Dynamic Online Learning via Frank-Wolfe Algorithm

Deepak S. Kalhan, Amrit S. Bedi, Alec Koppel, Ketan Rajawat, Hamed Hassani, Abhishek Gupta, and Adrish Banerjee .

***Abstract*—Online optimization is a framework for learning when training sets are large-scale or dynamic, and has grown essential as data has proliferated. In this setting, a complex optimization problem is broken down into a sequence of smaller problems, each of which must be solved with limited information and in the presence of distributional drift. To ensure safe model adaption or to avoid overfitting, constraints are often imposed, which are classically addressed with high complexity projections. To avoid this bottleneck, we propose a projection-free scheme based on Frank-Wolfe: instead of online gradient steps, we use steps that are collinear with the gradient but guaranteed to be feasible. We establish performance in terms of dynamic regret, which quantifies cost accumulation as compared with the optimal at each individual time slot. Specifically, for convex losses, we establish $\mathcal{O}(T^{1/2})$ dynamic regret up to metrics of non-stationarity. We relax the algorithm's required information to only noisy gradient estimates, i.e., partial feedback. To improve performance under partial feedback, we propose to use the 'Meta-Frank Wolfe' which uses multiple samples per step, and characterize its dynamic regret. Experiments on matrix completion problem and background separation in video demonstrate favorable performance of the proposed scheme.**

***Index Terms*—Online learning, Frank-Wolfe algorithm, convex optimization, gradient descent.**

I. Introduction

Many learning problems may be formulated as complex data-dependent optimization problems, as in the design of methods for speech recognition [1], perception [2], and increasingly, locomotion [3]. These technologies upend several orthodoxies in the design of optimization algorithms: finite time performance is prioritized, updates must be memory-efficient despite the scale of training sets, and drift in data distributions must be mitigated. Recently, online optimization has gained popularity as a way to meet these specifications in disparate contexts such as nonparametric regression [4], [5], portfolio management [6], control in robotics [7]. The framework of online optimization decomposes a complex problem into a sequence of sub-problems, which inherently arises when one operates on subsets of data per step due to the sheer scale of full training sets. Alternatively, in many problems, the cost is an expectation of a collection of loss functions parameterized by data only accessible via samples [8], [9].

Deepak, K. Rajawat, A. Gupta, and A. Banerjee are with the department of electrical engineering, IIT Kanpur, India {Email: {dskalhan,ketan,gkrabhi,adrish}@iitk.ac.in}. A. S. Bedi and A. Koppel are with US Army Research Laboratory, Adelphi, MD, USA {Email: alec.e.koppel.civ@mail.mil, amrit0714@gmail.com}. H. Hasani is with the department of electrical engineering, University of Pennsylvania, Philadelphia, PA, USA {Email: hassani@seas.upenn.edu}

Static and Dynamic Regret: To be specific, in online convex optimization (OCO), at each time t, a learner selects an action $\mathbf{x}_t$ after which an arbitrary convex cost F_t is revealed. The standard performance metric for this setting is to compare the action sequence $\{\mathbf{x}_t\}_{t=1}^T$ up to some time-horizon T with a *single* best action in hindsight, defined as the regret $\mathbf{Reg}_T^S = \sum_{t=1}^T F_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T F_t(\mathbf{x})$. However, whenever training data defines trajectories, as is the case in increasingly salient learning problems in dynamical systems or reinforcement learning [10], [11], then hypothesizing that samples come from a stationary distribution is invalid. While the use of buffers experimentally sidestep this issue [12], rigorously addressing it requires treating learning as non-stationary stochastic optimization [13]. This leads the the time varying optimal actions as compared to the single best actions in the static regret settings.

In general, this perspective requires tuning algorithms to mixing rates of the data distribution [14], [15], which substantially impact performance but mixing rates are typically unknown. Online optimization in the presence of non-stationarity avoids these difficulties by instead defining an alternative quantifier of performance called *dynamic regret*: the difference between the instantaneous cost accumulation and the cost of the best action at each time slot [16].

$$\mathbf{Reg}_T^D = \sum_{t=1}^T F_t(\mathbf{x}_t) - \sum_{t=1}^T \min_{\mathbf{x}\in\mathcal{X}} F_t(\mathbf{x}). \tag{1}$$

OCO concerns the design of methods such that $\mathbf{Reg}_T^D$ grows sublinearly in horizon T for a given sequence of loss function F_t, i.e., the average regret goes to null with T (no-regret [17]). Unfortunately, exactly tracking the optimizer defined by an arbitrarily varying optimization problem is impossible [13], [18], and the best one may hope for is to be competitive up to metrics of non-stationarity such as the loss variation V_T and gradient variation D_T defined as [19], [13]

$$\begin{aligned} V_T &:= \sum_{t=1}^T \sup_{\mathbf{x}\in\mathcal{X}} \mid F_t(\mathbf{x}) - F_{t-1}(\mathbf{x}) \mid, \\ D_T &:= \sum_{t=1}^T \left\| \nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1}) \right\|^2. \end{aligned} \tag{2}$$

Our goal in this work is the design of algorithms such that dynamic regret grows sublinearly in T up to multiplicative factors of the variable and gradient variations defined in (2), i.e.,

$\mathbf{Reg}_T^D = o(T(V_T + D_T))$. Before continuing, we contextualize our approach with related works.

Related Work Central to online optimization is online gradient descent [17], whose static regret is $\mathcal{O}(T^{1/2})$. Improvements are possible for strongly convex losses [20], for a detailed review, see [21]. Often, operating under constraints is required to, e.g., safely adapt models or avoid overfitting. Constraint satisfaction at each time slot poses challenges: methods based on Lagrangian relaxation such as ADMM [22] or saddle point [23] cannot ensure feasibility of individual actions. In contrast, projections do so but require a quadratic problem to be solved at each step [24]. Frank-Wolfe (conditional gradient) method moves in a feasible direction that is collinear with the gradient through the solution of a linear program [25], and has gained attention recently as a way to avoid projections in online constrained settings [26], [27]. We build upon these successes to characterize the behavior of Frank-Wolfe method in non-stationary settings.

Non-stationary learning problems cannot be solved exactly [13], [18]: as previously mentioned, dynamic regret is an irreducible of the problem dynamics such as (2) or the variable variation W_T (how much the optimizer changes across time). Thus, several works characterize sublinear growth of dynamic regret up to factors depending on V_T and W_T, i.e., $\mathcal{O}(T^{1/2}(1 + W_T))$ for OGD or mirror descent with convex losses [17], [28], expressions that depend on multiple metrics of non-stationarity [13], [16], and improved rates $\mathcal{O}(1+W_T)$ in strongly convex cases [19]. These works, however, all execute projections, which owing to the complexity requirements, may prohibit them from yielding solutions in a timely fashion when data drifts, in contrast to Frank-Wolfe [26].

An additional challenge is that in practice, exact online gradients may be unavailable, due to dependence on unknown distributions or latency required for sampling. Thus there is a great need for optimization tools that are robust to non-stationarity and noisy gradient estimates. When only estimates of the online gradient are available, i.e., partial feedback, augmentations of Frank-Wolfe that involve sampling multiple gradient estimates per time slot exist [29]. In this work, we take inspiration from [29] to propose such methods that may operate effectively in the presence of non-stationarity.

Contributions In this work, we put forth a collection of online optimization schemes that obviate the need for projection and are robust to gradient estimation error, leveraging recently developed averaging techniques [30], and characterize their performance amidst non-stationarity. In particular:

- We generalize Frank-Wolfe method to non-stationary problems (Sec. II) and establish $\mathcal{O}(T^{1/2})$ dynamic regret when losses are convex (Sec. III).
- We generalize the algorithm to the setting where we only have access noisy estimates of online gradients (partial feedback, Sec. II-A), and establish that its dynamic regret growth is also sublinear (Sec. III-A).
- To close the gap between partial feedback and the aforementioned $\mathcal{O}(T^{1/2})$ rate, we propose to use Meta-Frank Wolfe which allows for multiple samples per action update. We establish its dynamic regret can match rates where one has exact online gradient information.
- Experimentally, we observe that Frank-Wolfe and Meta-Frank Wolfe attain favorable performance relative to alternatives [21] on non-stationary matrix completion and background extraction in video (Sec. IV). In particular, Frank-Wolfe yields a significant reduction in the computational time, while attaining comparable performance, to existing approaches.

The paper is organized as follows. We describe the Frank-Wolfe algorithm for the dynamic settings in Sec.II. All the theoretical results are detailed in Sec.III. The proposed algorithm in applied to the practical problems of interest and the results are presented in IV. In the end, Sec.V concludes the paper.

II. Frank-Wolfe Method

We begin by deriving standard Frank-Wolfe (conditional gradient) algorithm adapted to the setting of online optimization. For time t, assuming that action $\mathbf{x}_t$ has been chosen and the instantaneous cost F_t is revealed, we may evaluate the online gradient as $\nabla F_t(\mathbf{x}_t)$. Based upon this information, we define directional vector d_t by the recursion:

$$\mathbf{d}_t = (1-\rho)\mathbf{d}_{t-1} + \rho \nabla F_t(\mathbf{x}_t) \tag{3}$$

with initial vector $\mathbf{d}_0 = 0$, and $\rho \in (0, 1]$ is a constant momentum parameter. The smoothing step (3) permits us to gracefully apply the algorithm to the more challenging setting of partial feedback or non-convex losses discussed later [31]. Then, we seek a direction $\mathbf{v}_t$ that is parallel to $\mathbf{d}_t$ inside feasible set $\mathcal{X}$, the source of the name *conditional* gradient. This is accomplished by solving the following linear program (LP)

$$\mathbf{v}_t = \arg\min_{\mathbf{v}\in\mathcal{X}} \langle \mathbf{d}_t, \mathbf{v}\rangle. \tag{4}$$

Then, the action $\mathbf{x}_{t+1}$ for subsequent time $t+1$ is given by

$$\mathbf{x}_{t+1} = (1-\gamma)\mathbf{x}_t + \gamma \mathbf{v}_t, \tag{5}$$

where $\gamma < 1$ is a time-invariant step-size. In the following subsection, we discuss a generalization to partial feedback. The method is summarized as Alg. 1.

A. Partial Feedback

To implement the Algorithm 1, the exact gradient $\nabla F_t(\mathbf{x}_t)$ must be computed at each iteration t. In practice, this computation may be unavailable or prohibitively costly to obtain. For instance, in expected risk minimization [8], $\nabla F_t(\mathbf{x}_t)$ denotes the *full batch* gradient, which, if the number N of samples $\{\mathbf{z}_n\}_{n=1}^N$ in the training set is large, is costly to evaluate [32], [33]. Alternatively, one may simply receive only noisy samples of the gradient, but not its true value, as is the case with received signal strength-based localization [34] or learned models of mismatched kinematics in optimal control [35]. For such situations, only a noisy estimate $\nabla f_t(\mathbf{x}_t, \mathbf{z}_t)$ of the online gradient $\nabla F_t(\mathbf{x}_t)$ is available such that $\nabla F_t(\mathbf{x}) = \mathbb{E}\left[\nabla f_t(\mathbf{x}, \mathbf{z}_t)\right]$. Here $\mathbf{z}_t$ denotes a realization of random variable $\mathbf{z}$ that parameterizes the noisy online gradient.

