# OPTIMA 95

Mathematical Optimization Society Newsletter


## MOS Chair's Column

September 15, 2014. We are moving into the prize-nomination season of the society's three-year cycle. Over the past several months, working together with the American Mathematical Society, the Chinese Mathematical Programming Society, and SIAM, all of our prize committees have been assembled. The action now shifts to our society members to put together nomination packages for the great work that has been published in recent years.

I want to thank the 25 people who have graciously agreed to serve on our prize committees. And in particular I thank the chairs of the committees: Karen Aardal (Tucker Prize), Mihai Anitescu (Lagrange Prize), Robert Bixby (Beale-Orchard-Hays Prize), Michele Conforti (Fulkerson Prize), Margaret Wright (Dantzig Prize), and Yinyu Ye (Tseng Lectureship). The chairs are ready and waiting for your nomination letters. Details can be found in the calls for nominations published in this issue of Optima.

ISMP 2015 in Pittsburgh is only a year away, and so it is also time to start the planning for ISMP 2018. Jan Karel Lenstra is chairing the Symposium Advisory Committee that will make the recommendation to the MOS Council for the site of the next symposium. By tradition, only locations outside of Canada and the USA will be eligible to host ISMP 2018. I'm hoping for exciting bids!


Bill Cook, University of Waterloo
bico@uwaterloo.ca


## Note from the Editors

Dear members of MOS:
Martin Jaggi (ETH Zürich) and Zaid Harchaoui (INRIA Grenoble) jointly with Federico Pierucci organized a workshop on "Greedy Algorithms, Frank-Wolfe and Friends – A modern perspective" at the annual conference of the Neural Information Processing Systems Foundation in 2013 (NIPS 2013) at Lake Tahoe, Nevada. On that occasion they had the chance to interview Marguerite Frank about the invention of the Frank–Wolfe algorithm she developed together with Philip Wolfe in the 1950's. We are very happy to have a column in this issue in which she shares with us her memories on this important moment in the history of mathematical optimization dating back some 60 years. How influential her's and Philip Wolfe's invention was can be seen also from the scientific article in this issue, in which Martin Jaggi and Zaid Harchaoui describe recent succesful applications of the method.

We hope that you will enjoy both the historical and the current aspects of this issue – and that you don't stop reading after the scientific part, because this might cause you to miss important information about the upcoming ISMP to be held July 12–19, 2015 in Pittsburgh and the prizes to be awarded there.

Sam Burer, Co-Editor
Volker Kaibel, Editor
Jeff Linderoth, Co-Editor

**Contents of Issue 95 / September 2014**




Martin Jaggi and Zaid Harchaoui


## Preluding Remarks

In 1956, Marguerite Frank and Philip Wolfe published a paper entitled "An Algorithm for Quadratic Programming" [1] in Naval Research Logistics Quarterly. The paper's title was a strong understatement, given that it introduces a new algorithm not only for quadratic convex, but general convex optimization, and proves the $O(1/t)$ convergence rate. The *Frank–Wolfe* algorithm is widely considered the very first method for general constrained convex optimization.

Originating in the famous Logistics Project group at Princeton, which was the most prominent place in linear programming and game theory at the time, the new collaboration between Marguerite Frank and Phil Wolfe (both coming from quite different scientific backgrounds) aimed at the following research question: Given the recent successes and tools of linear programming, is it possible to derive a method for the more general case, that is quadratic convex programming, or even general convex programming? The answer is well known to the reader at this point, and in our opinion marks some of the birth hours of the research field of convex optimization, and a significant and brave departure from the linear programming territory.

Since then, the powerful core idea of the Frank–Wolfe method – to decompose a (hard) general optimization problem into a sequence of (easy) linear problems – has proven fertile numerous times in a surprisingly large variety of applications, and is continuing to have impact in many fields.

We feel extremely honored that Marguerite Frank, co-author of the original paper, and one of the first female PhD students in mathematics at Harvard University, has agreed to share her thoughts on these interesting birth years of mathematical optimization.

We would also like to thank Hallie Wolfe, P. Wolfe's wife, and Isabelle Frank, M. Frank's daughter, and Anna Nagourney for their constant kindness and generous help in preparing this article. We

are indebted to the editors of Optima, Volker Kaibel, Sam Burer, Jeff Linderoth and Katya Scheinberg for many valuable comments, as well as to to Alexandre d'Aspremont, Rob Freund, Anatoli Juditsky, Simon Lacoste-Julien, Claude Lemaréchal and Arkadi Nemirovski for fruitful discussions.

### Reference

[1] Marguerite Frank and Philip Wolfe. An Algorithm for Quadratic Programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.

Marguerite Frank

## The History of the Frank–Wolfe Algorithm

As far as I can remember – from my idealized early past in Paris, then Toronto, I was interested in math (Euclid, algebra), reading and ideas. When majoring in math and physics at the University of Toronto, I was mostly inspired by Harold Scott MacDonald Coxeter, Leopold Infeld and especially Richard Brauer. Graduating in 1947, I went on to Radcliffe-Harvard as a Teaching Fellow at the age of 20. When I left in 1949 with all but a PhD thesis, neither the logic of Willard Van Orman Quine, nor the algebraic geometry of Oscar Zariski worked as substitutes for algebra (I was only 23 years old back then).

During a break of two Parisian years, I got a diploma in History of Mathematics with Alexandre Koyré, and incidentally met my future husband, whom I then followed to the University of Chicago, where I had the great fortune to be accepted as an auditor by the algebraist Abraham Adrian Albert. This led to my defining of new simple Lie algebras, the first in several years, and to more such new classes later [14, 12].

Then, in 1955, I followed my husband, Joseph Frank, to Princeton, where he was invited as a Gauss Lecturer on the topic of "Existentialism and Dostoevsky". During this time I was fortunate to become a Visitor in Princeton's prestigious Logistics Project, run by Albert W. Tucker and Harold W. Kuhn. These were the exciting days of Game Theory (von Neumann & Morgenstern, Nash et al.), and of the reigning technique of Linear Programming and the Simplex method of George Dantzig, which served as strong inspirations. There was talk of computers, but none to be actually seen as I recall. A complete neophyte in these new esoteric disciplines, I immersed myself as best I could, attempting to master Linear Programming, Lagrange Multipliers and their new Kuhn-Tucker optimality conditions for inequality constraints, as well as getting used to the subtleties of competitive games.

I was assigned to work with the then post-doc instructor Philip Wolfe – who was already well-trained in the general field of optimization. Our assigned joint goal was to seek a procedure that might yield the optimum solution for a suitably shaped quadratic function under linear inequality constraints. It all occurred a very long time ago, and what I remember mostly is – working at home – I had no office – covering, as usual, sheets of paper with symbols related – now – by linear inequalities instead of equations. I also attended the famous Fine Hall Afternoon Teas, a daily and informal departmental meeting.

I seem to remember presenting some jottings to Phil in his office-and his exclaiming "I think Marguerite has solved the quadratic problem" – then reproaching me for not having established the "convergence" – and immediately doing so himself... But this evokes more specifically the later general "convex problem" so my memory may be at fault. In any case, Phil rewrote both procedures in their present form [13], and chose the (at the time) original title. He then continued to work brilliantly in the field.

For me this brief foray in optimization made it easier later to obtain – after various research and part time jobs – a tenured position in the nearby Rider University Business School, and a later one as Fellow in Stanford's Engineering School O.R. department, where I finally met George Dantzig.

I never received any special acknowledgment or a single bitcoin for the article – only the usual professional interest from Anna Nagourney, and Stella Dafermos, who befriended me in my new guise. Hence my astonishment at being alerted by Michel Balinski, in the fall of 2013, about the NIPS Workshop,[1] at nearby Lake Tahoe – and then invited there by Martin Jaggi and Zaid Harchaoui. I was stunned to learn, as was Phil Wolfe whom I contacted by phone, that Frank–Wolfe when translated into computer code had become useful in various places, including major companies in the current digital economy.

Thinking about the current times of digitalization in general, I find it ironic that abstract Boolean algebra – that could not possibly have emerged fully formed from a mind-deprived brain, nor from a data cloud – is at the root of what is surely our present technological / cultural paradigm shift.

### Note

1. NIPS'13 Workshop on Greedy algorithms, Frank–Wolfe, and Friends: https://sites.google.com/site/nips13greedyfrankwolfe/

Martin Jaggi and Zaid Harchaoui

## The Recent Revival of the Frank–Wolfe Algorithm

In this article, we attempt to give a brief, non-comprehensive overview of the Frank–Wolfe algorithm, also called the conditional gradient method, its variants and properties, and some recent successful applications of this class of first-order optimization methods. The aim of this review is to complement the preceding fascinating first-hand account of the birth of this algorithm, by Marguerite Frank, co-author of the groundbreaking original paper [13] from 1956 together with Phil Wolfe.

