Frank-Wolfe Algorithms for Saddle Point Problems

Gauthier Gidel
INRIA - Sierra team
ENS Paris

Tony Jebara
Department of CS
Columbia University, NYC

Simon Lacoste-Julien
Department of CS & OR (DIRO)
Université de Montréal, Montréal

Abstract

We extend the Frank-Wolfe (FW) optimization algorithm to solve constrained smooth convex-concave saddle point (SP) problems. Remarkably, the method only requires access to linear minimization oracles. Leveraging recent advances in FW optimization, we provide the first proof of convergence of a FW-type saddle point solver over polytopes, thereby partially answering a 30 year-old conjecture. We verify our convergence rates empirically and observe convergence under more general conditions with a heuristic step-size, paving the way for future work.

1 Introduction

The Frank-Wolfe (FW) optimization algorithm [7], also known as the conditional gradient method [5], is a first-order method for smooth constrained optimization over a compact set. It has recently enjoyed a surge in popularity thanks to its ability to cheaply exploit the structured constraint sets appearing in machine learning applications [12, 15]. A known forte of FW is that it only requires access to a *linear minimization oracle* (LMO) over the constraint set, i.e., the ability to minimize linear functions over the set, in contrast to projected gradient methods which require the minimization of *quadratic* functions or other nonlinear functions. In this paper, we extend the applicability of the FW algorithm to solve the following convex-concave saddle point (SP) problems:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}), \qquad \text{with only access to} \quad \text{LMO}(\mathbf{r}) \in \arg\min_{\boldsymbol{s}\in\mathcal{X}\times\mathcal{Y}} \langle \boldsymbol{s}, \mathbf{r}\rangle\,, \tag{1}$$

where $\mathcal{L}$ is a smooth (with L-Lipschitz continuous gradient) *convex-concave function*, that is, $\mathcal{L}(\cdot, \boldsymbol{y})$ is convex for all $\boldsymbol{y} \in \mathcal{Y}$ and $\mathcal{L}(\boldsymbol{x}, \cdot)$ is concave for all $\boldsymbol{x} \in \mathcal{X}$. We also assume that $\mathcal{X} \times \mathcal{Y}$ is a convex compact set such that its LMO is cheap to compute. A *saddle point solution* to (1) is a pair $(\boldsymbol{x}^*, \boldsymbol{y}^*) \in \mathcal{X} \times \mathcal{Y}$ [10, VII.4] such that:

$$\mathcal{L}(\boldsymbol{x}^*, \boldsymbol{y}) \leq \mathcal{L}(\boldsymbol{x}^*, \boldsymbol{y}^*) \leq \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}^*) \quad \forall \boldsymbol{x} \in \mathcal{X},\, \forall \boldsymbol{y} \in \mathcal{Y}. \tag{2}$$

Examples of saddle point problems. Taskar et al. [21] cast the maximum-margin estimation of structured output models as a bilinear saddle point problem $\mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{x}^\top M \boldsymbol{y}$, where $\mathcal{X}$ is the regularized set of parameters and $\mathcal{Y}$ is an encoding of the set of possible structured outputs. They considered settings where projection on $\mathcal{X}$ and $\mathcal{Y}$ were efficient but one can imagine many situations where only LMO's are efficient. For example, we could use a structured sparsity inducing norm [18] for the parameter $\boldsymbol{x}$, such as the overlapping group lasso for which the projection is expensive [2], while $\mathcal{Y}$ could be a combinatorial object such as a the ground state of a planar Ising model (without external field) which admits an efficient oracle [3] but has potentially intractable projection. Similarly, two-player games [22] can often be solved as bilinear minimax problems. In situations such as the Matching Duel [1], the strategy space is intractably large and defined by an exponential number of linear constraints. Fortunately, some linear minimization oracles such as the blossom algorithm [6] can efficiently optimize over matching polytopes despite an exponential number of linear constraints.

