The Online Saddle Point Problem: Applications to Online Convex Optimization with Knapsacks

Adrian Rivera Cardoso
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332
adrian.riv@gatech.edu

He Wang
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332
he.wang@isye.gatech.edu

Huan Xu
School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332
huan.xu@isye.gatech.edu

Abstract

We study the online saddle point problem, an online learning problem where at each iteration a pair of actions need to be chosen without knowledge of the current and future (convex-concave) payoff functions. The objective is to minimize the gap between the cumulative payoffs and the saddle point value of the aggregate payoff function, which we measure using a metric called “SP-regret”. The problem generalizes the online convex optimization framework and can be interpreted as finding the Nash equilibrium for the aggregate of a sequence of two-player zero-sum games. We propose an algorithm that achieves \( \tilde{O}(\sqrt{T}) \) SP-regret in the general case, and \( O(\log T) \) SP-regret for the strongly convex-concave case. We then consider an online convex optimization with knapsacks problem motivated by a wide variety of applications such as: dynamic pricing, auctions, and crowdsourcing. We relate this problem to the online saddle point problem and establish \( O(\sqrt{T}) \) regret using a primal-dual algorithm.

1 Introduction

In this paper, we study the online saddle point (OSP) problem. The OSP problem involves a sequence of two-player zero-sum convex-concave games which are selected arbitrarily by Nature. In each iteration, player 1 chooses an action to minimize its payoffs, while player 2 chooses an action to maximize its payoffs. Both players choose actions without knowledge of the current and future payoff functions. Our goal is to jointly choose a pair of actions for both players at each iteration, such that each player’s cumulative payoff at the end is as close as possible to that of the Nash equilibrium (i.e. saddle point) of the aggregate game.

More formally, we define the OSP problem as follows. There is a sequence of unknown functions \( \{\mathcal{L}_t(x, y)\}_{t=1}^T \) that are convex in \( x \in X \) and concave in \( y \in Y \). Here, \( X \) and \( Y \) are compact convex sets in Euclidean space. As a result, there exists a saddle point \((x^*, y^*) \in X \times Y\) such that
\[
\min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) = \sum_{t=1}^T \mathcal{L}_t(x^*, y) = \max_{y \in Y} \min_{x \in X} \sum_{t=1}^T \mathcal{L}_t(x, y).
\]

At each iteration \( t \), the decision makers jointly choose a pair of actions \((x_t, y_t) \in X \times Y\), and then the function \( \mathcal{L}_t \) is revealed. The goal is to design an algorithm to minimize the cumulative
saddle-point regret (SP-regret), defined as

\[
\text{SP-Regret}(T) = \left| \sum_{t=1}^{T} \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y) \right| .
\] (1)

In other words, we would like to obtain a cumulative payoff that is as close as possible to the saddle-point value if we had known all the functions \(\{\mathcal{L}_t\}_{t=1}^{T}\) in advance.

We would like to emphasize an important distinction between the OSP problem and the standard Online Convex Optimization (OCO) problem [21]. In the OCO problem, Nature selects an arbitrary sequence of convex functions \(\{f_t(\cdot)\}_{t=1}^{T}\), and the decision maker chooses an action \(x_t \in X\) before each function \(f_t(\cdot)\) is revealed. The objective is to minimize the regret defined as

\[
\sum_{t=1}^{T} f_t(x_t) - \min_{x \in X} \sum_{t=1}^{T} f_t(x).
\]

The objective in the OSP problem is to choose the actions of two players jointly such that the payoffs of both players are close to the Nash equilibrium. In contrast, OCO involves only an individual player against Nature. The OCO framework can be viewed as a special case of the OSP problem where the action set of the second player \(Y\) is a singleton. Moreover, the standard OCO setting is applicable to the OSP problem when only one of the players’ payoff is optimized at a time. To be specific, we define the individual-regret of players 1 and 2 as

\[
\text{Ind-Regret}_x(T) = \sum_{t=1}^{T} \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \sum_{t=1}^{T} \mathcal{L}_t(x, y_t), \quad (2a)
\]

\[
\text{Ind-Regret}_y(T) = \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x_t, y) - \sum_{t=1}^{T} \mathcal{L}_t(x_t, y_t). \quad (2b)
\]

The individual-regret measures each player’s own regret while fixing the other player’s actions. It is easy to see that minimizing individual-regret \((2a)\) or \((2b)\) can be cast as a standard OCO problem. However, we will show that SP-regret and individual-regret do not imply one another, so existing OCO algorithms cannot be directly applied to the OSP problem. More surprisingly, we show that any OCO algorithm with a sublinear \(o(T)\) individual-regret will inevitably have a linear \(\Omega(T)\) SP-regret in the general OSP problem (see details in [3]).