### 1 Introduction

The Frank–Wolfe algorithm [13], also known as the *conditional gradient* method, was historically the earliest algorithm for general constrained convex optimization, that is for problems of the form

$$\min_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})\,. \tag{1}$$

Here we assume that the objective function $f$ is convex and differentiable, and the domain (or feasible set) $\mathcal{D}$ is a bounded convex subset of some arbitrary vector space (that can be infinite-dimensional).

The method is given by the following simple first-order optimization scheme:

**Algorithm 1:** Frank–Wolfe / Conditional Gradient

Let $\boldsymbol{x}^{(0)} \in \mathcal{D}$
**for** $t = 0 \dots T$ **do**
  Compute $\boldsymbol{s} := \mathrm{LMO}_{\mathcal{D}}\left(\nabla f(\boldsymbol{x}^{(t)})\right)$
  Let $\gamma := \frac{2}{t+2}$
  Update $\boldsymbol{x}^{(t+1)} := (1-\gamma)\boldsymbol{x}^{(t)} + \gamma \boldsymbol{s}$
**end**

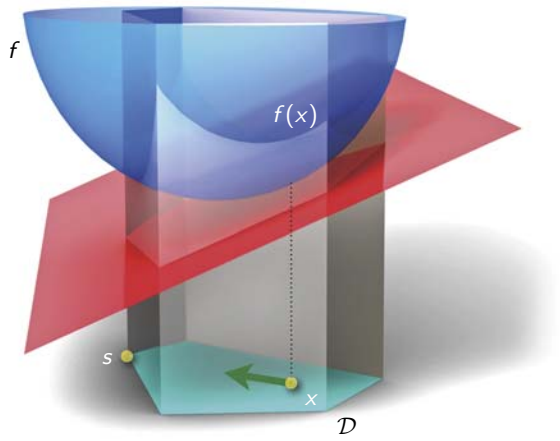



Figure 1. *Linearization (middle) of a convex function $f : \mathbb{R}^2 \to \mathbb{R}$ (top) at $\boldsymbol{x}$, and the corresponding Frank–Wolfe step. The domain $\mathcal{D} \subset \mathbb{R}^2$ is visualized at the bottom.*

The core element of the algorithm is the $\text{LMO}_{\mathcal{D}}(.)$ procedure, that is the *Linear Minimization Oracle* associated with the domain $\mathcal{D}$. This routine is defined to return a minimizer of a linear subproblem over $\mathcal{D}$, i.e. a point satisfying

$$\text{LMO}_{\mathcal{D}}(\boldsymbol{d}) \;\in\; \arg\min_{\boldsymbol{s}\in\mathcal{D}} \langle \boldsymbol{d}, \boldsymbol{s} \rangle \;.$$

We allow the choice of any minimizer, in case there are several.

Interpreting the Frank–Wolfe (FW) algorithm, we see that the method in each iteration considers the linearization of the objective at the current iterate, that is $f(\boldsymbol{x}^{(t)}) + \langle \boldsymbol{y} - \boldsymbol{x}^{(t)}, \nabla f(\boldsymbol{x}^{(t)}) \rangle$, and employs this as a surrogate of the objective. The update direction $\boldsymbol{s}$ is obtained as the minimizer of the linearization.

The name *conditional gradient* comes from the following interpretation of the main step of the algorithm. In projected gradient (gradient projection) algorithms, first a gradient step is taken *unconditionally* to the constraint set (*i.e.*, that may move to a non-feasible point), which is then projected onto the constraint set by a projection operator. In contrast, the conditional gradient algorithm makes steps that are *conditional* to the constraint set (*i.e.*, that stay feasible, by construction), and does not require any further projection step.

*Properties, and Comparison to Proximal Gradient Methods.* Given that the FW method only uses first-order information, it is important to discuss the relation to classical gradient or proximal gradient methods. The two classes of methods perform fundamentally different updates in each iteration: FW moves towards a feasible point of best inner product with the gradient, which in general is different from the gradient direction itself (except, e.g., in the special case when the domain $\mathcal{D}$ is the Euclidean ball).

We briefly summarize a few more distinctive properties of FW methods, in contrast to classical gradient based methods:

1. *Iteration Cost.* Another fundamental difference is that the FW subproblem ($\text{LMO}_{\mathcal{D}}$) is *linear*, whereas classical proximal (or projected) gradient methods require solving a *quadratic* problem in every iteration. For several classes of domains – some of which we will mention below – the linear oracle can be much easier to compute than a proximal operator or a projection step. For example for the domain being the matrices of bounded nuclear norm, projection onto $\mathcal{D}$ requires computing a full singular value decomposition (SVD), while the linear problem $\text{LMO}_{\mathcal{D}}$ only needs the top singular vector pair, as we explain in more detail in Section 5.
2. *Sparse Iterates.* After $t$ iterations, the iterate $\boldsymbol{x}^{(t)}$ will always be represented as a convex combination of at most $t$ extreme points of the domain. In several applications, this can be very beneficial for efficiently storing the iterate for large scale problems, and also benefits interpretability. The sparse representation is in strong contrast to gradient based methods, where no such compact representation is possible in general.
3. *Infinite dimensional domains.* The FW algorithm allows application to infinite dimensional domains, in particular when those are represented as convex hulls of a set of atoms (as we discuss in Section 2 below), such as e.g. wavelet functions in the time-frequency domain in signal processing and harmonic analysis. The only requirement for application of the method is the availability of an $\text{LMO}_{\mathcal{D}}$ oracle.

*Convergence.* In terms of convergence, it has been known from the early papers that the optimization error attained by the iterates[1] of Algorithm 1 decreases with rate $O(1/t)$, as shown in [13, 6, 10], under the assumptions of convexity, smoothness of $f$, and boundedness of $\mathcal{D}$. Also, this rate is optimal for the class of algorithms moving towards one vertex of $\mathcal{D}$ per iteration [26]. We give a compact proof of the convergence rate in Section 4 below, and also give a modern view on the algorithm properties, highlighting primal-dual convergence and affine invariance (independence of the algorithms and proofs from the geometry or choice of a norm).

Significant parts of the material presented here are taken from the recent overview papers [26] and [21], and ICML 2014 tutorial https://sites.google.com/site/frankwolfegreedytutorial.

## 2 Atomic Domains

A significant part of the renewed interest in Frank–Wolfe type methods is owed to its suitability for optimizing over atomic domains, that is domains of the form $\mathcal{D} = \text{conv}(\mathcal{A})$ for a set of atoms $\mathcal{A}$, chosen as any set of points in a vector space.

Traditionally, the usual way of specifying the optimization domain $\mathcal{D}$ was by imposing (e.g., linear) constraints, such as in linear programming and in the original paper [13]. When instead considering domains represented as the convex hull of a set of atoms, it became more clear in recent years that FW type methods are even more convenient to apply, for the following reason. Writing $\mathcal{D} = \text{conv}(\mathcal{A})$ for a set of atoms $\mathcal{A}$, the solutions of the linear subproblems are always attained by one of the atoms $\boldsymbol{s} \in \mathcal{A}$, meaning that we can assume $\boldsymbol{s} := \text{LMO}_{\mathcal{D}}(\boldsymbol{d}) \in \mathcal{A}$. In consequence, the algorithm in every iteration moves towards one of the atoms, and maintains the iterate $\boldsymbol{x}^{(t)}$ as a (convex) combination of at most $t$ atoms after $t$ iterations.

This simple yet powerful interpretation makes FW an instance of the general type of greedy optimization methods of the form

$$\boldsymbol{x}^{(t+1)} := \alpha \boldsymbol{x}^{(t)} + \beta \boldsymbol{s} \quad \text{for } \boldsymbol{s} \in \mathcal{A} \,.$$

where $\alpha, \beta \in \mathbb{R}$ are some specified weights, and $\boldsymbol{s}$ is the new atom selected in the current iteration. Greedy methods of this type have seen very significant research interest recently e.g. in signal processing, where the most prominent application is *matching pursuit*, corresponding to the case where $\mathcal{D}$ is the unit ball of the $\ell_1$-norm, i.e. that $\mathcal{A}$ is the set of unit basis vectors together with their negatives.

We consider this interpretation of FW methods for the application to convex hulls as one of the most significant reasons for the recent surge of renewed interest in this kind of method. The work of [5, 8, 39, 25, 38, 37, 26] and [9, Appendix D] were among the ones popularizing this algorithmic perspective for atomic sets, while [4] has first considered the atomic norm concept.

## 3 Algorithm Variants, and a Brief Historical Overview

Since the original breakthrough paper [13] (see the accompanying first-hand account of Marguerite Frank), there has been a lot of research on FW related methods in the last 60 years, and unfortunately the available space here does not permit us to give credit to all the important developments.

