Conditional Gradient Methods: Frank-Wolfe-type Algorithms

Naomi Graham*, Emma Hansen**

I. Introduction

In this report we primarily look at the work of Martin Jaggi [1] in defining, and obtaining convergence results for, the general Frank-Wolfe algorithm. This algorithm, which was first introduced by Frank and Wolfe in 1956, has several variants. By viewing this algorithm under a more general framework, Jaggi obtains convergence results which apply to the broader class of Frank-Wolfe type algorithms, unifying existing convergence results as well as demonstrating the broad applicability of Frank-Wolfe. Jaggi notes that Frank-Wolfe-type methods are of particular interest in their behaviour of encouraging sparsity of solutions by maintaining iterates as a convex combination of few atoms. We investigate this aspect of the algorithm by specifically looking at the problem:

$$\min f(x) \text{ subject to } x \in D, \tag{1}$$

where $f(x) = ||Ax - b||_2^2$. In other words, a least squares problem with domain $D = \left\{ x \in \mathbb{R}^{n^2} |\ ||x||_1 \leq \tau \right\}$.

A. Mathematical Background

Before diving in to the inner workings of the algorithm we first present some mathematical. The heart of this approach lies in the representation of the domain as an atomic set, in particular as the closed convex hull of a set of atoms. Recall that the dual norm of the 1-norm:

$$||x||_1 = \sum_i |x_i| \tag{2}$$

is the infinity norm:

$$||x||_\infty = \max_i |x_i|. \tag{3}$$

As well, recall that the gauge and support functions over a set $\mathcal{A}$ are defined as:

$$\gamma_{\mathcal{A}}(x) = \inf \left\{ \lambda \geq 0 |\ x \in \lambda \mathcal{A} \right\}, \tag{4}$$

$$\sigma_{\mathcal{A}}(x) = \sup \left\{ \langle c, d \rangle | d \in \textbf{conv}(\mathcal{A}) \right\}, \tag{5}$$

respectively.

Specific to our application it's important to give background on image blurring. Given an image $\mathcal{I}$ (represented by an array), and a blurring kernel (we used Gaussian), the blurred image is given by:

$$\mathcal{I}_B = R\mathcal{I}[:] \tag{6}$$

where R is determined from the circulant Toeplitz matrix of the vectorized kernel (with rows padded with zeros to match the length of $\mathcal{I}[:]$ - see [2]). The wavelet representation of the image $\mathcal{I}$ is left out as it can quickly become an entire report on its own.

* PhD Student - University of British Columbia, Dept. Computer Science
** PhD Student - University of British Columbia, Dept. Mathematics

II. Frank-Wolfe-type Algorithms

A. History and Variants

In 1956 Marguerite Frank and Philip Wolfe published their work on a new iterative first order method for quadratic programming over polyhedral domains [3], [1]. The name "conditional gradient" was coined by Levitin and Polyak in 1966, to describe the condition that the gradient at the minimizer defines a support functional to the domain at the minimizer [4]. In 1978 a variant using only approximate linear minimizers in the subproblems was shown, by J.C. Dunn and S. Harshbarger, to have $\frac{1}{k}$ convergence [5], [1]. And more recently in 1993 and 2003, M. Patrikson and T. Zhang, respectively, proposed variants which use nonlinear subproblems [1].

B. Significance of the work by Martin Jaggi

In his work, Jaggi proposes a more general framework for Frank-Wolfe algorithms and provides a convergence analysis for this class of algorithms. And, he provides certificates for approximation quality, guaranteeing that each iterate is within a specified range from the optimal value [1]. The convergence results are applicable to a range of sparse greedy methods, and primal-dual convergence is proven for the variants: approximating linear subproblems, line-search for step-size, and "fully corrective". An important to mention, but not discussed variant was the away steps variant (sounds pretty neat!) which not only adds a helpful atom on each iteration, but removes a hindering atom [1].

A particularly useful property of the Frank-Wolfe algorithm, and the convergence analysis presented by Jaggi, is that it is invariant under affine transformations [1]. This is particularly useful for the chosen application of image deblurring where the images are represented in the wavelet domain for efficiency in calculations.

III. Frank-Wolfe Algorithm and Numerical Experiments

The most basic Frank-Wolfe type algorithm is given in Fig. 1. Which is the version of the algorithm which was implemented for this project. The precise problem statement, as given above, is:

$$\min ||Ax - b||_2^2 \ \text{ s.t. } ||x||_1 \leq \tau, \tag{7}$$

Algorithm 1 Frank-Wolfe (1956)

Let $\boldsymbol{x}^{(0)} \in \mathcal{D}$
for $k = 0 \dots K$ **do**
  Compute $\boldsymbol{s} := \arg\min_{\boldsymbol{s} \in \mathcal{D}} \langle \boldsymbol{s}, \nabla f(\boldsymbol{x}^{(k)}) \rangle$
  Update $\boldsymbol{x}^{(k+1)} := (1-\gamma)\boldsymbol{x}^{(k)} + \gamma \boldsymbol{s}, \quad \text{for } \gamma := \frac{2}{k+2}$
end for

Fig. 1. Basic Frank-Wolfe [1].

Where $A = RW$ represents the blurring of image Wx (x is in the wavelet domain), and b is the original blurry image.

The crux of the implementation comes in computing the `argmin`. Luckily, we have convex optimisation to help with that! First note that $\texttt{argmin}\langle s, \nabla f(x)\rangle = \texttt{argmax}\langle s, -\nabla f(x)\rangle$ the value of which is the support of the domain at $-\nabla f(x)$. And knowing the dual norm of the 1-norm, we can determine that $s = -\tau \texttt{sign}(\nabla_{i^*} f(x)) e_{i^*}$ where i^* is the index of the entry of $\nabla f(x)$ which has largest absolute value [6].

A. Simulations

The image deblurring problem was attempted to deblur the following 128×128px image:

Fig. 2. Blurry female climber.

We direct the reader to the jupyter notebook file A2-final in the GitHub repository (in which you found this report) for the remainder of the analysis of the algorithm implementation and simulations.

REFERENCES

[1] M. Jaggi, "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," vol. 28 of *Proceedings of Machine Learning Research*, (Atlanta, Georgia, USA), pp. 427–435, PMLR, 17–19 Jun 2013.
[2] G. Raposo, "Can you recover a blurred image?," *Medium.com*, Sept. 2020.
[3] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, pp. 95–110, Mar. 1956.
[4] E. Levitin and B. Polyak, "Constrained minimization methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 6, pp. 1–50, Jan. 1966.
[5] J. Dunn and S. Harshbarger, "Conditional gradient algorithms with open loop step size rules," *Journal of Mathematical Analysis and Applications*, vol. 62, pp. 432–444, Feb. 1978.
[6] R. Tibshirani, "Lecture 23: Conditional gradient method," *Carnegie Melon University*, Apr. 2015.