Randomized Block Frank-Wolfe for Convergent Large-Scale Learning

Liang Zhang, *Student Member, IEEE,* Gang Wang, *Student Member, IEEE,*
Daniel Romero, *Member, IEEE*, and Georgios B. Giannakis, *Fellow, IEEE*

***Abstract*—Owing to their low-complexity iterations, Frank-Wolfe (FW) solvers are well suited for various large-scale learning tasks. When block-separable constraints are present, randomized block FW (RB-FW) has been shown to further reduce complexity by updating only a fraction of coordinate blocks per iteration. To circumvent the limitations of existing methods, the present work develops step sizes for RB-FW that enable a flexible selection of the number of blocks to update per iteration while ensuring convergence and feasibility of the iterates. To this end, convergence rates of RB-FW are established through computational bounds on a primal sub-optimality measure and on the duality gap. The novel bounds extend the existing convergence analysis, which only applies to a step-size sequence that does not generally lead to feasible iterates. Furthermore, two classes of step-size sequences that guarantee feasibility of the iterates are also proposed to enhance flexibility in choosing decay rates. The novel convergence results are markedly broadened to encompass also nonconvex objectives, and further assert that RB-FW with exact line-search reaches a stationary point at rate $\mathcal{O}(1/\sqrt{t})$. Performance of RB-FW with different step sizes and number of blocks is demonstrated in two applications, namely charging of electrical vehicles and structural support vector machines. Extensive simulated tests demonstrate the impressive performance improvement of RB-FW relative to existing randomized single-block FW methods.**

***Index Terms*—Conditional gradient descent, nonconvex optimization, block coordinate, parallel optimization.**

I. Introduction

The Frank-Wolfe (FW) algorithm [1], also known as conditional gradient descent [2], has well-documented merits as a first-order solver especially for smooth constrained optimization tasks over convex compact sets. FW has recently received revived interest due to its simplicity and versatility in handling structured constraint sets in various signal processing and machine learning applications [3]. This growing popularity is due to its per-iteration simplicity that only entails minimizing a linear function over the feasible set, whereas competing first-order alternatives, such as projected gradient descent [4] and their accelerated versions [5], involve minimizing a quadratic function over the feasible set per iteration. Typically, solving a constrained linear optimization is considerably easier than finding the aforementioned projections per iteration. This is critical in diverse large-scale learning tasks, including matrix completion [6], multi-class classification [7], image reconstruction [7], structural support vector machines (SVMs) [8], particle filtering [9], and scheduling electric vehicle (EV) charging [10].

Despite the simplicity, FW can become prohibitively expensive when dealing with high-dimensional data. For this reason, *randomized* single-block FW has been advocated for solving large-scale convex constrained programs [8], where only a randomly selected block of variables is updated per iteration. At the price of obtaining the duality gap, convergence of randomized single-block FW has been improved [11]. Furthermore, randomized multiple-block FW was devised by updating multiple blocks per iteration in *parallel* [12]. Unfortunately, feasibility of the resulting iterates is in general not guaranteed by the original parallel randomized block (RB)-FW [12]. Moreover, all results on randomized FW focus on convex objectives, and convergence of RB-FW for nonconvex programs remained hitherto an open problem.

The present paper is the first to introduce a broad class of step sizes for RB-FW that offer: (i) guaranteed convergence and feasibility of the iterates along with (ii) flexibility to select a step-size sequence whose decay rate is attuned to the problem at hand. RB-FW with this rich class of step sizes subsumes the classical FW as well as the randomized single-block FW solvers as special cases. We further broaden the scope of RB-FW by allowing for nonconvex smooth objective functions. Specifically, we establish that RB-FW with typical step sizes attains a stationary point at rate $\mathcal{O}(1/\log t)$, whereas line-search-based step sizes enjoys an improved rate of $\mathcal{O}(1/\sqrt{t})$. Remarkably, the latter coincides with the rate afforded by the classical FW for nonconvex problems [13]. Finally, simulated tests on optimal coordination of EV charging and structural SVMs corroborate the merits of RB-FW with our novel step sizes relative to the single-block FW.

The remainder of this paper is organized as follows. Section II outlines the FW and RB-FW algorithms. Section III describes two novel families of step sizes for RB-FW, and establishes their feasibility and convergence. Section IV derives the RB-FW convergence rates for non-convex programs, whereas Section V highlights the implications of the results in Section III to the classical FW. Section VI shows the merits of RB-FW in two application settings, whereas Section VII tests the RB-FW performance numerically. Finally, Section VIII

Manuscript received December 26, 2016; revised May 27, 2017; accepted September 4, 2017. Paper no. T-SP-21600-2016. Work in this paper was supported in part by NSF grants 1423316, 1442686, 1508993, and 1509040.

L. Zhang, G. Wang, and G. B. Giannakis are with the Digital Technology Center and the Department of Electrical and Computer Engineering at the University of Minnesota, Minneapolis, MN 55455, USA. D. Romero is with the Department of Information and Communication Technology, University of Agder, Grimstad 4879, Norway. G. Wang is also with the State Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing 100081, P. R. China. E-mails: {zhan3523, gangwang, georgios}@umn.edu, daniel.romero@uia.no.

Color versions of one or more of the figures is this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier XXXXXX

Algorithm 1 Frank-Wolfe [1]

1: Initialize $t = 0$, $\mathbf{x}^0 \in \mathcal{X}$
2: **while** stopping_criterion not met **do**
3: Compute $\mathbf{s}^t = \arg\min_{\mathbf{s}\in\mathcal{X}} \ \mathbf{s}^\top \nabla f(\mathbf{x}^t)$
4: Update $\mathbf{x}^{t+1} = (1-\gamma_t)\mathbf{x}^t + \gamma_t \mathbf{s}^t$
5: $t \leftarrow t+1$
6: **end while**

concludes the paper.

Regarding common notation, lower- (upper-) case boldface letters represent column vectors (matrices). Sets are denoted by calligraphic letters, $|\mathcal{B}|$ stands for the cardinality of set $\mathcal{B}$, and $\mathcal{N} \setminus \mathcal{B} := \{x \in \mathcal{N} : x \notin \mathcal{B}\}$ denotes set difference. Symbol $^\top$ is reserved for transposition of vectors and matrices, whereas $\mathbf{0}$ and $\mathbf{1}$ denote the all-zero and all-one vectors of suitable dimensions, respectively. Operator $\lceil x \rceil$ gives the smallest integer greater than or equal to x, and $\log(x)$ returns the natural logarithm of x.

II. Preliminaries

The classical FW algorithm [1] aims at solving the following generic constrained optimization problem

$$\begin{aligned} \underset{\mathbf{x}\in\mathbb{R}^d}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x} \in \mathcal{X} \end{aligned} \tag{1}$$

where $f(\mathbf{x})$ is convex and differentiable, while the feasible set $\mathcal{X}$ is convex and compact. A number of problems in signal processing and machine learning, e.g., ridge regression or basis pursuit [14], can be expressed in this form. Listed as Algorithm 1, FW is initialized with a feasible $\mathbf{x}^0$. Given iterate $\mathbf{x}^t$, it then solves the following so-termed "linear oracle"

$$\mathbf{s}^t := \arg\min_{\mathbf{s}\in\mathcal{X}} \ \mathbf{s}^\top \nabla f(\mathbf{x}^t) \tag{2}$$

and uses a convex combination of $\mathbf{s}^t$ with $\mathbf{x}^t$ to obtain

$$\mathbf{x}^{t+1} = (1-\gamma_t)\mathbf{x}^t + \gamma_t \mathbf{s}^t \tag{3}$$

where the step size $\gamma_t \in (0,1]$ is typically selected as [3]

$$\gamma_t = \frac{2}{t+2} \ . \tag{4}$$

Alternatively, γ_t can be chosen via line search, which picks $\mathbf{x}^{t+1}$ as the best point on the line segment between $\mathbf{x}^t$ and $\mathbf{s}^t$:

$$\gamma_t = \arg\min_{0\le\gamma\le 1} f\left((1-\gamma)\mathbf{x}^t + \gamma \mathbf{s}^t\right). \tag{5}$$

In either case, Algorithm 1 converges at rate $\mathcal{O}(1/t)$ [3].

When d is large, updating all d entries of $\mathbf{x}$ at each t is computationally challenging. Randomized FW alleviates this difficulty by updating only a subset of the d entries [8], [12]. Splitting $\mathbf{x}$ into N_b blocks $\{\mathbf{x}_n\}_{n=1}^{N_b}$ with respective feasible sets $\{\mathcal{X}_n\}_{n=1}^{N_b}$ assumed convex and compact, (1) becomes

$$\begin{aligned} \underset{\mathbf{x}\in\mathbb{R}^d}{\text{minimize}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{x}_1 \in \mathcal{X}_1, \ \ldots, \ \mathbf{x}_{N_b} \in \mathcal{X}_{N_b} \end{aligned} \tag{6}$$

where $\mathbf{x}^\top := [\mathbf{x}_1^\top, \mathbf{x}_2^\top, \cdots, \mathbf{x}_{N_b}^\top]$. Note that if $N_b = 1$, then (6) boils down to (1).

Algorithm 2 Randomized Block Frank-Wolfe

1: Initialize $t = 0$, $\mathbf{x}^0 \in \mathcal{X}$
2: **while** stopping_criterion not met **do**
3: Randomly pick $\mathcal{B}_t \subseteq \mathcal{N}_b$ such that $|\mathcal{B}_t| = B$
4: Compute $\mathbf{s}_n^t = \underset{\mathbf{s}_n\in\mathcal{X}_n}{\arg\min} \ \mathbf{s}_n^\top \nabla_{\mathbf{x}_n} f(\mathbf{x}^t), \ \forall n \in \mathcal{B}_t$
5: Update

$$\mathbf{x}_n^{t+1} = \begin{cases} (1-\gamma_t)\mathbf{x}_n^t + \gamma_t \mathbf{s}_n^t, & \forall n \in \mathcal{B}_t \\ \mathbf{x}_n^t, & \forall n \in \mathcal{N}_b \setminus \mathcal{B}_t \end{cases}$$

6: $t \leftarrow t+1$
7: **end while**

The decomposition $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_{N_b}$ entails no loss of generality since any $\mathcal{X}$ can be expressed in this form by setting $N_b = 1$. It also emerges naturally in a number of applications, including the dual of structural SVMs [8], trace-norm regularized tensor completion [15], EV charging [10], the dual of group fused Lasso [16], and structured sub-modular minimization [17]. Thanks to the separable structure of $\mathcal{X}$, the linear oracle in (2) decouples across N_b blocks as

$$\mathbf{s}_n^t = \arg\min_{\mathbf{s}_n\in\mathcal{X}_n} \ \langle \mathbf{s}_n, \nabla_{\mathbf{x}_n} f(\mathbf{x}^t)\rangle, \quad n = 1, 2, \ldots, N_b \tag{7}$$

where $\nabla_{\mathbf{x}_n} f(\mathbf{x}^t)$ comprises the partial derivatives of $f(\mathbf{x})$ with respect to the entries of $\mathbf{x}_n$.

Instead of solving the N_b problems in (7), RB-FW reduces complexity by solving just B of them, where $B \in \{1, \ldots, N_b\}$ is a pre-selected constant. Let $\mathcal{N}_b := \{1, \ldots, N_b\}$ be the index set of all blocks, and let $\mathcal{B}_t$ be chosen at iteration t uniformly at random among all subsets of $\mathcal{N}_b$ with B elements. The RB-FW solver of (6) is summarized as Algorithm 2. To save computation time, step 4 of Algorithm 2 can be run in parallel [12] as illustrated in Fig. 1. In this case, B can be selected according to the number of physical processor cores in the control center.

The only step-size sequence for RB-FW available in the literature is [12]

$$\gamma_t = \frac{2\alpha}{\alpha^2 t + 2/N_b}, \quad t = 0, 1, \ldots \tag{8}$$

where $\alpha := B/N_b$ is the fraction of updated blocks. For $\alpha = 1$ and $N_b \neq 1$, note that (8) is different from (4); hence, FW is not generally a special case of the parallel RB-FW in [12]. Interestingly, Sec. III will introduce a family of RB-FW step sizes that subsumes the one in (4) as a special case.