Still, one should definitely mention the important contributions of Levitin and Polyak [31], Demyanov and Rubinov [6, 7], Pshenichny and Danilin [36], Dunn [10], and Patriksson [34]. The term *conditional gradient method* for the same algorithm was coined by [31], and from then on was used most frequently in the optimization literature.

The analysis was generalized to arbitrary Banach spaces in [6, 7], as in the setting here (note that the algorithm as well as the analysis presented in Section 4 only depend on inner products). In the 70's, [10] investigated approximate linear minimizers of the subproblems, as we describe in Section 3.4 below.

In recent years, Frank–Wolfe / conditional gradient methods have seen both renewed research interest as well as a revival in several application areas. We will outline some of these more recent results as well as active research directions in Section 6 below.

### 3.1 Step-Sizes, and Line Search

Instead of the fixed sequence of step-sizes $2/(t+2)$ in Algorithm 1, a natural alternative is to move to the best possible point on the line-segment between the iterate $\boldsymbol{x}^{(t)}$ and $\boldsymbol{s}$. This is obtained by replacing the choice of $\gamma$ in the algorithm by

$$\gamma := \operatorname*{arg\,min}_{\gamma\in[0,1]} f\Big((1-\gamma)\boldsymbol{x}^{(t)} + \gamma\boldsymbol{s}\Big) \,.$$

The same convergence results as presented here also hold for the line-search variant, by using the fact that the decrease in objective is always at least as good as for the original method. Traditionally, the earlier literature has not used the simpler $2/(t+2)$ step length, but instead employed line-search on the objective $f$, or on a quadratic upper bound on $f$. Such a bound is given for example by a Lipschitz-gradient assumption with constant $L$, for which line-search then gives an explicit step-size $\gamma = g(\boldsymbol{x})/L\,\|\boldsymbol{x}-\boldsymbol{s}\|^2$ as introduced by [6], with $g(\boldsymbol{x}) := \langle \boldsymbol{x}-\boldsymbol{s}, \nabla f(\boldsymbol{x})\rangle$. The original paper [13] used the tighter non-uniform quadratic upper bound given by the Hessian of $f$, which then appears in the denominator instead of $L$.

### 3.2 Accelerated Frank–Wolfe by Simplicial Decomposition

After the addition of a new atom (or search direction) $\boldsymbol{s}$, the classical FW method updates the iterate as $\boldsymbol{x}^{(t+1)} := (1-\gamma)\boldsymbol{x}^{(t)} + \gamma\boldsymbol{s}$. An improved and slightly harder-working variant of the FW method replaces this step by re-optimizing the objective $f$ over the previously used atoms [24, 33], i.e. using the update

$$\boldsymbol{x}^{(t+1)} := \operatorname*{arg\,min}_{\boldsymbol{x}\in\operatorname{conv}(\boldsymbol{s}^{(0)},\ldots,\boldsymbol{s}^{(t+1)})} f(\boldsymbol{x}) \,.$$

Compared to the original FW algorithm, the idea is that the variant here will hopefully make more progress per iteration, and therefore result in iterates $\boldsymbol{x}$ being combinations of even fewer atoms (i.e. better sparsity). This however comes at a price, namely that the internal problem in each iteration now becomes harder to solve. The presented update variant is often called the "simplicial decomposition method" [2], or *fully corrective*. Alternatively, algorithm variants which only re-optimize over the convex hull of *some* of the previously used atoms together with the current iterate, are often called *partially corrective* or *memory-based* FW. In the case of an $\ell_1$-domain, fully corrective FW corresponds to orthogonal matching pursuit [32] in signal processing. For the same convergence rates to hold, the re-optimization subproblems do not have to be solved exactly – it is enough that the point returned has an objective value at least as good as the one obtained by the simple FW step $\boldsymbol{x}^{(t+1)} := (1-\gamma)\boldsymbol{x}^{(t)} + \gamma\boldsymbol{s}$ for $\gamma := 2/(t+2)$.

### 3.3 Composite Frank–Wolfe

In several machine learning applications, the optimization problem of interest is a norm-regularized composite minimization problem

$$\min_{\boldsymbol{x}\in K} f(\boldsymbol{x}) + \lambda\|\boldsymbol{x}\|_{\mathcal{A}} \,,$$

where $K$ is a closed convex cone (such as $\mathbb{R}^d$), and $\|.\|_{\mathcal{A}}$ is an atomic norm in an ambient Hilbert space. More formally, $\|\boldsymbol{x}\|_{\mathcal{A}} := \inf_{r\geq 0}\{r \mid \boldsymbol{x}\in r\mathcal{D}\}$ is the gauge function associated with the convex set ("unit-ball") $\mathcal{D} = \operatorname{conv}(\mathcal{A})$. The function $f$ is assumed to satisfy the same properties as in the previous sections. The term $\lambda\|\boldsymbol{x}\|_{\mathcal{A}}$ usually corresponds to a regularization or roughness penalty, parametrized by $\lambda$, that allows a flexible control of the regularity properties of the solution to the problem. Classical FW cannot be applied in general to such a problem, as the norm $\|.\|_{\mathcal{A}}$ might be non-smooth, and $K$ is unbounded.

For such a composite minimization problem with atomic-norm regularization, a *Composite Conditional Gradient* algorithm (or Composite Frank–Wolfe) can be designed, again using a linear minimization oracle and first-order information. The algorithm works on the epigraph form of the composite minimization problem

$$\min_{\boldsymbol{z}} F_\lambda(\boldsymbol{z}) := f(\boldsymbol{x}) + \lambda r \,,$$

using the shorthand variable $\boldsymbol{z} := [\boldsymbol{x}, r]$. In the following, $\bar{R}$ is an a priori guess on the norm of the optimal solution. This algorithm summarized below was introduced in [21], where a $O(1/t)$ convergence rate, independent of the choice of $\bar{R}$, was proved. An accelerated version in the sense of Section 3.2, called *AtomDescent*, was proposed in [8, 9], along with an equivalent coordinate-descent formulation.

**Algorithm 2:** Composite Frank–Wolfe / Conditional Gradient

Let $\boldsymbol{z}^{(0)} = [0, \bar{R}]$, with $\bar{R}$ such that $\|\boldsymbol{x}^\star\|_{\mathcal{A}} \leq \bar{R}$
**for** $t = 0 \ldots T$ **do**
  Compute $\boldsymbol{s} := \bar{R}\cdot[\operatorname{LMO}_{\mathcal{D}}\Big(\nabla f(\boldsymbol{x}^{(t)})\Big), 1]$ Let
  $(\alpha,\beta) := \operatorname{arg\,min}_{\alpha,\beta\geq 0;\ \alpha+\beta\leq 1} \ F_\lambda\Big(\alpha\boldsymbol{s} + \beta\boldsymbol{z}^{(t)}\Big)$
  Update $\boldsymbol{z}^{(t+1)} := \beta\boldsymbol{z}^{(t)} + \alpha\boldsymbol{s}$
**end**

In fact, classical FW can be recovered from the composite FW algorithm when working in a more general setting with a regularization penalty that can be the indicator function of a convex set. Therefore, the composite FW algorithm can be seen as generalizing FW for both composite minimization and constrained minimization problems, only requiring first-order and linear minimization oracles.

### 3.4 Approximate Linear Minimization Oracles

An exact linear minimization oracle – which is needed in every iteration of the FW algorithm – might not be available for all applications. We can replace the exact oracle $\operatorname{LMO}_{\mathcal{D}}(\boldsymbol{d})$ by an approximate linear oracle $\operatorname{LMO}_{\mathcal{D},\varepsilon_t}(\boldsymbol{d})$ which returns $\boldsymbol{s}$ s.t.

$$\langle \boldsymbol{d}, \boldsymbol{s}\rangle \leq \min_{\boldsymbol{s}\in\mathcal{D}} \langle \boldsymbol{d}, \boldsymbol{s}\rangle + \varepsilon_t \,,$$

a criterion which was first proposed by [10]. As it turns out, the same convergence rate will hold for this approximate algorithm variant, if $\varepsilon_t := C_f/(t+2)$, decreasing with the iteration counter $t$. The

only effect will be that the error decrease given by the convergence Theorem 1 below will be multiplied by a factor of two. See the next section for the definition of the constant $C_f$.

Alternatively, it is also possible to assume multiplicative approximation quality for the internal oracle. This is particularly suitable if the domain is symmetric, such as, e.g., a norm ball. Examples of such cases were considered in [6] or [29, Appendix C].