Related work. The standard approaches to solve smooth constrained SP problems are projection-type methods (surveyed in Xiu and Zhang [23]), with in particular variations of Korpelevich's

29th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

extragradient method [14], such as [19] which was used to solve the structured prediction problem [21] mentioned above. There is surprisingly little work on FW-type methods for saddle point problems, although they were briefly considered for the more general *variational inequality* problem (VIP):

$$\text{find} \quad \boldsymbol{z}^* \in \mathcal{Z} \quad \text{s.t.} \quad \langle \boldsymbol{r}(\boldsymbol{z}^*), \boldsymbol{z} - \boldsymbol{z}^* \rangle \geq 0 \quad \text{for all } \boldsymbol{z} \in \mathcal{Z}, \tag{3}$$

where $\boldsymbol{r}$ is a Lipschitz mapping from $\mathbb{R}^p$ to itself and $\mathcal{Z} \subseteq \mathbb{R}^p$. By using $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ and $\boldsymbol{r}(\boldsymbol{z}) = (\nabla_x \mathcal{L}(\boldsymbol{z}), -\nabla_y \mathcal{L}(\boldsymbol{z}))$, (3) reduces to the equivalent optimality conditions for the SP problem (1). Hammond [9] showed that a FW algorithm with a step size of $O(1/t)$ converges for the VIP (3) when the set $\mathcal{Z}$ is strongly convex, while FW with a generalized line-search on a saddle point problem is sometimes non-convergent when $\mathcal{Z}$ is a polytope (see also [20, § 3.1.1]). She conjectured though that using a step size of $O(1/t)$ was also convergent when $\mathcal{Z}$ is a polytope – a problem left open up to this point. More recently, Juditsky and Nemirovski [13] (see also Cox et al. [4]) proposed a method to transform a VIP on $\mathcal{Z}$ where one has only access to a LMO, to a "dual" VIP on which they can use a projection-type method. Lan [16] proposes to solve the SP problem (1) by running FW on $\mathcal{X}$ on the *smoothed* version of the problem $\max_{\boldsymbol{y} \in \mathcal{Y}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{y})$, thus requiring a projection oracle on $\mathcal{Y}$. In contrast, in this paper we study simple approaches that do not require any transformations of (1) nor any projection oracle on $\mathcal{X}$ or $\mathcal{Y}$.

Contributions. In § 2, we extend several variants of the FW algorithm to solve the saddle point problem (1) that we think could be of interest to the machine learning community. In § 3, we state our convergence results for these methods over polytope domains giving a partial answer to the conjecture from Hammond [9]. We finally present illustrative experiments for our theory noticing that the convergence theory is still incomplete for these methods.

2 Saddle point Frank-Wolfe

The algorithms. This paper will explore two SP extensions of the classical *Frank-Wolfe* algorithm which are summarized in Algorithm 1 and Algorithm 2. In the following the point computed by these algorithms after t steps will be noted $\boldsymbol{z}^{(t)} = (\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})$. We first obtain the *saddle point FW* (SP-FW) algorithm by simultaneously doing a FW update on both convex functions $\mathcal{L}(\cdot, \boldsymbol{y}^{(t)})$ and $-\mathcal{L}(\boldsymbol{x}^{(t)}, \cdot)$ with a properly chosen step size. Hence the point $\boldsymbol{z}^{(t)}$ has a sparse representation as a convex combination of the points previously given by the oracle. This set of points is called *active set*. If we assume that $\mathcal{X}$ and $\mathcal{Y}$ are the convex hulls of two finite sets of points $\mathcal{A}$ and $\mathcal{B}$, we can also extend the *away-step Frank-Wolfe* (AFW) algorithm [15] to saddle point problems. As for AFW, this new algorithm is able to remove mass from "bad" atoms in the active set to avoid the zig-zagging problem that slows down standard FW [15]. Because of the special product structure of the domain, we consider more away directions than proposed in [15] for AFW. Namely, for every corner $\boldsymbol{v} = (\boldsymbol{v}_x, \boldsymbol{v}_y)$ and $\boldsymbol{v}' = (\boldsymbol{v}'_x, \boldsymbol{v}'_y)$ already picked, $\boldsymbol{x} - \boldsymbol{v}_x$ ($\boldsymbol{y} - \boldsymbol{v}_y$) is a feasible direction in $\mathcal{X}$ ($\mathcal{Y}$). Thus every combination $(\boldsymbol{x} - \boldsymbol{v}_x, \boldsymbol{y} - \boldsymbol{v}'_y)$ is a feasible direction even if this particular pair of corners have never been picked together. We thus maintain the iterates on $\mathcal{X}$ and $\mathcal{Y}$ as independent convex combination of their respective active sets of corners (Line 13 of Algorithm 2), i.e., $\boldsymbol{x}^{(t)} = \sum_{\boldsymbol{v}_x \in \mathcal{S}_x^{(t)}} \alpha_{\boldsymbol{v}_x} \boldsymbol{v}_x$ (and similarly for $\boldsymbol{y}^{(t)}$).