As far as convergence of FW solvers, two quantities play instrumental roles. The first one is the curvature constant, which for a differentiable $f(\mathbf{x})$ over $\mathcal{X}$ is defined as [18], [3]

$$C_f := \sup_{\substack{\gamma\in[0,1] \\ \mathbf{x},\mathbf{s}\in\mathcal{X} \\ \mathbf{y}:=(1-\gamma)\mathbf{x}+\gamma\mathbf{s}}} \frac{2}{\gamma^2}\left[f(\mathbf{y}) - f(\mathbf{x}) - \langle \mathbf{y}-\mathbf{x}, \nabla f(\mathbf{x})\rangle\right]. \tag{9}$$

C_f is the least upper bound of a scaled difference between $f(\mathbf{y})$ and its linear approximation around $\mathbf{x}$. Throughout, C_f is assumed bounded. This property is closely related to the L-Lipschitz continuity of $\nabla f(\mathbf{x})$ over $\mathcal{X}$, which is defined as

$$\exists L > 0: \ \ \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{s})\| \le L\|\mathbf{x}-\mathbf{s}\|, \ \ \forall \mathbf{x}, \ \mathbf{s} \in \mathcal{X}. \tag{10}$$

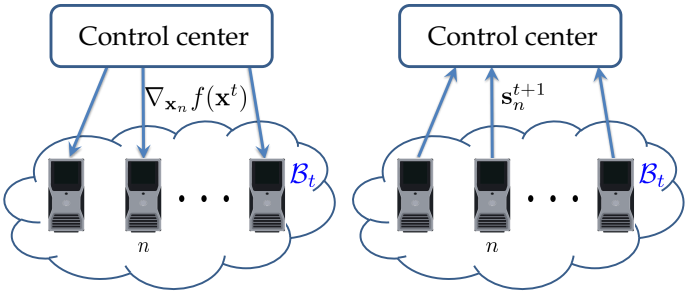


Fig. 1. Parallel implementation for Algorithm 2 at iteration $t \geq 0$. *Left:* The control center sends gradient $\nabla_{\mathbf{x}_n} f(\mathbf{x}^t)$ to processor $n \in \mathcal{B}_t$. *Right:* The updated $\{\mathbf{s}_n^{t+1}\}_{n\in\mathcal{B}_t}$ are sent to the control center.

If (10) holds, it is easy to check that [3, Appendix D]

$$C_f \leq LD_{\mathcal{X}}^2 \tag{11}$$

where $D_{\mathcal{X}} := \sup_{\mathbf{x},\mathbf{s}\in\mathcal{X}} \|\mathbf{x}-\mathbf{s}\|$ is the diameter of $\mathcal{X}$ that is finite for $\mathcal{X}$ compact. Equation (11) evidences that C_f is bounded whenever $\nabla f(\mathbf{x})$ is L-Lipschitz continuous over $\mathcal{X}$.

When it comes to RB-FW, the set curvature for an index set $\mathcal{B} \subseteq \mathcal{N}_b$ is commonly used instead of the constant C_f [12]

$$C_f^{\mathcal{B}} := \sup_{\substack{\gamma\in[0,1]\\ \mathbf{x}\in\mathcal{X}\\ \{\mathbf{s}_n\in\mathcal{X}_n\}_{n\in\mathcal{B}}}} \frac{2}{\gamma^2}\Big(f(\mathbf{y})-f(\mathbf{x})-\sum_{n\in\mathcal{B}}\langle \mathbf{y}_n-\mathbf{x}_n, \nabla_{\mathbf{x}_n} f(\mathbf{x})\rangle\Big) \tag{12}$$

where

$$\mathbf{y}_n := \begin{cases} (1-\gamma)\mathbf{x}_n + \gamma\mathbf{s}_n, & n \in \mathcal{B} \\ \mathbf{x}_n, & n \in \mathcal{N}_b \setminus \mathcal{B} \end{cases}$$

and $\mathbf{y}^\top := [\mathbf{y}_1^\top, \ldots, \mathbf{y}_{N_b}^\top]$. The *expected set curvature* for the uniformly randomly selected $\mathcal{B}$ can thus be expressed as

$$\bar{C}_f^B := \mathbb{E}_{\mathcal{B}}\left[C_f^{\mathcal{B}}\right] = \binom{N_b}{B}^{-1} \sum_{\{\mathcal{B}:\ \mathcal{B}\subseteq\mathcal{N}_b, |\mathcal{B}|=B\}} C_f^{\mathcal{B}} \tag{13}$$

with $\binom{N_b}{B} := N_b!/\big(B!(N_b-B)!\big)$. It is easy to verify that $\bar{C}_f^B \leq C_f$ by observing from (9) and (12) that $C_f^{\mathcal{B}} \leq C_f$, $\forall \mathcal{B} \subseteq \mathcal{N}_b$. Note however that $\bar{C}_f^B = C_f^{\mathcal{B}} = C_f$, when $B = N_b$.

The second quantity of interest is the so-termed *duality gap*

$$g(\mathbf{x}) := \sup_{\mathbf{s}\in\mathcal{X}} (\mathbf{x}-\mathbf{s})^\top \nabla f(\mathbf{x}), \quad \mathbf{x} \in \mathcal{X} \tag{14}$$

whose name stems from Fenchel duality; see [8, Appendix D], [3, Section 2]. Clearly, for the constrained problem (1), $\mathbf{x}$ is a stationary point if and only if $g(\mathbf{x}) = 0$. In addition, it holds that $g(\mathbf{x}) \geq 0$, $\forall \mathbf{x} \in \mathcal{X}$, since $(\mathbf{x}-\mathbf{s})^\top \nabla f(\mathbf{x}) = 0$ for $\mathbf{s} = \mathbf{x}$. Thus, $g(\mathbf{x})$ quantifies the distance of $\mathbf{x}$ from a stationary point of $f(\mathbf{x})$ [13].

III. Feasibility-Ensuring Step Sizes for RB-FW

To motivate the need for novel step sizes, this section starts by showing that γ_t in (8) does not guarantee feasibility of the iterates $\{\mathbf{x}^t\}$. It then introduces two families of feasibility-ensuring step size sequences, and proves that the iterates they generate are convergent for convex objectives. Moreover, these families are shown to offer a gamut of decay rates, thereby allowing for a flexible selection of the most suitable step size for a given problem.

With e.g., $N_b = 10^3$ and $\alpha = 2 \times 10^{-3}$, the step size in (8) will be $\gamma_t > 1$, $\forall t < 500$. As a result, step 5 of Algorithm 2 can generate infeasible iterates $\mathbf{x}^t$, which render RB-FW unstable since the gradient of the objective at the resulting $\mathbf{x}^t$ may not be even defined. For example, consider applying Algorithm 2 with $N_b = 100$, $B = 10$ and step size as in (8) to solve the smooth and convex program

$$\begin{aligned} \underset{\{x_n\}_{n=1}^{100}}{\text{minimize}} \quad & \sum_{n=1}^{100} x_n^2 - \log x_n \\ \text{subject to} \quad & 2 \leq x_n \leq 3, \quad n = 1, \ldots, 100. \end{aligned} \tag{15}$$

Initializing with $\{x_n^1 = 3\}_{n\in\mathcal{N}_b}$, it is easy to verify that $\{s_n^1 = 2\}_{n\in\mathcal{B}_1}$ and $\{x_n^2 = -11/3\}_{n\in\mathcal{B}_1}$, implying that $f(\mathbf{x}^2)$ and $\nabla f(\mathbf{x}^2)$ do not exist. Thus, the parallel RB-FW algorithm in [12], whose step size is given by (8), can not solve (15).

In a nutshell, existing step sizes do not guarantee feasibility of RB-FW iterates. Besides, the decay rates of existing step sizes can not be attuned to a given problem. To fill this gap, convergence analysis of RB-FW will be pursued first for a rich class of step sizes.

A. Convergence of RB-FW for convex programs

For randomized FW, convergence analysis typically focuses on $f(\mathbf{x}^t)$ and the $g(\mathbf{x}^t)$ in (14) [8], [12]. Let $\mathbf{x}^*$ denote one globally optimal solution of (6), and define the *primal sub-optimality* of point $\mathbf{x}^t$ as $h(\mathbf{x}^t) := f(\mathbf{x}^t) - f(\mathbf{x}^*)$. The next lemma, which quantifies the improvement of $h(\mathbf{x}^t)$ per iteration, will be handy in the ensuing convergence analysis.

Lemma 1. *If $\{\mathbf{x}^t\}_{t=0,1,\ldots}$ is generated by Algorithm 2 with an arbitrary predefined step-size sequence $\{\gamma_t\}_{t=0,1,\ldots}$ satisfying $\gamma_t \in [0,1]$ $\forall t$, then it holds that*

$$\mathbb{E}\left[h(\mathbf{x}^{t+1})\right] \leq \mathbb{E}\left[h(\mathbf{x}^t)\right] - \alpha\gamma_t \mathbb{E}\left[g(\mathbf{x}^t)\right] + \gamma_t^2 \bar{C}_f^B/2 \tag{16}$$

for $t \geq 0$, where the expectation is taken over $\{\mathcal{B}_\tau\}_{\tau=0}^t$.

A detailed proof can be found in [8], [12]; see also part A of the Appendix for an outline. Note that Lemma 1 can be applied regardless of whether $f(\mathbf{x})$ is convex or not.

Aiming to upper bound $\mathbb{E}\left[h(\mathbf{x}^t)\right]$, we will consider that $\{\gamma_t\}_{t=0,1,\ldots}$ satisfy

$$0 < \gamma_t \leq 1, \quad \forall t \geq 0 \tag{17a}$$

$$\frac{1-\alpha\gamma_{t+1}}{\gamma_{t+1}^2} \leq \frac{1}{\gamma_t^2}, \quad \forall t \geq 0. \tag{17b}$$

It can be easily seen that (17b) is equivalent to

$$\gamma_{t+1} \geq \frac{\gamma_t}{2}\left(\sqrt{\alpha^2\gamma_t^2 + 4} - \alpha\gamma_t\right)$$

which implies that (17b) limits how rapidly $\{\gamma_t\}_{t=0,1,\ldots}$ can decrease. Condition (17) is very general and subsumes existing step sizes as special cases. For example, if $B = N_b$, Algorithm 2 boils down to Algorithm 1, for which (4) is typically adopted [3]. For γ_t as in (4), it is clear that (17a) is satisfied, whereas (17b) follows from $(t+1)(t+3) \leq (t+2)^2$. Another

example arises if $B = 1$, in which case Algorithm 2 reduces to Algorithm 4 in [8]. The sequence

$$\gamma_t = \frac{2N_b}{t + 2N_b} \tag{18}$$

which was proposed in [8] for Algorithm 2, clearly satisfies (17a), and also (17b) since it holds that $(t+1+2N_b)^2-2(t+1+2N_b) \leq (t+2N_b)$. The step size (8) also satisfies (17b) since $(\alpha^2 t+\alpha^2+2/N_b)^2-2\alpha^2(\alpha^2 t+\alpha^2+2/N_b) \leq (\alpha^2 t+2/N_b)^2$, but fails to satisfy (17a), which ensures feasible iterates. However, upon observing that γ_t in (8) satisfies $\gamma_t \leq 1$ for $t \geq \tilde{t} := (2BN_b - 2N_b)/B^2$, one deduces that the shifted sequence

$$\tilde{\gamma}_t := \gamma_{t+\tilde{t}} = \frac{2}{\alpha t + 2} \tag{19}$$

does satisfy (17a), and therefore constitutes a feasible alternative to (8). Furthermore, it also satisfies (17b) because $(\alpha t + 2 + \alpha)(\alpha t + 2 - \alpha) \leq (\alpha t + 2)^2$.

To proceed with convergence rate analysis for a broad class of step sizes, an upper bound on $\mathbb{E}\left[h(\mathbf{x}^t)\right]$ for step sizes satisfying (17) will be developed.