### 3.5 Approximate Gradients

Irrespective of the linear oracle being exact or not, the gradient information used at each iteration might also be noisy or inexact in some applications. If we assume that we only have access to a noisy and randomized estimate $\hat{\boldsymbol{d}}_x$ of the true gradient, and we suppose that this vector $\hat{\boldsymbol{d}}_x$ satisfies

$$\mathrm{E}\Big[\, \big\| \hat{\boldsymbol{d}}_x - \boldsymbol{d}_x \big\|^* \,\Big] \leq \frac{\varepsilon_t}{\mathrm{diam}_{\|.\|}(\mathcal{D})} \,, \tag{2}$$

for some arbitrary choice of norm $\|.\|$, and $\boldsymbol{d}_x := \nabla f(\boldsymbol{x})$. Then we can use $\mathrm{LMO}_{\mathcal{D}}(\hat{\boldsymbol{d}}_x)$ instead of the true $\mathrm{LMO}_{\mathcal{D}}(\boldsymbol{d}_x)$ in the FW algorithm. Here the expectation is over the randomness in $\hat{\boldsymbol{d}}_x$.

Again, analogous to the case of approximate oracles, the FW algorithm in the setting here – using $\hat{\boldsymbol{d}}_x$ instead of the true $\boldsymbol{d}_x$ in each iteration – is known to have the same convergence rate [26] (up to a factor of two), assuming the quality of the estimator is chosen as $\varepsilon_t := C_f/(t+2)$. More recently, [15] used an affine invariant counterpart of (2), also maintaining the convergence rate.

## 4 A Modern Perspective and Convergence Analysis

We present a short analysis of the convergence rate of the FW method, which also extends to those algorithm variants which we have detailed above. Furthermore, the analysis here has the advantage of being affine invariant, i.e., independent of the parameterization of the input problem, in terms of choice of geometry or norm.

**Theorem 1** (Primal Convergence Rate). *For each $t \geq 1$, the iterates $\boldsymbol{x}^{(t)}$ of the Frank–Wolfe algorithm satisfy*

$$f(\boldsymbol{x}^{(t)}) - f(\boldsymbol{x}^\star) \leq \frac{2C_f}{t+2} \,.$$

*Proof.* Let $C_f$ be a constant s.t.

$$f(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \gamma \underbrace{\langle \boldsymbol{s}-\boldsymbol{x}, \nabla f(\boldsymbol{x}) \rangle}_{-g(\boldsymbol{x})} + \tfrac{\gamma^2}{2} C_f$$

for all $\boldsymbol{x}, \boldsymbol{s} \in \mathcal{D}$ and $\boldsymbol{y} := \boldsymbol{x} + \gamma(\boldsymbol{s}-\boldsymbol{x})$ for $\gamma \in [0,1]$.

Writing $h(\boldsymbol{x}^{(t)}) := f(\boldsymbol{x}^{(t)}) - f(\boldsymbol{x}^\star)$ for the objective error, we have

$$\begin{aligned} h(\boldsymbol{x}^{(t+1)}) &\leq h(\boldsymbol{x}^{(t)}) - \gamma g(\boldsymbol{x}^{(t)}) + \frac{\gamma^2}{2} C_f \\ &\leq h(\boldsymbol{x}^{(t)}) - \gamma h(\boldsymbol{x}^{(t)}) + \frac{\gamma^2}{2} C_f \\ &= (1-\gamma) h(\boldsymbol{x}^{(t)}) + \frac{\gamma^2}{2} C_f \,, \end{aligned}$$

where the first inequality is by definition of $C_f$, and the second one $h(\boldsymbol{x}) \leq g(\boldsymbol{x})$ follows from convexity of $f$, using that the graph of $f$ lies above its linearization, see also the discussion below.

From here, the decrease rate follows from a simple induction for the sequence of numbers $h(\boldsymbol{x}^{(t)})$ for $\gamma = 2/(t+2)$, see, e.g., [26]. □

*The Curvature Constant.* The convergence analysis given above relied on the following curvature constant, measuring the non-linearity of the objective function $f$ over the domain $\mathcal{D}$.

$$C_f := \sup_{\substack{\boldsymbol{x},\boldsymbol{s}\in\mathcal{D},\\ \gamma\in[0,1],\\ \boldsymbol{y}=\boldsymbol{x}+\gamma(\boldsymbol{s}-\boldsymbol{x})}} \frac{2}{\gamma^2}\big(f(\boldsymbol{y}) - f(\boldsymbol{x}) - \langle \boldsymbol{y}-\boldsymbol{x}, \nabla f(\boldsymbol{x}) \rangle\big) \,. \tag{3}$$

The curvature constant $C_f$ provides us a single "condition" number to measure the complexity of a constrained smooth optimization problem of the form (1), by incorporating both degrees of "difficulty" coming from the objective function $f$, as well as well as from the domain $\mathcal{D}$. The quantity was introduced by [5] for $\mathcal{D}$ being the unit simplex, and by [25, 26] for general domains.

Interestingly, the curvature constant $C_f$ (as well as the algorithm) does not depend on the choice of a norm. But still, the assumption of bounded curvature $C_f$ closely relates to a Lipschitz assumption on the gradient of $f$ (sometimes called $C_f$-*strong smoothness*), as formulated in following Lemma 2 taken from [26].

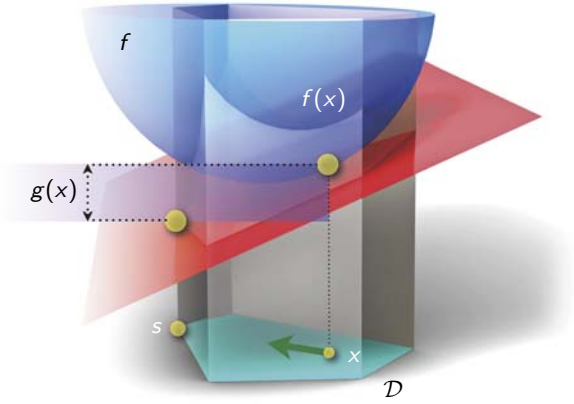
**Lemma 2.** *Let $f$ be a convex and differentiable function with its gradient $\nabla f$ being Lipschitz-continuous w.r.t. some norm $\|.\|$ over the domain $\mathcal{D}$ with Lipschitz-constant $L_{\|.\|} > 0$. Then*

$$C_f \leq \mathrm{diam}_{\|.\|}(\mathcal{D})^2 L_{\|.\|} \,.$$

*Duality Gap, and Certificates of Optimization Accuracy.* For a constrained convex optimization problem of the form (1) and a feasible point $\boldsymbol{x} \in \mathcal{D}$, one can define the following simple duality gap function [6, 7, 23, 5, 26]

$$g(\boldsymbol{x}) := \max_{\boldsymbol{s}\in\mathcal{D}} \big\langle \boldsymbol{x}-\boldsymbol{s}, \nabla f(\boldsymbol{x}) \big\rangle \,.$$

The quantity $g(\boldsymbol{x})$ serves as a useful *certificate* for the current optimization quality, by upper-bounding the (unknown) error $g(\boldsymbol{x}) \geq f(\boldsymbol{x}) - f(\boldsymbol{x}^\star)$. This property (weak duality) is directly implied by convexity of $f$, giving that the linearization $f(\boldsymbol{x}) + \big\langle \boldsymbol{s}-\boldsymbol{x}, \nabla f(\boldsymbol{x}) \big\rangle$ always lies below the graph of the function $f$, as again illustrated in the figure. While the optimal value $f(\boldsymbol{x}^\star)$ is unknown in most problems of interest, the quantity $g(\boldsymbol{x})$ for a candidate $\boldsymbol{x}$ is often easy to compute.




For example, the duality gap is "automatically" computed as a by-product of every iteration of the FW Algorithm 1: Whenever $\boldsymbol{s}$ is a minimizer of the linearized problem at $\boldsymbol{x}$ as returned by $\mathrm{LMO}_{\mathcal{D}}$, then this same $\boldsymbol{s}$ is a certificate for the current gap $g(\boldsymbol{x}) = \big\langle \boldsymbol{x}-\boldsymbol{s}, \nabla f(\boldsymbol{x}) \big\rangle$. Such certificates for the optimization error are useful not only for the algorithms considered here, but in fact for any optimizer of a

constrained problem of the form (1), e.g. as a stopping criterion, or to verify the numerical stability. The $O(1/t)$ convergence rate of the FW algorithm is known to not only hold for the primal error $f(\boldsymbol{x}^{(t)}) - f(\boldsymbol{x}^\star)$, but also for the duality gap $g(\boldsymbol{x}^{(t)})$ [5, 26]. The only change to the rate of Theorem 1 above is that the constant will be 6.75 $C_f$ instead of $2C_f$, and the error is guaranteed for one of the iterates up to $t$, not necessarily for the last one.

