0 + 1. \end{cases}$$

After  $t$  iterations, Adaptive-BCFW has evaluated  $f$  and  $\nabla f$  at-most  $2 + \lceil \log_\tau(L_f / \eta^t M_0) \rceil$  times.

## Convex setting: flexible stepsizes

### Theorem

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $\tau > 1 \geq \eta$  and  $M_0 > 0$  be approximation parameters, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , let  $\mathbf{x}_0 \in \mathbb{R}^N$ , let  $\mathbf{x}^*$  solve (1), and assume  $(\star)$ . Set  $n_0 := \max\{\lceil \log(\tau L_f / (\eta M_0)) / (K \log \eta) \rceil, 0\}$ . Then,

$$f(\mathbf{x}_{nK}) - f(\mathbf{x}^*) \leq \begin{cases} \min_{0 \leq p \leq n-1} \left\{ \frac{K\eta^{pK} M_0 D^2}{2} - A_{pK} \right\} & \text{if } 1 \leq n \leq n_0 + 1 \\ \frac{2K\tau L_f D^2}{n - n_0 + \sum_{p=n_0}^n \frac{2A_{pK}}{f(\mathbf{x}_{n_0}) - f(\mathbf{x}^*)} + \left( \frac{A_{pK}}{f(\mathbf{x}_{n_0}) - f(\mathbf{x}^*)} \right)^2} & \text{if } n > n_0 + 1. \end{cases}$$

After  $t$  iterations, Adaptive-BCFW has evaluated  $f$  and  $\nabla f$  at-most  $2 + \lceil \log_\tau(L_f / \eta^t M_0) \rceil$  times.

→ After  $t$  iterations, matches  $\mathcal{O}(K/t)$  rate for convex cyclic setting

## Corollary: Parallelized short-step BCFW

### Corollary

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , let  $\mathbf{x}^*$  solve (1), and assume  $(\star)$ . Then,

$$(\forall n \in \mathbb{N}) \quad f(\mathbf{x}_{nK}) - f(\mathbf{x}^*) \leq \begin{cases} \frac{KL_f D^2}{2} - A_0 & \text{if } n = 1 \\ \frac{2KL_f D^2}{n - 1 + \sum_{p=1}^n \frac{2A_{pK}}{f(\mathbf{x}_1) - f(\mathbf{x}^*)} + \left( \frac{A_{pK}}{f(\mathbf{x}_1) - f(\mathbf{x}^*)} \right)^2} & \text{if } n \geq 2. \end{cases}$$

Furthermore, Short-step BCFW requires one gradient evaluation per iteration.

## Corollary: Parallelized short-step BCFW

### Corollary

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets, let  $f$  be convex and  $L_f$ -smooth, let  $D$  be the diameter of  $\times_{i \in I} C_i$ , let  $\mathbf{x}^*$  solve (1), and assume  $(\star)$ . Then,

$$(\forall n \in \mathbb{N}) \quad f(\mathbf{x}_{nK}) - f(\mathbf{x}^*) \leq \begin{cases} \frac{KL_f D^2}{2} - A_0 & \text{if } n = 1 \\ \frac{2KL_f D^2}{n - 1 + \sum_{p=1}^n \frac{2A_{pK}}{f(\mathbf{x}_1) - f(\mathbf{x}^*)} + \left( \frac{A_{pK}}{f(\mathbf{x}_1) - f(\mathbf{x}^*)} \right)^2} & \text{if } n \geq 2. \end{cases}$$

Furthermore, Short-step BCFW requires one gradient evaluation per iteration.

→ Matches rate **and** constant for non-block Short-step FW.

→ Easier to parallelize than Adaptive BCFW.

# Nonconvex convergence

## Theorem (Nonconvex convergence)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets with diameter  $D$ . Let  $\nabla f$  be  $L_f$ -Lipschitz continuous on  $\times_{i \in I} C_i$ , set  $H_0 = f(\mathbf{x}_0) - \inf f(\times_{i \in I} C_i)$ . Suppose that  $(\star)$  holds. Then, for every  $n \in \mathbb{N}$ , Short-step BCFW guarantees

$$\min_{0 \leq p \leq n-1} G_I(\mathbf{x}_{pK}) \leq \frac{1}{n} \sum_{p=0}^{n-1} G_I(\mathbf{x}_{pK}) \leq \begin{cases} \frac{2H_0 - \sum_{p=0}^{n-1} A_{pK}}{n} + \frac{KL_f D^2}{2} & \text{if } n \leq \frac{2H_0}{KL_f D^2} \\ 2D \sqrt{\frac{H_0 KL_f}{n}} - \frac{\sum_{p=0}^{n-1} A_{pK}}{n} & \text{otherwise.} \end{cases}$$

In particular, there exists a subsequence  $(n_k)_{k \in \mathbb{N}}$  such that  $G_I(\mathbf{x}_{n_k K}) \rightarrow 0$ , and every accumulation point of  $(\mathbf{x}_{n_k K})_{k \in \mathbb{N}}$  is a stationary point of (1).