Theorem 1 (Primal convergence)**.** *If $f(\mathbf{x})$ is convex and $\{\gamma_t\}_{t=0,1,\ldots}$ satisfies* (17)*, the iterates of Algorithm 2 satisfy*

$$\mathbb{E}\left[h(\mathbf{x}^t)\right] \leq \frac{1-\alpha\gamma_0}{\gamma_0^2}\gamma_{t-1}^2 h(\mathbf{x}^0) + \frac{t\bar{C}_f^B}{2}\gamma_{t-1}^2, \quad t \geq 1. \tag{20}$$

Proof. Since $f(\mathbf{x})$ is differentiable, convexity of $f(\mathbf{x})$ implies that

$$f(\mathbf{x}^t) - f(\mathbf{x}^*) \leq (\mathbf{x}^t - \mathbf{x}^*)^\top \nabla f(\mathbf{x}^t) \tag{21}$$

where $\mathbf{x}^*$ denotes any solution to (6). Combining (14) and (21) yields

$$g(\mathbf{x}^t) \geq f(\mathbf{x}^t) - f(\mathbf{x}^*) = h(\mathbf{x}^t) \geq 0. \tag{22}$$

Thus, $\mathbb{E}[g(\mathbf{x}^t)] \geq \mathbb{E}[h(\mathbf{x}^t)]$ and (16) can be rewritten as

$$\mathbb{E}\left[h(\mathbf{x}^{t+1})\right] \leq (1-\alpha\gamma_t)\mathbb{E}\left[h(\mathbf{x}^t)\right] + \gamma_t^2\bar{C}_f^B/2. \tag{23}$$

Dividing both sides of (23) by γ_t^2 gives rise to

$$\frac{1}{\gamma_t^2}\mathbb{E}\left[h(\mathbf{x}^{t+1})\right] \leq \frac{1-\alpha\gamma_t}{\gamma_t^2}\mathbb{E}\left[h(\mathbf{x}^t)\right] + \frac{\bar{C}_f^B}{2}. \tag{24}$$

Utilizing successively (17b) and (24) yields

$$\begin{aligned}\frac{1}{\gamma_t^2}\mathbb{E}[h(\mathbf{x}^{t+1})] &\leq \frac{1}{\gamma_{t-1}^2}\mathbb{E}[h(\mathbf{x}^t)] + \frac{1}{2}\bar{C}_f^B \\ &\leq \frac{1-\alpha\gamma_{t-1}}{\gamma_{t-1}^2}\mathbb{E}[h(\mathbf{x}^{t-1})] + \frac{1}{2}\bar{C}_f^B + \frac{1}{2}\bar{C}_f^B \\ &\leq \ldots \leq \frac{1-\alpha\gamma_0}{\gamma_0^2}h(\mathbf{x}^0) + \frac{t+1}{2}\bar{C}_f^B \end{aligned} \tag{25}$$

where the last inequality uses $\mathbb{E}[h(\mathbf{x}^0)] = h(\mathbf{x}^0)$. Therefore,

$$\mathbb{E}\left[h(\mathbf{x}^{t+1})\right] \leq \frac{1-\alpha\gamma_0}{\gamma_0^2}\gamma_t^2 h(\mathbf{x}^0) + \frac{t+1}{2}\gamma_t^2\bar{C}_f^B$$

which establishes (20). □

Theorem 1 generalizes existing results on the convergence of Algorithm 2, which apply only for specific step sizes either assume $B = 1$ [8] or $B = N_b$ [3]. Thus, Theorem 1 sheds light on step size design for arbitrary B by providing computational guarantees for Algorithm 2 with any step-size sequence satisfying (17).

Another quantity of interest to characterize the convergence of Algorithm 2 is $g(\mathbf{x}^t)$, which can be used to assess how close is $\mathbf{x}^t$ from being a solution [8], [12] since $g(\mathbf{x}^t) \geq h(\mathbf{x}^t)$; cf. (22). However, since finding upper bounds on $g(\mathbf{x}^t)$ is difficult [8], [13], [12], bounds on the minimal expected duality gap until iteration t, defined as [8], [12]

$$g_t := \min_{k\in\{0,1,\ldots t\}} \mathbb{E}\left[g(\mathbf{x}^k)\right] \tag{26}$$

are pursued next.

Theorem 2 (Primal-dual convergence)**.** *Let $\{\gamma_t\}_{t=0,1,\ldots}$ satisfy* (17) *and $\gamma_{t+1} \leq \gamma_t,\ \forall t \geq 0$. If $f(\mathbf{x})$ is convex and $\{\mathbf{x}^t\}_{t=1,2,\ldots}$ is generated by Algorithm 2, then for all $K \in \{1,\ldots,t\}$ it holds that*

$$g_t \leq \frac{\mathbb{E}\left[h(\mathbf{x}^K)\right]}{\alpha(t-K+1)\gamma_t} + \frac{\bar{C}_f^B\gamma_K^2}{2\alpha\gamma_t}\ . \tag{27}$$

Proof. Lemma 1 asserts that

$$\alpha\gamma_k\mathbb{E}[g(\mathbf{x}^k)] \leq \mathbb{E}\left[h(\mathbf{x}^k)\right] - \mathbb{E}\left[h(\mathbf{x}^{k+1})\right] + \gamma_k^2\bar{C}_f^B/2. \tag{28}$$

From $g_t \leq \mathbb{E}[g(\mathbf{x}^k)]$ and (28), it follows that

$$\begin{aligned}\alpha g_t\sum_{k=K}^{t}\gamma_k \leq& \alpha\sum_{k=K}^{t}\gamma_k\mathbb{E}[g(\mathbf{x}^k)] \\ \leq& \sum_{k=K}^{t}\left(\mathbb{E}\left[h(\mathbf{x}^k)\right] - \mathbb{E}\left[h(\mathbf{x}^{k+1})\right]\right) + \frac{\bar{C}_f^B}{2}\sum_{k=K}^{t}\gamma_k^2 \\ =& \mathbb{E}\left[h(\mathbf{x}^K)\right] - \mathbb{E}\left[h(\mathbf{x}^{t+1})\right] + \frac{\bar{C}_f^B}{2}\sum_{k=K}^{t}\gamma_k^2 \\ \leq& \mathbb{E}\left[h(\mathbf{x}^K)\right] + \frac{\bar{C}_f^B}{2}(t-K+1)\gamma_K^2 \end{aligned} \tag{29}$$

where the last inequality follows from $\mathbb{E}[h(\mathbf{x}^{t+1})] \geq 0$ and $\gamma_{t+1} \leq \gamma_t$. But since $\gamma_k \geq \gamma_t,\ \forall k \leq t$, one arrives at

$$\sum_{k=K}^{t}\gamma_k \geq (t-K+1)\gamma_t. \tag{30}$$

Finally, (27) follows after combining (29) with (30), and dividing both sides of the resulting inequality by $\alpha(t-K+1)\gamma_t$. □

Theorem 2 characterizes the primal-dual convergence of RB-FW for any non-increasing step size satisfying (17). Plugging (20) into (27) and fixing the step-size sequence yields an upper bound on g_t that can be minimized with respect to K to obtain the convergence rate of g_t. This approach will be pursued in Section III-B.

B. Proposed step sizes

This section develops two classes of step sizes obeying (17) for arbitrary values of B. Theorems 1 and 2 will then be invoked to derive the resulting convergence rates. To start with, consider the following general family of diminishing step-size sequences for fixed $q \in (0, \alpha]$ and decay rate $\rho \in (0.5, 1]$:

$$\boxed{\gamma_t = \frac{2}{qt^\rho + 2}, \quad \forall t \geq 0.} \tag{31}$$

As will be seen, this family includes, as special cases, the step sizes in (4), (18), and (19).

Lemma 2. *If* $\{\gamma_t\}_{t=0,1,\ldots}$ *is given by* (31)*, it satisfies* (17b).

Proof. See part B of the Appendix. □

Upon noticing that $0 < \gamma_t \leq 1$ and $\gamma_{t+1} \leq \gamma_t$ for $\{\gamma_t\}_{t=0,1,\ldots}$ in (31), the convergence rate of RB-FW can be derived by appealing to Theorems 1 and 2 as follows.

Corollary 1. *For convex* $f(\mathbf{x})$*, the iterates* $\{\mathbf{x}^t\}_{t=1,2,\ldots}$ *of Algorithm 2 with step size* (31) *satisfy*

$$\mathbb{E}\left[h(\mathbf{x}^t)\right] \leq \frac{4\left(1-\alpha\right)h(\mathbf{x}^0)}{\left[q(t-1)^\rho+2\right]^2} + \frac{2t\bar{C}_f^B}{\left[q(t-1)^\rho+2\right]^2} \tag{32}$$

and

$$g_t \leq \frac{(2\rho+1)^{2\rho+1}(qt^\rho+2)}{\alpha q^2(2\rho)^{2\rho}} \cdot \frac{(t+1)\bar{C}_f^B + 2(1-\alpha)h(\mathbf{x}^0)}{t^{2\rho+1}}. \tag{33}$$

Proof. See part C of the Appendix. □

Corollary 1 subsumes existing convergence results as special cases. Indeed, when $B = N_b$, one has that $\mathcal{B}_t = \mathcal{N}_b$ $\forall t$, which implies that $\bar{C}_f^B = C_f$, and Algorithm 2 reduces to the traditional FW solver. By selecting $q = 1$ and $\rho = 1$, the classical step size in (4) is retrieved. From Corollary 1, the resulting computational bounds are

$$h(\mathbf{x}^t) \leq \frac{2tC_f}{(t+1)^2} \leq \frac{2C_f}{t+2} \tag{34}$$

and

$$g_t \leq \frac{27C_f}{4} \cdot \frac{(t+1)(t+2)}{t^3}. \tag{35}$$

The resulting convergence rate of $h(\mathbf{x}^t)$ coincides with the one in [3, Theorem 1]. As for g_t, the bound in (35) is of the same order as that in [3, Theorem 2].

In addition, with $B = 1$, $q = 1/N_b$, and $\rho = 1$, the step size (18) proposed in [8] is recovered. From Corollary 1, the primal computational bound is

$$\begin{aligned}\mathbb{E}[h(\mathbf{x}^t)] &\leq \frac{4(N_b^2-N_b)h(\mathbf{x}^0)}{(t-1+2N_b)^2} + \frac{2tN_b^2C_f^1}{(t-1+2N_b)^2} \\ &\leq \frac{4(N_b^2-N_b)h(\mathbf{x}^0)}{(t-1+2N_b)^2} + \frac{2N_b^2C_f^1}{t+4N_b-2}\end{aligned} \tag{36}$$

where the last inequality follows from

$$\frac{t}{t+2N_b-1} \leq \frac{t+2N_b-1}{t+4N_b-2}.$$

Meanwhile, g_t is bounded by

$$g_t \leq \frac{27N_b(t+2N_b)}{4t^3}\left[(t+1)N_bC_f^1 + 2(N_b-1)h(\mathbf{x}^0)\right]. \tag{37}$$

Notably, the bound in (36) is *tighter* than the one reported in [8, Theorem 2], while the bound on g_t in (37) is of the same order as that in [8, Theorem 2].

Finally, note that Corollary 1 also characterizes convergence for the step size $\tilde{\gamma}_t$ in (19), since $\tilde{\gamma}_t$ is recovered from (31) upon setting $q = \alpha$ and $\rho = 1$.

The decreasing rates of the bounds in Theorems 1 and 2 are determined by the decay rates of the step size sequence. The faster γ_t diminishes, the more rapidly the upper bound in Theorem 1 vanishes. However, the sequence in (31) decreases at most as fast as $2/(\alpha t+2)$. To improve the bound in Theorem 1, a more rapidly vanishing sequence is proposed next. Specifically, consider the sequence

$$\boxed{\gamma_0 = 1, \text{ and } \gamma_{t+1} = \frac{\sqrt{\alpha^2\gamma_t^4 + 4\gamma_t^2} - \alpha\gamma_t^2}{2}, \quad \forall t \geq 0.} \tag{38}$$

It is then possible to establish the following.

Lemma 3 (Recursive step size)**.** *If* $\{\gamma_t\}_{t=0,1,\ldots}$ *is chosen as in* (38)*, it then holds that*

$$\frac{1}{\alpha t+1} \leq \gamma_t \leq \frac{2}{\alpha t+2}, \quad \forall t \geq 0 \tag{39a}$$

$$\gamma_{t+1} \leq \gamma_t, \quad \forall t \geq 0. \tag{39b}$$

Proof. See part D of the Appendix. □

The upper bound in (39a) confirms that the step size in (38) vanishes at least as fast as $\frac{2}{\alpha t+2}$. To check whether (38) meets (17), note that (17a) follows from (39a), whereas (38) implies that (17b) holds with equality. Because (38) satisfies (17) and (39b), the following computational bounds for (38) can be derived by plugging (39a) into Theorems 1 and 2.

Corollary 2. *For convex* $f(\mathbf{x})$*, the iterates* $\{\mathbf{x}^t\}_{t=1,2,\ldots}$ *of Algorithm 2 with step size as in* (38)*, satisfy*

$$\mathbb{E}\left[h(\mathbf{x}^t)\right] \leq \frac{4(1-\alpha)h(\mathbf{x}^0)}{(\alpha t+2-\alpha)^2} + \frac{2t\bar{C}_f^B}{(\alpha t+2-\alpha)^2} \tag{40}$$

and

$$g_t \leq \frac{27(\alpha t+1)}{2\alpha^3t^3}\left[(t+1)\bar{C}_f^B + 2(1-\alpha)h(\mathbf{x}^0)\right]. \tag{41}$$

Proof. See part E of the Appendix. □

To recap, this section put forth two families of step sizes for Algorithm 2 with arbitrary B, namely (31) and (38). Corollaries 1 and 2 establish convergence of Algorithm 2 for these step sizes, which also guarantee feasibility of the iterates since they satisfy (17a). When $\{\gamma_t\}_{t=0,1,\ldots}$ is given by (31) with $q = \alpha$ and $\rho = 1$ or when it is defined as in (38), the convergence rates of Algorithm 2 are in the order of $\mathcal{O}\left(1/t\right)$, thus matching those of the traditional FW algorithm.