In addition to establishing general results for the OSP problem, we focus on one of its prominent applications: the online convex optimization with knapsacks (OCOwK) problem. Several variants of the OCOwK problem have recently received a lot of attention in recent literature, but we found its connection to the OSP problem has not been well exploited. We show that the OCOwK problem is closely related to the OSP problem through Lagrangian duality; thus, we are able to apply our results for the OSP problem to the OCOwK problem.

In the OCOwK problem, a decision maker is endowed with a fixed budget of resource at the beginning of \(T\) periods. In each period \(t = 1, \ldots, T\), the decision maker chooses an action \(x_t \in X\), and then the Nature reveals a reward function \(r_t\) and a budget consumption function \(c_t\). The objective is to maximize total reward \(\sum_{t=1}^{T} r_t(x_t)\) while keeping the total consumption \(\sum_{t=1}^{T} c_t(x_t)\) within the given budget.

The OCOwK model also generalizes the standard OCO problem by having an additional budget constraint. Additionally, it also has a wide range of practical applications (see more discussion in [3]), some notable examples include:

- Dynamic pricing: a retailer is selling a fixed amount of goods in a finite horizon. The actions correspond to pricing decisions, the reward is the retailer’s revenue, and the budget represents finite item inventory. The reward functions are unknown initially due to high uncertainty in customer demand.

- Online ad auction: a firm is bidding for advertising on a platform (e.g. Google) with limited daily budget. The actions refer to auction bids, and the reward represents impressions received from displayed ads. The reward function is unknown because the firm is unaware of other firms’ bidding strategies.
We first propose an algorithm called SP-FTL (Saddle-Point Follow-the-Leader) for the online saddle point problem, and show that this algorithm has a SP-Regret of \( \tilde{O}(\sqrt{T}) \), which matches the lower bound of \( \Omega(\sqrt{T}) \) up to a logarithmic factor. In the special case where the payoff function \( L_t(x, y) \) is strongly-convex in \( x \) and strongly-concave in \( y \), the algorithm has a SP-regret of \( O(\log T) \), which is optimal.

In addition, we show that no algorithm can simultaneously achieve sublinear (i.e. \( o(T) \)) SP-regret and sublinear individual-regrets (defined in Equation 1 and 2) in the general OSP problem. This impossibility result further illustrates the contribution of the SP-FTL algorithm, as existing OCO algorithms designed to achieve sublinear individual-regret are not able to achieve sublinear SP-regret.

Then, we consider the OCOwK problem. We show that this problem is related to the OSP problem by Lagrangian duality, and a sufficient condition to achieve a sublinear regret for OCOwK is that an algorithm must have both sublinear SP-regret and sublinear individual-regret for the OSP problem. In light of the previous impossibility result, we consider the OCOwK problem in a stochastic setting where the reward and consumption functions are sampled i.i.d. from some unknown distribution. By applying the SP-FTL algorithm and exploiting the connection between OCOwK and OSP problems, we obtain a \( \tilde{O}(T^{5/6}) \) regret. We then propose a new algorithm called PD-FTL (Primal-Dual Follow-the-Leader) that achieves an \( O(\sqrt{T}) \) regret bound. The result matches the lower bound \( \Omega(\sqrt{T}) \) for the OCOwK problem in the stochastic setting. We then provide numerical experiments to compare the empirical performances of SP-FTL and PD-FTL.

1.1 Main Contributions

We first propose an algorithm called SP-FTL (Saddle-Point Follow-the-Leader) for the online saddle point problem, and show that this algorithm has a SP-Regret of \( \tilde{O}(\sqrt{T}) \), which matches the lower bound of \( \Omega(\sqrt{T}) \) up to a logarithmic factor. In the special case where the payoff function \( L_t(x, y) \) is strongly-convex in \( x \) and strongly-concave in \( y \), the algorithm has a SP-regret of \( O(\log T) \), which is optimal.