Example 1. For instance, consider the following nonlinear dynamical system [35]

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t) + g(\mathbf{x}_t, \mathbf{z}_t) \tag{6}$$

Algorithm 1 Online Frank-Wolfe Algorithm (OFW)

1: **Require** step sizes $0 < \rho < 1$ and $0 < \gamma < 1$.
2: **Initialize** $t = 0$, $d_0 = 0$ and choose $x_0 \in \mathcal{X}$.
3: **for** t = 1,2.......**do**.
4: **Update** gradient estimate $\mathbf{d}_t = (1-\rho)\mathbf{d}_{t-1} + \rho \nabla F_t(\mathbf{x}_t)$
5: **Compute** $\mathbf{v}_t = \arg\min_{\mathbf{v}\in\mathcal{X}} \langle \mathbf{d}_t, \mathbf{v} \rangle$
6: **Update** $\mathbf{x}_{t+1} = (1-\gamma)\mathbf{x}_t + \gamma \mathbf{v}_t$
7: **end for**

In practice, one observes state estimates $\{\hat{\mathbf{x}}_t\}$ via on-board sensing, which are subtracted from known kinematics $f(\mathbf{x}_t)$ to form unknown model disturbance $g(\mathbf{x}_t, \mathbf{z}_t)$ which is environmentally dependent and non-stationary. In (6), $\mathbf{z}_t$ is interpreted as a sensor feed at time t, and which may be used to learn a model of uncertainty $g(\mathbf{x}_t, \mathbf{z}_t)$, i.e., training pairs take the form $\{\mathbf{z}_t, g(\mathbf{x}_t, \mathbf{z}_t)\}$. The true model disturbance $\mathbb{E}_{\mathbf{z}}[g(\mathbf{x}, \mathbf{z})]$ is unknown, and hence only partial feedback is available.

Under partial feedback, we require use of stochastic online gradients $\nabla f_t(\mathbf{x}_t, \mathbf{z}_t)$ rather than exact online gradients $\nabla F_t(\mathbf{x}_t)$ in step 4 of Algorithm 1. The significance of parameter $\rho \in (0,1)$ will become clear in the context of establishing regret bounds of Frank-Wolfe, which is discussed in later sections.

Batching and Meta-Frank Wolfe While noisy unbiased samples of the online gradient may be the only available, often it is practical to take multiple samples of the gradient per algorithm iterate. This means that we could fix a batch size K and then perform the updated (5) processed on samples $\{\nabla f_t(\mathbf{x}_t, \mathbf{z}_t^k)\}_{k=1}^{K}$, and output only the last action performed at $K+1$. This is summarized in step 7 of Algorithm 2. This improves the quality of the gradient (by reducing the variance) and as a result attains improved dynamic regret, as detailed in the next section. We call the variant of Frank-Wolfe that incorporate two time-scale procedure Meta-Frank Wolfe, which is summarized in Algorithm 2. The steps in Algorithm 2 are similar to the that proposed in [29] but here specified for the non-stationary setting. Next, we shift focus to establishing convergence of Algorithms 1 - 2. The main results are summarized in comparison to the state of the art results in Table I.

III. Dynamic Regret Analysis

In this section, we characterize the performance of Algorithm 1-2 in the presence of non-stationarity as quantified by dynamic regret. First, we state some required technical assumptions.

Assumption 1. *The set $\mathcal{X}$ is convex and compact with diameter D, i.e., for all $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, it holds that $\|\mathbf{x} - \mathbf{y}\| \leq D$.*

Assumption 2. *The gradient of loss $\nabla F_t(\cdot)$ is Lipschitz with parameter L_1, which implies that*

$$\left\|\nabla F_t(\mathbf{x}) - \nabla F_t(\mathbf{y})\right\| \leq L_1 \|\mathbf{x} - \mathbf{y}\| \text{ for all } t \text{ and } (\mathbf{x}, \mathbf{y}) \in \mathcal{X}. \tag{7}$$

The Assumptions 1-2 are standard in online learning [19], [31]. Assumption 1 ensures constrained set $\mathcal{X}$ is compact. Assumption 2 bounds the loss function gradient. Next, we present the dynamic regret analysis of Algorithm 1. To do so,

Algorithm 2 Meta-Frank Wolfe Algorithm

1: **INPUT:** convex set $\mathcal{X}$, time horizon T, linear optimization oracles $\mathcal{E}^{(1)}$........$\mathcal{E}^{K}$, step sizes $\rho \in (0,1)$ and $\gamma \in (0,1)$, and initial point $\mathbf{x_1}$.
2: **OUTPUT:** $\{\mathbf{x}_t : 1 \leq t \leq T\}$
3: **Initialize** online linear optimization oracles $\mathcal{E}^{(1)}$........$\mathcal{E}^{K}$.
4: **Initialize** $\mathbf{d}_t^0 = 0$ and $\mathbf{x}_t^1 = \mathbf{x}_{t-1}$
5: **for** t = 1,2.......T **do**.
6: $\mathbf{v}_t^k \leftarrow$ output of oracle $\mathcal{E}^k$ in round $t-1$.
7: $\mathbf{x}_t^{(k+1)} \leftarrow (1-\gamma)\mathbf{x}_t^k + \gamma \mathbf{v}_t^k$ for $k = 1, \cdots, K$
8: Select $\mathbf{x}_t = \mathbf{x}_t^{(K+1)}$, then obtain $f_t(\cdot, \mathbf{z}_t^k)$ and online stochastic grad. $\nabla f_t(\cdot, \mathbf{z}_t^k)$ for each k
9: $\mathbf{d}_t^k \leftarrow (1-\rho)\mathbf{d}_t^{(k-1)} + \rho \nabla f_t(\mathbf{x}_t^k, \mathbf{z}_t^k)$ for $k = 1, \cdots, K$
10: Feedback $\langle \mathbf{v}_t^k, \mathbf{d}_t^k \rangle$ to $\mathcal{E}^k$ for $k = 1, \cdots, K$
11: **end for**

we establish some technical lemmas that characterize descent-like properties.

Lemma 1. *Under Assumptions 1-2, Algorithm 1 satisfies the following descent relations: when loss functions F_t are convex,*

$$\begin{aligned} F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \leq & F_{t,t-1}^{sup}(\mathbf{x}) + (1-\gamma)(F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}^\star)) \\ & + F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star) + \gamma^2 \frac{3L_1}{2} D^2 \\ & + \gamma D \left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\|. \end{aligned} \tag{8}$$

where we define $F_{t,t-1}^{sup}(\mathbf{x}) := \sup_{\mathbf{x}\in\mathcal{X}} |F_t(\mathbf{x}) - F_{t-1}(\mathbf{x})|$ as the instantaneous maximum cost variation.

Our first result, Lemma 1, shows that the sub-optimality $F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)$ depends on $(1-\gamma)$ times the sub-optimality at previous instant and maximum variation between the two consecutive functions, since the remaining terms can be very small with proper selection of step-size γ. The detailed proof is provided in Appendix A.

Theorem 1. *Under the Assumptions 1-2, for the iterates generated by Algorithm 1, under step-size selection $\gamma = \frac{1}{\sqrt{T}}$, it holds that*

$$\boldsymbol{Reg}_T^D \leq \mathcal{O}\left(\sqrt{T}\left(1 + V_T + \sqrt{D_T}\right)\right). \tag{9}$$

Proof. Taking Summation on both sides from $t = 1$ to T of the statement of Lemma 1, we get

$$\begin{aligned} &\sum_{t=1}^{T}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right] \\ &\quad \leq \sum_{t=1}^{T} F_{t,t-1}^{\text{sup}}(\mathbf{x}) + F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star) \\ &\qquad + (1-\gamma)\sum_{t=1}^{T-1}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right] + (1-\gamma)\left[F_0(\mathbf{x}_0) - F_0(\mathbf{x}_0^\star)\right] \\ &\qquad + \gamma D \sum_{t=1}^{T}\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\| + \gamma^2 T \frac{3L_1 D^2}{2} \end{aligned} \tag{10}$$

Reference	Loss func.	Step-size	Batch	Regret definition	Rate
[26]	$(L/D)t^{-1/4}$-strongly convex	diminishing	$\mathcal{O}(t)$	$\sum_{t=1}^{T} F(\mathbf{x}_t) - F(\mathbf{x}^\star)$	$\mathcal{O}\left(T^{3/4}\right)$
[31]	convex	diminishing	$\mathcal{O}(1)$	$\mathbb{E}\left[F(\mathbf{x}_T) - F(\mathbf{x}^\star)\right]$	$\mathcal{O}(1/T^{1/3})$
[36]	convex	depends on σ_2 & C_T	-	$\frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\left[F_t(\mathbf{x}_{i,t}) - F_t(\mathbf{x}_t^\star)\right]$	$\mathcal{O}\left(\sqrt{\frac{(1+C_T)T}{1-\sigma_2(W)}}\right)$
[21]	1-strongly convex	diminishing	$\mathcal{O}(1)$	$\sum_{t=1}^{T}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}^\star)\right]$	$\mathcal{O}\left(T^{3/4}\right)$
This work	convex	constant	$\mathcal{O}(1)$	$\sum_{t=1}^{T} F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)$	$\mathcal{O}\left(\sqrt{T}\left(1 + V_T + \sqrt{D_T}\right)\right)$
This work	convex	constant	$\mathcal{O}(1)$	$\sum_{t=1}^{T}\mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right]$	$\mathcal{O}\left(1 + T^{\frac{5}{6}} + \sqrt{T}V_T + T^{\frac{5}{6}}\sqrt{D_T}\right)$
This work	convex	constant	$\mathcal{O}(T^a)$	$\sum_{t=1}^{T} F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)$	$\mathcal{O}\left(1 + V_T + \mathcal{R}_T^{\mathcal{E}} + T^{(1-a)}\right)$
This work	convex	constant	$\mathcal{O}(T^a)$	$\sum_{t=1}^{T}\mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right]$	$\mathcal{O}\left(1 + V_T + \mathcal{R}_T^{\mathcal{E}} + T^{\max\{(2.5-a),0.5\}}\right)$

TABLE I: Summary of the related works compared to the present work.

where $F_{t,t-1}^{\sup}(\mathbf{x}) := \sup_{\mathbf{x}\in\mathcal{X}} |F_t(\mathbf{x}) - F_{t-1}(\mathbf{x})|$ as in Lemma 1. Taking the term $(1-\gamma)\sum_{t=1}^{T-1}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)|\right]$ to the left side of (10), and utilizing $\left(\sum_{t=1}^{T} F_{t-1}(\mathbf{x}_{t-1}^\star) - \sum_{t=1}^{T} F_t(\mathbf{x}_t^\star)\right) = \left(F_0(\mathbf{x}_0^\star) - F_T(\mathbf{x}_T^\star)\right)$, we get

$$\gamma\sum_{t=1}^{T-1}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right] + \left[F_T(\mathbf{x}_T) - F_T(\mathbf{x}_T^\star)\right] \tag{11}$$
$$\leq \sum_{t=1}^{T} F_{t,t-1}^{\sup}(\mathbf{x}) + (1-\gamma)\left[F_0(\mathbf{x}_0) - F_0(\mathbf{x}_0^\star)\right] + F_0(\mathbf{x}_0^\star)$$
$$- F_T(\mathbf{x}_T^\star) + \gamma D\sum_{t=1}^{T}\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\| + \gamma^2 T\frac{3L_1D^2}{2}.$$

Since $\gamma < 1$, the left hand side of (11) is lower bounded by $\gamma\sum_{t=1}^{T}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right]$ and $(1-\gamma) < 1$. We can write (11) as

$$\gamma\mathbf{Reg}_T^D \leq \sum_{t=1}^{T} F_{t,t-1}^{\sup}(\mathbf{x}) + \left[F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star)\right] \tag{12}$$
$$+ \gamma D\sum_{t=1}^{T}\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\| + \gamma^2 T\frac{3L_1D^2}{2}.$$

Divide both sides of (12) by γ and substitute the identity

$$\sum_{t=1}^{T}\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\|$$
$$\leq \sqrt{T\sum_{t=1}^{T}\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\|^2} = \sqrt{TD_T} \tag{13}$$

into (12) with the definition of V_T (2) to obtain

$$\mathbf{Reg}_T^D \leq \frac{1}{\gamma}V_T + \frac{1}{\gamma}\left[F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star)|\right]$$
$$+ D\sqrt{TD_T} + \gamma T\frac{3L_1D^2}{2}. \tag{14}$$

Select $\gamma = \frac{1}{\sqrt{T}}$ and use $(1-\gamma) < 1$ to write

$$\mathbf{Reg}_T^D \leq \sqrt{T}\left(V_T + K_1 + D\sqrt{D_T}\right), \tag{15}$$

where $K_1 := \left[F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star)\right] + \frac{3L_1D^2}{2}$, completing the proof. □

Theorem 1 establishes convergence of Algorithm 1 for non-stationary problems in terms of dynamic regret up to factors depending on V_T and D_T [cf. (2)], as defined in Section I. This is the first time a projection-free scheme has been demonstrated as theoretically effective for dynamic learning problems, which paves the way for use in applications with data drift across time. Note, however, that Algorithm 1 requires exact gradient information at each step, which in applications to learning control such as (6), may be unavailable. This motivates the partial feedback setting which we analyze next.