Remark 1. The step size of RB-FW can also be chosen through line search, which prescribes

$$\gamma_t = \arg\min_{0\leq\gamma\leq1} f\left((1-\gamma)\mathbf{x}^t + \gamma\hat{\mathbf{s}}^t\right) \tag{42}$$

with $\hat{\mathbf{s}}^{t\top} := [\hat{\mathbf{s}}_1^{t\top}, \ldots, \hat{\mathbf{s}}_{N_b}^{t\top}]$ and

$$\hat{\mathbf{s}}_n^t := \begin{cases} \mathbf{s}_n^t, & n \in \mathcal{B}_t \\ \mathbf{x}_n^t, & n \in \mathcal{N}_b \setminus \mathcal{B}_t. \end{cases}$$

Let $\{\check{\mathbf{x}}^t\}_{t=0,1,\ldots}$ be the iterates generated by Algorithm 2 with γ_t given by (42). By (16) and (42), it holds that

$$\mathbb{E}\left[h(\check{\mathbf{x}}^{t+1})\right] \leq \mathbb{E}\left[h(\check{\mathbf{x}}^t)\right] - \alpha\gamma_t\mathbb{E}\left[g(\check{\mathbf{x}}^t)\right] + \gamma_t^2\bar{C}_f^B/2. \tag{43}$$

for any predefined step-size sequence $\{\gamma_t \in [0,1]\}$ [8], [12]. Particularly, (43) holds for $\{\gamma_t := 2/(\alpha t+2)\}_{t=0,1,\ldots}$. It can then be shown that $\{\check{\mathbf{x}}^t\}_{t=0,1,\ldots}$ satisfy for $t \geq 1$

$$\mathbb{E}\left[h(\check{\mathbf{x}}^t)\right] \leq \frac{4(1-\alpha)h(\mathbf{x}^0)}{(\alpha t+2-\alpha)^2} + \frac{2t\bar{C}_f^B}{(\alpha t+2-\alpha)^2}$$

and

$$\check{g}_t \leq \frac{27(\alpha t+2)}{4\alpha^3} \cdot \frac{(t+1)\bar{C}_f^B + 2(1-\alpha)h(\mathbf{x}^0)}{t^3}$$

where $\check{g}_t := \min_{k\in\{0,1,\ldots t\}} \mathbb{E}\left[g(\check{\mathbf{x}}^k)\right]$. The proof follows the steps of the one for Corollary 1. The convergence rate of line-search-based Algorithm 2 therefore remains in the order of $\mathcal{O}(1/t)$. Note however that extra computational cost is incurred for finding γ_t via (42).

Remark 2. At this point, it is worth discussing the choice of the step size leading to the fastest convergence in a given problem. Even though the bounds in this section suggest that the more rapid decrease of the step sizes, the quicker decrease of $h(\mathbf{x}^t)$, this is not always the case in practice. This is because step sizes with large decay rates become small after the first few iterations, and small step sizes lead to slow changes in $h(\mathbf{x}^t)$. Conversely, small decay rates tend to yield rapidly decreasing $h(\mathbf{x}^t)$ in the first few iterations since the step sizes remain relatively large. Hence, it is difficult to provide universal guidelines since rapidly or slowly diminishing step sizes may be preferred depending on the specific optimization problem at hand. For example, if optimal solutions lie in the interior of the feasible set, rapidly diminishing step sizes can help reduce oscillations around optimal solutions, thus improving the overall convergence rates. On the other hand, if $f(\mathbf{x})$ is monotone on $\mathcal{X}$, the solution lies on the boundary, which means that no oscillatory behavior is produced and hence, slowly diminishing step sizes will be preferable.

IV. RB-FW for Nonconvex Programs

The objective function of (6) is nonconvex in certain applications, such as constrained multilinear decomposition [19] and power system state estimation [20], [21]. Yet, convergence of RB-FW has never been investigated for this case. The rest of this section fills this gap by analyzing the convergence rate of RB-FW in problems involving a nonconvex objective. Similar to Sec. III, computational bounds are first derived for a wide class of step sizes, and are subsequently tailored for γ_t as in (31), and also for the exact line search.

Recall that Section II introduced $g(\mathbf{x})$ as a non-stationarity measure of point $\mathbf{x}$ with respect to $f(\mathbf{x})$. In the sequel, RB-FW with be analyzed in terms of upper bounds on g_t [cf. (26)].

Theorem 3. *If $\{\gamma_t\}_{t=0,1,\ldots}$ satisfy $0 \leq \gamma_t \leq 1$ $\forall t$, it holds for the iterates $\{\mathbf{x}^t\}_{t=0,1,\ldots}$ of Algorithm 2 that*

$$g_t \leq \frac{h(\mathbf{x}^0)}{\alpha\sum_{k=0}^t \gamma_k} + \frac{\bar{C}_f^B \sum_{k=0}^t \gamma_k^2}{2\alpha\sum_{k=0}^t \gamma_k}, \quad t \geq 0. \tag{44}$$

Proof. Using $0 \leq g_t \leq \mathbb{E}[g(\mathbf{x}^k)]$ and (28), we deduce that

$$\begin{aligned}
\alpha g_t \sum_{k=0}^t \gamma_k \leq& \alpha \sum_{k=0}^t \gamma_k \mathbb{E}[g(\mathbf{x}^k)] \\
\leq& \sum_{k=0}^t \Big(\mathbb{E}\big[h(\mathbf{x}^k)\big] - \mathbb{E}\big[h(\mathbf{x}^{k+1})\big]\Big) + (\bar{C}_f^B/2)\sum_{k=0}^t \gamma_k^2 \\
=& \mathbb{E}\left[h(\mathbf{x}^0)\right] - \mathbb{E}\left[h(\mathbf{x}^{t+1})\right] + (\bar{C}_f^B/2)\sum_{k=0}^t \gamma_k^2 \\
\leq& h(\mathbf{x}^0) + (\bar{C}_f^B/2)\sum_{k=0}^t \gamma_k^2
\end{aligned}$$

where the last inequality follows from $\mathbb{E}\left[h(\mathbf{x}^{t+1})\right] \geq 0$. Dividing both sides by $\alpha\sum_{k=0}^t \gamma_k$ leads to (44). □

Clearly, Theorem 3 affirms that $\lim_{t\to\infty} g_t = 0$ if the step-size sequence $\{\gamma_t\}_{t=0,1,\ldots}$ satisfies

$$\lim_{t\to\infty}\sum_{k=0}^t \gamma_k = \infty, \quad \text{and} \quad \lim_{t\to\infty}\sum_{k=0}^t \gamma_k^2 = S$$

for some finite $S > 0$. In other words, if $\{\gamma_t\}_{t=0,1,\ldots}$ is not summable and $\{\gamma_t^2\}_{t=0,1,\ldots}$ is summable, then either $\mathbf{x}_t$ is a stationary point for some t, or, a subsequence of $\{\mathbf{x}_t\}_{t=0,1,\ldots}$ converges to a stationary point.

For any given step size, the convergence rates of RB-FW can be derived through (44). To start with, consider $\{\gamma_t\}_{t=0,1,\ldots}$ in (31) with $q = \alpha,\ \rho = 1$; that is, $\gamma_t = 2/(\alpha t+2)$, and note that

$$\sum_{k=0}^t \frac{2}{\alpha k+2} \geq \int_{x=0}^t \frac{2}{\alpha x+2}dx = \frac{2}{\alpha}\log\Big(\frac{\alpha t+2}{2}\Big) \tag{45a}$$

$$\sum_{k=0}^t \frac{4}{(\alpha k+2)^2} \leq \int_{x=-1}^t \frac{4}{(\alpha x+2)^2}dx = \frac{4}{\alpha}\Big(\frac{1}{2-\alpha} - \frac{1}{\alpha t+2}\Big). \tag{45b}$$

By substituting (45) into Theorem 3, it follows that Algorithm 2 attains a stationary point of a nonconvex program at rate $\mathcal{O}(1/\log t)$.

This rather slow rate can be substantially improved upon adopting exact line search for RB-FW.

Theorem 4. *If $\{\gamma_t\}_{t=0,1,\ldots}$ is chosen as in* (42)*, it holds for the iterates $\{\mathbf{x}^t\}_{t=0,1,\ldots}$ of Algorithm 2 that*

$$g_t \leq \frac{\max\left\{2h(\mathbf{x}^0), \bar{C}_f^B\right\}}{\alpha\sqrt{t+1}}, \quad t \geq 0. \tag{46}$$

Proof. The right-hand side of (16) is minimized for

$$\begin{aligned}
\hat{\gamma}_k &= \arg\min_{\gamma\in[0,1]} \mathbb{E}\left[h(\mathbf{x}^k)\right] - \alpha\gamma\mathbb{E}\left[g(\mathbf{x}^k)\right] + \gamma^2\bar{C}_f^B/2 \\
&= \min\left\{1, \alpha\mathbb{E}\left[g(\mathbf{x}^k)\right]/\bar{C}_f^B\right\}.
\end{aligned} \tag{47}$$

Thus, if $\mathbb{E}\left[g(\mathbf{x}^k)\right] \geq \bar{C}_f^B/\alpha$, then $\hat{\gamma}_k = 1$ and (16) becomes

$$\begin{aligned}
\mathbb{E}\left[h(\mathbf{x}^{k+1})\right] &\leq \mathbb{E}\left[h(\mathbf{x}^k)\right] - \alpha\mathbb{E}\left[g(\mathbf{x}^k)\right] + \bar{C}_f^B/2 \\
&\leq \mathbb{E}\left[h(\mathbf{x}^k)\right] - \alpha\mathbb{E}\left[g(\mathbf{x}^k)\right]/2
\end{aligned} \tag{48}$$

where the second inequality follows from $\bar{C}_f^B \leq \alpha \mathbb{E}\left[g(\mathbf{x}^k)\right]$. Similarly, if $\mathbb{E}\left[g(\mathbf{x}^k)\right] < \bar{C}_f^B/\alpha$, then $\hat{\gamma}_k = \alpha \mathbb{E}\left[g(\mathbf{x}^k)\right]/\bar{C}_f^B$ and (16) becomes

$$\mathbb{E}\left[h(\mathbf{x}^{k+1})\right] \leq \mathbb{E}\left[h(\mathbf{x}^k)\right] - \alpha^2 \mathbb{E}\left[g(\mathbf{x}^k)\right]^2 / 2\bar{C}_f^B. \quad (49)$$

Combining both cases, (48) and (49) establish that

$$\mathbb{E}\left[h(\mathbf{x}^{k+1})\right] \leq \mathbb{E}\left[h(\mathbf{x}^k)\right] - \min\left\{\frac{\alpha \mathbb{E}\left[g(\mathbf{x}^k)\right]}{2}, \frac{\alpha^2 \mathbb{E}^2\left[g(\mathbf{x}^k)\right]}{2\bar{C}_f^B}\right\}. \quad (50)$$

When γ_k is given by (42) with $t = k$, $h(\mathbf{x}^{k+1})$ is not greater than when $\gamma_k = \hat{\gamma}_k$. Therefore, (50) still holds in the former case. Thus, for $\{\gamma_k\}_{k=0,1,\ldots}$ as in (42), it follows that

$$\min\left\{\frac{\alpha \mathbb{E}\left[g(\mathbf{x}^k)\right]}{2}, \frac{\alpha^2 \mathbb{E}\left[g(\mathbf{x}^k)\right]^2}{2\bar{C}_f^B}\right\} \leq \mathbb{E}\left[h(\mathbf{x}^k)\right] - \mathbb{E}\left[h(\mathbf{x}^{k+1})\right]. \quad (51)$$

Summing (51) from $k = 0$ to t yields

$$(t+1)\min\left\{\frac{\alpha g_t}{2}, \frac{\alpha^2 g_t^2}{2\bar{C}_f^B}\right\} \leq h(\mathbf{x}^0) - \mathbb{E}\left[h(\mathbf{x}^{t+1})\right]. \quad (52)$$

Therefore,

$$g_t \leq \max\left\{\frac{2h(\mathbf{x}^0)}{\alpha(t+1)}, \frac{\sqrt{2\bar{C}_f^B h(\mathbf{x}^0)}}{\alpha\sqrt{t+1}}\right\}. \quad (53)$$

Since $t+1 \geq \sqrt{t+1}$ and $\sqrt{2\bar{C}_f^B h(\mathbf{x}^0)} \leq \max\{2h(\mathbf{x}^0), \bar{C}_f^B\}$, (46) holds. □

Theorem 4 generalizes the recent result in [13], which only applies to the classical FW method. The improved bound in (46) is attained at the price of performing exact line search, which requires the solution to a potentially nonconvex univariate optimization subproblem (42). It is worth mentioning that an optimal solution to this subproblem can be readily found in a number of cases. For example, if $f((1-\gamma)\mathbf{x}^t + \gamma \mathbf{s}^t)$ is quadratic in γ, then γ_t can be readily found by evaluating this function at three points.