In addition, we show that no algorithm can simultaneously achieve sublinear (i.e. \( o(T) \)) SP-regret and sublinear individual-regrets (defined in Equation 1 and 2) in the general OSP problem. This impossibility result further illustrates the contribution of the SP-FTL algorithm, as existing OCO algorithms designed to achieve sublinear individual-regret are not able to achieve sublinear SP-regret.

Then, we consider the OCOwK problem. We show that this problem is related to the OSP problem by Lagrangian duality, and a sufficient condition to achieve a sublinear regret for OCOwK is that an algorithm must have both sublinear SP-regret and sublinear individual-regret for the OSP problem. In light of the previous impossibility result, we consider the OCOwK problem in a stochastic setting where the reward and consumption functions are sampled i.i.d. from some unknown distribution. By applying the SP-FTL algorithm and exploiting the connection between OCOwK and OSP problems, we obtain a \( \tilde{O}(T^{5/6}) \) regret. We then propose a new algorithm called PD-FTL (Primal-Dual Follow-the-Leader) that achieves an \( O(\sqrt{T}) \) regret bound. The result matches the lower bound \( \Omega(\sqrt{T}) \) for the OCOwK problem in the stochastic setting. We then provide numerical experiments to compare the empirical performances of SP-FTL and PD-FTL.

2 Literature Review

Saddle point problems emerge from a variety of fields such as machine learning, statistics, computer science, and economics. Some applications of the saddle point problem include: minimizing the maximum of smooth convex functions, minimizing the maximal eigenvalue, \( l_1 \)-minimization (an important tool in sparsity-oriented Signal processing), nuclear norm minimization, robust learning problems, and two-player zero-sum games [35, 29, 17, 6, 33, 28, 16, 27].

A few papers have studied the saddle point problem in online learning settings. Motivated by joint optimization and estimation problems, Ho-Nguyen and Kilinc-Karzan [24] consider an online saddle point problem similar to ours. They show that the so-called “online saddle point gap,” or using our terminology, the sum of both players’ individual-regret, is sublinear. However, they did not consider SP-regret. Cesa-Bianchi and Lugosi [14] provide a detailed overview of online learning for static two-player zero-sum games, where the (convex-concave) payoffs are given by \( L_t(x, y) = L(x, y) \) for all \( t = 1, \ldots, T \). They show that if both players minimize their individual-regrets, then the average of actions \((\bar{x}, \bar{y})\) satisfy \(|L(\bar{x}, \bar{y}) - L(x^*, y^*)| \rightarrow 0\) as \( T \rightarrow \infty \), where \((x^*, y^*)\) is a Nash equilibrium. Abernethy and Wang [2], using the same scheme, establish a connection between online learning and projection-free optimization by considering a specific static two-player zero-sum game, where both players apply individual regret minimization algorithms. They also mention this idea has had applications in boosting, differential privacy, linear programming and flow optimization. This line of research has been continued in [40, 1].

The Online Convex Optimization with Knapsacks (OCOwK) problem studied in this paper is related to several previous works on constrained multi-armed bandit problems, online linear programming, and online convex programming. We next give an overview of the work related to OCOwK. Agrawal et al. [5] and Agrawal and Devanur [4] consider online linear/convex programming problems. A key difference between the online linear/convex programming problems and the OCOwK problem is that we assume the action must be chosen without knowledge of the function associated with the current iteration. In [5, 4], it is assumed that these functions are revealed before the action is chosen. Related work is that of Buchbinder and Naor [13], where they study an online fractional covering/packing
We introduce some notation and definitions that will be used in later sections. By default, all vectors \( x \), \( y \) are column vectors. A vector with entries \( x_1, \ldots, x_n \) is written as \( x = [x_1; \ldots; x_n] = [x_1, \ldots, x_n]^\top \), where \( \top \) denotes the transpose. Let \( \| \cdot \| \) be the Euclidean norm of a vector.

We say a function \( L(x, y) \) is convex-concave if it is convex in \( x \in X \) and concave in \( y \in Y \). A pair \((x^*, y^*)\) is called a saddle point for \( L \) if for any \( x \in X \) and any \( y \in Y \), we have
\[
L(x^*, y) \leq L(x^*, y^*) \leq L(x, y^*).
\] (3)

It is well known that if \( L \) is convex-concave, and \( X \) and \( Y \) are convex compact sets, there always exists at least one saddle point (see e.g. [12]).

