A Multistep Frank-Wolfe Method

Zhaoyue Chen, Yifan Sun
OptiML Lab, Stony Brook University

Stony Brook University
Computer Science

Summary

- Frank-Wolfe algorithm is slow because of the zig-zagging.
- Continuous Time Frank-Wolfe does not zig-zag.
- We try to imitate the Continuous Time Frank-Wolfe.

Frank-Wolfe (FW) Algorithm [1,2]

Constrained optimization problem:
f is differentiable, convex.
D is a convex compact constraint set.

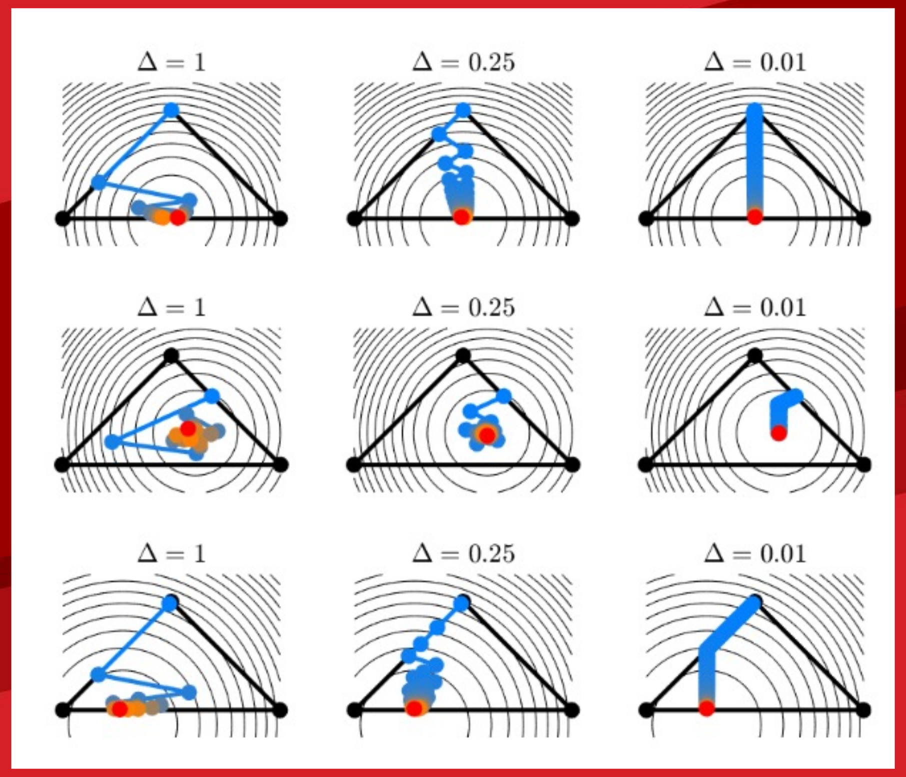
$$\min_{x\in\mathcal{D}} f(x)$$

Let $x^{(0)} \in \mathcal{D}$
for $k = 0 \ldots K$ **do**
  $s^{(k)} = \operatorname{argmin}_{s\in\mathcal{D}} \nabla f(x^{(k)})^T s$,   ▷ *linear minimization oracle (LMO)*
  $x^{(k+1)} = \gamma^{(k)} s^{(k)} + (1-\gamma^{(k)}) x^{(k)}$.
end for

- $\gamma(k) = \frac{c}{c+k}$.
- If $\gamma(k) = O\left(\frac{1}{k^p}\right)$, with $p > 1$, a sequence becomes summable.