A. Regret Analysis under Partial Feedback

To analyze performance in when feedback is partial, before proceeding, we state an additional required assumption that limits the variance of stochastic approximation error.

Assumption 3. *The variance of the unbiased stochastic gradients $\nabla\tilde{F}_t(\mathbf{x}, \mathbf{z})$ is bounded above by σ^2*

$$\mathbb{E}\left[\left\|\nabla f_t(\mathbf{x},\mathbf{z}) - \nabla F_t(\mathbf{x})\right\|^2\right] \leq \sigma^2, \quad \textit{for all } t. \tag{16}$$

We are ready to state the convergence of Algorithm 1 in terms of dynamic regret. We begin by characterizing the error associated with using partial feedback, i.e., stochastic gradients, rather than true gradients, in the following lemma.

Lemma 2. *Let the Assumptions 1-3 hold, then the iterates generated by Algorithm 1 satisfy*

$$\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t) - \mathbf{d}_t\right\|^2\right] \leq \rho^2\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t) - \nabla f_t(\mathbf{x}_t,\mathbf{z}_t)\right\|^2\right] \tag{17}$$
$$+ \frac{(1-\rho)}{\rho}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\|^2\right]$$
$$+ (1-\rho)\mathbb{E}\left[\left\|\nabla F_{t-1}(\mathbf{x}_{t-1}) - \mathbf{d}_{t-1}\right\|^2\right].$$

The proof is provided in Appendix B. The result in Lemma 2 shows that the squared error of gradient estimation $\left\|\nabla F_t(\mathbf{x}_t) - \mathbf{d}_t\right\|^2$ depends on $(1-\rho)$ times the squared error in previous gradient estimate and variation in gradient at previous instant. The other terms are negligible with proper choice of ρ.

We will use this characterization of gradient estimation error to establish decrement properties of Algorithm 1 in the convex settings. Before doing so, we analyze the error accumulation over time horizon T for partial feedback, as stated next.

Corollary 1. *Using the result of Lemma 1, it holds that*

$$\sum_{t=1}^{T}\mathbb{E}\big[\,\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\|\,\big] \tag{18}$$
$$\leq \sqrt{\left(\rho T^2\sigma^2+\frac{T(1-\rho)}{\rho^2}D_T+\frac{T(1-\rho)}{\rho}\left\|\nabla F_0(\mathbf{x}_0)\right\|^2\right)}.$$

The result in Corollary 1 (proof is provided in Appendix C) shows that the error in gradient estimate over the entire time horizon is bounded asymptotically if we choose ρ properly. We note that if we select $\rho=1$, i.e., use stochastic gradients, then the gradient estimation error over entire time horizon diverges due to the variance of estimates. Now, we are ready to establish the decrement properties of Algorithm 1 in the convex case, which is formalized in the following lemma.

Lemma 3. *Under Assumptions 1-2, the iterates generated by the Algorithm 1 under the partial feedback satisfy: when loss functions F_t are convex, it holds that :*

$$F_t(\mathbf{x}_t)-F_t(\mathbf{x}_t^\star)\leq(1-\gamma)\left(F_{t-1}(\mathbf{x}_{t-1})-F_{t-1}(\mathbf{x}_{t-1}^\star)\right) \tag{19}$$
$$+2F_{t,t-1}^{sup}(\mathbf{x})+\gamma D\left\|\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\right\|$$
$$+\gamma D\left\|\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\right\|+\gamma^2\frac{3L_1D^2}{2}.$$

The detailed proof of Lemma 3 is provided in Appendix D. Lemma 3 shows that the sub-optimality $F_t(\mathbf{x}_t)-F_t(\mathbf{x}_t^\star)$ depends on $(1-\gamma)$ times the sub-optimality at previous instant and maximum variation between the two consecutive functions and other terms which can be made small with appropriate choice of averaging parameter γ.

Theorem 2. *Under the Assumptions 1-3, for the iterates generated by Algorithm 1, the following expected dynamic regret bounds hold:*

$$\sum_{t=1}^{T}\left[\mathbb{E}\left[F_t(\mathbf{x}_t)\right]-F_t(\mathbf{x}_t^\star)\right] \tag{20}$$
$$\leq\mathcal{O}\left(1+T^{\frac{5}{6}}+\sqrt{T}V_T+T^{\frac{5}{6}}\sqrt{D_T}\right),$$

under step-size and inertia selections $\gamma=\frac{1}{\sqrt{T}}$, $\rho=\frac{1}{T^{1/3}}$.

Proof. Firstly, compute the total expectation on both sides of the statement of Lemma 3 for a given $\mathcal{F}_{t-1}$, we get,

$$\mathbb{E}\left[F_t(\mathbf{x}_t)\right]-F_t(\mathbf{x}_t^\star) \tag{21}$$
$$\leq F_{t,t-1}^{\text{sup}}(\mathbf{x})+(1-\gamma)\big[\mathbb{E}[F_{t-1}(\mathbf{x}_{t-1})]-F_{t-1}(\mathbf{x}_{t-1}^\star)\big]$$
$$+\mathbb{E}\big[F_{t-1}(\mathbf{x}_{t-1}^\star)-F_t(\mathbf{x}_t^\star)\big]+\gamma D\mathbb{E}\left\|\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\right\|$$
$$+\gamma D\mathbb{E}\big[\left\|\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\right\|\big]+\gamma^2\frac{3L_1D^2}{2}$$

Next, following the steps similar to (10) to (14), we obtain

$$\sum_{t=1}^{T}\left[\mathbb{E}\left[F_t(\mathbf{x}_t)\right]-F_t(\mathbf{x}_t^\star)\right]$$
$$\leq\frac{1}{\gamma}V_T+\frac{1}{\gamma}\left[F_0(\mathbf{x}_0)-F_T(\mathbf{x}_T^\star)|\right]+D\sqrt{TD_t}$$
$$+D\sum_{t=1}^{T}\mathbb{E}\big[\left\|\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\right\|\big]+\gamma T\frac{3L_1D^2}{2}. \tag{22}$$

Using the result of Corollary 1 into (22), utilize $\sqrt{u_1+u_2}\leq\sqrt{u_1}+\sqrt{u_2}$, and the upper bound of $(1-\rho)<1$, we get

$$\sum_{t=1}^{T}\Big[\mathbb{E}\left[F_t(\mathbf{x}_t)\right]-F_t(\mathbf{x}_t^\star)\Big]\leq\frac{1}{\gamma}V_T+\frac{1}{\gamma}\left[F_0(\mathbf{x}_0)-F_T(\mathbf{x}_T^\star)\right]+D\sqrt{\rho}T\sigma$$
$$+D\left(\frac{1}{\rho}+1\right)\sqrt{TD_T}+D\sqrt{\frac{T}{\rho}}\left\|\nabla F_0(\mathbf{x}_0)\right\|$$
$$+D\left\|\nabla F_0(\mathbf{x}_0)\right\|+\gamma T\frac{3L_1D^2}{2}. \tag{23}$$

Substituting $\gamma=\frac{1}{\sqrt{T}}$ and $\rho=\frac{1}{T^c}$ for any $0<c<\frac{1}{2}$, we obtain

$$\sum_{t=1}^{T}\left[\mathbb{E}\left[F_t(\mathbf{x}_t)\right]-F_t(\mathbf{x}_t^\star)\right]\leq\sqrt{T}\left(V_T+K_1\right)+DT^{\left(1-\frac{c}{2}\right)}\sigma \tag{24}$$
$$+D\left(T^c+1\right)\sqrt{TD_T}+T^{\left(\frac{1+c}{2}\right)}B+B$$

where, $B:=\left(D\left\|\nabla F_0(\mathbf{x}_0)\right\|\right)$, $K_1=\left[F_0(\mathbf{x}_0)-F_T(\mathbf{x}_T^\star)+\frac{3L_1D^2}{2}\right]$ and we utilize the definition of D_T and V_T.

In order to ensure sublinearity of (20), we require $0<c<\frac{1}{2}$ and with $c=\frac{1}{3}$ we have,

$$\sum_{t=1}^{T}\left[\mathbb{E}\left[F_t(\mathbf{x}_t)\right]-F_t(\mathbf{x}_t^\star)\right]$$
$$\leq B+\sqrt{T}\left(K_1+V_T+D\sqrt{D_T}\right) \tag{25}$$
$$+T^{\left(\frac{5}{6}\right)}D\big(\sigma+\sqrt{D_T}\big)+T^{\left(\frac{2}{3}\right)}(B). \tag{26}$$

□

Theorem 2 establishes that the dynamic regret for Algorithm 1 is sublinear despite only having access to noisy estimates of online gradients, given appropriate stepsize and averaging parameter selections.

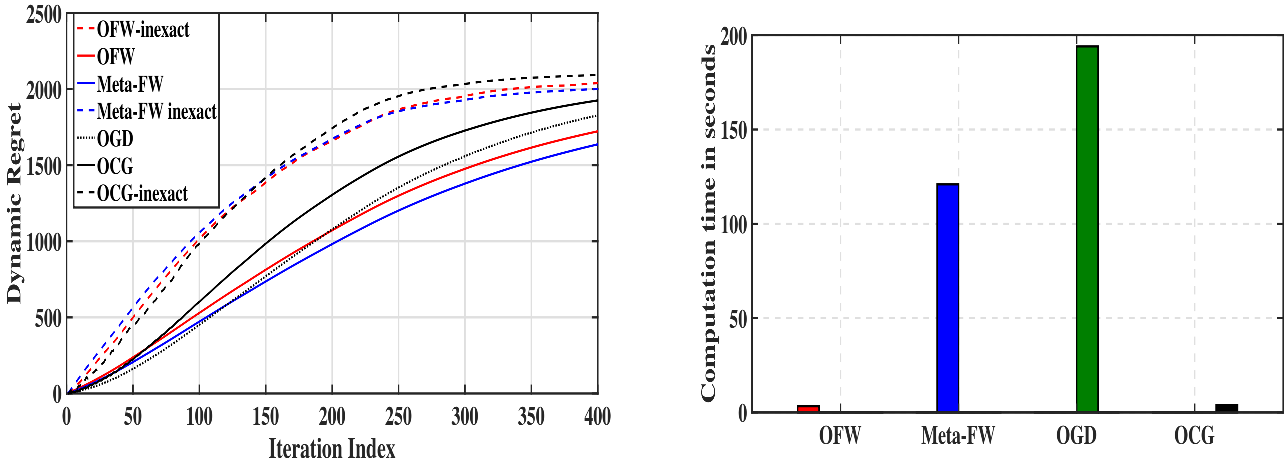
B. Improved Results for Meta-Frank Wolfe

We may tighten the regret bounds by sampling multiple stochastic gradients between time slots, as described in Algorithm 2. To establish these results, several prerequisite lemmas must be established which characterize the gradient estimation error and decrement properties. Firstly, we analyze the noise in gradient approximation at a particular instant k and provide an upper bound as stated in following Lemma 4.