We say that a function \( f : X \to \mathbb{R} \) is \( H \)-strongly convex if for any \( x_1, x_2 \in X \), it holds that
\[
f(x_1) \geq f(x_2) + \nabla f(x_2)^\top (x_1 - x_2) + \frac{H}{2} \|x_1 - x_2\|^2.
\]

Mannor et al. [32] consider a variant of the online convex optimization (OCO) problem where the adversary may choose extra constraints that must be satisfied. They construct an example such that no algorithm can attain an \( \epsilon \)-approximation to the offline problem. In view of such result, several papers [31] [34] [42] study problems similar to [32] with further restrictions on how constraints are selected by the adversary. The objective in this line of work is to choose a sequence of decisions to achieve the offline optimum while making sure the constraints are (almost) satisfied. In this line of research, the most relevant work to ours is that of [34]. They study OCO with time-varying constraints, the model is similar to that of [32], however in view of the existing negative results they consider three different settings. In the first one, both the cost functions and the constraints are arbitrary sequences of convex functions, however in view of the negative result from [32], the constraints must all be non positive over a common subset of \( \mathbb{R}^n \). In the second setting the sequences of loss functions remain adversarially chosen however the constraints are sampled i.i.d. from some unknown distribution. Finally, in the third setting both the sequences of loss functions and constraints are sampled i.i.d. from some unknown distribution. They develop algorithms for all the three different settings that ensure the total loss incurred by the algorithm is not too far from the offline optimum and such that the constraints are almost satisfied. The setup and results are different than ours because they only require the cumulative constraint violation to be sublinear whereas in OCOwK, once the player exceeds the budget it can no longer collect rewards. Closely related is the problem of “Online Convex Optimization with Long Term Constraints”. The setup is similar to that of OCO where the functions are chosen adversarially with the difference that it is not required that the decisions the player makes at each step belong to the set. Instead, it is required that the average decision lies in the set (which is fully known in advance). As the authors explain, this problem is useful to avoid the projection step of online gradient descent (OGD) and it allows to solve problems such as multi-objective online classification [10], and for using the popular online-to-batch conversion. The algorithms they develop consist on simultaneously running two copies of variants of OGD on convex-concave functions. Better rates and slightly different guarantees were obtained for the same problem in [44] [25] [43]. In [37] they study a continuous time version of a problem similar to that of [31] and show that a continuous time version of primal-dual online gradient descent in continuous time guarantees small regret. In [15] motivated by an application in low-latency fog computing they consider a problem similar to that in [30] however there is bandit feedback in the loss function. The algorithm they provide is primal-dual online gradient descent that combines ideas from [19] to deal with bandit feedback.

Most closely related to our model is the constrained multi-armed bandit problem studied by Badanidiyuru et al. [8] and Wu et al. [41]. In this problem, there is a finite set of arms, and each arm yields a random reward and consumes resources when it is pulled. The goal is to maximize total reward without exceeding a total budget. The Bandits with Knapsacks problem can be viewed as a special case of the OCOwK problem, where the reward and consumption functions are both linear. Agrawal and Devanur [3] study a generalization of bandits with concave rewards and convex knapsack constraints. Similar problems have also been studied in specific application contexts, such as online ad auction [9] and dynamic pricing [11] [18].

3 Preliminaries

We introduce some notation and definitions that will be used in later sections. By default, all vectors are column vectors. A vector with entries \( x_1, \ldots, x_n \) is written as \( x = [x_1; \ldots; x_n] = [x_1, \ldots, x_n]^\top \), where \( \top \) denotes the transpose. Let \( \| \cdot \| \) be the Euclidean norm of a vector.

We say a function \( L(x, y) \) is convex-concave if it is convex in \( x \in X \) and concave in \( y \in Y \). A pair \((x^*, y^*)\) is called a saddle point for \( L \) if for any \( x \in X \) and any \( y \in Y \), we have
\[
L(x^*, y) \leq L(x^*, y^*) \leq L(x, y^*).
\] (3)

It is well known that if \( L \) is convex-concave, and \( X \) and \( Y \) are convex compact sets, there always exists at least one saddle point (see e.g. [12]).