*Affine Invariance.* Interestingly, the Frank–Wolfe algorithm – as well as the convergence analysis presented here – is invariant under affine transformations and linear re-parameterizations of the input problem, and in this sense independent of the chosen geometry [26]. If one re-parameterizes the domain $\mathcal{D}$, by a *surjective* linear or affine map $M : \hat{\mathcal{D}} \to \mathcal{D}$, then the "old" and "new" optimization problem variants $\min_{\boldsymbol{x}\in\mathcal{D}} f(\boldsymbol{x})$ and $\min_{\hat{\boldsymbol{x}}\in\hat{\mathcal{D}}} \hat{f}(\hat{\boldsymbol{x}})$ for $\hat{f}(\hat{\boldsymbol{x}}) := f(M\hat{\boldsymbol{x}})$ look completely the same to the FW algorithm: More precisely, every iteration will remain exactly the same, and also the convergence with $C_f/t$ is unchanged, since the curvature constant $C_f$ by its definition (3) is also invariant (using that $\nabla\hat{f} = M^T\nabla f$). A natural example of such a re-parameterization is the use of *bary-centric coordinates*, when $\mathcal{D}$ is a convex hull of finitely many vectors (then $M$ contains these vectors as columns, and $\hat{\mathcal{D}}$ is the unit simplex).

Up to our knowledge, FW is the only first-order method for general constrained problems which features this property. Note that classical (proximal) gradient methods are *not* affine invariant, since they depend on the choice of a norm or a proximal function.

*Optimality in Terms of Sparsity and Iteration Complexity.* We have seen above that the FW algorithm in $t$ iterations delivers an $O(1/t)$-approximate solution of sparsity $t$ (meaning the iterate is a convex combination of at most $t$ atoms). This trade-off between the *optimization error* and the *sparsity of the iterate* is crucial of the success of FW methods, as the efficiency and memory requirement usually scales directly with the sparsity.

Interestingly, the trade-off is optimally characterized by $\Theta(1/t)$. No algorithm adding only one new atom (vertex of the domain $\mathcal{D}$) per iteration can obtain a better accuracy in general. The following lemma formalizes this by providing a lower bound of $\Omega(1/t)$ on the optimization error, for any solutions of sparsity at most $t$. This result also shows that the FW algorithm is optimal in terms of number of iterations, and cannot be accelerated in general.

For the lower bound, the domain is chosen as the unit simplex, $\mathcal{D} := \Delta_n \subseteq \mathbb{R}^n$. The same matching sparsity upper and lower bounds will also hold for optimizing over the $\ell_1$-ball instead, and also for the *rank* in trace-norm constrained optimization [25].

**Lemma 3** ([26]). *For $f(\boldsymbol{x}) := \|\boldsymbol{x}\|_2^2$, and $1 \leq t \leq n$, we have*

$$\min_{\substack{\boldsymbol{x}\in\Delta_n \\ \text{card}(\boldsymbol{x})\leq t}} f(\boldsymbol{x}) = \frac{1}{t} .$$

In other words, for any vector $\boldsymbol{x}$ of sparsity $\leq t$, the primal error $f(\boldsymbol{x}) - f(\boldsymbol{x}^\star)$ is always at least $1/t - 1/n$. A similar lower bound holds for the duality gap in the same setting. Furthermore, we observe that the convergence with $C_f/t$ as attained by the FW algorithm is optimal, since the curvature $C_f$ scales in the same way as in the bound here, when multiplying the objective function by a constant.

Note that the bound presented here assumes that the problem dimension $n$ exceeds the number of iterations. Specifically for the FW algorithm, and without considering sparsity, [3] has proved an asymptotic lower bound of $\Omega(1/t^{1+\mu})$ on the primal error of the FW when run on a quadratic objective, for all $\mu > 0$.

## 5 Applications

Since the early applications to network problems such as traffic assignment [16], FW-type methods have been applied to a surprisingly large variety of applications in completely different fields. In the following we briefly discuss a non-representative but relatively recent example, focusing on the aspect that the FW algorithm returns compact/sparse representations. This aspect is particularly interesting for domains $\mathcal{D}$ which are the convex hull of an infinite set of atoms $\mathcal{A}$.

### 5.1 Matrix Factorizations via Frank–Wolfe methods

An interesting class of atomic domains is obtained when we consider rank-one matrices or dyads of the form

$$\mathcal{A} := \left\{ \boldsymbol{u}\boldsymbol{v}^T \;\middle|\; \begin{matrix} \boldsymbol{u}\in\mathcal{A}_{\text{left}} \\ \boldsymbol{v}\in\mathcal{A}_{\text{right}} \end{matrix} \right\} \subset \mathbb{R}^{n\times m}$$

for two given sets of vectors $\mathcal{A}_{\text{left}} \subset \mathbb{R}^n$ and $\mathcal{A}_{\text{right}} \subset \mathbb{R}^m$. By definition, every iteration of a Frank–Wolfe type algorithm optimizing over such a domain $\mathcal{D}$ will consist of a rank-one update. In other words, the iterate $\boldsymbol{x}^{(t)} \in \mathbb{R}^{n\times m}$ is always conveniently represented as a low rank *matrix factorization*, more precisely of rank $\leq t$ after $t$ iterations. If the domain admits a linear minimization oracle $\text{LMO}_{\mathcal{D}}$, then FW algorithms are therefore a natural choice for obtaining such low rank-matrix factorizations. While the resulting trade-off between rank and optimization accuracy is worst-case optimal, not all types of such domains admit an efficient $\text{LMO}_{\mathcal{D}}$ oracle [26]. Using more than two atomic sets leads to a generalization to tensor factorizations.

*The Matrix Nuclear Norm Case.* The most natural example of this class of problems comes from the choice of $\mathcal{A}_{\text{left}}$ and $\mathcal{A}_{\text{right}}$ as the Euclidean unit-length vectors, in which case the domain

$$\mathcal{D} := \text{conv}(\mathcal{A}) = \text{conv}\left( \left\{ \boldsymbol{u}\boldsymbol{v}^T \;\middle|\; \begin{matrix} \boldsymbol{u}\in\mathbb{R}^n,\ \|\boldsymbol{u}\|_2=1 \\ \boldsymbol{v}\in\mathbb{R}^m,\ \|\boldsymbol{v}\|_2=1 \end{matrix} \right\} \right)$$

is known to be the unit ball of the matrix *nuclear norm* [11]. The nuclear norm of a matrix is defined as the sum of its singular values, or in other words the $\ell_1$-norm applied to the spectrum.

Proximal gradient algorithms require a projection onto the nuclear norm ball for this class of problems, which amounts to computing a full SVD in each iteration. In contrast, the linear oracle for the FW method is computationally much more efficient here: $\text{LMO}_{\mathcal{D}}(\boldsymbol{d})$ is attained by the top singular vector pair of $\boldsymbol{d} \in \mathbb{R}^{n\times m}$. Furthermore, memory efficiency is significantly improved, as FW only needs to store the $t$ rank-1 factors of $\boldsymbol{x}^{(t)}$, while the traditional methods would need to store a dense matrix $\boldsymbol{x}^{(t)} \in \mathbb{R}^{n\times m}$, which does not scale well to very large problems as for example in the following application:

*Recommender Systems.* Recommender systems have become increasingly important for suggesting relevant items of all sorts to users, readers or potential customers. In an open competition from 2007 to 2009, the movie rental company Netflix has offered a 1M$ prize for the algorithm predicting the ratings of movies given by users (measured on a scale from 1 to 5 stars). Mathematically, this problem can be formulated as follows: Given the known entries $\Omega \subseteq [n] \times [m]$ of a matrix $Y \in \mathbb{R}^{n\times m}$, where $n$ is the number of users, and $m$ is the number of movies, try to predict the unknown entries of $Y$.

Matrix factorization techniques were among the most successful approaches in the competition. Formally, those methods try to approximate the known entries of $Y$ by a low-rank factorization $UV^T$ with factors $U \in \mathbb{R}^{n\times k}$ and $V \in \mathbb{R}^{m\times k}$.

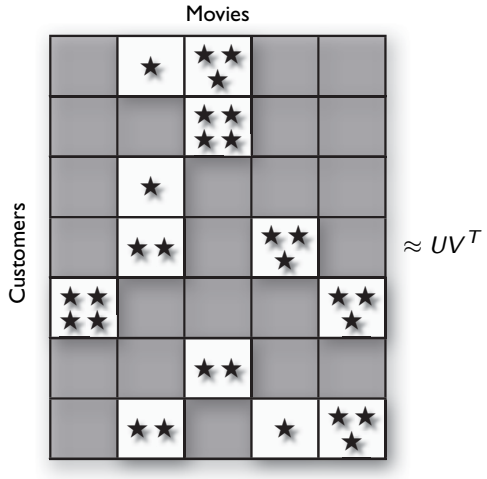



Figure 2. *Matrix Completion for Recommender Systems*

While finding the best approximation to the known entries $\Omega$ under a rank constraint is NP-hard [19] (when using e.g. squared error), the corresponding nuclear norm constrained problem – as defined above – offers an attractive alternative. The nuclear norm is the tightest convex relaxation of the rank in the spectral-norm unit-ball [11].