→ Reactivated **gap** terms reappear!

## Nonconvex convergence

### Theorem (Nonconvex convergence)

Let  $\times_{i \in I} C_i \subset \mathcal{H}$  be a product of  $m$  nonempty compact convex sets with diameter  $D$ . Let  $\nabla f$  be  $L_f$ -Lipschitz continuous on  $\times_{i \in I} C_i$ , set  $H_0 = f(\mathbf{x}_0) - \inf f(\times_{i \in I} C_i)$ . Suppose that  $(\star)$  holds. Then, for every  $n \in \mathbb{N}$ , Short-step BCFW guarantees

$$\min_{0 \leq p \leq n-1} G_I(\mathbf{x}_{pK}) \leq \frac{1}{n} \sum_{p=0}^{n-1} G_I(\mathbf{x}_{pK}) \leq \begin{cases} \frac{2H_0 - \sum_{p=0}^{n-1} A_{pK}}{n} + \frac{KL_f D^2}{2} & \text{if } n \leq \frac{2H_0}{KL_f D^2} \\ 2D \sqrt{\frac{H_0 KL_f}{n}} - \frac{\sum_{p=0}^{n-1} A_{pK}}{n} & \text{otherwise.} \end{cases}$$

In particular, there exists a subsequence  $(n_k)_{k \in \mathbb{N}}$  such that  $G_I(\mathbf{x}_{n_k K}) \rightarrow 0$ , and every accumulation point of  $(\mathbf{x}_{n_k K})_{k \in \mathbb{N}}$  is a stationary point of (1).

→ Reactivated **gap** terms reappear!

→ After  $t$  iterations, minimal F-W gap converges like  $\mathcal{O}(K/\sqrt{t})$ .

# Flexible Block-Coordinate Frank-Wolfe Algorithm

1. Motivation
2. Our approach
3. Analysis
- 4. Numerical experiments**

# Experiments

Toy intersection problem (convex)

Find a matrix in the intersection of the spectrahedron  $C_1 = \{X \in \mathbb{S}_+^{r \times r} \mid \text{Trace}(X) = 1\}$  and the hypercube  $C_2 = [-5, \mu]^{r \times r}$  ( $\mu = 1/r$ ).

$$\underset{\mathbf{x} \in C_1 \times C_2}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{x}^1 - \mathbf{x}^2\|^2$$



# Experiments

Toy intersection problem (convex)

Find a matrix in the intersection of the spectrahedron  $C_1 = \{X \in \mathbb{S}_+^{r \times r} \mid \text{Trace}(X) = 1\}$  and the hypercube  $C_2 = [-5, \mu]^{r \times r}$  ( $\mu = 1/r$ ).

$$\underset{\mathbf{x} \in C_1 \times C_2}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{x}^1 - \mathbf{x}^2\|^2$$

→  $\text{LMO}_{C_1}$  is far more expensive than  $\text{LMO}_{C_2}$ .

→ We use Short-step BCFW to compare the following **block activations**: full, cyclic, permuted-cyclic, and “ $q$ -lazy”:

$$(\forall t \in \mathbb{N}) \quad l_t = \begin{cases} \{1, 2\} & \text{if } t \equiv 0 \pmod{q}; \\ \{2\} & \text{otherwise.} \end{cases} \quad (q\text{-Lazy})$$

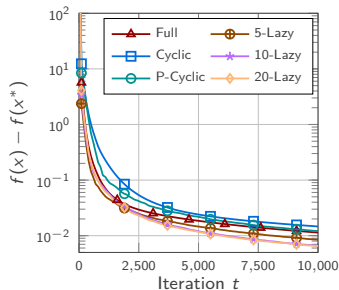
# Experiments

Toy intersection problem (convex)

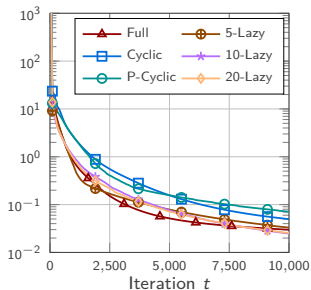
comparing **block-activations**: full, cyclic, permuted-cyclic, and

$$\underset{\mathbf{x} \in C_1 \times C_2}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{x}^1 - \mathbf{x}^2\|^2$$