All in all, the main contribution here is convergence rate analysis of RB-FW for minimizing (6) with nonconvex $f(\mathbf{x})$. Interestingly, when RB-FW relies on step sizes obtained using line search, a stationary point is reached with rate $\mathcal{O}(1/\sqrt{t})$.

V. Generalized Step Sizes for FW

The availability of satisfactory step sizes for FW is rather limited. Indeed, besides line search, convergence rate of FW has only been established for $\gamma_t = \frac{2}{t+2}$ [3], and $\gamma_t = \frac{1}{t+1}$ [22]. This limits the user's control on convergence of FW iterates; cf. Remark 2. To alleviate this limitation, this section examines the usage of step sizes in (31) and (38) in the classical FW solver, namely Algorithm 1. Since the latter can be viewed as a special case of Algorithm 2 with $B = N_b$, Corollaries 1 and 2 can be leveraged to derive the convergence rates of FW for convex programs with the novel step sizes. Specifically, the ensuing computational bounds hold.

Corollary 3. *If $f(\mathbf{x})$ is convex and the step size sequence $\{\gamma_t\}_{t=0,1,\ldots}$ is chosen as in* (31) *with $\alpha = 1$, $q \in (0,1]$ and $\rho \in (0.5, 1]$, then the successive iterates $\{\mathbf{x}^t\}_{t=1,2,\ldots}$ of Algorithm 1 satisfy for $t \geq 1$*

$$h(\mathbf{x}^t) \leq \frac{2tC_f}{\left[q(t-1)^\rho + 2\right]^2} \quad (54)$$

and

$$g_t \leq \frac{(2\rho+1)^{2\rho+1}(qt^\rho+2)}{q^2(2\rho)^{2\rho}} \frac{(t+1)C_f}{t^{2\rho+1}}. \quad (55)$$

Proof. This is a special case of Corollary 1 for $\alpha = 1$. □

Corollary 4. *If $f(\mathbf{x})$ is convex and the step-size sequence $\{\gamma_t\}_{t=0,1,\ldots}$ is chosen as in* (38) *with $\alpha = 1$, then the successive iterates $\{\mathbf{x}^t\}_{t=1,2,\ldots}$ of Algorithm 1 satisfy for $t \geq 1$*

$$h(\mathbf{x}^t) \leq \frac{2C_f}{t+2} \quad (56)$$

and

$$g_t \leq \frac{27C_f}{2}\left(\frac{1}{t} + \frac{2}{t^2} + \frac{1}{t^3}\right). \quad (57)$$

Proof. Corollary 4 follows directly from Corollary 2. □

Corollaries 3 and 4 establish convergence rates in terms of both $h(\mathbf{x}^t)$ and g_t for the classical FW method with step sizes of different decay rates. For a given problem, the most suitable step size can be selected following the guidelines in Remark 2. Interestingly, comparing Corollaries 3 and 4 with Corollaries 1 and 2 reveals that the initial optimality gap $h(\mathbf{x}^0)$ no longer affects the bounds for FW.

VI. Applications

Two applications where RB-FW exhibits significant computational advantages over existing alternatives will be delineated in this section.

A. Coordination of EV charging

The convex setup of optimal schedules for EV charging in [23] is briefly reviewed next. Suppose that a load aggregator coordinates the charging of N EVs over the T consecutive time slots $\mathcal{T} := \{1, \ldots, T\}$ of length Δ_τ. Let $\mathcal{T}_n \subseteq \mathcal{T}$ denote the time slots in which vehicle n is connected to the power grid, and let $p_n(\tau)$ be the charging rate of EV n at time τ to be scheduled by the load aggregator. If $\bar{p}_n$ is the charging rate limitation imposed by the battery of vehicle n, then $p_n(\tau)$ should lie in the interval $[0, \bar{p}_n(\tau)]$ with

$$\bar{p}_n(\tau) := \begin{cases} \bar{p}_n, & \tau \in \mathcal{T}_n, \\ 0, & \text{otherwise.} \end{cases}$$

The charging profile for vehicle n, denoted by $\mathbf{p}_n^\top := [p_n(1), \cdots, p_n(T)]$, should therefore belong to the convex and compact set

$$\mathcal{P}_n := \left\{\mathbf{p}_n : \Delta_\tau \mathbf{p}_n^\top \mathbf{1} = R_n,\ 0 \leq p_n(\tau) \leq \bar{p}_n(\tau),\ \forall \tau \in \mathcal{T}\right\}$$

where R_n represents the total energy needed by EV n.

Given $\{R_n\}_{n=1}^N$, $\{\bar{p}_n\}_{n=1}^N$, and $\{\mathcal{T}_n\}_{n=1}^N$, the problem solved by the aggregator is to find the charging profiles minimizing its electricity cost [23]; that is,

$$\begin{aligned} \mathbf{p}^* \in \arg\min_{\mathbf{p}} \quad & f(\mathbf{p}) \\ \text{subject to} \quad & \mathbf{p}_n \in \mathcal{P}_n,\ \forall\ n \in \mathcal{N} \end{aligned} \tag{58}$$

where $\mathbf{p}^\top := [\mathbf{p}_1^\top, \cdots, \mathbf{p}_N^\top]$ and $\mathcal{N} := \{1, \ldots, N\}$. With $\{D(\tau)\}_{\tau=1}^T$ denoting additional known loads, the total cost $f(\mathbf{p})$ is

$$f(\mathbf{p}) = \sum_{\tau=1}^{T} \Big(D(\tau) + \sum_{n=1}^{N} p_n(\tau) \Big)^2. \tag{59}$$

Note that $f(\mathbf{p})$ is convex but *not strongly convex* in $\mathbf{p}$. The feasible set for (58) is the Cartesian product $\mathcal{P} := \mathcal{P}_1 \times \ldots \times \mathcal{P}_N$, which is convex and compact. Thus, problem (58) is convex and of the form (6).

Assuming that the aggregator can only afford updating the charging profiles of B out of the N vehicles in parallel due to a limited number of processors, the ensuing B linear subproblems arise when solving (58) via Algorithm 2:

$$\mathbf{s}_n^t \in \arg\min_{\mathbf{s}_n \in \mathcal{P}_n} \langle \mathbf{s}_n, \mathbf{c}^t \rangle, \quad n \in \mathcal{B}_t \tag{60}$$

where $|\mathcal{B}_t| = B$ and $\mathbf{c}^t := \nabla_{\mathbf{p}_n} f(\mathbf{p}^t)$. The latter does not depend on n since the gradient $\nabla_{\mathbf{p}_n} f(\mathbf{p}^t)$ is identical across the N vehicles. Its τ-th entry is given by

$$c^t(\tau) := 2\big(D(\tau) + \sum_{n=1}^{N} p_n^t(\tau)\big). \tag{61}$$

The subproblem (60) can be solved in closed form [24]. To find a solution, sort the entries of $\mathbf{c}^t$ in non-decreasing order by finding $\{\tau_i^t\}_{i=1}^T$ such that $c^t(\tau_1^t) \leq c^t(\tau_2^t) \leq \ldots \leq c^t(\tau_T^t)$. Subsequently, one needs to find the index $\bar{\tau}_n^t \geq 1$ for which

$$\sum_{i=1}^{\bar{\tau}_n^t - 1} \bar{p}_n(\tau_i^t) \leq R_n \text{ and } \sum_{i=1}^{\bar{\tau}_n^t} \bar{p}_n(\tau_i^t) > R_n. \tag{62}$$

Finally, the entries of the minimizer $\mathbf{s}_n^t$ are found as

$$s_n^t(\tau_i^t) = \begin{cases} \bar{p}_n(\tau_i^t), & i = 1, \ldots, \bar{\tau}_n^t - 1 \\ R_n - \sum_{j=1}^{\bar{\tau}_n^t} \bar{p}_n(\tau_j^t), & i = \bar{\tau}_n^t \\ 0, & i = \bar{\tau}_n^t + 1, \ldots, T. \end{cases} \tag{63}$$

The computational advantage of RB-FW for solving (58) stems from the fact that the solution to the subproblems (60) can be obtained efficiently via (63) upon receiving the $\mathbf{c}_t$ entry order, whereas competing alternatives require projections onto $\{\mathcal{P}_n\}_{n \in \mathcal{B}_t}$ per iteration [10]. Our RB-FW-based charging scheme is summarized in Algorithm 3.

Algorithm 3 EV charging coordination solver

Input: $\{R_n\}_{n=1}^N$, $\{\bar{p}_n\}_{n=1}^N$, $\{\mathcal{T}_n\}_{n=1}^N$, and B

Initialize $\{\mathbf{p}_n^0\}$ and $t = 0$
while stopping_criterion not met **do**
  Randomly pick $\mathcal{B}_t \subseteq \mathcal{N}$ such that $|\mathcal{B}_t| = B$
  Evaluate $\mathbf{c}^t$ via (61) and broadcast $\mathbf{c}^t$ entry order
  Calculate $\{\mathbf{s}_n^t\}_{n \in \mathcal{B}_t}$ via (62) and (63)
  Update $\{\mathbf{p}_n^{t+1}\}_{n \in \mathcal{N}}$ via

$$\mathbf{p}_n^{t+1} = \begin{cases} (1 - \gamma_t)\mathbf{p}_n^t + \gamma_t \mathbf{s}_n^t, & \forall n \in \mathcal{B}_t \\ \mathbf{p}_n^t, & \forall n \in \mathcal{N} \setminus \mathcal{B}_t \end{cases}$$

  $t \leftarrow t + 1$
end while

B. Structural SVMs

The term *structured prediction* comprises a family of machine learning problems, where the output to the *predictors* have variable sizes [25]. An example is the optical character recognition (OCR) task, where one is given a vector $z \in \mathbb{R}^P$ containing the P-pixel image of an M-letter word. The goal is to produce a vector $\mathbf{y} \in \{1, \ldots, 26\}^M$, whose m-th entry indicates which of the 26 letters of the alphabet corresponds to the m-th character in that word. This problem is challenging because the output $\mathbf{y}$ may take 26^M values, and also the same predictor must work for different values of M.

Structural SVMs have been widely adopted to carry out the aforementioned structured prediction tasks [26], [27]. Upon defining the application-dependent feature map $\boldsymbol{\phi}$ [27] that encodes the relevant information for the input-output pair $(\mathbf{z}, \mathbf{y})$ in the d-dimensional vector $\boldsymbol{\phi}(\mathbf{z}, \mathbf{y})$, a vector $\mathbf{w}$ is learned so that $\langle \mathbf{w}, \boldsymbol{\phi}(\mathbf{z}, \mathbf{y}) \rangle$ when seen as a function of $\mathbf{y}$ is maximized at the correct $\mathbf{y}$ for a given input $\mathbf{z}$. Given N training pairs $\{(\mathbf{z}_n, \mathbf{y}_n)\}_{n=1}^N$, $\mathbf{w}$ is learned by solving

$$\underset{\mathbf{w}, \boldsymbol{\xi}}{\text{minimize}} \quad \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{N}\sum_{n=1}^{N} \xi_n \tag{64a}$$

$$\text{subject to} \quad \langle \mathbf{w}, \boldsymbol{\psi}_n(\tilde{\mathbf{y}}) \rangle \geq L_n(\tilde{\mathbf{y}}) - \xi_n \tag{64b}$$
$$\forall \tilde{\mathbf{y}} \in \mathcal{Y}_n,\ \forall n \in \mathcal{N}$$

where $\mathcal{N} := \{1, \ldots, N\}$, $\boldsymbol{\psi}_n(\tilde{\mathbf{y}}) := \boldsymbol{\phi}(\mathbf{z}_n, \mathbf{y}_n) - \boldsymbol{\phi}(\mathbf{z}_n, \tilde{\mathbf{y}})$, $L_n(\tilde{\mathbf{y}})$ is the incurred loss by predicting $\tilde{\mathbf{y}}$ instead of the given label $\mathbf{y}_n$, $\{\xi_n\}_{n=1}^N$ are slack variables, λ is a nonnegative constant, and $\mathcal{Y}_n$ is the set of all possible outputs for input $\mathbf{z}_n$. In the OCR example, $\mathcal{Y}_n = \{1, \ldots, 26\}^{M_n}$, where M_n is the number of characters of the n-th word.