Algorithm 1 Saddle point FW algorithm

1: Let $\boldsymbol{z}^{(0)} = (\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}) \in \mathcal{X} \times \mathcal{Y}$
2: **for** $t = 0 \dots T$ **do**
3: Compute $\boldsymbol{r}^{(t)} := \begin{pmatrix} \nabla_x \mathcal{L}(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) \\ -\nabla_y \mathcal{L}(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)}) \end{pmatrix}$
4: Compute $\boldsymbol{s}^{(t)} \in \operatorname*{argmin}_{\boldsymbol{z} \in \mathcal{X} \times \mathcal{Y}} \left\langle \boldsymbol{z}, \boldsymbol{r}^{(t)} \right\rangle$
5: Compute $g_t := \left\langle \boldsymbol{z}^{(t)} - \boldsymbol{s}^{(t)}, \boldsymbol{r}^{(t)} \right\rangle$
6: Let $\gamma = \min\left(1, \frac{\nu}{2C} g_t\right)$ **or** $\gamma = \frac{2}{2+t}$
7: Update $\boldsymbol{z}^{(t+1)} := (1-\gamma)\boldsymbol{z}^{(t)} + \gamma \boldsymbol{s}^{(t)}$
8: **end for**

The proposed algorithms preserve several nice properties of previous FW methods (in addition to only requiring LMO's): simplicity of implementation, affine invariance [12], gap certificates computed for free, sparse representation of the iterates and the possibility to have adaptive step sizes using the gap.

The suboptimality error and the gap. We define the following *suboptimality error* h_t for our saddle point problem:

$$h_t := \mathcal{L}(\boldsymbol{x}^{(t)}, \widehat{\boldsymbol{y}}^{(t)}) - \mathcal{L}(\widehat{\boldsymbol{x}}^{(t)}, \boldsymbol{y}^{(t)}), \quad \text{where} \begin{cases} \widehat{\boldsymbol{x}}^{(t)} := \operatorname*{arg\,min}_{\boldsymbol{x} \in \mathcal{X}} \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}^{(t)}) \\ \widehat{\boldsymbol{y}}^{(t)} := \operatorname*{arg\,max}_{\boldsymbol{y} \in \mathcal{Y}} \mathcal{L}(\boldsymbol{x}^{(t)}, \boldsymbol{y}). \end{cases} \tag{4}$$

Algorithm 2 Saddle point away-step Frank-Wolfe algorithm: **SP-AFW**$(\boldsymbol{z}^{(0)}, \mathcal{A} \times \mathcal{B}, \epsilon)$