Lemma 4. *Under Assumption, 1-2, it holds for the iterates generated by Algorithm 2 that*

$$\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2\Big]\leq\rho^2\mathbb{E}\left\|\nabla F_t(\mathbf{x}_t^k)-\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k)\right\|^2 \tag{27}$$
$$+\frac{(1-\rho)}{\rho}\mathbb{E}\left\|\nabla F_t(\mathbf{x}_t^k)-\nabla F_t(\mathbf{x}_t^{k-1})\right\|^2$$
$$+(1-\rho)\mathbb{E}\left\|\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}_t^{k-1}\right\|^2.$$

The result in Lemma 4 (proof in Appendix E)shows that the squared error of gradient estimation $\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2|\mathcal{F}_t\Big]$ depends on $(1-\rho)$ times the squared error in previous gradient estimate and variation in gradient at previous instant, when the



(a) (b)

Fig. 1: (a) Comparison of dynamic regrets of different algorithms for the matrix completion. (b) Runtime comparison of Frank-Wolfe and Meta Frank-Wolfe compared to alternatives on matrix complication. Observe that the OFW performs better than the Meta-FW and OGD. The performance is similar to OCG but OCG has a higher dynamic regret [see Fig. 1a].

rest of the terms are made negligible relative to these terms under proper selection of averaging parameter ρ.

We use the preceding relationship (Lemma 4) to establish the following corollary which characterizes the gradient estimation error associated with partial feedback over time horizon T.

Corollary 2. *Utilizing the statement of Lemma 4, it holds that*

$$\sum_{k=1}^{K} \mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k) - \mathbf{d}_t^k\right\|^2 \Big] \leq \rho\sigma^2 K + \frac{L_1^2(1-\rho)}{\rho^2}\gamma^2 K D^2 + \frac{(1-\rho)}{\rho}\left\|\nabla F_t(\mathbf{x}_t^0)\right\|^2. \quad (28)$$

If we use the stochastic gradient directly in place of its estimate $\mathbf{d}_t^k$ by setting $\rho = 1$, the error in gradient approximation over entire inner loop time horizon diverges due to non-vanishing variance of gradient approximation

Next, similar to the analysis of Algorithm 1, we present a decrement property satisfied by the iterates of Algorithm 2.

Lemma 5. *With all the Assumption 1-3 satisfied, for the sub-optimality $F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)$ generated by the actions of Algorithm 2, the following hold true, when loss functions F_t are convex,*

$$F_t(\mathbf{x}_t^{k+1}) - F_t(\mathbf{x}_t^\star) \leq (1-\gamma)(F_t(\mathbf{x}_t^k) - F_t(\mathbf{x}_t^\star)) + \frac{\gamma}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^k) - d_t^k\right\|^2 + \gamma\frac{\beta}{2}D^2 + \gamma\langle d_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star\rangle + \gamma^2\frac{L_1}{2}D^2. \quad (29)$$

Lemma 5 shows that the sub-optimality $F_t(\mathbf{x}_t^{k+1}) - F_t(\mathbf{x}_t^\star)$ decreases at each iteration by the factor $(1-\gamma)$, when the remaining terms are made small under proper selection of averaging parameter γ and constant $\beta > 0$.

With those in place, the dynamic regret performance of Meta-Frank Wolfe is presented next.

Theorem 3. *Consider the proposed Algorithm 2, with all the Assumptions 1-3 satisfied, and the online linear optimization oracles have regret $\mathcal{R}_t^{\mathcal{E}}$ at instant t for any k. Then, with step-size $\gamma = \frac{1}{K}$, inertia value $\rho = \frac{1}{T}$, and inner-loop $K = \mathcal{O}(T^a)$ with $a >$ 1.5, it holds that*

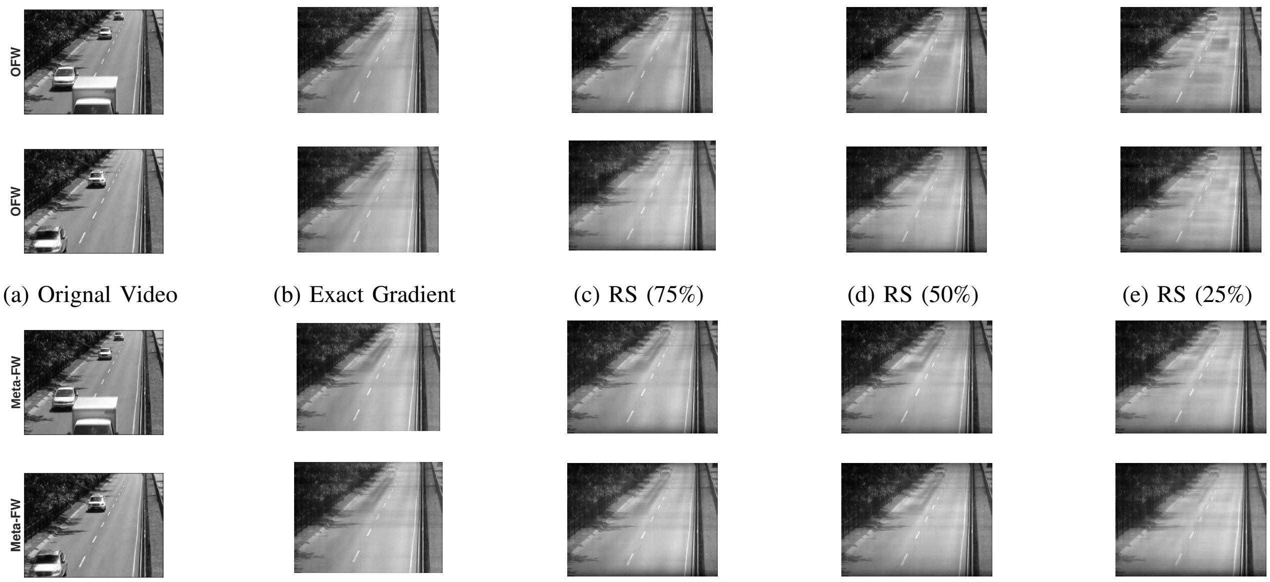
$$\sum_{t=1}^{T}\Big[\mathbb{E}\left[F_t(\mathbf{x}_t)\right] - F_t(\mathbf{x}_t^\star)\Big] \leq \mathcal{O}\left(1 + V_T + \mathcal{R}_T^{\mathcal{E}} + T^{\max\{(2.5-a), 0.5\}}\right). \quad (30)$$

Here, $\mathbb{E}$ denotes the total expectation with respect to randomness in the algorithm updates and noise realizations of $\mathbf{z}$ *and $\mathcal{R}_T^{\mathcal{E}} = \sum_{t=1}^{T}\mathcal{R}_t^{\mathcal{E}}$.*

Theorem 3 (proof in Appendix H) establishes that one may improve performance in terms of dynamic regret by using Algorithm 2, i.e., multiple gradient samples per action update, rather than a single one (Alg. 1), as in Theorem 2. Meta-Frank Wolfe (Algorithm 2) makes the use of K online linear optimization oracles in each round which in turn requires K gradient samples for each round and hence attains superior dynamic regret compared to its standard counterpart which decreases with increase in K. In particular, attains $\mathcal{O}(T^{1/2})$ dynamic regret as compared to the one shot algorithm which attains a slower rate (depending on the choice of c that parameterizes the step-size. With the performance of Algorithms 1 - 2 established, we turn to experimental evaluation of the proposed schemes.

IV. Experiments

In this section, we experimentally evaluate Algorithms 1-2 on matrix completion and background subtraction in video, both of which demonstrate the merits of online Frank-Wolfe. That is, we observe a favorable tradeoff between complexity and accuracy by virtue of avoiding computationally costly projections. In particular, we compare Algorithm 1 (basic Frank-Wolfe) with full and partial feedback, i.e., exact and stochastic estimates of the online gradient, as well as Algorithm 2, Meta Frank-Wolfe, in the full and partial (25% of the possible gradient samples)



(a) Orignal Video (b) Exact Gradient (c) RS (75%) (d) RS (50%) (e) RS (25%)

(f) Orignal Video (g) Exact Gradient (h) RS (75%) (i) RS (50%) (j) RS (25%)

Fig. 2: Background Extraction Problem: $1st$ and $3rd$ row represents results for instant 1 of the video; the $2nd$ and 4th row represent instant 2 of the video, which is clear from the $1st$ column. In the figure, RS denotes random sampling and the percentage denotes how many samples from the full gradient are utilized for the algorithm updates. The proposed algorithm performs really well for this application since the cars are effectively removed from the frame, as clear from 2nd to 5th columns.

feedback setting. The aforementioned approaches obviate the need for projections, whereas online gradient descent [17], which we also implement for comparison, requires projections. We shift to detailing the benchmarks.

Matrix Completion: Consider the problem of online matrix completion, which seeks the best possible low rank approximation of a given matrix $\mathbf{M_t} \in \mathbb{R}^{m\times n}$. Denote as $\mathbf{X}_t \in \mathbb{R}^{m\times n}$ the low-rank approximation. In each round, the entries of matrix are updated. The problem is then defined as [21, Chap. 7]

$$\min_{\mathbf{X}_{ij}} \sum_{(ij)\in OB} (\mathbf{X}_{ij} - \mathbf{M}_{ij})^2 \quad \text{such that } \|\mathbf{X}\|_* \leq k. \tag{31}$$

We solve (31) using Algorithms 1-2 and compare performance with alternatives such as OGD in Fig. 1a. We have presented the results for both exact as well as inexact gradient. For OFW-inexact and Meta-FW-inexact, we have considered only 25% of the samples full gradient from random locations at each iteration. As presented in Fig. 1a, OFW performs better than online conditional gradient (OCG), a projection-free algorithm of [21] when full gradient information is available. We remark that the Meta-FW algorithm performs best among all the algorithms when full gradient is available. A similar behavior is observed with the partial information availability too. Also Fig. 1b shows that OGD is the slowest as compared to all the algorithms due to the required projection. For the experiments, we have considered $m = n = 20$. To implement the Meta-Frank Wolfe algorithm, we fix $K = 30$.

Background extraction problem: In this experiment, we extend the matrix completion problem on real dataset from [37]. At each instant we observe a video frame and collect it into matrix $\mathbf{M}_t$. The goal of the problem is to extract the background from the video which is conceptually the low-rank estimate $\mathbf{M}_t$ of the data matrix. The problem is then given as:

Algorithm	Exact Gradient	RS(75%)	RS(50%)	RS(25%)
OFW	4.6436	4.2325	3.1949	3.1396
Meta-FW	26.5808	26.5810	22.8794	21.1206

TABLE II: Summary of computation time in seconds for background extraction problem.

$$\min_{\mathbf{L}_t} \left\|\mathbf{M}_t - \mathbf{L}_t\right\|_F^2 + \frac{1}{2}\left\|\mathbf{L}_t\right\|_F^2 \ \text{ such that } \left\|\mathbf{L}_t\right\|_* \leq k \tag{32}$$

The results in Fig. 2 are generated using OFW with different samples of gradient at different instants. Note that online Frank-Wolfe yields effective performance for this application as demonstrated in Fig. 2 – the cars are removed from the frame. We summarize execution times in Table II, where we observe obviating projections yields quick completion. Please see the video appended to the submission to observe Frank-Wolfe implementing online background subtraction.

V. Conclusions and Future Work

In this work, we focused on learning problems amidst non-stationarity, which we addressed with the formalism of dynamic regret minimization. Due to the nature of non-stationary learning, the best one may hope to achieve is sublinear regret up to fundamental metrics of non-stationarity. Existing approaches which achieve these results, such as online gradient descent, require projections to ensure constraint satisfaction, which can cause a computational bottleneck that reduces adaptivity. To avoid this issue, we proposed using Frank-Wolfe (conditional gradient method), which involves executing updates in directions that are collinear with the gradient of the instantaneous cost that are guaranteed to be feasible. These directions are found as the solution of a linear program.