Problem (64) is difficult since the number of constraints explodes with $|\mathcal{Y}_n|$. If $\beta_n(\tilde{\mathbf{y}})$ is the Lagrange dual variable associated with (64b), vector $\boldsymbol{\beta}_n$ is formed with entries $\{\beta_n(\tilde{\mathbf{y}})\}_{\tilde{\mathbf{y}} \in \mathcal{Y}_n}$, and vector $\boldsymbol{\beta}$ has entries $\{\boldsymbol{\beta}_n\}_{n \in \mathcal{N}_b}$, the dual of (64) is [8]

$$\begin{aligned} \underset{\substack{\boldsymbol{\beta} \in \mathbb{R}^m \\ \boldsymbol{\beta} \geq 0}}{\text{minimize}} \quad & f(\boldsymbol{\beta}) := \frac{\lambda}{2}\|\mathbf{A}\boldsymbol{\beta}\|^2 - \mathbf{b}^\top \boldsymbol{\beta} \\ \text{subject to} \quad & \mathbf{1}^\top \boldsymbol{\beta}_n = 1, \quad \forall n \in \mathcal{N} \end{aligned} \tag{65}$$

where $m := \sum_n |\mathcal{Y}_n|$, $\mathbf{A} \in \mathbb{R}^{d \times m}$ is formed with columns $\{\frac{1}{\lambda N}\boldsymbol{\psi}_n(\tilde{\mathbf{y}}) \in \mathbb{R}^d |\ \tilde{\mathbf{y}} \in \mathcal{Y}_n, n \in \mathcal{N}\}$, and vector $\mathbf{b} \in \mathbb{R}^m$ has entries $\{\frac{1}{N} L_n(\tilde{\mathbf{y}})\}_{\tilde{\mathbf{y}} \in \mathcal{Y}_n, n \in \mathcal{N}}$.

A randomized single-block FW is adopted by [8], to solve (65). Extending this approach to $B > 1$ yields Algorithm 4. To avoid storing the high-dimensional vector $\boldsymbol{\beta}^t$, auxiliary variables $\tilde{\mathbf{w}}^t := \mathbf{A}\boldsymbol{\beta}^t$, $t = 0, 1, \ldots$ are introduced. It can be shown that iterates $\{\tilde{\mathbf{w}}^t\}_{t=0,1\ldots}$ converge to the global minimizer of (64); see [8] for additional details.

Algorithm 4 Structural SVMs solver

Input: $\{(\mathbf{z}_n, \mathbf{y}_n)\}_{n=1}^N$, $\{\mathcal{Y}_n\}_{n=1}^N$, and B

Initialize $\boldsymbol{\beta}^0$, $\ell^0 = \ell_1^0 = \ldots = \ell_{N_b}^0 = 0$, and $t = 0$

Calculate $\tilde{\mathbf{w}}^0 = \tilde{\mathbf{w}}_1^0 = \ldots = \tilde{\mathbf{w}}_{N_b}^0 = \mathbf{A}\boldsymbol{\beta}^0$

while stopping_criterion not met **do**

  Randomly pick $\mathcal{B}_t \subseteq \mathcal{N}$ such that $|\mathcal{B}_t| = B$

  for $n \in \mathcal{B}_t$ **do**

    Compute

$$\mathbf{y}_n^* := \arg\max_{\mathbf{y}\in\mathcal{Y}_n} \; L_n(\mathbf{y}) - \langle \tilde{\mathbf{w}}^t, \boldsymbol{\psi}_n(\mathbf{y})\rangle$$

    Update $\tilde{\mathbf{w}}_n^{t+1} = (1-\gamma_t)\tilde{\mathbf{w}}_n^t + \frac{\gamma_t}{\lambda N}\boldsymbol{\psi}_n(\mathbf{y}_n^*)$

    Update $\ell_n^{t+1} = (1-\gamma_t)\ell_n^t + \frac{\gamma_t}{N} L_n(\mathbf{y}_n^*)$

    Update $\tilde{\mathbf{w}}^{t+1} = \tilde{\mathbf{w}}^t + \tilde{\mathbf{w}}_n^{t+1} - \tilde{\mathbf{w}}_n^t$

    Update $\ell^{t+1} = \ell^t + \ell_n^{t+1} - \ell_n^t$

  end for

  $t \leftarrow t+1$

end while

VII. Simulated Tests

This section demonstrates the efficacy of the novel step sizes, and our parallel RB-FW solvers, in the context of the large-scale applications of Sec. VI.

A. Coordination of EV charging

In the first experiment, 63 EVs with maximum charging power $\bar{p}_n = 3.45$ kW $\forall n$, were scheduled. The simulation comprises $T = 96$ time slots ranging from 12:00 pm to 12:00 pm of the next day. The values of $\{\mathcal{T}_n\}_{n=1}^N$ and $\{R_n\}_{n=1}^N$ were set according to real travel data of the National Household Travel Survey [28], [10]. The base load $\{D_\tau\}_{\tau=1}^T$ were obtained by averaging the 2014 residential load data from Southern California Edison [29].

Convergence is assessed in terms of the relative error $\epsilon(\mathbf{p}^t) := (f(\mathbf{p}^t) - f(\mathbf{p}^*))/f(\mathbf{p}^*)$, where $\mathbf{p}*$ is obtained using the off-the-shelf solver SeDuMi.

The following step-size sequences were compared.

$$\begin{aligned}
&\text{(S1)}: \quad \gamma_t := \frac{2}{\alpha t + 2} \qquad (66)\\
&\text{(S2)}: \quad \gamma_t := \frac{\sqrt{\alpha^2\gamma_{t-1}^4 + 4\gamma_{t-1}^2} - \alpha\gamma_{t-1}^2}{2}, \;\; \gamma_0 = 1\\
&\text{(S3)}: \quad \gamma_t := \frac{2}{0.5\alpha t + 2}\\
&\text{(S4)}: \quad \gamma_t := \frac{2}{0.5\alpha t^{0.9} + 2}\\
&\text{(S5)}: \quad \gamma_t := \frac{2}{0.5\alpha t^{0.8} + 2}.
\end{aligned}$$

S2 is the sequence in (38), whereas S1 and S3-S5 are special cases of (31). Sequences S1-S5 cover a wide range of decay rates. S2 vanishes faster than S1 [cf. (39a)], whereas the decay rates of S3-S5 are smaller than that of S1. Note that S1 boils down to the step size in (18) when setting $B = 1$. For all $n = 1, \ldots, N$, $\mathbf{p}_n^0$ was initialized as

$$p_n^0(\tau) = \begin{cases} \bar{p}_n(\tau), & \tau = 1, \ldots, \bar{\tau}_n^0 - 1\\ R_n - \sum_{j=1}^{\bar{\tau}_n^0} \bar{p}_n(j), & \tau = \bar{\tau}_n^0\\ 0, & \tau = \bar{\tau}_n^0 + 1, \ldots, T \end{cases}$$

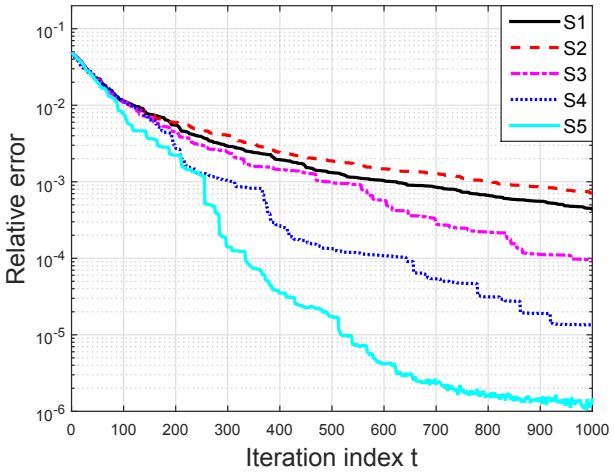


Fig. 2. Convergence performance of Algorithm 3 with $B = 1$.

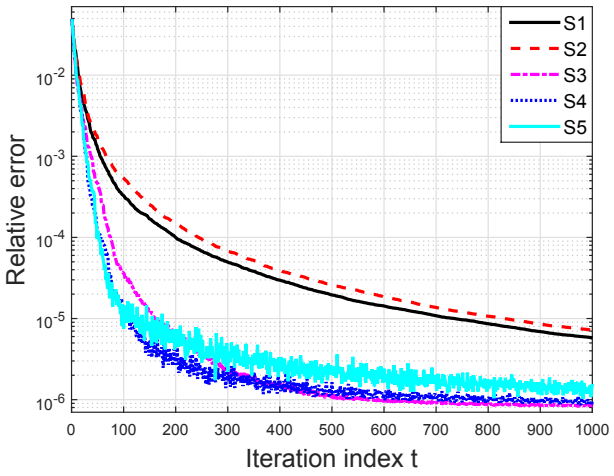


Fig. 3. Convergence performance of Algorithm 3 with $B = 10$.

where the index $\bar{\tau}_n^0 \geq 1$ was found as

$$\sum_{\tau=1}^{\bar{\tau}_n^0 - 1} \bar{p}_n(\tau) \leq R_n \text{ and } \sum_{\tau=1}^{\bar{\tau}_n^0} \bar{p}_n(\tau^0) > R_n.$$

The first experiment assumed that only one vehicle was randomly selected to update its charging profile per iteration. Algorithm 3 with $B = 1$ was run with the step sizes S1-S5 for 1,000 iterations. Fig. 2 depicts the evolution of $\epsilon(\mathbf{p}^t)$ across the iteration index t for Algorithm 3 with step sizes S1-S5 when $B = 1$. It is observed that the algorithm converges towards a global minimum for all the tested step sizes. In this scenario, the more slowly the step size diminishes, the faster the relative error decreases. Since $B = 1$ and Algorithm 3 is a special instance of Algorithm 2, Fig. 2 therefore highlights how randomized single-block FW can benefit from the proposed step sizes. Specifically, the proposed step sizes S3-S5 lead to a much faster convergence than S1, which coincides with the step size in (18) since $B = 1$.

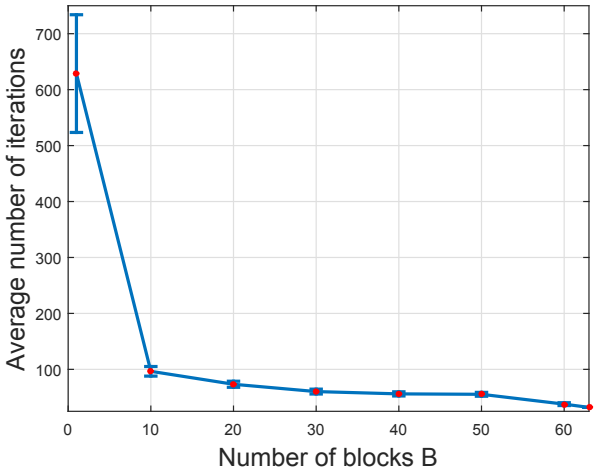


Fig. 4. Number of iterations to achieve $\epsilon(\mathbf{p}^t) \leq 10^{-5}$.

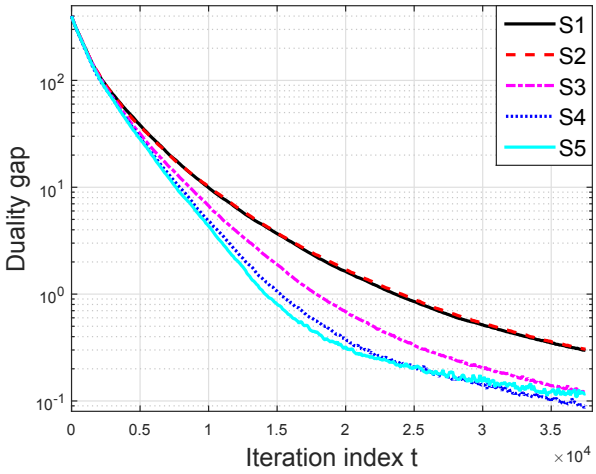


Fig. 6. Progress of $g(\boldsymbol{\beta}^t)$ for Algorithm 4 with $B = 1$.

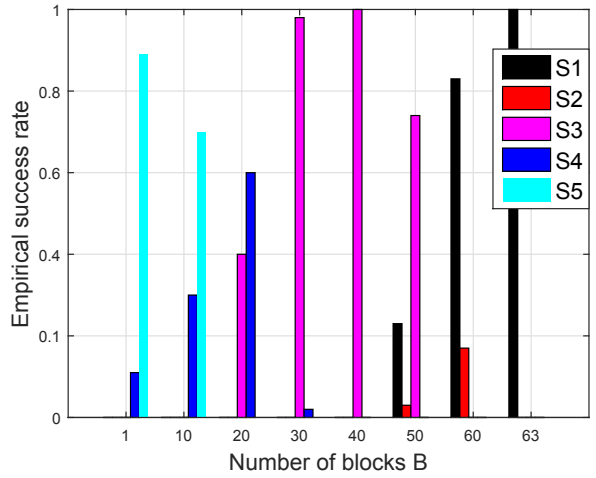


Fig. 5. Empirical success rate for S1-S5 with different values of B.