We say that a function \( f : X \to \mathbb{R} \) is \( H \)-strongly convex if for any \( x_1, x_2 \in X \), it holds that
\[
f(x_1) \geq f(x_2) + \nabla f(x_2)^\top (x_1 - x_2) + \frac{H}{2} \|x_1 - x_2\|^2.
\]
Here, $\nabla f(x)$ denotes a subgradient of $f$ at $x$. Strong convexity implies that the problem $\min_{x \in X} f(x)$ has a unique solution. We say a function $g$ is $H$-strongly concave if $-g$ is $H$-strongly convex. Furthermore, we say a function $\mathcal{L}(x, y)$ is $H$-strongly convex-concave if for any fixed $y_0 \in Y$, the function $\mathcal{L}(x, y_0)$ is $H$-strongly convex in $x$, and for any fixed $x_0 \in X$, the function $\mathcal{L}(x_0, y)$ is $H$-strongly concave in $y$. If $\mathcal{L}$ is $H$-strongly convex-concave, then there exists a unique saddle point.

We say a function $\mathcal{L}(x, y)$ is $G$-Lipschitz continuous if
\[ |\mathcal{L}(x_1, y_1) - \mathcal{L}(x_2, y_2)| \leq G||x_1 - x_2||. \]
It is well known that the previous inequality holds if and only if
\[ ||\nabla_x \mathcal{L}(x, y); \nabla_y \mathcal{L}(x, y)|| \leq G \]
for all $x \in X, y \in Y$ \cite{12}.

Throughout the paper we will use the big $O$ notation to hide constant factors. For two functions $f(T)$ and $g(T) > 0$, we write $f(T) = O(g(T))$ if there exists a constant $M_1$ and a constant $T_1$ such that $f(T) \leq M_1 g(T)$ for all $T \geq T_1$; we write $f(T) = \Omega(g(T))$ if there exists a constant $M_2$ and a constant $T_2$ such that $f(T) \geq M_2 g(T)$ for all $T \geq T_2$. We use the $\tilde{O}$ notation to hide constant factors and poly-logarithmic factors. More specifically, for two functions $f(T)$ and $g(T) > 0$, we write $f(T) = \tilde{O}(g(T))$ if there exists constants $M_3, T_3$ and an integer $k \geq 0$ such that $f(T) \leq M_3 g(T) \log^k(g(T))$ for all $T \geq T_3$.

4 The Online Saddle Point Problem

4.1 The Strongly Convex-Concave Case

We now present algorithms for the OSP problem with guaranteed sublinear SP-regret. Recall that the SP-regret defined in (1) measures the gap between the cumulative value achieved by an online algorithm and the value of the game under the Nash equilibrium if all functions are known in hindsight.

For simplicity we assume $T$ is known in advance (this assumption can be relaxed using the well known doubling trick from \cite{14, 38}). We first consider the case where the functions $(\mathcal{L}_t)_{t=1}^T$ are strongly convex-concave. We show that the following simple algorithm Saddle-Point Follow-the-Leader (SP-FTL), which is a variant of the Follow-the-Leader (FTL) algorithm by Kalai and Vempala \cite{26}, attains sublinear SP-regret.

\begin{algorithm}
\textbf{Algorithm 1} Saddle-Point Follow-the-Leader (SP-FTL)
\begin{algorithmic}
  \State \textbf{input:} $x_1 \in X, y_1 \in Y$
  \For{$t = 1, \ldots, T$} \Comment{for each round $t$}
    \State Choose actions $(x_t, y_t)$
    \State Observe function $\mathcal{L}_t$
    \State $x_{t+1} \leftarrow \arg \min_{x \in X} \max_{y \in Y} \sum_{\tau=1}^t \mathcal{L}_\tau(x, y)$
    \State $y_{t+1} \leftarrow \arg \max_{y \in Y} \min_{x \in X} \sum_{\tau=1}^t \mathcal{L}_\tau(x, y)$
  \EndFor
\end{algorithmic}
\end{algorithm}

The main difference between SP-FTL and FTL is that in SP-FTL both players update jointly and play the (unique) saddle point of the sum of the games observed so far. In contrast, the updates for Follow-the-Leader would be $x_{t+1} \leftarrow \arg \min_{x \in X} \sum_{\tau=1}^t \mathcal{L}_\tau(x, y_{FTL}^\tau)$ and $y_{t+1} \leftarrow \arg \max_{y \in Y} \sum_{\tau=1}^t \mathcal{L}_\tau(x_{FTL}^\tau, y)$ for $t = 2, \ldots, T$ and $x_{FTL}^1, y_{FTL}^1$ are arbitrarily chosen from their respective sets $X$ and $Y$. It is easy to see that the sequence of iterates is in general not the same. In fact, in view of Theorem 3 we will see that FTL can not achieve sublinear SP-Regret when the sequence of functions is chosen arbitrarily.