Overall, we established that Frank-Wolfe enjoys comparable performance in terms of dynamic regret to existing methods with substantially reduced complexity, thus improving adaptivity. Moreover, when only noisy estimates of the stochastic gradient are available, we established the dynamic regret of Frank-Wolfe, which can be improved through multiple gradient samples per time-slot (Meta Frank-Wolfe). These conceptual results translated well into experimental performance competitive with the state of the art with significant reductions in runtime. Future directions include tuning hyper-parameters to the dynamics of the learning problem, as well as addressing constraints that cannot be satisfied easily using projections or the solution of linear programs.

Appendix A
Proof of Lemma 1

Proof. From the L_1 smoothness of the loss function F_t (Assumption 2), it holds that

$$F_t(\mathbf{x}_t) \leq F_t(\mathbf{x}_{t-1}) + \langle \nabla F_t(\mathbf{x}_{t-1}), \mathbf{x}_t - \mathbf{x}_{t-1}\rangle + \frac{L_1}{2}\left\|\mathbf{x}_t - \mathbf{x}_{t-1}\right\|^2. \quad (33)$$

From the updated in Algo. 1, we write $(\mathbf{x}_t - \mathbf{x}_{t-1}) = \gamma(\mathbf{v}_{t-1} - \mathbf{x}_{t-1})$ and substitute into (33), we get

$$\begin{aligned} F_t(\mathbf{x}_t) \leq & F_t(\mathbf{x}_{t-1}) + \gamma\langle \nabla F_t(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \\ & + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2. \end{aligned} \quad (34)$$

Adding and subtracting the terms $\gamma\langle \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle$ to the right hand side of (34) and after rearranging, we obtain

$$\begin{aligned} F_t(\mathbf{x}_t) \leq & F_t(\mathbf{x}_{t-1}) + \gamma\langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \quad (35) \\ & + \gamma\langle \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2. \end{aligned}$$

Next, utilizing the optimality condition i.e.

$$\langle \mathbf{x}^\star_{t-1}, \nabla F_{t-1}(\mathbf{x}_{t-1})\rangle \geq \min_{\mathbf{v}\in\mathcal{X}}\langle \mathbf{v}, \nabla F_{t-1}(\mathbf{x}_{t-1})\rangle = \langle \mathbf{v}_{t-1}, \nabla F_{t-1}(\mathbf{x}_{t-1})\rangle$$

into (35), we get

$$\begin{aligned} F_t(\mathbf{x}_t) \leq & F_t(\mathbf{x}_{t-1}) + \gamma\langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \quad (36) \\ & + \gamma\langle \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{x}^\star_{t-1} - \mathbf{x}_{t-1}\rangle + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2. \end{aligned}$$

Using first order convexity condition of $F_{t-1}(\cdot)$ in (36), we get

$$\begin{aligned} F_t(\mathbf{x}_t) \leq & F_t(\mathbf{x}_{t-1}) + \gamma\langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \quad (37) \\ & + \gamma(F_{t-1}(\mathbf{x}^\star_{t-1}) - F_{t-1}(\mathbf{x}_{t-1})) + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2. \end{aligned}$$

Subtracting $F_t(\mathbf{x}^\star_t)$ from both sides of (37) leads to

$$\begin{aligned} F_t(\mathbf{x}_t) - F_t(\mathbf{x}^\star_t) \leq & F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}^\star_t) \\ & + \gamma\langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \\ & + \gamma(F_{t-1}(\mathbf{x}^\star_{t-1}) - F_{t-1}(\mathbf{x}_{t-1})) \\ & + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2. \end{aligned} \quad (38)$$

Next, consider the term $F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}^\star_t)$ from the right hand side of (38), we can write

$$\begin{aligned} F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}^\star_t) = & F_t(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}) + F_{t-1}(\mathbf{x}_{t-1}) \quad (39) \\ & - F_{t-1}(\mathbf{x}^\star_{t-1}) + F_{t-1}(\mathbf{x}^\star_{t-1}) - F_t(\mathbf{x}^\star_t). \end{aligned}$$

Bounding the $F_t(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1})$ by its absolute value, and then taking the supremum over $\mathbf{x} \in \mathcal{X}$, we get

$$\begin{aligned} F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}^\star_t) \leq & F^{\sup}_{t,t-1}(\mathbf{x}) + F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}^\star_{t-1}) \\ & + F_{t-1}(\mathbf{x}^\star_{t-1}) - F_t(\mathbf{x}^\star_t). \end{aligned} \quad (40)$$

where we use $F^{\sup}_{t,t-1}(\mathbf{x}) := \sup_{\mathbf{x}\in\mathcal{X}} |F_t(\mathbf{x}) - F_{t-1}(\mathbf{x})|$. Using the inequality from (40) into (38) and after regrouping the terms, we get

$$\begin{aligned} F_t(\mathbf{x}_t) - F_t(\mathbf{x}^\star_t) \leq & F^{\sup}_{t,t-1}(\mathbf{x}) + (1-\gamma)(F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}^\star_{t-1})) \\ & + F_{t-1}(\mathbf{x}^\star_{t-1}) - F_t(\mathbf{x}^\star_t) \\ & + \gamma\langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \\ & + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2 \end{aligned} \quad (41)$$

Further, the inner product term in (41) can be written as

$$\begin{aligned} & \langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \quad (42) \\ & = \langle \nabla F_t(\mathbf{x}_{t-1}) - \nabla F_t(\mathbf{x}_t) + \nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1}), \mathbf{v}_{t-1} - \mathbf{x}_{t-1}\rangle \end{aligned}$$

Substituting (42) into (41) and applying the Cauchy-Schwartz inequality, we get

$$\begin{aligned} F_t(\mathbf{x}_t) - F_t(\mathbf{x}^\star_t) \leq & F^{\sup}_{t,t-1}(\mathbf{x}) + (1-\gamma)(F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}^\star_{t-1})) \\ & + F_{t-1}(\mathbf{x}^\star_{t-1}) - F_t(\mathbf{x}^\star_t) \quad (43) \\ & + \gamma\left\|\nabla F_t(\mathbf{x}_{t-1}) - \nabla F_t(\mathbf{x}_t)\right\|\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\| \\ & + \gamma\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\|\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\| \\ & + \gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2. \end{aligned}$$

Using Assumption 2 and the update for $\mathbf{x}_t$, we get

$$\begin{aligned} F_t(\mathbf{x}_t) - F_t(\mathbf{x}^\star_t) \leq & F^{\sup}_{t,t-1}(\mathbf{x}) + (1-\gamma)(F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}^\star_{t-1})) \quad (44) \\ & + F_{t-1}(\mathbf{x}^\star_{t-1}) - F_t(\mathbf{x}^\star_t) + \gamma^2\frac{3L_1}{2}\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|^2 \\ & + \gamma\left\|\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})\right\|\left\|\mathbf{v}_{t-1} - \mathbf{x}_{t-1}\right\|. \end{aligned}$$

Utilizing the statement of Assumption 1 and upper bounding the right hand side of (24) provides the result in Lemma 1. □

Appendix B
Proof of Lemma 2

Proof. Let us start by analyzing the term $\left\|\nabla F_t(\mathbf{x}_t) - \mathbf{d}_t\right\|^2$. Using the definition of $\mathbf{d}_t$ in (3), we can write

$$\left\|\nabla F_t(\mathbf{x}_t) - \mathbf{d}_t\right\|^2 = \left\|\nabla F_t(\mathbf{x}_t) - (1-\rho)\mathbf{d}_{t-1} - \rho\nabla f_t(\mathbf{x}_t, \mathbf{z}_t)\right\|^2. \quad (45)$$

Add and subtract the term $(1-\rho)\nabla F_{t-1}(x_{t-1})$ to inside the norm to the right hand side of (45) and after rearranging the terms, we get

$$\begin{aligned} & \left\|\nabla F_t(\mathbf{x}_t) - \mathbf{d}_t\right\|^2 \\ & = \|\rho(\nabla F_t(\mathbf{x}_t) - \nabla f_t(\mathbf{x}_t, \mathbf{z}_t)) + (1-\rho)(\nabla F_t(\mathbf{x}_t) - \nabla F_{t-1}(\mathbf{x}_{t-1})) \\ & \quad + (1-\rho)(\nabla F_{t-1}(\mathbf{x}_{t-1}) - \mathbf{d}_{t-1})\|^2. \end{aligned} \quad (46)$$

Next, on expanding the square on right hand side, we obtain

$$\begin{aligned}&\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2 \quad (47)\\&=\left\|\rho(\nabla F_t(\mathbf{x}_t)-\nabla f_t(\mathbf{x}_t,\mathbf{z}_t))\right\|^2+\left\|(1-\rho)(\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1}))\right\|^2\\&\quad+\left\|(1-\rho)(\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1})\right\|^2\\&\quad+2\rho(1-\rho)\langle\nabla F_t(\mathbf{x}_t)-\nabla f_t(\mathbf{x}_t,\mathbf{z}_t),\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\rangle\\&\quad+2(1-\rho)^2\langle\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1}),\nabla f_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\rangle\\&\quad+2(1-\rho)^2\langle\nabla F_t(\mathbf{x}_t)-\nabla f_t(\mathbf{x}_t,\mathbf{z}_t),\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\rangle.\end{aligned}$$

Let us define a sigma algebra $\mathcal{F}_t$ such that it represents the algorithm history which contains $\{\mathbf{x}_k\}_{k=1}^t$ and $\{\mathbf{z}_k\}_{k=1}^{t-1}$. This implies that the conditional expectation of the unbiased stochastic gradient estimate is equal to $\mathbb{E}[\nabla f(\mathbf{x}_t,\mathbf{z}_t)|\mathcal{F}_t]=\nabla F_t(\mathbf{x}_t)$. Now compute the conditional expectation $\mathbb{E}[\cdot|\mathcal{F}_t]$ on both sides of (47), we obtain

$$\begin{aligned}&\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2 \mid \mathcal{F}_t\right]=\mathbb{E}\left[\left\|\rho(\nabla F_t(\mathbf{x}_t)-\nabla f_t(\mathbf{x}_t,\mathbf{z}_t))\right\|^2 \mid \mathcal{F}_t\right]\\&\quad+\left\|(1-\rho)(\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1}))\right\|^2\\&\quad+\left\|(1-\rho)(\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1})\right\|^2 \quad (48)\\&\quad+2(1-\rho)^2\langle\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1}),\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\rangle.\end{aligned}$$

From Young's inequality, it holds that

$$\begin{aligned}&\langle\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1}),\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\rangle \quad (49)\\&\leq\frac{\beta}{2}\left\|\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\right\|^2+\frac{1}{2\beta}\left\|\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\right\|^2.\end{aligned}$$

Using upper bound from (49) into (48) and considering $\beta=\rho$, utilize results $(1-\rho^2)<1$, and $\left(1+\frac{1}{\rho}\right)(1-\rho)<\frac{1}{\rho}$, we get

$$\begin{aligned}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2|\mathcal{F}_t\right]&\leq\rho^2\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\nabla f_t(\mathbf{x}_t,\mathbf{z}_t)\right\|^2|\mathcal{F}_t\right] \quad (50)\\&\quad+((1-\rho)/\rho)\left\|\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\right\|^2\\&\quad+(1-\rho)\left\|\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\right\|^2.\end{aligned}$$

After taking the total expectation on both the sides of (50), we obtain the result stated in Lemma 2. □