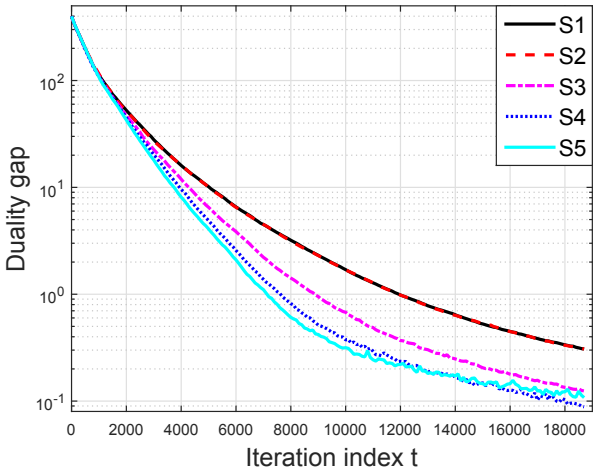


Fig. 7. Progress of $g(\boldsymbol{\beta}^t)$ for Algorithm 4 with $B = 2$.

The second experiment tested Algorithm 3 with $B = 10$. Fig. 3 further confirms that slowly diminishing step sizes lead to fast convergence in the first few iterations. However, as the iterates approach a minimum, the slowly diminishing step sizes yield larger oscillations; see e.g. S5 in Fig. 3. This phenomenon has already been described in Remark 2. Comparing Figs. 2 and 3 reveals that considerably less iterations are required to achieve a target accuracy for larger B. For example, about one fifth of iterations are now required for Algorithm 3 with S5 to reach $\epsilon(\mathbf{p}) \leq 10^{-5}$. Thus, if the ten blocks can be processed in parallel, setting $B = 10$ roughly reduces the computation time by a factor of five, which further corroborates the merits of parallel RB-FW.

The next experiment highlights the impact of B on the convergence of Algorithm 3. Five copies of Algorithm 3, each one with a different step size S1-S5, are executed for 100 independent trials. The minimum value of t such that at least one of these copies satisfies $\epsilon(\mathbf{p}^t) \leq 10^{-5}$ is recorded. Fig. 4 represents the sample mean and standard derivation of this minimum t averaged over the 100 trials for different values of B. It is observed that both mean and standard derivation decrease for increasing B. If each iteration of Algorithm 3 is run in B cores in parallel, then the number of iterations constitutes a proxy for runtime. Fig. 4 adopts this proxy to showcase the benefit of adopting $B > 1$. Nonetheless, observe that the influence of B on the number of iterations decreases for large B. Fig. 5 depicts the fraction of trials that each copy of Algorithm 3 is the first among the five copies in achieving $\epsilon(\mathbf{p}^t) \leq 10^{-5}$. This figure reveals that slowly diminishing step sizes are preferable for small values of B.

B. Structural SVMs

The structural SVMs experiment was conducted on a subset of the OCR dataset [26], [30]. The feature mapping $\boldsymbol{\phi}(\mathbf{z}, \mathbf{y})$, loss function $L_n(\tilde{\mathbf{y}})$, and solution to the subproblems in step 5 of Algorithm 4 were evaluated using the open source code [31] released by the authors in [8]. The dimension of $\boldsymbol{\phi}(\mathbf{z}, \mathbf{y})$ is $d = 4,028$, and the number of training examples is $N = 6,251$. To initialize each $\boldsymbol{\beta}_n^0$, one of its entries chosen

uniformly at random was set to one, whereas all the remaining entries were set to zero. Algorithm 4 with $\lambda = 0.1$ and step sizes S1-S5 was run for six passes through all the training examples. The duality gap $g(\boldsymbol{\beta}^t)$ in (14) is depicted in Figs. 6 and 7 for $B = 1$ and $B = 2$, respectively. In both cases, Algorithm 4 with S5 outperforms all other variants in the first few iterations. Furthermore, it can be seen that the required number of iterations to achieve a target accuracy almost halves when increasing B from one to two.

VIII. Concluding Summary

The RB-FW algorithm is especially suited to solve high-dimensional constrained learning problems whose feasible set is block separable. For convex programs, the present contribution developed feasibility-ensuring step sizes that enable parallel updates of provably convergent RB-FW iterates. The novel step sizes admit various decay rates, leading to flexible convergence rates of RB-FW. Convergence of RB-FW is also established for constrained nonconvex problems. Numerical tests using real-world datasets corroborated the speed-up advantage of parallel RB-FW with the proposed step sizes over randomized single-block FW. In addition, single-block FW with the developed slowly diminishing step sizes converges markedly faster than that with existing step sizes.

Appendix

A. Proof of Lemma 1

Using (12) together with steps 4 and 5 of Algorithm 2, we find

$$\begin{aligned} f(\mathbf{x}^{t+1}) &\leq f(\mathbf{x}^t) + \sum_{n\in\mathcal{B}_t} \langle \mathbf{x}_n^{t+1} - \mathbf{x}_n^t, \nabla_{\mathbf{x}_n} f(\mathbf{x}^t)\rangle + \gamma_t^2 C_f^{\mathcal{B}_t}/2 \\ &= f(\mathbf{x}^t) + \sum_{n\in\mathcal{B}_t} \gamma_t \langle \mathbf{s}_n^t - \mathbf{x}_n^t, \nabla_{\mathbf{x}_n} f(\mathbf{x}^t)\rangle + \gamma_t^2 C_f^{\mathcal{B}_t}/2. \end{aligned}$$

Subtracting $f(\mathbf{x}^*)$ from both sides yields

$$h(\mathbf{x}^{t+1}) \leq h(\mathbf{x}^t) + \sum_{n\in\mathcal{B}_t} \gamma_t \langle \mathbf{s}_n^t - \mathbf{x}_n^t, \nabla_{\mathbf{x}_n} f(\mathbf{x}^t)\rangle + \gamma_t^2 C_f^{\mathcal{B}_t}/2.$$

Taking conditional expectation with respect to $\mathcal{B}_t$, we arrive for a given $\mathbf{x}^t$ at

$$\begin{aligned} &\mathbb{E}_{\mathcal{B}_t}\left[h(\mathbf{x}^{t+1})|\mathbf{x}^t\right] \\ &\leq h(\mathbf{x}^t) + \alpha \sum_{n\in\mathcal{N}_b} \gamma_t \langle \mathbf{s}_n^t - \mathbf{x}_n^t, \nabla_{\mathbf{x}_n} f(\mathbf{x}^t)\rangle + \gamma_t^2 \bar{C}_f^B/2 \\ &= h(\mathbf{x}^t) + \alpha\gamma_t \langle \mathbf{s}^t - \mathbf{x}^t, \nabla f(\mathbf{x}^t)\rangle + \gamma_t^2 \bar{C}_f^B/2 \\ &= h(\mathbf{x}^t) - \alpha\gamma_t g(\mathbf{x}^t) + \gamma_t^2 \bar{C}_f^B/2 \end{aligned} \tag{67}$$

where the last equality follows from (14) and step 4 of Algorithm 2. Since $\mathbf{x}^t$ is determined by $\{\mathcal{B}_\tau\}_{\tau=0}^{t-1}$, taking expectations in (67) with respect to $\{\mathcal{B}_\tau\}_{\tau=0}^{t-1}$ yields (16).

B. Proof of Lemma 2

Plugging (31) into the left-hand side of (17b) yields

$$\begin{aligned} \frac{1-\alpha\gamma_{t+1}}{\gamma_{t+1}^2} &= \frac{[q(t+1)^\rho + 2 - \alpha]^2 - \alpha^2}{4} \\ &\leq \frac{[q(t+1)^\rho + 2 - \alpha]^2}{4} \\ &\leq \frac{[q(t+1)^\rho - q + 2]^2}{4}. \end{aligned} \tag{68}$$

where the last inequality follows from $q \leq \alpha \leq 1$. Consider the auxiliary function $\varphi(x) := (x+c)^\rho - x^\rho - c, \quad x \geq 0$ for some constant $c \geq 1$, and its first-order derivative

$$\varphi'(x) = \rho(x+1)^{\rho-1} - \rho x^{\rho-1}, \quad x \geq 0.$$

Since $\rho \leq 1$, it holds that $\varphi'(x) \leq 0$, and thus,

$$\varphi(x) \leq \varphi(0) = c^\rho - c \leq 0, \quad \forall x \geq 0$$

or,

$$(x+c)^\rho - c \leq x^\rho, \quad \forall x \geq 0. \tag{69}$$

Multiplying both sides of (69) by q, and setting $c = 1$ and $x = t$ gives rise to

$$0 \leq q(t+1)^\rho - q \leq qt^\rho, \quad \forall t \geq 0. \tag{70}$$

Combining (68) and (70) yields

$$\frac{1-\alpha\gamma_{t+1}}{\gamma_{t+1}^2} \leq \frac{[qt^\rho + 2]^2}{4} = \frac{1}{\gamma_t^2}$$

which concludes the proof.

C. Proof of Corollary 1

Expression (32) follows directly by substituting (31) into (20). To show (33), apply Theorem 2 to verify that

$$\begin{aligned} \alpha g_t &\leq \frac{\mathbb{E}\left[h(\mathbf{x}^K)\right]}{\gamma_0^2(t-K+1)\gamma_t} + \frac{\bar{C}_f^B \gamma_K^2}{2\gamma_t} \\ &\leq \frac{(1-\alpha\gamma_0)\gamma_{K-1}^2 h(\mathbf{x}^0)}{\gamma_0^2(t-K+1)\gamma_t} + \frac{K\bar{C}_f^B\gamma_{K-1}^2}{2(t-K+1)\gamma_t} + \frac{\bar{C}_f^B\gamma_K^2}{2\gamma_t} \\ &\leq \frac{(1-\alpha)\,\gamma_{K-1}^2 h(\mathbf{x}^0)}{(t-K+1)\gamma_t} + \frac{\gamma_{K-1}^2\bar{C}_f^B(t+1)}{2\gamma_t(t-K+1)} \\ &\leq \frac{\gamma_{K-1}^2}{t-K+1} \cdot \frac{(t+1)\bar{C}_f^B + 2(1-\alpha)h(\mathbf{x}^0)}{2\gamma_t} \end{aligned} \tag{71}$$

for all $K \in \{1, \ldots, t\}$, where the second inequality stems from (20) and the third one follows from $\gamma_K \leq \gamma_{K-1}$ and $\gamma_0 = 1$. The next step is to bound the first quotient in the right-hand side of (71). To this end, set $c = 2/q$ and $x = K-1$ in (69) to deduce that

$$\gamma_{K-1} = \frac{2}{q(K-1)^\rho + 2} \leq \frac{2}{q(K-1+2/q)^\rho}. \tag{72}$$

Now set $K = \lceil \mu(t+2/q) \rceil$, where μ is an arbitrary constant. Since $K \in \{1, \ldots, t\}$, μ needs to satisfy

$$0 < \mu \leq \frac{t}{t+2/q}. \tag{73}$$

Since

$$\lceil \mu(t+2/q) \rceil - 1 + 2/q \geq \mu(t+2/q) - 2 + 2/q \geq \mu(t+2/q) > 0$$

it follows from (72) that

$$\gamma_{K-1} \leq \frac{2}{q\mu^{\rho}(t+2/q)^{\rho}}.$$

Therefore,

$$\begin{aligned}\frac{\gamma_{K-1}^2}{t-K+1} \leq & \frac{4}{q^2(t-K+1)\mu^{2\rho}(t+2/q)^{2\rho}} \\ \leq & \frac{4}{q^2[t-\mu(t+2/q)]\mu^{2\rho}(t+2/q)^{2\rho}}. \end{aligned} \tag{74}$$

Minimizing the right-hand side with respect to μ in the interval (73) yields

$$\frac{\gamma_{K-1}^2}{t-K+1} \leq \frac{4(2\rho+1)^{2\rho+1}}{q^2(2\rho)^{2\rho}t^{2\rho+1}} \tag{75}$$

for $\mu = \frac{2\rho}{2\rho+1}\frac{t}{t+2/q}$.

From (71), g_t can be upper bounded as

$$\begin{aligned} g_t \leq & \frac{\gamma_{K-1}^2}{t-K+1} \cdot \frac{(t+1)\bar{C}_f^B + 2(1-\alpha)h(\mathbf{x}^0)}{2\alpha\gamma_t} \\ \leq & \frac{(2\rho+1)^{2\rho+1}(qt^{\rho}+2)}{\alpha q^2(2\rho)^{2\rho}} \cdot \frac{(t+1)\bar{C}_f^B + 2(1-\alpha)h(\mathbf{x}^0)}{t^{2\rho+1}}. \end{aligned}$$

where the second inequality follows from (75) and (31).