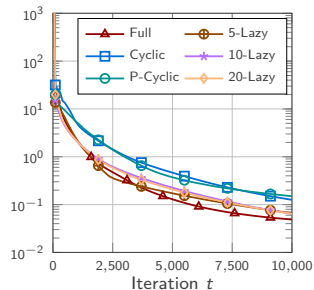
$$(\forall t \in \mathbb{N}) \quad l_t = \begin{cases} \{1, 2\} & \text{if } t \equiv 0 \pmod{q}; \\ \{1\} & \text{otherwise.} \end{cases} \quad (q\text{-lazy})$$



(a)  $r = 100$



(b)  $r = 300$



(c)  $r = 500$

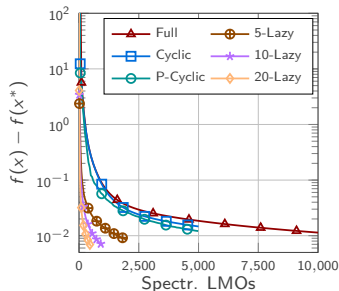
# Experiments

Toy intersection problem (convex)

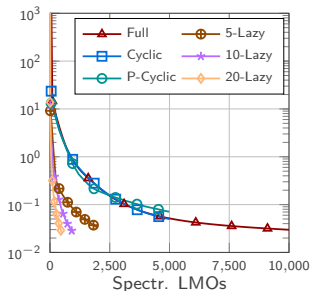
comparing **block-activations**: full, cyclic, permuted-cyclic, and

$$\underset{\mathbf{x} \in C_1 \times C_2}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{x}^1 - \mathbf{x}^2\|^2$$

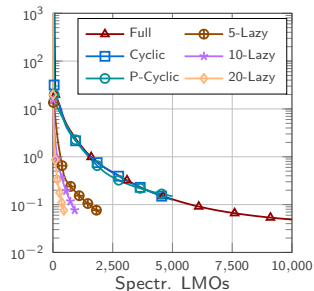
$$(\forall t \in \mathbb{N}) \quad I_t = \begin{cases} \{1, 2\} & \text{if } t \equiv 0 \pmod{q}; \\ \{1\} & \text{otherwise.} \end{cases} \quad (q\text{-lazy})$$



(d)  $r = 100$



(e)  $r = 300$



(f)  $r = 500$

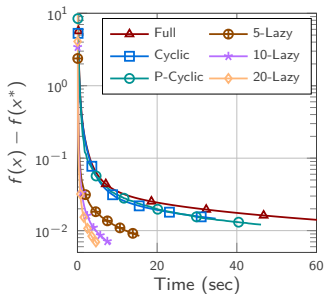
# Experiments

Toy intersection problem (convex)

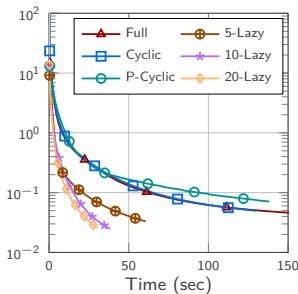
comparing **block-activations**: full, cyclic, permuted-cyclic, and

$$\underset{\mathbf{x} \in C_1 \times C_2}{\text{minimize}} \quad \frac{1}{2} \|\mathbf{x}^1 - \mathbf{x}^2\|^2$$

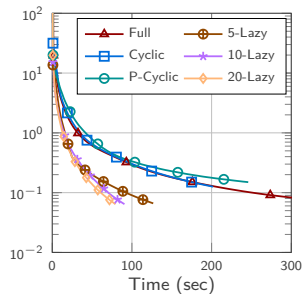
$$(\forall t \in \mathbb{N}) \quad I_t = \begin{cases} \{1, 2\} & \text{if } t \equiv 0 \pmod{q}; \\ \{1\} & \text{otherwise.} \end{cases} \quad (q\text{-lazy})$$



(g)  $r = 100$



(h)  $r = 300$



(i)  $r = 500$

# Experiments

## Toy Difference-of-Convex quadratic problem

Find a  $2r \times r$  matrix such that its first  $r \times r$  submatrix satisfies  $\|X\|_\infty \leq 1$ , and its second submatrix satisfies  $\|X\|_{\text{nuc}} \leq 1$ . To investigate BCFW when the number of components is large, we set  $C_1 = \dots = C_r = \{x \in \mathbb{R}^r \mid \|x\|_\infty \leq 1\}$  and  $C_{r+1} = \{X \in \mathbb{R}^{r \times r} \mid \|X\|_{\text{nuc}} \leq 1\}$ . For PSD  $2r \times r$  matrices  $A$  and  $B$ , we seek to solve

$$\underset{x \in \bigtimes_{1 \leq i \leq r+1} C_i}{\text{minimize}} \quad \langle [x] \mid [x]A \rangle - \langle [x] \mid [x]B \rangle$$

→ For each instance, we verify  $A - B$  is indefinite.