\textbf{Theorem 1.} Let $(\mathcal{L}_t(x, y))_{t=1}^T$ be an arbitrary sequence of $H$-strongly convex-concave, $G$-Lipschitz functions. Then, the SP-FTL algorithm guarantees
\[ \text{SP-Regret}(T) = \sum_{t=1}^T \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) \leq \frac{8G^2}{H} (1 + \log T). \]
We now assume the following claim holds for $T$.

We now show by induction that

$$-G\sum_{t=1}^{T}||x_t - x_{t+1}|| \leq \sum_{t=1}^{T} \mathcal{L}_t(x_{t+1}, y_{t+1}) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y) \leq G\sum_{t=1}^{T} ||y_t - y_{t+1}||. \quad (4)$$

**Proof.** We first prove the second inequality. We proceed by induction. The base case $t = 1$ holds by definition of $(x_2, y_2)$, indeed

$$L_1(x_2, y_2) + G||y_1 - y_2|| \geq L_1(x_2, y_2) := \min_{x \in X} \max_{y \in Y} L_1(x, y).$$

We now assume the following claim holds for $T - 1$:

$$\min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T-1} \mathcal{L}_t(x, y) \geq \sum_{t=1}^{T-1} \mathcal{L}_t(x_{t+1}, y_{t+1}) - G\sum_{t=1}^{T-1} ||y_t - y_{t+1}||, \quad (5)$$

and show it holds for $T$.

$$\min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y)
= \sum_{t=1}^{T-1} \mathcal{L}_t(x_{T+1}, y_{T+1}) + \mathcal{L}_T(x_{T+1}, y_{T+1})
\geq \sum_{t=1}^{T-1} \mathcal{L}_t(x_{T+1}, y_{T}) + \mathcal{L}_T(x_{T+1}, y_T)
\geq \sum_{t=1}^{T} \mathcal{L}_t(x_{T+1}, y_{T}) - G\sum_{t=1}^{T} ||y_t - y_{T+1}|| + \mathcal{L}_T(x_{T+1}, y_T)
\geq \sum_{t=1}^{T} \mathcal{L}_t(x_{T+1}, y_{T}) - G\sum_{t=1}^{T} ||y_t - y_{T+1}|| - G||y_T - y_{T+1}||
= \sum_{t=1}^{T} \mathcal{L}_t(x_{T+1}, y_{T}) - G\sum_{t=1}^{T} ||y_t - y_{T+1}||.$$

We now show by induction that

$$\min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y) \leq \sum_{t=1}^{T} \mathcal{L}_t(x_{t+1}, y_{t+1}) + G\sum_{t=1}^{T} ||x_t - x_{t+1}||.$$
Indeed, $t = 1$ follows from the definition of $(x_2, y_2)$. We now assume the claim holds for $T - 1$ and prove it for $T$:

$$
\begin{align*}
\min_{x} \max_{y} & \sum_{t=1}^{T} \mathcal{L}_t(x, y) \\
= & \sum_{t=1}^{T} \mathcal{L}_t(x_{t+1}, y_{t+1}) \\
\leq & \sum_{t=1}^{T} \mathcal{L}_t(x_t, y_{t+1}) + \mathcal{L}_T(x_T, y_{T+1}) \\
\leq & \sum_{t=1}^{T-1} \mathcal{L}_t(x_t, y_{t+1}) + \mathcal{L}_T(x_{T-1}, y_{T+1}) + \mathcal{L}_T(x_{T-1}, y_{T+1}) - \mathcal{L}_T(x_T, y_{T+1}) \\
\leq & \sum_{t=1}^{T-1} \mathcal{L}_t(x_{t+1}, y_{t+1}) + G \sum_{t=1}^{T-1} \|x_t - x_{t+1}\| \\
& + \mathcal{L}_T(x_{T-1}, y_{T+1}) + \mathcal{L}_T(x_{T-1}, y_{T+1}) - \mathcal{L}_T(x_T, y_{T+1}) \\
\leq & \sum_{t=1}^{T} \mathcal{L}_t(x_{t+1}, y_{t+1}) + G \sum_{t=1}^{T} \|x_t - x_{t+1}\| \\
& \text{by induction claim} \\
\leq & \sum_{t=1}^{T} \mathcal{L}_t(x_{t+1}, y_{t+1}) + G \sum_{t=1}^{T} \|x_t - x_{t+1}\| \\
& \text{since } \mathcal{L}_T \text{ is } G\text{-Lipschitz.}
\end{align*}
$$