Let $\boldsymbol{z}^{(0)} = (\boldsymbol{x}^{(0)}, \boldsymbol{y}^{(0)}) \in \mathcal{A} \times \mathcal{B}$, $\mathcal{S}_x^{(0)} := \{\boldsymbol{x}^{(0)}\}$ and $\mathcal{S}_y^{(0)} := \{\boldsymbol{y}^{(0)}\}$
for $t = 0 \dots T$ **do**
  Let $\boldsymbol{s}^{(t)} := \mathrm{LMO}_{\mathcal{A}\times\mathcal{B}}\big(\boldsymbol{r}^{(t)}\big)$ and $\boldsymbol{d}_{\mathrm{FW}}^{(t)} := \boldsymbol{s}^{(t)} - \boldsymbol{z}^{(t)}$ *($\boldsymbol{r}^{(t)}$ as defined in L3 in Algorithm 1)*
  Let $\boldsymbol{v}^{(t)} \in \underset{\boldsymbol{v}\in\mathcal{S}_x^{(t)}\times\mathcal{S}_y^{(t)}}{\arg\max} \big\langle \boldsymbol{r}^{(t)}, \boldsymbol{v}\big\rangle$ and $\boldsymbol{d}_{\mathrm{A}}^{(t)} := \boldsymbol{z}^{(t)} - \boldsymbol{v}^{(t)}$ *(the away direction)*
  if $g_t^{\mathrm{FW}} := \Big\langle -\boldsymbol{r}^{(t)}, \boldsymbol{d}_{\mathrm{FW}}^{(t)}\Big\rangle \le \epsilon$ **then return** $\boldsymbol{z}^{(t)}$ *(FW gap is small enough, so return)*
  if $\Big\langle -\boldsymbol{r}^{(t)}, \boldsymbol{d}_{\mathrm{FW}}^{(t)}\Big\rangle \ge \Big\langle -\boldsymbol{r}^{(t)}, \boldsymbol{d}_{\mathrm{A}}^{(t)}\Big\rangle$ **then**
    $\boldsymbol{d}^{(t)} := \boldsymbol{d}_{\mathrm{FW}}^{(t)}$, and $\gamma_{\max} := 1$ *(choose the FW direction)*
  else
    $\boldsymbol{d}^{(t)} := \boldsymbol{d}_{\mathrm{A}}^{(t)}$, and $\gamma_{\max} := \min\left\{\frac{\alpha_{\boldsymbol{v}_x^{(t)}}}{1-\alpha_{\boldsymbol{v}_x^{(t)}}}, \frac{\alpha_{\boldsymbol{v}_y^{(t)}}}{1-\alpha_{\boldsymbol{v}_y^{(t)}}}\right\}$ *(a* drop step *is when $\gamma_t = \gamma_{\max}$)*
  end if
  Let $g_t^{\mathrm{PFW}} = \langle -\boldsymbol{r}^{(t)}, \boldsymbol{d}_{\mathrm{FW}}^{(t)} + \boldsymbol{d}_{\mathrm{A}}^{(t)}\rangle$ **and** $\gamma_t = \min\left\{\gamma_{\max}, \frac{\nu}{2C} g_t^{\mathrm{PFW}}\right\}$ *(ν^{PFW} and C set as in Thm. 1)*
  Update $\boldsymbol{z}^{(t+1)} := \boldsymbol{z}^{(t)} + \gamma_t \boldsymbol{d}^{(t)}$ *(and accordingly for the weights $\boldsymbol{\alpha}^{(t+1)}$, see [15])*
  Update $\mathcal{S}_x^{(t+1)} := \{\boldsymbol{v}_x \in \mathcal{A} \text{ s.t. } \alpha_{\boldsymbol{v}_x}^{(t+1)} > 0\}$ and $\mathcal{S}_y^{(t+1)} := \{\boldsymbol{v}_y \in \mathcal{B} \text{ s.t. } \alpha_{\boldsymbol{v}_y}^{(t+1)} > 0\}$
end for

By convex-concavity, h_t can be upper-bounded by the following FW linearization gap [11, 12, 17, 24]:

$$g_t^{\mathrm{FW}} := \underbrace{\max_{\boldsymbol{s}_x\in\mathcal{X}} \Big\langle \boldsymbol{x}^{(t)} - \boldsymbol{s}_x, \nabla_x \mathcal{L}(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\Big\rangle}_{:=g_t^{(x)}} + \underbrace{\max_{\boldsymbol{s}_y\in\mathcal{Y}} \Big\langle \boldsymbol{y}^{(t)} - \boldsymbol{s}_y, -\nabla_y \mathcal{L}(\boldsymbol{x}^{(t)}, \boldsymbol{y}^{(t)})\Big\rangle}_{:=g_t^{(y)}}. \tag{5}$$

This gap is easy to compute and gives a stopping criterion since $g_t^{\mathrm{FW}} \ge h_t$.

3 SP-FW for strongly convex functions

In this section, we will assume that $\mathcal{L}$ is uniformly $(\mu_{\mathcal{X}}, \mu_{\mathcal{Y}})$-strongly convex-concave, which means that the following function is convex-concave:

$$(\boldsymbol{x}, \boldsymbol{y}) \mapsto \mathcal{L}(\boldsymbol{x}, \boldsymbol{y}) - \tfrac{\mu_{\mathcal{X}}}{2}\|\boldsymbol{x}\|^2 + \tfrac{\mu_{\mathcal{Y}}}{2}\|\boldsymbol{y}\|^2. \tag{6}$$