→ Problem is nonseparable

# Experiments

Toy Difference-of-Convex quadratic problem

→  $\text{LMO}_{C_{r+1}}$  is far more expensive than  $(\text{LMO}_{C_i})_{1 \leq i \leq r}$ .

→ We use Short-step BCFW to compare the following **block activations**: full, cyclic, permuted-cyclic, and “ $(p, q)$ -lazy”:

$$(\forall t \in \mathbb{N}) \quad I_t = \begin{cases} I & \text{if } t \equiv 0 \pmod{q} \\ \{i_1, \dots, i_p\} \subset_R I \setminus \{r+1\} & \text{otherwise.} \end{cases} \quad ((p, q)\text{-Lazy})$$

Full update every  $q$  iterations; otherwise, update a random subset of  $p$  “cheap” coordinates in parallel.

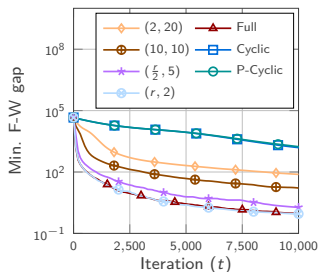
# Experiments

Toy Difference-of-Convex quadratic problem

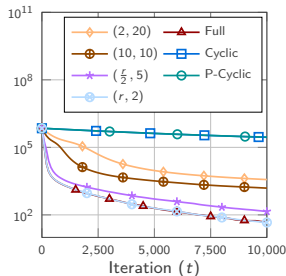
comparing full, cyclic, perm.-cyclic, and “ $(p, q)$ -lazy”:

$$\underset{\mathbf{x} \in \bigcap_{1 \leq i \leq r+1} C_i}{\text{minimize}} \quad \langle [\mathbf{x}] \mid [\mathbf{x}]A \rangle - \langle [\mathbf{x}] \mid [\mathbf{x}]B \rangle$$

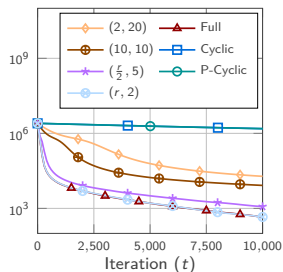
$$I_t = \begin{cases} I & \text{if } t \equiv 0 \pmod{q} \\ \{i_1, \dots, i_p\} \subset_R I \setminus \{r+1\} & \text{otherwise.} \end{cases}$$



(j)  $r = 100$



(k)  $r = 300$



(l)  $r = 500$

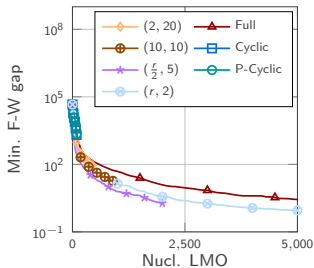
# Experiments

Toy Difference-of-Convex quadratic problem

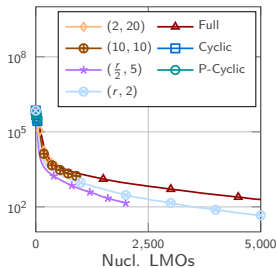
comparing full, cyclic, perm.-cyclic, and “ $(p, q)$ -lazy”:

$$\underset{\mathbf{x} \in \bigtimes_{1 \leq i \leq r+1} C_i}{\text{minimize}} \quad \langle [x] \mid [x]A \rangle - \langle [x] \mid [x]B \rangle$$

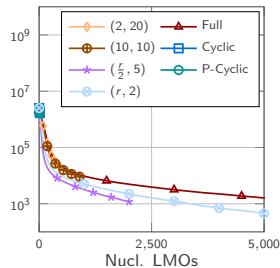
$$I_t = \begin{cases} I & \text{if } t \equiv 0 \pmod{q} \\ \{i_1, \dots, i_p\} \subset_R I \setminus \{r+1\} & \text{otherwise.} \end{cases}$$