In this light, FW here offers an attractive trade-off between low rank and optimization error, and scales nicely to large problems due to the cheap iterations and the low memory requirements, avoiding dense matrices. Usually, the convex objective function is chosen as the squared error $f(\boldsymbol{x}) := \sum_{ij\in\Omega}(Y_{ij} - \boldsymbol{x}_{ij})^2$. In the case of the Netflix competition where $|\Omega| = 10^8$, $n \approx 500k$, and $m \approx 20k$, the FW method is easy to implement, whereas projected gradient methods are difficult to run on a single machine due to the memory requirements and expensive SVD computations per iteration. While as far as we know, the Frank–Wolfe algorithm was not part of the winning algorithms in the competition, it is arguably the simplest method which still comes with a guaranteed convergence rate [27].

## 6 Current Research Topics, and Outlook

Interest in first-order optimization algorithms of FW-type (i.e., using a linear minimization oracle) has increased significantly in recent years.

Recent work has addressed primal-dual rates [5, 25], non-smooth problems [30, 35], online variants [22], interpretation of the iterations in terms of duality [1] and mirror descent, or variations using different (non-)linear subproblems [34]. Other adaptations of FW to more structured problems include norm-regularized instead of constrained problem formulations [21], and block-structured problems [29]. Finally, a significant recent line of research addresses accelerated convergence rates under additional assumptions on the objective and the domain [31, 20, 17, 30, 28, 18]. We are looking forward to following the developments of this exciting research area in the next years.

Martin Jaggi, Department of Computer Science, ETH Zurich, Switzerland.
jaggi@inf.ethz.ch
Zaid Harchaoui, LEAR team, INRIA Rhone-Alpes Grenoble, France.
zaid.harchaoui@inria.fr

## Notes

1. We note that most existing literature about Frank–Wolfe/conditional gradient is not using the simple $2/(t+2)$ step-size, but instead performs a line-search for $\gamma$ in every iteration. We refer the reader to Section 3.1 for more details.

## Bibliography

[1] Francis Bach. Duality between subgradient and conditional gradient methods. *arXiv.org*, 2013.
[2] Dimitri P Bertsekas. Supplementary Chapter 6 on Convex Optimization Algorithms. In *Convex Optimization Theory*. Athena Scientific, 2010.
[3] M D Canon and C D Cullum. A Tight Upper Bound on the Rate of Convergence of Frank-Wolfe Algorithm. *SIAM Journal on Control*, 6, 1968.
[4] Venkat Chandrasekaran, Benjamin Recht, Pablo A Parrilo, and Alan S Willsky. The Convex Geometry of Linear Inverse Problems. *Foundations of Computational Mathematics*, 12(6):805–849, 2012.
[5] Kenneth L Clarkson. Coresets, Sparse Greedy Approximation, and the Frank-Wolfe Algorithm. *ACM Transactions on Algorithms*, 6(4), 2010.
[6] Vladimir F Demyanov and Aleksandr M Rubinov. The Minimization of a Smooth Convex Functional on a Convex Set. *SIAM Journal on Control*, 5(2), 1967.
[7] Vladimir F Demyanov and Aleksandr M Rubinov. Minimization of Functionals in Normed Spaces. *SIAM Journal on Control*, 6(1):73–88, 1968.
[8] Miroslav Dudik, Zaid Harchaoui, and Jerome Malick. Learning with matrix gauge regularizers. *NIPS Optimization Workshop*, 2011.
[9] Miroslav Dudik, Zaid Harchaoui, and Jerome Malick. Lifted coordinate descent for learning with trace-norm regularization. *AISTATS*, 2012.
[10] Joseph C Dunn and S Harshbarger. Conditional Gradient Algorithms with Open Loop Step Size Rules. *Journal of Mathematical Analysis and Applications*, 62(2):432–444, 1978.
[11] Maryam Fazel, Haitham Hindi, and Stephen P Boyd. A Rank Minimization Heuristic with Application to Minimum Order System Approximation. *Proceedings American Control Conference*, 6:4734–4739, 2001.
[12] Marguerite Frank. Two new classes of simple Lie algebras. *Transactions of the American Mathematical Society*, 112(3):456–482, 1958.
[13] Marguerite Frank and Philip Wolfe. An Algorithm for Quadratic Programming. *Naval Research Logistics Quarterly*, 3:95–110, 1956.
[14] Marguerite Frank. A New Class of Simple Lie Algebras. *Proceedings of the National Academy of Sciences*, 40(8):713–719, 1954.
[15] Robert M Freund and Paul Grigas. New Analysis and Results for the Frank-Wolfe Method. *arXiv.org*, 2013.
[16] Masao Fukushima. A modified Frank-Wolfe algorithm for solving the traffic assignment problem. *Transportation Research*, 18(2):169–177, 1984.
[17] Dan Garber and Elad Hazan. Playing Non-linear Games with Linear Oracles. *FOCS*, pages 420–428, 2013.
[18] Dan Garber and Elad Hazan. Faster Rates for the Frank-Wolfe Method over Strongly-Convex Sets. *arXiv*, 2014.
[19] Nicolas Gillis and François Glineur. Low-Rank Matrix Approximation with Weights or Missing Data Is NP-Hard. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1149, 2011.
[20] Jacques Guélat and Patrice Marcotte. Some Comments on Wolfe's 'Away Step'. *Mathematical Programming*, 35(1):110–119, 1986.
[21] Zaid Harchaoui, Anatoli Juditsky, and Arkadi Nemirovski. Conditional gradient algorithms for norm-regularized smooth convex optimization. *Mathematical Programming*, 2014.
[22] Elad Hazan and Satyen Kale. Projection-free Online Learning. *ICML*, 2012.
[23] Donald W Hearn. The gap function of a convex program. *Operations Research Letters*, 1(2):67–71, 1982.
[24] Charles A Holloway. An Extension of the Frank and Wolfe Method of Feasible Directions. *Mathematical Programming*, 6(1):14–27, 1974.
[25] Martin Jaggi. *Sparse Convex Optimization Methods for Machine Learning*. PhD thesis, ETH Zürich, October 2011.
[26] Martin Jaggi. Revisiting Frank-Wolfe: Projection-Free Sparse Convex Optimization. *ICML*, 2013.
[27] Martin Jaggi and Marek Sulovský. A Simple Algorithm for Nuclear Norm Regularized Problems. *ICML*, 2010.
[28] Simon Lacoste-Julien and Martin Jaggi. An Affine Invariant Linear Convergence Analysis for Frank-Wolfe Algorithms. In *NIPS 2013 Workshop on Greedy Algorithms, Frank-Wolfe and Friends*, 2013.
[29] Simon Lacoste-Julien, Martin Jaggi, Mark Schmidt, and Patrick Pletscher. Block-Coordinate Frank-Wolfe Optimization for Structural SVMs. *ICML*, 2013.
[30] Guanghui Lan. The Complexity of Large-scale Convex Programming under a Linear Optimization Oracle. *optimization-online.org*, 2013.
[31] Evgenij S Levitin and Boris T Polyak. Constrained Minimization Methods. *USSR Computational Mathematics and Mathematical Physics*, 6(5), 1966.
[32] Stéphane Mallat. *A Wavelet Tour of Signal Processing: The Sparse Way*. Academic Press, 3rd edition, 2008.
[33] Gerard G L Meyer. Accelerated Frank–Wolfe Algorithms. *SIAM Journal on Control*, 12(4):655–663, 1974.
[34] Michael Patriksson. Partial linearization methods in nonlinear programming. *Journal of Optimization Theory and Applications*, 78(2):227–246, 1993.
[35] Federico Pierucci, Zaid Harchaoui, and Jerome Malick. A smoothing approach for composite conditional gradient with nonsmooth loss. In *Conférence d'Apprentissage Automatique – Actes CAP'14*, 2014.

[36] Boris Nikolaevich Pshenichny, Iuri Mikhalovich Danilin. Numerical methods in extremal problems. *Mir Publishers*, 1978.
[37] Nikhil Rao, Parikshit Shah, Stephen J Wright, and Robert Nowak. A Greedy Forward-Backward Algorithm For Atomic Norm Constrained Minimization. *ICASSP*, 2013.
[38] Vladimir N Temlyakov. Greedy approximation in convex optimization. *arXiv.org*, 2012.
[39] Ambuj Tewari, Pradeep Ravikumar, and Inderjit S Dhillon. Greedy Algorithms for Structurally Constrained High Dimensional Problems. *NIPS*, 2011.

## Calls for Nominations

The following prizes will be presented at the *22nd International Symposium on Mathematical Programming* (ISMP 2015) to take place in Pittsburgh, PA, USA, July 12–19, 2015.