Convergence result. We now provide a theorem that establishes convergence in two situations: (I) when the SP belongs to the interior of $\mathcal{X} \times \mathcal{Y}$; (P) when the set is a polytope, i.e. when there exist two finite sets such that $\mathcal{X} = \mathrm{conv}(\mathcal{A})$ and $\mathcal{Y} = \mathrm{conv}(\mathcal{B})$). Our convergence result holds when (roughly) the strong convex-concavity of $\mathcal{L}$ is big enough in comparison to the largest Lipschitz constants L_{XY}, L_{YX} respectively of $\nabla_x \mathcal{L}(\boldsymbol{x}, \cdot)$ and $\nabla_x \mathcal{L}(\cdot, \boldsymbol{y})$ multiplied by geometric "condition numbers" of each set. The condition number of $\mathcal{X}$ (and similarly for $\mathcal{Y}$) is defined as the ratio of its diameter $D_{\mathcal{X}} := \sup_{\boldsymbol{x},\boldsymbol{x}'\in\mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}'\|$ over the following appropriate notions of "width":

$$\text{border distance: } \delta_{\mathcal{X}} := \min_{\boldsymbol{s}_x\in\partial\mathcal{X}} \|\boldsymbol{x}^* - \boldsymbol{s}\| \text{ for (I)}, \quad \text{pyramidal width: } \delta_{\mathcal{A}} := PWidth(\mathcal{A}) \text{ for (P)}.$$

The pyramidal width is formally defined in Eq. 9 of Lacoste-Julien and Jaggi [15]. We present below a non-affine invariant version of our theorem (for simplicity), a fully affine invariant version is given in the longer version [8].

Theorem 1. *Let $\mathcal{L}$ be a convex-concave function and $\mathcal{X} \times \mathcal{Y}$ a convex and compact set. Assume that the gradient of $\mathcal{L}$ is L-Lipschitz continuous, that $\mathcal{L}$ is $(\mu_{\mathcal{X}}, \mu_{\mathcal{Y}})$-strongly convex-concave, and that we are in one of the two following situations:*

(I) *The saddle point belongs to the interior of $\mathcal{X} \times \mathcal{Y}$. In this case, set $g_t = g_t^{\mathrm{FW}}$ (as in L5 of Alg. 2), $\delta_\mu := \sqrt{\min(\mu_{\mathcal{X}}\delta_{\mathcal{X}}^2, \mu_{\mathcal{Y}}\delta_{\mathcal{Y}}^2)}$ and $a := 1$. "Algorithm" then refers to SP-FW.*

(P) *The sets $\mathcal{X}$ and $\mathcal{Y}$ are polytopes. In this case, set $g_t = g_t^{\mathrm{PFW}}$ (as in L11 of Alg. 2), $\delta_\mu := \sqrt{\min(\mu_{\mathcal{X}}\delta_{\mathcal{A}}^2, \mu_{\mathcal{Y}}\delta_{\mathcal{B}}^2)}$ and $a := \frac{1}{2}$. "Algorithm" then refers to SP-AFW.*

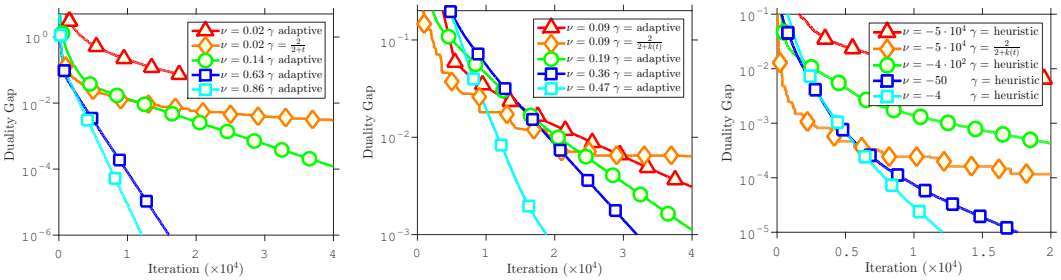
In both cases, if $\nu := a - \frac{\sqrt{2}}{\delta_\mu} \max\left\{\frac{D_{\mathcal{X}} L_{XY}}{\sqrt{\mu_{\mathcal{Y}}}}, \frac{D_{\mathcal{Y}} L_{YX}}{\sqrt{\mu_{\mathcal{X}}}}\right\}$ is positive, then the errors h_t (4) of the iterates of the algorithm with step size $\gamma_t = \min\{\gamma_{\max}, \frac{\nu}{2C} g_t\}$ decrease geometrically as