Appendix C
Proof for Corollary 1

Proof. First, taking the summation over t on both sides of (50), second, taking the term $(1-\rho)\sum_{t=1}^{T-1}\mathbb{E}[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2]$ to left hand side, and then using $\mathbb{E}\left[\left\|\nabla F_T(\mathbf{x}_T)-\mathbf{d}_T\right\|^2\right]>\rho\mathbb{E}\left[\left\|\nabla F_T(\mathbf{x}_T)-\mathbf{d}_T\right\|^2\right]$ since $\rho<1$, we get

$$\begin{aligned}\rho\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2\right]&\leq\rho^2\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\nabla f_t(\mathbf{x}_t,\mathbf{z}_t)\right\|^2\right]\\&\quad+\frac{1-\rho}{\rho}\sum_{t=1}^{T}\left\|\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\right\|^2\\&\quad+(1-\rho)\left\|\nabla F_0(\mathbf{x}_0)-\mathbf{d}_0\right\|^2.\end{aligned} \quad (51)$$

Divide both the sides by ρ in (51), use the initialization $\mathbf{d}_0=\mathbf{0}$, utilize the bounded variance assumption of Assumption 3, and invoking the definition of D_T, we obtain

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2\right]\leq\rho T\sigma^2+\frac{1-\rho}{\rho^2}D_T+\frac{1-\rho}{\rho}\left\|\nabla F_0(\mathbf{x}_0)\right\|^2. \quad (52)$$

Note that

$$\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|\right]\leq\sqrt{T\sum_{t=1}^{T}\left(\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|\right]\right)^2}. \quad (53)$$

and using the inequality $[\mathbb{E}[X]]^2\leq\mathbb{E}\left[X^2\right]$, we get

$$\sqrt{T\sum_{t=1}^{T}\left(\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|\right]\right)^2}\leq\sqrt{T\sum_{t=1}^{T}\mathbb{E}\left[\left\|\nabla F_t(\mathbf{x}_t)-\mathbf{d}_t\right\|^2\right]}. \quad (54)$$

From (54), we obtain the statement of Corollary 1. □

Appendix D
Proof for Lemma 3

Proof. We remark that the statement of Lemma 3 is similar to Lemma 1 except for a $\mathbf{d}_t$ dependent term in the right hand side. Hence, the proof of Lemma 3 is almost similar to the one for Lemma 1 and the major steps which are changed are explained next. Adding and subtracting the term $\gamma\langle\mathbf{d}_{t-1},\mathbf{v}_{t-1}-\mathbf{x}_{t-1}\rangle$ (note that we use $\mathbf{d}_{t-1}$ instead of $\nabla F_{t-1}(\mathbf{x}_{t-1})$ in the proof of Lemma 1) to the right hand side of (34) and after rearranging, we obtain

$$\begin{aligned}F_t(\mathbf{x}_t)\leq&F_t(\mathbf{x}_{t-1})+\gamma\langle\nabla F_t(\mathbf{x}_{t-1})-\mathbf{d}_{t-1},\mathbf{v}_{t-1}-\mathbf{x}_{t-1}\rangle\\&+\gamma\mathbf{d}_{t-1}^T(\mathbf{v}_{t-1}-\mathbf{x}_{t-1})+\gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1}-\mathbf{x}_{t-1}\right\|^2. \quad (55)\end{aligned}$$

Next, utilizing the optimality condition i.e.

$$\langle\mathbf{x}_{t-1}^\star,\mathbf{d}_{t-1}\rangle\geq\min_{v\in\mathcal{X}}\langle\mathbf{v},\mathbf{d}_{t-1}\rangle=\langle\mathbf{v}_{t-1},\mathbf{d}_{t-1}\rangle, \quad (56)$$

into (35), we get

$$\begin{aligned}F_t(\mathbf{x}_t)\leq&F_t(\mathbf{x}_{t-1})+\gamma(\nabla F_t(\mathbf{x}_{t-1})-\mathbf{d}_{t-1})^T(\mathbf{v}_{t-1}-\mathbf{x}_{t-1})\\&+\gamma\mathbf{d}_{t-1}^T(\mathbf{x}_{t-1}^\star-\mathbf{x}_{t-1})+\gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1}-\mathbf{x}_{t-1}\right\|^2. \quad (57)\end{aligned}$$

Utilizing the similar steps used in (36) to (44), we obtain

$$\begin{aligned}&F_t(\mathbf{x}_t)-F_t(\mathbf{x}_t^\star) \quad (58)\\&\leq F_{t,t-1}^{\sup}(\mathbf{x})+(1-\gamma)\left(F_{t-1}(\mathbf{x}_{t-1})-F_{t-1}(\mathbf{x}_{t-1}^\star)\right)+F_{t-1}(\mathbf{x}_{t-1}^\star)-F_t(\mathbf{x}_t^\star)\\&\quad+\gamma\left(L_1\left\|\mathbf{x}_{t-1}-\mathbf{x}_t\right\|+\left\|\nabla F_t(\mathbf{x}_t)-\nabla F_{t-1}(\mathbf{x}_{t-1})\right\|\right)\left\|\mathbf{v}_{t-1}-\mathbf{x}_{t-1}\right\|\\&\quad+\gamma\left\|\nabla F_{t-1}(\mathbf{x}_{t-1})-\mathbf{d}_{t-1}\right\|\left\|\mathbf{v}_{t-1}-\mathbf{x}_{t-1}^\star\right\|+\gamma^2\frac{L_1}{2}\left\|\mathbf{v}_{t-1}-\mathbf{x}_{t-1}\right\|^2.\end{aligned}$$

Using the update step of $\mathbf{x}_t$ and applying the upper bound from Assumption 1 in (58), we obtain the result of Lemma 3. □

Appendix E
Proof of Lemma 4

Proof. 4 Using the definition of $\mathbf{d}_t^k$ from the update step 9 of Algorithm 2, we can write

$$\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2=\left\|\nabla F_t(\mathbf{x}_t^k)-(1-\rho)\mathbf{d}_t^{k-1}-\rho\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k)\right\|^2. \quad (59)$$

Add and subtract the term $(1-\rho)\nabla F_t(\mathbf{x}_t^{k-1})$ to (59) as follows

$$\begin{aligned}\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2=&\|\nabla F_t(\mathbf{x}_t^k)-(1-\rho)\nabla F_t(\mathbf{x}_t^{k-1}) \quad (60)\\&+(1-\rho)\nabla F_t(\mathbf{x}_t^{k-1})-(1-\rho)\mathbf{d}_t^{k-1}\\&-\rho\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k)\|^2.\end{aligned}$$

Rearranging and expanding the squares, the right hand side of (59) can be written as

$$\begin{aligned}\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2 &= \rho\left\|\nabla F_t(\mathbf{x}_t^k)-\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k)\right\|^2\\ &+(1-\rho)^2\left\|\nabla F_t(\mathbf{x}_t^k)-\nabla F_t(\mathbf{x}_t^{k-1})\right\|^2+(1-\rho)^2\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2\\ &+2\rho(1-\rho)\langle\nabla F_t(\mathbf{x}_t^k)-\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k),\nabla F_t(\mathbf{x}_t^k)-\nabla F_t(\mathbf{x}_t^{k-1})\rangle\\ &+2(1-\rho)^2\langle\nabla F_t(\mathbf{x}_t^k)-\nabla F_t(\mathbf{x}_t^{k-1}),\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}^{k-1}\rangle\\ &+2(1-\rho)^2\langle\nabla F_t(\mathbf{x}_t)-\nabla f(\mathbf{x}_t^k,\mathbf{z}_t^k),\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}_t^{k-1}\rangle.\end{aligned}$$

Now compute the conditional expectation $\mathbb{E}\left[(.)|\mathcal{F}_t^k\right]$ for both sides, where $\mathcal{F}_t^k$ denotes the filtration containing algorithm history $\{\mathbf{x}_1^1,\mathbf{x}_1^2,\cdots,\mathbf{x}_1^K,\cdots,\mathbf{x}_t^1,\cdots,\mathbf{x}_t^k\}$ and $\{\mathbf{z}_1^1,\mathbf{z}_1^2,\cdots,\mathbf{z}_1^K,\cdots,\mathbf{z}_t^1,\cdots,\mathbf{z}_t^{k-1}\}$. Note that we have $\mathbb{E}\left[\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k)|\mathcal{F}_t^k\right]=\nabla F_t(\mathbf{x}_t^k)$, we obtain

$$\begin{aligned}\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2 \mid \mathcal{F}_t^k\Big] &= \mathbb{E}\Big[\Big[\left\|\rho(\nabla F_t(\mathbf{x}_t^k)-\nabla f_t(\mathbf{x}_t^k,\mathbf{z}_t^k))\right\|^2\\ &+\left\|(1-\rho)(\nabla F_t(\mathbf{x}_t^k)-\nabla F_t(\mathbf{x}_t^{k-1}))\right\|^2+\left\|(1-\rho)(\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}_t^{k-1})\right\|^2\\ &+2(1-\rho)^2\langle\nabla F_t(\mathbf{x}_t^k)-\nabla F_t(\mathbf{x}_t^{k-1}),\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}_t^{k-1}\rangle\Big]\mid\mathcal{F}_t^k\Big]. \quad (61)\end{aligned}$$

Using analogous logic to that which proceeds from (49)-(50), we may conclude Lemma 4. □

APPENDIX F
PROOF OF COROLLARY 2

Proof. From Assumption 1-3 and using them into the statement of Lemma 4, we get

$$\begin{aligned}\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2\Big] &\le \rho^2\sigma^2+\frac{L_1^2(1-\rho)}{\rho}\mathbb{E}\left\|\mathbf{x}_t^k-\mathbf{x}_t^{k-1}\right\|^2 \quad (62)\\ &+(1-\rho)\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}_t^{k-1}\right\|^2\Big].\end{aligned}$$

Utilizing the update in step 7 of Algorithm 2, we get

$$\begin{aligned}\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2\Big] &\le \rho^2\sigma^2+\frac{L_1^2(1-\rho)}{\rho}\gamma^2\mathbb{E}\left\|\mathbf{x}_t^{k-1}-\mathbf{v}_t^{k-1}\right\|^2\\ &+(1-\rho)\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^{k-1})-\mathbf{d}_t^{k-1}\right\|^2\Big]. \quad (63)\end{aligned}$$

Taking summation w.r.t. k and performing further simplifications, we get

$$\begin{aligned}\rho\sum_{k=1}^{K}\mathbb{E}\Big[\left\|\nabla F_t(\mathbf{x}_t^k)-\mathbf{d}_t^k\right\|^2\Big] &\le \rho^2\sigma^2K+\frac{L_1^2(1-\rho)}{\rho}\gamma^2\sum_{k=1}^{K}\mathbb{E}\left\|\mathbf{x}_t^{k-1}-\mathbf{v}_t^{k-1}\right\|^2\\ &+(1-\rho)\left\|\nabla F_t(\mathbf{x}_t^0)\right\|^2. \quad (64)\end{aligned}$$

Using Assumption 1 i.e. the boundedness of set $\mathcal{X}$, and taking ρ to the right hand side, we get the required result.