(m)  $r = 100$



(n)  $r = 300$



(o)  $r = 500$



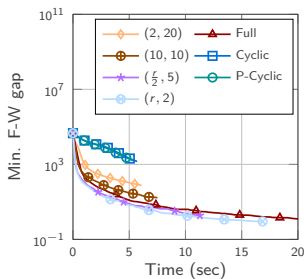
# Experiments

Toy Difference-of-Convex quadratic problem

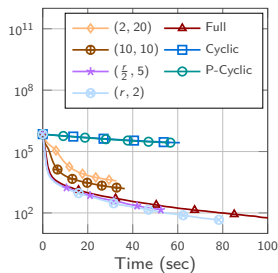
comparing full, cyclic, perm.-cyclic, and “ $(p, q)$ -lazy”:

$$\underset{\mathbf{x} \in \bigtimes_{1 \leq i \leq r+1} C_i}{\text{minimize}} \quad \langle [x] \mid [x]A \rangle - \langle [x] \mid [x]B \rangle$$

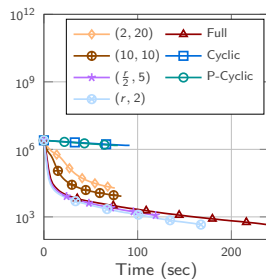
$$I_t = \begin{cases} I & \text{if } t \equiv 0 \pmod{q} \\ \{i_1, \dots, i_p\} \subset_R I \setminus \{r+1\} & \text{otherwise.} \end{cases}$$



(p)  $r = 100$



(q)  $r = 300$



(r)  $r = 500$

## Conclusion

Draft can be found here:









<https://zevwoodstock.github.io/media/publications/block.pdf>






Contact: woodstock@zib.de or woodstzcc@jmu.edu

**Thank you for your attention!**






## References

-  A. Beck, E. Pauwels, and S. Sabach, The cyclic block conditional gradient method for convex optimization problems  
*SIAM J. Optim.*, vol. 25, no. 4, pp. 2024–2049, 2015
-  C. Combettes and S. Pokutta, Complexity of linear minimization and projection on some sets  
*Oper. Res. Lett.*, vol. 49, no. 4, pp. 565–571, 2021
-  P. L. Combettes and ZW, Signal recovery from inconsistent nonlinear observations  
*Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, pp 5872—5876, 2022.
-  P. L. Combettes and ZW, A variational inequality model for the construction of signals from inconsistent nonlinear equations  
*SIAM J. Imaging Sci.*, vol. 15, no. 1, pp. 84–109, 2022
-  M. Frank and P. Wolfe, An algorithm for quadratic programming  
*Naval Res. Logist. Quart.*, vol. 3, iss. 1–2, pp. 95–110, 1956
-  E. Hazan and H. Luo, Variance-Reduced and Projection-Free Stochastic Optimization  
*Proc. ICML*, vol. 48, pp. 1263–1271, 2016

# References

-  C. T. Kelley, Iterative Methods for Linear and Nonlinear Equations  
SIAM, Philadelphia, 1995.
-  S. Lacoste-Julien, M. Jaggi, M. Schmidt, P. Pletscher, Block-Coordinate Frank-Wolfe  
Optimization for Structural SVMs  
*Proc. ICML*, vol. 28, pp. 53–61, 2013
-  A. Osokin, J.-B. Alayrac, I. Lukasewitz, P. Dokania, S. Lacoste-Julien, Minding the Gaps for  
Block Frank-Wolfe Optimization of Structured SVMs  
*Proc. ICML*, vol. 48, pp. 593–602, 2016
-  N. Ottavay, Strong convergence of projection-like methods in Hilbert spaces  
*J. Optim. Theory Appl.*, vol. 56, pp. 433–461, 1988
-  M. Patriksson, Decomposition methods for differentiable optimization problems over Cartesian  
product sets  
*Comput. Optim. Appl.*, vol. 9, pp. 5–42, 1998

# References

-  F. Pedregosa, G. Negiar, A. Askari, and M. Jaggi, Linearly convergent Frank-Wolfe with backtracking line-search  
*ICML*, pp. 1–10, 2020
-  S. Pokutta, The Frank-Wolfe Algorithm: a Short Introduction  
*Jahresber. Dtsch. Math.-Ver.*, vol. 126, pp. 3—35, 2024
-  V. E. Shamanskii, A modification of Newton's method  
*Ukrain. Mat. Zh.*, vol. 19, pp. 133–138, 1967 (in Russian)
-  Y.-X. Wang, V. Sadhanala, W. Dai, W. Neiswanger, S. Sra, E. Xing, Parallel and Distributed Block-Coordinate Frank-Wolfe Algorithms  
*Proc. ICML*, vol. 48, pp. 1548–1557, 2016
-  ZW and S. Pokutta, Splitting the conditional gradient algorithm  
*arXiv:2311.05381*, 2024