$$h_t = O\left((1-\rho)^{\frac{k(t)}{2}}\right) \quad \textit{and moreover, the gaps:} \quad \min_{s\le t} g_s = O\left((1-\rho)^{\frac{k(t)}{2}}\right),$$

where $\rho := \frac{\nu^2 \mu}{2C}$, $C := \frac{LD_{\mathcal{X}}^2 + LD_{\mathcal{Y}}^2}{2}$ *and* $k(t)$ *is the number of non-drop step after* t *steps (see L9 in Alg. 2). In case (I) we have* $k(t) = t$ *and in case (P) we have* $k(t) \geq t/3$*. For both algorithms, if* $\delta_\mu > 2 \max\left\{\frac{D_{\mathcal{X}} L_{XY}}{\mu_{\mathcal{X}}}, \frac{D_{\mathcal{Y}} L_{YX}}{\mu_{\mathcal{Y}}}\right\}$*, we also obtain a sublinear rate with the universal choice* $\gamma_t = \min\{\gamma_{\max}, \frac{2}{2+k(t)}\}$*:*

$$\min_{s \leq t} h_s \leq \min_{s \leq t} g_s^{\text{FW}} = O\left(\frac{1}{t}\right). \tag{7}$$

Clearly, the sublinear rate seems less interesting than the linear one but has the added convenience that the step size can be set without knowledge of various constants that characterize $\mathcal{L}$. Moreover, it provides a partial answer to the conjecture from Hammond [9].



(a) SP in the interior, $d = 30$ (b) $\mathcal{X} \times \mathcal{Y}$ is a polytope, $d = 30$ (c) $\mathcal{X} \times \mathcal{Y}$ polytope, $d = 30$, $\nu < 0$

Figure 1: The best gap observed $\min_{s \leq t} g_s^{\text{FW}}$ is plotted as a function of t on a semilog scale.

Toy experiments. First, we test the empirical convergence of our algorithms on a simple saddle point problem over the unit cube in dimension d (whose pyramidal width has the explicit value $1/\sqrt{d}$ by Lemma 4 from Lacoste-Julien and Jaggi [15]). Thus $\mathcal{X} = \mathcal{Y} := [0, 1]^d$ and the linear minimization oracle is simply $\text{LMO}(\cdot) = -0.5 \cdot (\text{sign}(\cdot) - \mathbf{1})$. We consider the following objective function:

$$\frac{\mu}{2}\|\boldsymbol{x} - \boldsymbol{x}^*\|_2^2 + (\boldsymbol{x} - \boldsymbol{x}^*)^\top M(\boldsymbol{y} - \boldsymbol{y}^*) - \frac{\mu}{2}\|\boldsymbol{y} - \boldsymbol{y}^*\|_2^2 \tag{8}$$

for which we can control the location of the saddle point $(\boldsymbol{x}^*, \boldsymbol{y}^*) \in \mathcal{X} \times \mathcal{Y}$. We generate a matrix M randomly as $M \sim \mathcal{U}([-0.1, 0.1]^{d \times d})$ and keep it fixed for all experiments. For the interior point setup (I), we set $(\boldsymbol{x}^*, \boldsymbol{y}^*) \sim \mathcal{U}([0.25, 0.75]^{2d})$, while we set $\boldsymbol{x}^*$ and $\boldsymbol{y}^*$ to some fixed random vertex of the unit cube for the setup (P). With all these parameters fixed, the constant ν is a function of μ only. We thus vary the strong convexity parameter μ to test various ν's.

We verify the linear convergence expected for the SP-FW algorithm for case (I) in Figure 1a, and for the SP-AFW algorithm for case (P) in Figure 1b. As the adaptive step size (and rate) depends linearly on ν, the linear rate becomes quite slow for small ν. In this regime (in red), the step size $2/(2 + k(t))$ (in orange) can actually perform better, despite its theoretical sublinear rate.

Finally, figure 1c shows that we can observe a linear convergence of SP-AFW even if ν is negative by using a different step size. In this case, we use the heuristic adaptive step size $\gamma_t := g_t/\tilde{C}$ where $\tilde{C} := LD_{\mathcal{X}}^2 + LD_{\mathcal{Y}}^2 + L_{XY}L_{YX}\left(D_{\mathcal{X}}^2/\mu_{\mathcal{X}} + D_{\mathcal{Y}}^2/\mu_{\mathcal{Y}}\right)$. Here $\tilde{C}$ takes into account the coupling between the concave and the convex variable and is motivated from a different proof of convergence that we were not able to complete. The empirical linear convergence in this case is not yet supported by a complete analysis, highlighting the need for more sophisticated arguments.