### Dantzig Prize

Nominations are solicited for the 2015 George B. Dantzig Prize, administered jointly by the Mathematical Optimization Society (MOS) and the Society for Industrial and Applied Mathematics (SIAM). This prize is awarded to one or more individuals for original research which by its originality, breadth and depth, is having a major impact on the field of mathematical optimization. The contribution(s) for which the award is made must be publicly available and may belong to any aspect of mathematical optimization in its broadest sense. Past prize recipients are listed at www.mathopt.org/?nav=dantzig#winners.

The members of the 2015 Dantzig Prize committee are:

- John Birge, University of Chicago
- Laurence Wolsey, Université catholique de Louvain
- Margaret Wright (Chair), New York University
- Stephen Wright, University of Wisconsin

Nominations should consist of a letter describing the nominee's qualifications for the prize, and a current curriculum vitae of the nominee including a list of publications. They should be sent to

Professor Margaret Wright
Computer Science Department, Courant Institute
New York University
251 Mercer Street
New York, NY 10012, USA
mhw@cs.nyu.edu

and received by November 15, 2014. Submission of nomination materials in electronic form is strongly encouraged.

### Beale–Orchard Hays Prize

Nominations are invited for the 2015 Beale-Orchard Hays Prize for excellence in computational mathematical programming. The prize is sponsored by the Mathematical Optimization Society, in memory of Martin Beale and William Orchard-Hays, pioneers in computational mathematical programming. Nominated works must have been published between Jan 1, 2009 and Dec 31, 2014, and demonstrate excellence in any aspect of computational mathematical programming. "Computational mathematical programming" includes the development of high-quality mathematical programming algorithms and software, the experimental evaluation of mathematical programming algorithms, and the development of new methods for the empirical testing of mathematical programming techniques. Full details of prize rules and eligibility requirements can be found at www.mathopt.org/?nav=boh.

The members of the 2015 Beale–Orchard Hays Prize committee are:

- Robert Bixby (Chair), Gurobi Optimization
- Michael Ferris, University of Wisconsin
- Masakazu Kojima, Tokyo Institute of Technology
- Petra Mutzel, University of Dortmund
- Katja Scheinberg, Lehigh University

Nominations can be submitted electronically or in writing, and should include detailed publication details of the nominated work. Electronic submissions should include an attachment with the final published version of the nominated work. If done in writing, submissions should include five copies of the nominated work. Supporting justification and any supplementary material are strongly encouraged but not mandatory. The Prize Committee reserves the right to request further supporting material and justification from the nominees. Nominations should be submitted to:

Dr. Robert Bixby
3733-1 Westheimer Road, Box 1001
Houston, Texas 77027, USA
bixby@gurobi.com

The deadline for receipt of nominations is January 15, 2015.

### Fulkerson Prize

Nominations are invited for the Delbert Ray Fulkerson Prize, sponsored jointly by the Mathematical Optimization Society (MOS) and the American Mathematical Society. Up to three awards of US$ 1,500 each are presented at each (triennial) International Symposium of the MOS. The Fulkerson Prize is for outstanding papers in the area of discrete mathematics.

Eligible papers should represent the final publication of the main result(s) and should have been published in a recognized journal or in a comparable, well-refereed volume intended to publish final publications only, during the six calendar years preceding the year of the Symposium (thus, from January 2009 through December 2014). The prizes will be given for single papers, not series of papers or books, and in the event of joint authorship the prize will be divided.

The term "discrete mathematics" is interpreted broadly and is intended to include graph theory, networks, mathematical programming, applied combinatorics, applications of discrete mathematics to computer science, and related subjects. While research work in these areas is usually not far removed from practical applications, the judging of papers will be based only on their mathematical quality and significance.

Further information about the Fulkerson Prize (including a list of previous winners) can be found at mathopt.org/?nav=fulkerson.

The members of the 2015 Fulkerson Prize committee are:

- Michele Conforti (Chair), University of Padova
- Friedrich Eisenbrand, EPFL Lausanne
- Egon Schulte, Northeastern University, Department of Mathematics

Nominations (including reference to the nominated article and an evaluation of the work) should be sent to

Professor Michele Conforti
Universita Di Padova
Dipartimento di Matematica
Via Trieste 63
35121 Padova
Italy
conforti@math.unipd.it

by February 15, 2015. Electronic submissions are preferred.

## Tucker Prize

The A. W. Tucker Prize will be awarded to an outstanding doctoral thesis. The thesis can deal with any area of mathematical optimization. The Tucker Prize Committee will screen the nominations and select at most three finalists. The finalists will be invited to give oral presentations of their work at a special session of the International Symposium on Mathematical Programming 2015. The Tucker Prize Committee will select the winner before the symposium and present the award prior to the conclusion of the symposium.

The doctoral thesis must have been approved formally (with signatures) by the nominee's thesis committee between March 1, 2012 and March 1, 2015. The thesis may concern any aspect of mathematical optimization.

The winner will receive an award of US$ 1000 and a certificate. The other finalists will also receive certificates. The Society will also pay partial travel expenses for each finalist to attend the Symposium. Reimbursements will normally be limited to US$ 750. The nominee's doctoral institution will be encouraged to assist any nominee selected as a finalist with additional travel expense reimbursement.

The members of the 2015 Tucker Prize committee are:

- Karen Aardal (Chair), Delft University of Technology
- Satoru Iwata, University of Tokyo
- Benedetta Morini, Universita degli Studi di Firenze
- Danny Ralph, University of Cambridge
- Uday Shanbhag, Penn State University

Nominations must be made by electronic mail to:

Professor Karen Aardal
Faculteit EWI, Technische Universiteit Delft
Mekelweg 4
2627 CD Delft
The Netherlands
k.i.aardal@tudelft.nl

The nominator must be a faculty member at the institution that awards the nominee's doctoral degree, or a member of the nominee's thesis committee. Applications should consist of pdf files as follows: a letter of nomination; the nominee's thesis; a separate summary of the thesis' contributions, written by the nominee, no more than eight (8) pages in length; and a brief biographical sketch of the nominee. Nominations and the accompanying documentation must be written in a language acceptable to the Tucker Prize Committee (= English). The Tucker Prize Committee may request additional information.

The deadline for nominations is March 15, 2015.

## Lagrange Prize

The prize, established in 2002, is awarded jointly by the Mathematical Optimization Society and the Society for Industrial and Applied Mathematics. It is awarded for outstanding works in the area of continuous optimization. Judging of works will be based primarily on their mathematical quality, significance, and originality. Clarity and excellence of the exposition and the value of the work in practical applications may be considered as secondary attributes.

Works to be considered should form the final publication of the main result(s) and should have been published between the years of 2009 to 2014 either as an article in a recognized journal, or in a comparable, well-referenced volume intended to publish final publications only; or as a monograph consisting chiefly of original results rather than previously published material. Extended abstracts and prepublications, and articles in published journals, journal sections or proceedings that are intended to publish non-final papers, are not eligible.

The award will include a certificate containing the citation and a cash prize of US$ 1,500. In the event of joint authorship, the prize will be divided.

The members of the 2015 Lagrange Prize committee are:

- Mihai Anitescu (Chair), Argonne National Laboratory
- Kurt Anstreicher, University of Iowa
- Larry Biegler, Carnegie Mellon University
- Werner Römisch, Humboldt-Universität zu Berlin

Nominations should include a letter evaluating the contribution(s) of the work and citing the works to be considered. For questions, please contact via email:

Mihai Anitescu
Mathematics and Computer Science Division
Argonne National Laboratory, Building 240
9700 S. Cass Avenue Argonne, IL 60439, USA
anitescu@mcs.anl.gov

Nominations should be sent by e-mail to lagrange-prize@mcs.anl.gov, so as to arrive by February 1, 2015.

## Tseng Lectureship

The Mathematical Optimization Society invites nominations for the Paul Y. Tseng Memorial Lectureship in Continuous Optimization. This prize was established in 2011 and will be presented for the second time at the International Symposium on Mathematical Programming 2015. The lectureship was established on the initiative of family and friends of Professor Tseng, with financial contributions to the endowment also from universities and companies in the Asia-Pacific region. The purposes of the lectureship are to commemorate the outstanding contributions of Professor Tseng in continuous optimization and to promote the research and applications of continuous optimization in the Asia-Pacific region.

The lectureship is awarded to an individual for outstanding contributions in the area of continuous optimization, consisting of original theoretical results, innovative applications, or successful software development. The primary consideration in the selection process is the quality and impact of the candidate's work in continuous optimization. See more details at www.mathopt.org/?nav=tseng.