□

APPENDIX G
PROOF FOR LEMMA 5

Proof. Based on L_1-smoothness of function F_t (Assumption 1) at any time instant and the definition of the update, we have

$$\begin{aligned}F_t(\mathbf{x}_t^{k+1}) &= F_t(\mathbf{x}_t^{(k)}+\gamma(\mathbf{v}_t^{(k)}-\mathbf{x}_t^{(k)}))\\ &\le F_t(\mathbf{x}_t^{(k)})+\gamma\langle\nabla F_t(\mathbf{x}_t^{(k)}),\mathbf{v}_t^{(k)}-\mathbf{x}_t^{(k)}\rangle+\gamma^2\frac{L_1}{2}\left\|\mathbf{v}_t^{(k)}-\mathbf{x}_t^{(k)}\right\|^2\\ &\le F_t(\mathbf{x}_t^{(k)})+\gamma\langle\nabla F_t(\mathbf{x}_t^{(k)}),\mathbf{v}_t^{(k)}-\mathbf{x}_t^{(k)}\rangle+\gamma^2\frac{L_1}{2}D^2. \quad (65)\end{aligned}$$

We note that

$$\begin{aligned}\langle\nabla F_t(\mathbf{x}_t^{(k)}),\mathbf{v}_t^{(k)}-\mathbf{x}_t^{(k)}\rangle &= \langle\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle \quad (66)\\ &+\langle\nabla F_t(\mathbf{x}_t^{(k)}),\mathbf{x}_t^{\star}-\mathbf{x}_t^{(k)}\rangle+\langle\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle.\end{aligned}$$

Using Young's inequality for any $\beta>0$, it holds that

$$\begin{aligned}&\langle\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle\\ &\le\frac{1}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)}\right\|^2+\frac{\beta}{2}\left\|\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\right\|^2. \quad (67)\end{aligned}$$

From Assumption 1, i.e. the boundedness of set $\mathcal{X}$, we get

$$\langle\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle\le\frac{1}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)}\right\|^2+\frac{\beta}{2}D^2. \quad (68)$$

We know that $\nabla F_t(\mathbf{x}_t^{(k)})^T(\mathbf{x}_t^{\star}-\mathbf{x}_t^{(k)})$ is upper bounded by $F_t(\mathbf{x}_t^{\star})-F_t(\mathbf{x}_t^{(k)})$ via the first-order characterization of the convexity of F_t. Using this upper bound and substituting (68) into (66), we get

$$\begin{aligned}\langle\nabla F_t(\mathbf{x}_t^{(k)}),\mathbf{v}_t^{(k)}-\mathbf{x}_t^{(k)}\rangle &= \frac{1}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)}\right\|^2+\frac{\beta}{2}D^2+F_t(\mathbf{x}_t^{\star})\\ &-F_t(\mathbf{x}_t^{(k)})+\langle\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle. \quad (69)\end{aligned}$$

Using the upper bound from (69) into (65), it holds that

$$\begin{aligned}F_t(\mathbf{x}_t^{k+1}) &\le F_t(\mathbf{x}_t^{(k)})+\frac{\gamma}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)}\right\|^2+\gamma\frac{\beta}{2}D^2 \quad (70)\\ &+\gamma\langle\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle+\gamma F_t(\mathbf{x}_t^{\star})-\gamma F_t(\mathbf{x}_t^{(k)})+\gamma^2\frac{L_1}{2}D^2.\end{aligned}$$

Subtracting $F_t(\mathbf{x}_t^{\star})$ from both sides, we get

$$\begin{aligned}&F_t(\mathbf{x}_t^{k+1})-F_t(\mathbf{x}_t^{\star}) \quad (71)\\ &\le(1-\gamma)(F_t(\mathbf{x}_t^{(k)})-F_t(\mathbf{x}_t^{\star}))+\frac{\gamma}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^{(k)})-\mathbf{d}_t^{(k)}\right\|^2\\ &+\gamma\frac{\beta}{2}D^2+\gamma\langle\mathbf{d}_t^{(k)},\mathbf{v}_t^{(k)}-\mathbf{x}_t^{\star}\rangle+\gamma^2\frac{L_1}{2}D^2.\end{aligned}$$

which is as stated in Lemma 5. □

APPENDIX H
PROOF FOR THEOREM 3

Proof. Let $k=K$ in (71), we get

$$\begin{aligned}F_t(\mathbf{x}_t^K)-F_t(\mathbf{x}_t^{\star}) &\le(1-\gamma)\left(F_t(\mathbf{x}_t^{(K)})-F_t(\mathbf{x}_t^{\star})\right)\\ &+\frac{\gamma}{2\beta}\left\|\nabla F_t(\mathbf{x}_t^{(K)})-\mathbf{d}_t^{(K)}\right\|^2+\gamma\frac{\beta}{2}D^2\\ &+\gamma\langle\mathbf{d}_t^k,\mathbf{v}_t^k-\mathbf{x}_t^{\star}\rangle+\gamma^2\frac{L_1}{2}D^2. \quad (72)\end{aligned}$$

On expanding the term $F_t(\mathbf{x}_t^{(K)}) - F_t(\mathbf{x}_t^\star)$, we get

$$F_t(\mathbf{x}_t^K) - F_t(\mathbf{x}_t^\star) \leq (1-\gamma)^K \left(F_t(\mathbf{x}_t^{(1)}) - F_t(\mathbf{x}_t^\star) \right) + \frac{\gamma}{2\beta} \sum_{k=1}^K \left\| \nabla F_t(\mathbf{x}_t^k) - \mathbf{d}_t^k \right\|^2 + K\gamma \frac{\beta}{2} D^2 + \gamma \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + K\gamma^2 \frac{L_1}{2} D^2. \tag{73}$$

As $\mathbf{x}_t^1 = \mathbf{x}_{t-1}$, and $\mathbf{x}_t^K = \mathbf{x}_t$, we can write

$$F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \leq (1-\gamma)^K \left(F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}_t^\star) \right) + \frac{\gamma}{2\beta} \sum_{k=1}^K \left\| \nabla F_t(\mathbf{x}_t^k) - \mathbf{d}_t^k \right\|^2 + K\gamma \frac{\beta}{2} D^2 + \gamma \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + K\gamma^2 \frac{L_1}{2} D^2. \tag{74}$$

Next, adding and subtracting $F_{t-1}(\mathbf{x}_{t-1}) + F_{t-1}(\mathbf{x}_{t-1}^\star)$ to the term $F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}_t^\star)$ and after rearranging, we get

$$F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}_t^\star) = F_t(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}) + F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}^\star) + F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star). \tag{75}$$

In order to bring the right hand side of (75) in the form of V_T, we simplify the equation in (75) as

$$\begin{aligned} F_t(\mathbf{x}_{t-1}) - F_t(\mathbf{x}_t^\star) &\leq |F_t(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1})| + F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}^\star) + F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star) \\ &\leq F_{t,t-1}^{\sup}(\mathbf{x}) + F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}^\star) + F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star). \end{aligned} \tag{76}$$

Using (76) into (74), we get

$$F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \leq (1-\gamma)^K \left(F_{t,t-1}^{\sup}(\mathbf{x}) + F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}^\star)) \right) + (1-\gamma)^K \left(F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star) \right) + \frac{\gamma}{2\beta} \sum_{k=1}^K \left\| \nabla F_t(\mathbf{x}_t^k) - \mathbf{d}_t^k \right\|^2 + K\gamma \frac{\beta}{2} D^2 + \gamma \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + K\gamma^2 \frac{L_1}{2} D^2. \tag{77}$$

Take sum over t and compute the total expectation, we get

$$\begin{aligned} \sum_{t=1}^T \mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \right] &\leq (1-\gamma)^K \sum_{t=1}^T \left(F_{t,t-1}^{\sup}(\mathbf{x}) + \mathbb{E}\left[F_{t-1}(\mathbf{x}_{t-1}) - F_{t-1}(\mathbf{x}_{t-1}^\star) \right] \right) \\ &+ (1-\gamma)^K \sum_{t=1}^T \mathbb{E}\left[F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star) \right] \\ &+ \frac{\gamma}{2\beta} \sum_{t=1}^T \sum_{k=1}^K \mathbb{E}\left[\left\| \nabla F_t(\mathbf{x}_t^k) - \mathbf{d}_t^k \right\|^2 \right] + KT\gamma \frac{\beta}{2} D^2 \\ &+ \gamma \sum_{t=1}^T \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + KT\gamma^2 \frac{L_1}{2} D^2. \end{aligned} \tag{78}$$

Using $\sum_{t=1}^T \mathbb{E}\left[F_{t-1}(\mathbf{x}_{t-1}^\star) - F_t(\mathbf{x}_t^\star) \right] = F_0(\mathbf{x}_0^\star) - F_T(\mathbf{x}_T^\star)$, and rearranging the terms, we obtain

$$\begin{aligned} &\left(1 - (1-\gamma)^K\right) \sum_{t=1}^{T-1} \mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \right] + \mathbb{E}\left[F_T(\mathbf{x}_T) - F_T(\mathbf{x}_T^\star) \right] \\ &\leq (1-\gamma)^K \left(\sum_{t=1}^T F_{t,t-1}^{\sup}(\mathbf{x}) + F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star) \right) \\ &+ \frac{\gamma}{2\beta} \sum_{t=1}^T \sum_{k=1}^K \mathbb{E}\left[\left\| \nabla F_t(\mathbf{x}_t^k) - \mathbf{d}_t^k \right\|^2 \right] + KT\gamma \frac{\beta}{2} D^2 \\ &+ \gamma \sum_{t=1}^T \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + KT\gamma^2 \frac{L_1}{2} D^2. \end{aligned} \tag{79}$$

Since, $\left(1 - (1-\gamma)^K\right) < 1$, we can simplify the left hand side and further utilizing Corollary 2, we get

$$\begin{aligned} &\left(1 - (1-\gamma)^K\right) \sum_{t=1}^{T} \mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \right] \\ &\leq (1-\gamma)^K \left(\sum_{t=1}^T F_{t,t-1}^{\sup}(\mathbf{x}) + F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star) \right) \\ &+ \frac{\gamma}{2\beta} \sum_{t=1}^T \left(\rho\sigma^2 K + \frac{L_1^2(1-\rho)}{\rho^2} \gamma^2 K D^2 + \frac{(1-\rho)}{\rho} \left\| \nabla F_t(\mathbf{x}_t^0) \right\|^2 \right) \\ &+ KT\gamma \frac{\beta}{2} D^2 + \gamma \sum_{t=1}^T \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + KT\gamma^2 \frac{L_1}{2} D^2. \end{aligned} \tag{80}$$

From the definition of V_T and further simplification results in

$$\begin{aligned} &\left(1 - (1-\gamma)^K\right) \sum_{t=1}^{T} \mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \right] \\ &\leq (1-\gamma)^K \left(V_T + F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star) \right) + \frac{\gamma\rho}{2\beta} TK\sigma^2 \\ &+ \frac{L_1^2(1-\rho)\gamma^3}{2\beta\rho^2} TKD^2 + \frac{(1-\rho)\gamma}{2\beta\rho} T \left\| \nabla F_t(\mathbf{x}_t^0) \right\|^2 \\ &+ KT\gamma \frac{\beta}{2} D^2 + \gamma \sum_{t=1}^T \sum_{k=1}^K \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle + KT\gamma^2 \frac{L_1}{2} D^2. \end{aligned} \tag{81}$$

For a fixed k, the sequence $\{\mathbf{v}_t^k\}_{t=1}^T$ is produced by a online linear minimization oracle such that

$$\sum_{t=1}^T \langle \mathbf{d}_t^k, \mathbf{v}_t^k - \mathbf{x}_t^\star \rangle \leq \sum_{t=1}^T \langle \mathbf{d}_t^k, \mathbf{v}_t^k \rangle - \sum_{t=1}^T \min_{\mathbf{x}\in\mathcal{X}} \langle \mathbf{d}_t^k, \mathbf{x} \rangle \leq \sum_{t=1}^T \mathcal{R}_t^{\mathcal{E}}. \tag{82}$$

Using (82) in (81)

$$\begin{aligned} &\left(1 - (1-\gamma)^K\right) \sum_{t=1}^{T} \mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star) \right] \\ &\leq (1-\gamma)^K \left(V_T + F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star) \right) + \frac{\gamma\rho}{2\beta} TK\sigma^2 \\ &+ \frac{L_1^2(1-\rho)\gamma^3}{2\beta\rho^2} TKD^2 + \frac{(1-\rho)\gamma}{2\beta\rho} T \left\| \nabla F_t(\mathbf{x}_t^0) \right\|^2 \\ &+ KT\gamma \frac{\beta}{2} D^2 + \gamma \sum_{k=1}^K \sum_{t=1}^T \mathcal{R}_t^{\mathcal{E}} + KT\gamma^2 \frac{L_1}{2} D^2. \end{aligned} \tag{83}$$