Conclusion. We proposed FW-style algorithms for saddle-point optimization with the same attractive properties as FW, in particular only requiring access to a LMO. We gave the first convergence result for a FW-style algorithm towards a saddle point over polytopes by building on the recent developments on the linear convergence analysis of AFW. However, our experiments let us believe that the condition $\nu > 0$ is not required for the convergence of FW-style algorithms. We thus conjecture that a refined analysis could yield a linear rate for the general uniformly strongly convex-concave functions in both cases (I) and (P), paving the way for further theoretical work.

Acknowledgments. Thanks to N. Ruozzi and A. Benchaouine for helpful discussions. Work supported in part by DARPA N66001-15-2-4026, N66001-15-C-4032 and NSF III-1526914, IIS-1451500, CCF-1302269.

References

[1] A. Ahmadinejad, S. Dehghani, Hajiaghayi, B. Lucier, H. Mahini, and S. Seddighin. From duels to battlefields: Computing equilibria of Blotto and other games. In *AAAI*, 2016.

[2] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski. Optimization with sparsity-inducing penalties. *Foundations and Trends in Machine Learning*, 2012.

[3] F. Barahona. On the computational complexity of Ising spin glass models. *J. Phys. A: Math. Gen.*, 1982.

[4] B. Cox, A. Juditsky, and A. Nemirovski. Decomposition techniques for bilinear saddle point problems and variational inequalities with affine monotone operators on domains given by linear minimization oracles. *arXiv preprint arXiv:1506.02444*, 2015.

[5] V. F. Demyanov and A. M. Rubinov. *Approximate methods in optimization problems*. Elsevier, 1970.

[6] J. Edmonds. Paths, trees and flowers. *Canadian Journal of Mathematics*, 1965.

[7] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Nav. Res. Logist. Q.*, 1956.

[8] G. Gidel, T. Jebara, and S. Lacoste-Julien. Frank-Wolfe algorithms for saddle point problems. *arXiv preprint arXiv:1610.07797*, 2016.

[9] J. H. Hammond. *Solving asymmetric variational inequality problems and systems of equations with generalized nonlinear programming algorithms*. PhD thesis, Massachusetts Institute of Technology, 1984.

[10] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex analysis and minimization algorithms I: Fundamentals*. Springer, 2013.

[11] M. Jaggi. *Sparse convex optimization methods for machine learning*. PhD thesis, ETH Zürich, 2011.

[12] M. Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *ICML*, 2013.

[13] A. Juditsky and A. Nemirovski. Solving variational inequalities with monotone operators on domains given by linear minimization oracles. *Mathematical Programming*, 2016.

[14] G. Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 1976.

[15] S. Lacoste-Julien and M. Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *NIPS*, 2015.

[16] G. Lan. The complexity of large-scale convex programming under a linear optimization oracle. *arXiv preprint arXiv:1309.5550*, 2013.

[17] T. Larsson and M. Patriksson. A class of gap functions for variational inequalities. *Math. Prog.*, 1994.

[18] A. F. Martins, N. A. Smith, P. M. Aguiar, and M. A. Figueiredo. Structured sparsity in structured prediction. In *EMNLP*, 2011.

[19] Y. Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 2007.

[20] M. Patriksson. *Nonlinear Programming and Variational Inequality Problems: A Unified Approach*. Springer, 1999.

[21] B. Taskar, S. Lacoste-Julien, and M. I. Jordan. Structured prediction, dual extragradient and Bregman projections. *Journal of Machine Learning Research*, 2006.

[22] J. Von Neumann and O. Morgenstern. *Theory of games and economic behavior*. Princeton press, 1944.

[23] N. Xiu and J. Zhang. Some recent advances in projection-type methods for variational inequalities. *Journal of Computational and Applied Mathematics*, 2003.

[24] D. L. Zhu and P. Marcotte. Convergence properties of feasible descent methods for solving variational inequalities in Banach spaces. *Computational Optimization and Applications*, 1998.