D. Proof of Lemma 3

To prove (39a) by induction, it clearly holds for $t = 0$, and assume that it holds also for a fixed $t \geq 0$. Then, one needs to show that

$$\frac{1}{\alpha t+1+\alpha} \leq \gamma_{t+1} \leq \frac{2}{\alpha t+2+\alpha}. \tag{76}$$

To this end, define the auxiliary function

$$\hat{\varphi}(x) := \frac{\sqrt{\alpha^2x^4+4x^2}-\alpha x^2}{2}, \quad x \geq 0$$

which is monotonically increasing since

$$\hat{\varphi}'(x) = \frac{(\alpha^2x^2+2)-\alpha x\sqrt{\alpha^2x^2+4}}{\sqrt{\alpha^2x^2+4}} > 0.$$

Thus, by the induction hypothesis we have

$$\hat{\varphi}\left(\frac{1}{\alpha t+1}\right) \leq \gamma_{t+1} = \hat{\varphi}(\gamma_t) \leq \hat{\varphi}\left(\frac{2}{\alpha t+2}\right). \tag{77}$$

Note that

$$1 - \frac{2\alpha}{\alpha t+2+\alpha} \leq \left(1-\frac{\alpha}{\alpha t+2+\alpha}\right)^2 = \left(\frac{\alpha t+2}{\alpha t+2+\alpha}\right)^2$$

or, equivalently,

$$\begin{aligned} 1 \leq & \frac{2\alpha}{\alpha t+2+\alpha} + \left(\frac{\alpha t+2}{\alpha t+2+\alpha}\right)^2 \\ = & \left(\frac{\alpha}{\alpha t+2} + \frac{\alpha t+2}{\alpha t+2+\alpha}\right)^2 - \frac{\alpha^2}{(\alpha t+2)^2}. \end{aligned} \tag{78}$$

This inequality implies that

$$\begin{aligned}\hat{\varphi}\left(\frac{2}{\alpha t+2}\right) = & \frac{1}{\alpha t+2}\sqrt{\frac{4\alpha^2}{(\alpha t+2)^2}+4} - \frac{2\alpha}{(\alpha t+2)^2} \\ \leq & \frac{2}{\alpha t+2+\alpha}. \end{aligned} \tag{79}$$

Combining (79) with the second inequality in (77) proves the second inequality in (76).

On the other hand, since $(\alpha t+1+\alpha)^2 \geq \alpha(\alpha t+1+\alpha) + (\alpha t+1)^2$, it holds that

$$\begin{aligned}\frac{\alpha^2}{(\alpha t+1)^2}+4 \geq & \frac{\alpha^2}{(\alpha t+1)^2} + \frac{4\alpha}{\alpha t+1+\alpha} + \frac{4(\alpha t+1)^2}{(\alpha t+1+\alpha)^2} \\ = & \left(\frac{\alpha}{\alpha t+1} + \frac{2(\alpha t+1)}{\alpha t+1+\alpha}\right)^2. \end{aligned} \tag{80}$$

Thus,

$$\begin{aligned}\hat{\varphi}\left(\frac{1}{\alpha t+1}\right) = & \frac{1}{2(\alpha t+1)}\sqrt{\frac{\alpha^2}{(\alpha t+1)^2}+4} - \frac{\alpha}{2(\alpha t+1)^2} \\ \geq & \frac{1}{\alpha t+1+\alpha}. \end{aligned} \tag{81}$$

Combining (81) with the first inequality in (77) proves the first inequality in (76), thus concluding the proof of (39a).

To prove (39b), one can also proceed by induction. First, $\gamma_1 \leq \gamma_0$ since $\frac{\sqrt{\alpha^2+4}-\alpha}{2} \leq 1$. Assuming $\gamma_{t-1} \leq \gamma_t$, it follows that $\gamma_t \leq \gamma_{t+1}$ since $\hat{\varphi}(x)$ is nondecreasing.

E. Proof of Corollary 2

Inequality (40) readily follows from (20) and (39a). To prove (41), note that (71) holds because of $\gamma_0 = 1$ and (39). Meanwhile, by the second inequality in (39a), the step size in (38) satisfies (75) for $q = \alpha$ and $\rho = 1$, that is

$$\frac{\gamma_{K-1}^2}{t-K+1} \leq \frac{27}{\alpha^2t^3}. \tag{82}$$

Plugging (39a) together with (82) into (71), yields (41).

REFERENCES

[1] M. Frank and P. Wolfe, "An algorithm for quadratic programming," *Naval Research Logistics Quarterly*, vol. 3, no. 1–2, pp. 95–110, 1956.

[2] V. F. Deminaov and A. M. Rubinov, *Approximate Methods in Optimization Problems*. Amsterdam, Netherlands: Elsevier, 1970.

[3] M. Jaggi, "Revisiting Frank-Wolfe: Projection-free sparse convex optimization," in *Proc. Intl. Conf. on Machine Learning*, Atlanta, GA, Jun. 2013.

[4] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 1999.

[5] Y. Nesterov, *Introductory Lectures on Convex Optimization*. Boston, MA: Kluwer, 2004.

[6] M. Jaggi and M. Sulovsk, "A simple algorithm for nuclear norm regularized problems," in *Proc. Intl. Conf. on Machine Learning*, Haifa, Israel, Jun. 2010.

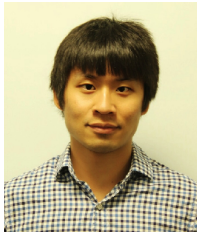
[7] Z. Harchaoui, A. Juditsky, and A. Nemirovski, "Conditional gradient algorithms for norm-regularized smooth convex optimization," *Math. Program.*, vol. 152, no. 1-2, pp. 75–112, 2015.

[8] S. Lacoste-Julien, M. Jaggi, M. Schmidt, and P. Pletscher, "Block-coordinate Frank-Wolfe optimization for structural SVMs," in *Proc. Intl. Conf. on Machine Learning*, Atlanta, GA, Jun. 2013.

[9] S. Lacoste-Julien, F. Lindsten, and F. R. Bach, "Sequential kernel herding: Frank-Wolfe optimization for particle filtering." in *Intl. Conf. Artificial Intelligence and Statistics*, San Diego, CA, May 2015.

[10] L. Zhang, V. Kekatos, and G. B. Giannakis, "Scalable electric vehicle charging protocols," *IEEE Trans. Power Syst.*, vol. 32, no. 2, pp. 1451–1462, Mar. 2017.
[11] A. Osokin, J.-B. Alayrac, I. Lukasewitz, P. K. Dokania, and S. Lacoste-Julien, "Minding the gaps for block Frank-Wolfe optimization of structured SVMs," in *Proc. Intl. Conf. on Machine Learning*, New York City, NY, Jun. 2016.
[12] Y. Wang, V. Sadhanala, W. Dai, W. Neiswanger, S. Sra, and E. P. Xing, "Parallel and distributed block-coordinate Frank-Wolfe algorithms," in *Proc. Intl. Conf. on Machine Learning*, New York City, NY, Jun. 2016.
[13] S. Lacoste-Julien, "Convergence rate of Frank-Wolfe for non-convex objectives," 2016. [Online]. Available: https://arxiv.org/abs/1607.00345
[14] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, Feb. 2001.
[15] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 208–220, Jan. 2013.
[16] C. M. Alaíz, Á. Barbero, and J. R. Dorronsoro, "Group fused lasso," in *Proc. Intl. Conf. on Artificial Neural Networks*, Sofia, Bulgaria, Mar. 2013, pp. 66–73.
[17] S. Jegelka, F. Bach, and S. Sra, "Reflection methods for user-friendly submodular optimization," in *Adv. Neural Inf. Process. Syst.*, Stateline, NV, Dec. 2013, pp. 1313–1321.
[18] K. L. Clarkson, "Coresets, sparse greedy approximation, and the Frank-Wolfe algorithm," *ACM Trans. Algorithms*, vol. 6, no. 4, p. 63, July 2010.
[19] E. E. Papalexakis, N. D. Sidiropoulos, and R. Bro, "From k-means to higher-way co-clustering: Multilinear decomposition with sparse latent factors," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 493–506, Dec. 2013.
[20] G. B. Giannakis, V. Kekatos, N. Gatsis, S.-J. Kim, H. Zhu, and B. Wollenberg, "Monitoring and optimization for power grids: A signal processing perspective," *IEEE Signal Processing Mag.*, vol. 30, no. 5, pp. 107–128, Sep. 2013.
[21] G. Wang, A. S. Zamzam, G. B. Giannakis, and N. D. Sidiropoulos, "Power system state estimation via feasible point pursuit: Algorithms and Crmér-Rao bound," *arXiv:1705.04031*, 2017.
[22] R. M. Freund and P. Grigas, "New analysis and results for the Frank–Wolfe method," *Math. Program.*, vol. 155, no. 1-2, pp. 199–230, Jan. 2016.
[23] L. Zhang, V. Kekatos, and G. B. Giannakis, "A generalized Frank-Wolfe approach to decentralized control of vehicle charging," in *Proc. Conf. on Decision and Control*, Las Vegas, NV, Dec. 2016.
[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY: Cambridge University Press, 2004.
[25] G. Bakır, T. Hofmann, B. Schölkopf, A. J. Smola, B. Taskar, and S. V. Vishwanathan, *Predicting Structured Data*. Cambridge, MA: MIT press, 2007.
[26] B. Taskar, C. Guestrin, and D. Koller, "Max-margin Markov networks," in *Adv. Neural Inf. Process. Syst.*, Vancouver, Canada, Dec. 2003.
[27] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *J. Mach. Lear. Res.*, vol. 6, pp. 1453–1484, Sep. 2005.
[28] Federal Highway Administration. US Department of Transportation. [Online]. Available: http://nhts.ornl.gov/2009/pub/stt.pdf
[29] Southern California Edison dynamic load profiles. Southern California Edison. [Online]. Available: https://www.sce.com/wps/portal/home/regulatory/load-profiles/
[30] OCR dataset. Stanford University. [Online]. Available: http://ai.stanford.edu/~btaskar/ocr/
[31] S. Lacoste-Julien, M. Jaggi, M. Schmidt, and P. Pletscher. Block-coordinate Frank-Wolfe solver for structural SVMs. [Online]. Available: https://github.com/ppletscher/BCFWstruct

Liang Zhang (S'13) received his B.Sc. and M.Sc. degrees in Electrical Engineering from Shanghai Jiao Tong University in 2012 and 2014, respectively. Since 2014, he has been working toward his Ph.D. degree in the Department of Electrical and Computer Engineering at the University of Minnesota. His research interests include the areas of large-scale optimization, and high-dimensional learning.



Gang Wang (S'12) received the B.Eng. degree in Electrical Engineering and Automation from the Beijing Institute of Technology, Beijing, China, in 2011. He is currently a Ph.D. candidate in the Department of Electrical and Computer Engineering at the University of Minnesota. His research interests focus on the areas of high-dimensional statistical learning, and stochastic (non-)convex optimization with applications to smart power grids and machine learning.

He received a National Scholarship from China in 2014, and a Best Student Paper Award at the 2017 European Signal Processing Conference.

Daniel Romero (M'16) received his M.Sc. and Ph.D. degrees in Signal Theory and Communications from the University of Vigo, Spain, in 2011 and 2015, respectively. From Jul. 2015 to Nov. 2016, he was a post-doctoral researcher with the Digital Technology Center and Department of Electrical and Computer Engineering, University of Minnesota, USA. In Dec. 2017, he joined the Department of Information and Communication Technology, University of Agder, Norway, as an associate professor. His main research interests lie in the areas of signal processing, communications, and machine learning.

Georgios B. Giannakis (Fellow'97) received his Diploma in Electrical Engr. from the Ntl. Tech. Univ. of Athens, Greece, 1981. From 1982 to 1986 he was with the Univ. of Southern California (USC), where he received his MSc. in Electrical Engineering, 1983, MSc. in Mathematics, 1986, and Ph.D. in Electrical Engr., 1986. He was with the University of Virginia from 1987 to 1998, and since 1999 he has been a professor with the Univ. of Minnesota, where he holds an Endowed Chair in Wireless Telecommunications, a University of Minnesota McKnight Presidential Chair in ECE, and serves as director of the Digital Technology Center.

His general interests span the areas of communications, networking and statistical signal processing - subjects on which he has published more than 400 journal papers, 700 conference papers, 25 book chapters, two edited books and two research monographs (h-index 127). Current research focuses on learning from Big Data, wireless cognitive radios, and network science with applications to social, brain, and power networks with renewables. He is the (co-) inventor of 30 patents issued, and the (co-) recipient of 8 best paper awards from the IEEE Signal Processing (SP) and Communications Societies, including the G. Marconi Prize Paper Award in Wireless Communications. He also received Technical Achievement Awards from the SP Society (2000), from EURASIP (2005), a Young Faculty Teaching Award, the G. W. Taylor Award for Distinguished Research from the University of Minnesota, and the IEEE Fourier Technical Field Award (2015). He is a Fellow of EURASIP, and has served the IEEE in a number of posts, including that of a Distinguished Lecturer for the IEEE-SP Society.