Motivation

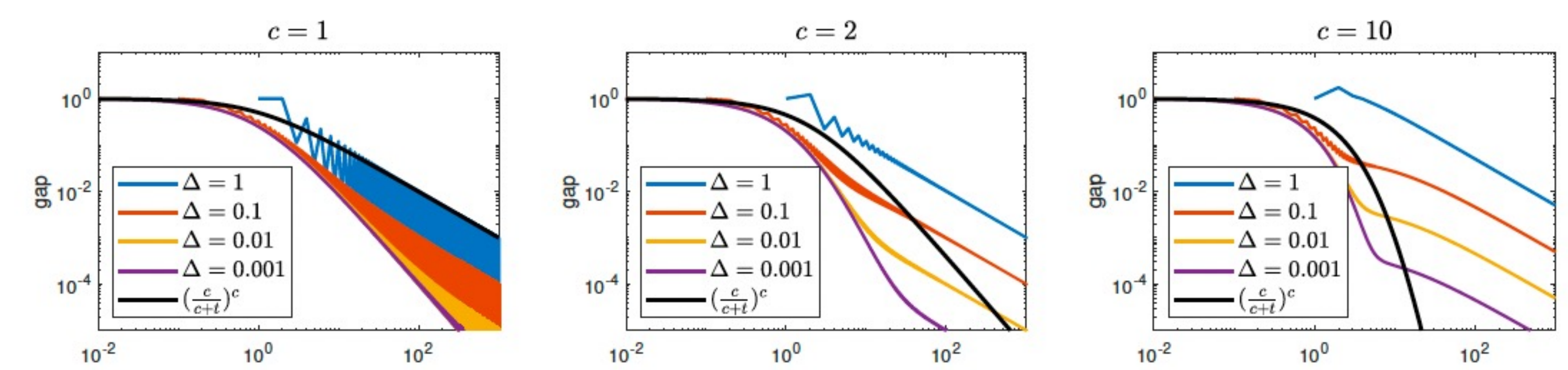
- Main drawback is the slow convergence rate [3].
- "Vanilla" FW method can only reach O(1/k) convergence
- FW Trajectory tends to zigzag.
- Zigzagging makes acceleration challenging.
- Continuous Time FW does not zigzag.



$$\min_{\mathbf{x}\in\mathcal{D}} \tfrac{1}{2}\|\mathbf{x}-\mathbf{x}^*\|_2^2, \quad \mathcal{D} := \mathbf{co}\{(-1,0),(1,0),(0,1)\}$$

Continuous Time Frank-Wolfe (FW) [4]

$$\dot{x}(t) = \gamma(t)(s(t) - x(t)),$$
$$s(t) \in \operatorname{argmin}_{s\in\mathcal{D}} \nabla f(x(t))^T (s - x(t))$$



Continuous vs discrete. A comparison of the numerical error vs compared with derived rate.

Proposition: Suppose $\gamma(t) = \frac{c}{c+t}$, and for some constant $c \geq 0$, the Continuous Time FW has an upper bound of

$$\frac{f(x(t)) - f^*}{f(x(0)) - f^*} \leq \left(\frac{c}{c+t}\right)^c = O\left(\frac{1}{t^c}\right)$$

Runge-Kutta Multistep Methods

- A Generalized Class of Higher Order Methods to Discretize Continuous Time FW:

$$\xi_i = \dot{x}\Big(k + \omega_i,\ \mathbf{x}^{(k)} + \sum_{j=1}^{q} A_{ij}\xi_j\Big),$$
$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \sum_{i=1}^{q} \beta_i \xi_i.$$

Proposition (Positive): All Runge-Kutta methods converge at worst with rate

$$f(x^k) - f^* \leq O\left(\frac{1}{k}\right)$$

Proposition (Negative): The worst best case bound for FW-RK, for any RK method, is of order $O\left(\frac{1}{k}\right)$.

Better Search Direction

- More Aggressive Line Search:

$$\gamma^{(k)} = \max\{\tfrac{2}{2+k}, \bar{\gamma}\},$$
$$\bar{\gamma} = \max_{0\leq\gamma\leq 1}\{\gamma : f(\mathbf{x}^{(k)} + \gamma^{(k)} d^{(k)}) \leq f(\mathbf{x}^{(k)})\}$$
$$\mathbf{x}^{(k+1)} = \gamma^{(k)}\mathbf{s}^{(k)} + (1-\gamma^{(k)})\mathbf{x}^{(k)}$$

Better Use of Momentum [5]:

$$\mathbf{y}^{(k)} = (1-\gamma_k)x^{(k)} + \gamma_k \mathbf{v}^{(k)},$$
$$\mathbf{z}^{(k+1)} = (1-\gamma_k)\mathbf{z}^{(k)} + \gamma_k \nabla f(\mathbf{y}^{(k)}),$$
$$\mathbf{v}^{(k+1)} = \mathrm{LMO}_{\mathcal{D}}(\mathbf{z}^{(k+1)}),$$
$$x^{(k+1)} = (1-\gamma_k)x^{(k)} + \gamma_k \mathbf{v}^{(k+1)}$$