The members of the 2015 Paul Y. Tseng Memorial Lectureship committee are:

- Xiaojun Chen, Hong Kong Polytechnic University
- Yu-Hong Dai, AMSS, Chinese Academy of Sciences
- Tom Luo, University of Minnesota
- Yinyu Ye (Chair), Stanford University

The nomination must include a nomination letter of no more than two pages and a short CV of the candidate (no more than two pages, including selected publications). In addition, the nominator should also arrange for 1-2 letters of recommendation. All nomination materials should be sent (preferably in electronic form, as pdf documents) to:

Professor Yinyu Ye
Department of Management Science and Engineering
Huang Engineering Center 308
475 Via Ortega
School of Engineering
Stanford University, CA 94305-4121, USA
yyye@stanford.edu

All nomination materials must be received by December 31, 2014.

# ISMP 2015 in Pittsburgh

The 22nd *International Symposium on Mathematical Programming* (ISMP 2015) will take place in Pittsburgh, PA, USA, July 12–19, 2015. ISMP is a scientific meeting held every three years on behalf of the *Mathematical Optimization Society* (MOS). It is the world congress of mathematical optimization where scientists as well as industrial users of mathematical optimization meet in order to present the most recent developments and results and to discuss new challenges from theory and practice.

## Conference Topics

The conference topics address all theoretical, computational and practical aspects of mathematical optimization including

- integer, linear, nonlinear, semidefinite, conic and constrained programming;
- discrete and combinatorial optimization;
- matroids, graphs, game theory, network optimization;
- nonsmooth, convex, robust, stochastic, PDE-constrained and global optimization;
- variational analysis, complementarity and variational inequalities;
- sparse, derivative-free and simulation-based optimization;
- implementations and software;
- operations research;
- logistics, traffic and transportation, telecommunications, energy systems, finance and economics.

## Conference Venue

The Symposium will take place at the Wyndham Grand Pittsburgh Downtown Hotel located at the confluence of Pittsburgh's famed Three Rivers. The opening ceremony will take place on Sunday, July 11, 2015.

The opening session will feature the presentation of awards by the Mathematical Optimization Society accompanied by riveting entertainment, and followed by the welcome reception.

## Plenary speakers

- Laurent El Ghaoui, University of California, Berkeley
- Jim Geelen, University of Waterloo, Canada
- Daniel Kuhn, EPFL, Switzerland
- Daniel A. Spielman, Yale University
- Stephen J. Wright, University of Wisconsin

## Semi-plenary speakers

- Samuel A. Burer, University of Iowa
- Roberto Cominetti, University of Chile, Chile
- Michelangelo Conforti, University of Padova, Italy
- Tamara G. Kolda, Sandia Labs
- Andrea Lodi, University of Bologna, Italy
- Asu Ozdaglar, Massachusetts Institute of Technology
- Werner Roemisch, Humboldt University Berlin, Germany
- Frank Vallentin, University of Koeln, Germany
- Pascal van Hentenryck, NICTA, Australia
- Yaxian Yuan, Chinese Academy of Sciences, China

## Registration and Important Dates

| | |
|---|---|
| September 1st, 2014 | Abstract submission opens. |
| December 15th, 2014 | Registration opens. |
| March 2nd, 2015 | Abstract submission deadline |
| March 27th, 2015 | Notification of acceptance |
| April 15th, 2015 | Early registration deadline |

*Early registration rate for 2015 MOS members*

| | |
|---|---|
| Student or retiree: | US$ 190 |
| Lifetime member: | US$ 290 |
| Regular member: | US$ 375 |

*Early registration rate for non-2015 MOS members*

| | |
|---|---|
| Student: | US$ 210 |
| Retiree: | US$ 220 |
| All others: | US$ 435 |

The registration rates for late registration will be higher.

## Web site

More details (including clusters, cluster chairs, all registration rates, discounted MOS membership for 2016–2018, hotel, sponsorship opportunities, exhibits etc.) will soon be available on the conference web site at www.ismp2015.org.

ISMP2015 PITTSBURGH
22nd INTERNATIONAL SYMPOSIUM on MATHEMATICAL PROGRAMMING
JULY12–17

*Pittsburgh skyline, taken from Mount Washington (Photo: Filipe Fortes, CC BY-SA 2.0)*

## Call for pre-proposals to organize and host ISMP 2018

The Symposium Advisory Committee of the *Mathematical Optimization Society* issues a call for pre-proposals to organize and host ISMP 2018, the triennial International Symposium on Mathematical Programming.

ISMP is the flagship event of our society, regularly gathering over a thousand scientists from around the world. The conference is usually held in or around the month of August. Hosting ISMP provides a vital service to the mathematical optimization community and often has a lasting effect on the visibility of the hosting institution. It also presents a significant challenge. This call for pre-proposals is addressed at local groups willing to take up that challenge. The tradition would be that only sites outside of USA and Canada are eligible to host ISMP 2018.

Preliminary bids will be examined by the Symposium Advisory Committee (SAC), which will then issue invitations for detailed bids. The final decision will be made and announced during ISMP 2015 in Pittsburgh. Members of the SAC are

- Michael Juenger, Germany, mjuenger@informatik.uni-koeln.de
- Jan Karel Lenstra, The Netherlands (chair), jkl@cwi.nl
- Jeff Linderoth, USA, linderot@cae.wisc.edu
- Andy Philpott, New Zealand, a.philpott@auckland.ac.nz
- Kim-Chuan Toh, Singapore, mattohkc@nus.edu.sg
- Luis Nunes Vicente, Portugal lnv@mat.uc.pt

Preliminary bids should be brief and contain information pertaining to the

1. location,
2. facilities,
3. logistics: accommodation and transportation, and
4. likely local organizers.

Further information can be obtained from any member of the advisory committee. Please address your preliminary bids until October 15, 2014 to Jan Karel Lenstra jkl@cwi.nl.

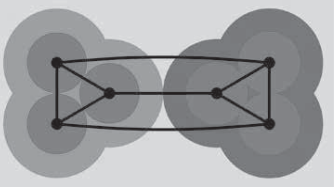


**IMPRINT**
**Editor:** Volker Kaibel, Institut für Mathematische Optimierung, Otto-von-Guericke Universität Magdeburg, Universitätsplatz 2, 39108 Magdeburg, Germany. kaibel@ovgu.de ■ **Co-Editors:** Samuel Burer, Department of Management Sciences, The University of Iowa, Iowa City, IA 52242-1994, USA. samuel-burer@uiowa.edu ■ Jeff Linderoth, Department of Industrial and Systems Engineering, University of Wisconsin-Madison, Madison, WI 53706-1572, USA. linderoth@wisc.edu ■ **Founding Editor:** Donald W. Hearn ■ Published by the Mathematical Optimization Society ■ Design and typesetting by Christoph Eyrich, Berlin, Germany. optima@0x45.de ■ Printed by Oktoberdruck AG, Berlin, Germany.

HAUSDORFF RESEARCH INSTITUTE FOR MATHEMATICS

universitätbonn

# September - December 2015



## Trimester Program

# Combinatorial Optimization

**Organizers:** András Frank, Satoru Iwata, Jochen Könemann, Jens Vygen

Combinatorial optimization is an active field leveraging ideas from many different areas including graph theory, combinatorics, matroid theory, submodularity, connectivity, network flows, approximation algorithms, mathematical programming, game theory, algebraic and geometric methods, and applications. This trimester program is intended to bring together the field's best researchers focusing on the discovery of new connections, and to establish new and deepen existing international collaborations.

We will host long-term visitors and plan four workshops during the program, on the broad topics

- Connectivity, Routing, and Network Design
- Rigidity, Submodularity, and Discrete Convexity
- Relaxations and Polyhedral Methods
- Algorithmic and Computational Game Theory

**Those planning to participate include:**

Nikhil Bansal
Joseph Cheriyan
William Cook
Gérard Cornuéjols
Friedrich Eisenbrand
Samuel Fiorini
Satoru Fujishige
Michel Goemans
Anupam Gupta
Nicole Immorlica
Kamal Jain
Tibor Jordán
Michael Jünger
Volker Kaibel
Monique Laurent
Jon Lee
Ruta Mehta
Kazuo Murota
Neil Olver
Britta Peis
R. Ravi
Bruce Reed
Thomas Rothvoß
Laura Sanità
Alexander Schrijver
Andreas Schulz
András Sebő
Bruce Shepherd
David Shmoys
Martin Skutella
Cliff Stein
Bernhard von Stengel
Ola Svensson
Chaitanya Swamy
Jan Vondrák
Robert Weismantel
David Williamson
Gerhard Woeginger
Laurence Wolsey
Rico Zenklusen

**Call for participation:** The Hausdorff Research Institute offers visiting positions for the whole period of the trimester program (for senior scientists, postdocs and PhD students). The deadline for applications is January 31, 2015. Please send applications (including CV and, for postdocs and PhD students, a letter of recommendation) using our online application form at **http://www.him.uni-bonn.de/combinatorial-optimization-2015/.** In addition numerous fellowships for shorter periods are available.