Let $K_1 := F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star)$, $B_1 := (1-\rho)\left\|\nabla F_t(\mathbf{x}_t^0)\right\|^2$, $C := L_1^2(1-\rho)D^2 := C$, and taking $\left(1-(1-\gamma)^K\right)$ to the right hand side, we obtain

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right] \\
&\leq \frac{1}{\left(1-(1-\gamma)^K\right)}\Big[(1-\gamma)^K\left(V_T + K_1\right) + \frac{\gamma\rho}{2\beta}TK\sigma^2 \\
&\quad + \frac{\gamma^3}{2\beta\rho^2}TKC + \frac{\gamma}{2\beta\rho}TB + KT\gamma\frac{\beta}{2}D^2 + \gamma\sum_{k=1}^{K}\sum_{t=1}^{T}\mathcal{R}_t^{\mathcal{E}} \\
&\quad + KT\gamma^2\frac{L_1D^2}{2}\Big].
\end{aligned} \tag{84}$$

which is as stated in Theorem 3. The term $\frac{1}{(1-(1-\gamma)^K)}$ From the expansion of $(1-x)^{-1} = 1 + x + x^2 + x^3 + \cdots$, we can say that $\frac{1}{(1-(1-\gamma)^K)} = 1 + (1-\gamma)^K + (1-\gamma)^{2K} + \cdots$ the terms $(1-\gamma)^K$ are diminishing and decreases with increase in K. Thus, with increase in K the term $\left(1-(1-\gamma)^K\right)$ approaches closer to 1. Taking $\gamma = \frac{1}{K}$, $\beta = \frac{1}{T^b}$, $\rho = \frac{1}{T^c}$, and using $\left(1-\frac{1}{K}\right)^K \leq \frac{1}{e}$ we get

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right] \\
&\leq \frac{1}{\left(1-\frac{1}{e}\right)}\Big[\frac{1}{e}\left(V_T + K_1\right) + \frac{T^{(1-c+b)}}{2}\sigma^2 + \frac{T^{(1+b+2c)}}{2K^2}C \\
&\quad + \frac{T^{(1+b+c)}}{2K}B + \frac{T^{(1-b)}}{2}D^2 + \frac{1}{K}\sum_{k=1}^{K}\sum_{t=1}^{T}\mathcal{R}_t^{\mathcal{E}} + \frac{T}{2K}L_1D^2\Big].
\end{aligned} \tag{85}$$

For $b = 0.5$, $c = 1$, and $K = T^a$, we have

$$\begin{aligned}
&\sum_{t=1}^{T}\mathbb{E}\left[F_t(\mathbf{x}_t) - F_t(\mathbf{x}_t^\star)\right] \\
&\leq \frac{1}{\left(1-\frac{1}{e}\right)}\Big[\frac{1}{e}\left(V_T + K_1\right) + \frac{\sqrt{T}}{2}\sigma^2 + \frac{C}{2}T^{(3.5-2a)} \\
&\quad + \frac{T^{(2.5-a)}}{2}B + \frac{\sqrt{T}}{2}D^2 + \mathcal{R}_T^{\mathcal{E}} + \frac{L_1D^2}{2}T^{(1-a)}\Big].
\end{aligned} \tag{86}$$

where, $\mathcal{R}_T^{\mathcal{E}} = \sum_{t=1}^{T}\mathcal{R}_t^{\mathcal{E}}$

□

Appendix I
Dynamic Regret of Meta Frank Wolfe algorithm (When full gradient is available)

With exact gradient available, the step 9 of Algo. 2 becomes $\mathbf{d}_t^k = \nabla F_t(\mathbf{x}_t)$ for all k. Since, exact gradient is available the error in gradient approximation becomes zero and similar to (84) we have

$$\begin{aligned}
\mathbf{Reg}_T^D \leq \frac{1}{\left(1-(1-\gamma)^K\right)}\Big[&(1-\gamma)^K\Big(\sum_{t=1}^{T}F_{t,t-1}^{\sup}(\mathbf{x}) + K_1\Big) \\
&+ \gamma\sum_{t=1}^{T}\sum_{k=1}^{K}\mathcal{R}_t^{\mathcal{E}} + KT\gamma^2\frac{L_1}{2}D^2\Big].
\end{aligned} \tag{87}$$

where, $K_1 := F_0(\mathbf{x}_0) - F_T(\mathbf{x}_T^\star)$, $F_{t,t-1}^{\sup}(\mathbf{x}) := \sup_{\mathbf{x}\in\mathcal{X}}|F_t(\mathbf{x}) - F_{t-1}(\mathbf{x})|$, and the online linear optimization oracles have regret $\mathcal{R}_t^{\mathcal{E}}$ at instant t for any k. Let $\gamma = \frac{1}{K}$, $K = T^a$, and using the inequality $\left(1-\frac{1}{K}\right)^K \leq \frac{1}{e}$ we get

$$\begin{aligned}
\mathbf{Reg}_T^D \leq \frac{1}{1-\left(1-\frac{1}{e}\right)}\Big[&\Big(1-\frac{1}{e}\Big)\left(V_T + K_1\right) + \mathcal{R}_T^{\mathcal{E}} \\
&+ \frac{L_1D^2}{2}T^{(1-a)}\Big].
\end{aligned} \tag{88}$$

where, $\mathcal{R}_T^{\mathcal{E}} = \sum_{t=1}^{T}\mathcal{R}_t^{\mathcal{E}}$ and $V_T := \sum_{t=1}^{T}F_{t,t-1}^{\sup}(\mathbf{x})$ as defined in (2). Finally, we get

$$\mathbf{Reg}_T^D = \mathcal{O}\left(1 + V_T + \mathcal{R}_T^{\mathcal{E}} + T^{(1-a)}\right). \tag{89}$$

References

[1] G. Hinton, L. Deng, D. Yu, G. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, B. Kingsbury *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, 2012.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[3] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv:1509.02971*, 2015.

[4] D. Calandriello, A. Lazaric, and M. Valko, "Second-order kernel online convex optimization with adaptive sketching," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. International Convention Centre, Sydney, Australia: PMLR, 06–11 Aug 2017, pp. 645–653.

[5] A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "Parsimonious online learning with kernels via sparse projections in function space," *The Journal of Machine Learning Research*, vol. 20, no. 1, pp. 83–126, 2019.

[6] A. Agarwal, E. Hazan, S. Kale, and R. E. Schapire, "Algorithms for portfolio management based on the newton method," in *Proceedings of the 23rd International Conference on Machine Learning*. ACM, 2006, pp. 9–16.

[7] N. Wagener, C.-A. Cheng, J. Sacks, and B. Boots, "An online learning approach to model predictive control," *arXiv preprint arXiv:1902.08967*, 2019.

[8] J. Friedman, T. Hastie, and R. Tibshirani, *The elements of statistical learning*. Springer Series in Statistics New York, 2001, vol. 1.

[9] A. Shapiro, D. Dentcheva *et al.*, *Lectures on Stochastic Programming: Modeling and Theory*. Siam, 2014, vol. 16.

[10] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press Cambridge, 1998, vol. 1, no. 1.

[11] K. J. Aström and R. M. Murray, *Feedback systems: an introduction for scientists and engineers*. Princeton university press, 2010.

[12] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," *arXiv preprint arXiv:1511.05952*, 2015.

[13] O. Besbes, Y. Gur, and A. Zeevi, "Non-stationary stochastic optimization," *Operations Research*, vol. 63, no. 5, pp. 1227–1244, 2015. [Online]. Available: https://doi.org/10.1287/opre.2015.1408

[14] V. S. Borkar, *Stochastic approximation: a dynamical systems viewpoint*. Springer, 2009, vol. 48.

[15] M. Mohri and A. Rostamizadeh, "Stability bounds for stationary φ-mixing and β-mixing processes," *Journal of Machine Learning Research*, vol. 11, no. Feb, pp. 789–814, 2010.

[16] A. Jadbabaie, A. Rakhlin, S. Shahrampour, and K. Sridharan, "Online optimization: Competing with dynamic comparators," in *Artificial Intelligence and Statistics*, 2015, pp. 398–406.

[17] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. 20th Int. Conf. on Machine Learning*, vol. 20, no. 2, Washington DC, USA, Aug. 21-24 2003, pp. 928–936.

[18] A. Simonetto, A. Mokhtari, A. Koppel, G. Leus, and A. Ribeiro, "A class of prediction-correction methods for time-varying convex optimization," *IEEE Transactions on Signal Processing*, vol. 64, no. 17, pp. 4576–4591.

[19] A. Mokhtari, S. Shahrampour, A. Jadbabaie, and A. Ribeiro, "Online optimization in dynamic environments: Improved regret rates for strongly convex problems," in *IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 7195–7201.

[20] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Machine Learning*, vol. 69, no. 2-3, pp. 169–192, 2007.

[21] E. Hazan *et al.*, "Introduction to online convex optimization," *Foundations and Trends® in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2016.

[22] H. Wang and A. Banerjee, "Online alternating direction method," in *Proceedings of the 29th International Conference on Machine Learning*. Omnipress, 2012, pp. 1699–1706.

[23] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: online convex optimization with long term constraints," *Journal of Machine Learning Research*, vol. 13, no. Sep, pp. 2503–2528, 2012.

[24] R. T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM journal on control and optimization*, vol. 14, no. 5, pp. 877–898, 1976.

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

[26] E. Hazan and S. Kale, "Projection-free online learning," in *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ser. ICML'12. USA: Omnipress, 2012, pp. 1843–1850. [Online]. Available: http://dl.acm.org/citation.cfm?id=3042573.3042808

[27] E. Hazan and H. Luo, "Variance-reduced and projection-free stochastic optimization." in *Proceesing of ICML*, vol. 16, 2016, pp. 1263–1271.

[28] E. C. Hall and R. M. Willett, "Online convex optimization in dynamic environments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 647–662, 2015.

[29] L. Chen, C. Harshaw, H. Hassani, and A. Karbasi, "Projection-free online optimization with stochastic gradient: From convexity to submodularity," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. Stockholmsmässan, Stockholm Sweden: PMLR, 10–15 Jul 2018, pp. 814–823. [Online]. Available: http://proceedings.mlr.press/v80/chen18c.html

[30] A. Mokhtari, H. Hassani, and A. Karbasi, "Conditional gradient method for stochastic submodular maximization: Closing the gap," in *International Conference on Artificial Intelligence and Statistics*, 2018, pp. 1886–1895.

[31] ——, "Stochastic conditional gradient methods: From convex minimization to submodular maximization," *arXiv preprint arXiv:1804.09554*, 2018.

[32] A. Mokhtari, A. Koppel, and A. Ribeiro, "A class of parallel doubly stochastic algorithms for large-scale learning," *Journal of Machine Learning Research (submitted)*, 2016.

[33] A. Koppel, A. Mokhtari, and A. Ribeiro, "Parallel stochastic successive convex approximation method for large-scale dictionary learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2771–2775.

[34] G. Wang, H. Chen, Y. Li, and M. Jin, "On received-signal-strength based localization with unknown transmit power and path loss exponent," *IEEE Wireless Communications Letters*, vol. 1, no. 5, pp. 536–539, 2012.

[35] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause, "Learning-based model predictive control for safe exploration," in *IEEE Conference on Decision and Control (CDC)*, 2018, pp. 6059–6066.

[36] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Transactions on Automatic Control*, vol. 63, no. 3, pp. 714–725, 2018.

[37] N. Goyette, P.-M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "Changedetection. net: A new change detection benchmark dataset," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 1–8.