Experiments

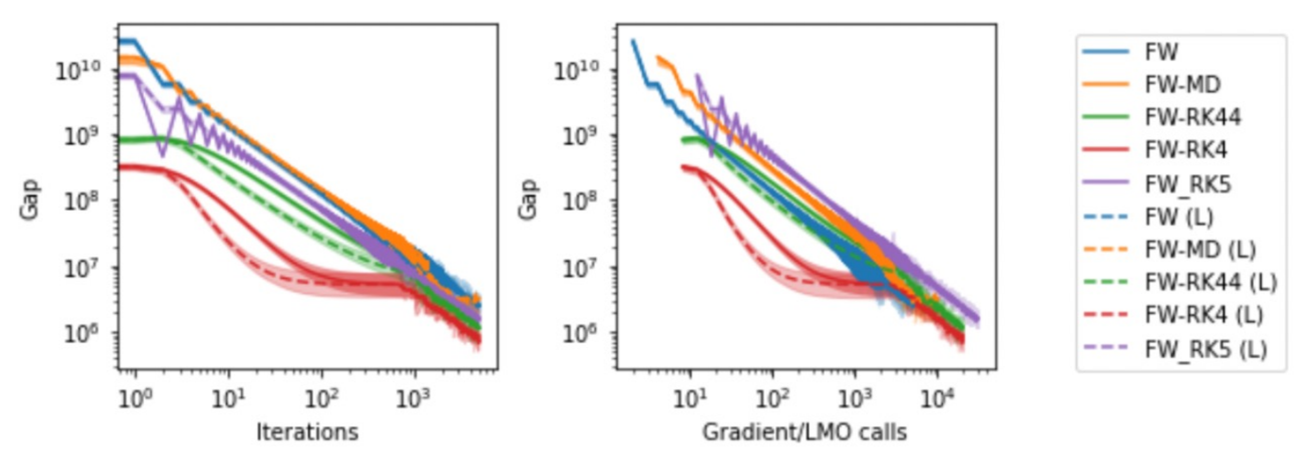


Figure 5: **Compressed sensing.** 500 samples, 100 features, 10% sparsity ground truth, $\alpha = 1000$. L = line search. Performed over 10 trials.

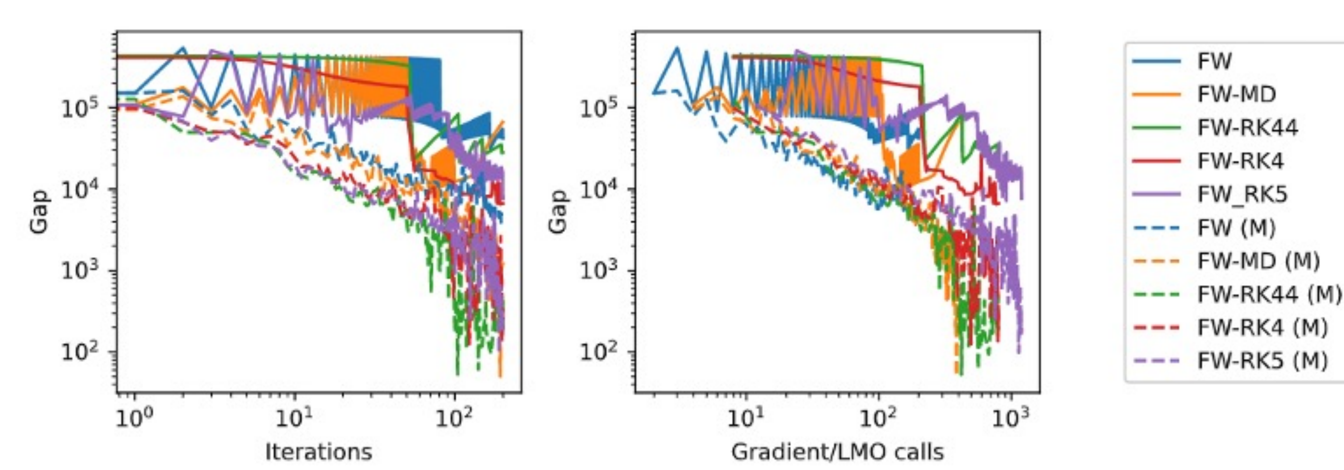


Figure 6: **Sparse logistic regression.** $m = 2000$ samples, $n = 5000$ features. $\alpha = 250$. M = momentum.

References

[1] Marguerite Frank, Philip Wolfe, et al. An algorithm for quadratic programming. Naval research logistics quarterly, 3(1-2):95–110, 1956.
[2] E.S. Levitin and B.T. Polyak. Constrained minimization methods. USSR Computational Mathematics and Mathematical Physics, 6(5):1–50, 1966.
[3] Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In International Conference on Machine Learning, pages 427–435. PMLR, 2013.
[4] Milojica Jacimovic and Andjelija Geary. A continuous conditional gradient method. Yugoslav journal of operations research, 9(2):169–182, 1999.
[5] Bingcong Li, Mario Couti.o, Georgios B. Giannakis, and Geert Leus. A momentum-guided frank-wolfe algorithm. IEEE Transactions on Signal Processing, 69:3597–3611, 2021. doi: 10.1109/TSP.2021.3087910.