**Lemma 2.** Let \( \{(x_t, y_t)\}_{t=1}^{T} \) be the iterates of SP-FTL. It holds that

$$
\|x_t - x_{t+1}\| + \|y_t - y_{t+1}\| \leq \frac{4G}{Ht}.
$$

**Proof.** Fix \( t \). Define

$$
J(x, y) \triangleq \sum_{t=1}^{T-1} \mathcal{L}_t(x, y) + \mathcal{L}_t(x, y)
$$

so that \((x_{t+1}, y_{t+1}) = \min_{x \in X} \max_{y \in Y} J(x, y)\). Since \( J \) is \( Ht \)-strongly convex it holds that for any \( x \in X \) and any \( y \in Y \)

$$
J(x, y) \geq J(x_{t+1}, y) + \nabla_x J(x_{t+1}, y) \top (x - x_{t+1}) + \frac{Ht}{2} \|x - x_{t+1}\|^2.
$$

Plugging in \( y = y_{t+1} \) and recalling the KKT condition \( \nabla_x J(x_{t+1}, y_{t+1}) \top (x - x_{t+1}) \geq 0 \), we have that for any \( x \in X \)

$$
\frac{2}{Ht} [J(x, y_{t+1}) - J(x_{t+1}, y_{t+1})] \geq \|x - x_{t+1}\|^2. \tag{6}
$$

Similarly, since \( J \) is \( Ht \) strongly concave. That is, for any \( y \in Y \)

$$
J(x_{t+1}, y) \leq J(x_{t+1}, y_{t+1}) + \nabla_y J(x_{t+1}, y_{t+1}) \top (y - y_{t+1}) - \frac{Ht}{2} \|y - y_{t+1}\|^2.
$$

Together with the KKT condition \( \nabla_y J(x_{t+1}, y_{t+1}) \top (y - y_{t+1}) \leq 0 \) we get that for any \( y \in Y \)

$$
\frac{2}{Ht} [J(x_{t+1}, y_{t+1}) - J(x_{t+1}, y)] \geq \|y - y_{t+1}\|^2. \tag{7}
$$
Adding up Equations (6) and (7), plugging $x = x_t$ and $y = y_t$ we get

$$\frac{2}{Ht} \left[ J(x_t, y_{t+1}) - J(x_{t+1}, y_t) \right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

\[\iff\] \[\iff\]

$$\frac{2}{Ht} \left[ \sum_{\tau=1}^{t-1} L_\tau(x_t, y_{t+1}) + L_t(x_t, y_{t+1}) - \sum_{\tau=1}^{t-1} L_\tau(x_{t+1}, y_t) - L_t(x_{t+1}, y_t) \right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

$$\iff\frac{2}{Ht} \left[ \sum_{\tau=1}^{t-1} L_\tau(x_t, y_{t+1}) + L_t(x_t, y_{t+1}) - \sum_{\tau=1}^{t-1} L_\tau(x_{t+1}, y_t) - L_t(x_{t+1}, y_t) \right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

since $\sum_{\tau=1}^{t-1} L_\tau(x_t, y_{t+1}) \leq \sum_{\tau=1}^{t-1} L_\tau(x_t, y_t)$.

Additionally, since $\sum_{\tau=1}^{t-1} L_\tau(x_t, y_t) \leq \sum_{\tau=1}^{t-1} L_\tau(x_{t+1}, y_t)$ we have

$$\frac{2}{Ht} \left[ \sum_{\tau=1}^{t-1} L_\tau(x_t, y_t) + L_t(x_t, y_{t+1}) - \sum_{\tau=1}^{t-1} L_\tau(x_{t+1}, y_t) - L_t(x_{t+1}, y_t) \right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

$$\iff\frac{2}{Ht} \left[ L_t(x_t, y_{t+1}) - L_t(x_{t+1}, y_t) \right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

$$\iff\frac{2}{Ht} G \left[ \|x_t; y_{t+1}\| - [x_{t+1}; y_t]\right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

$$\iff\frac{2}{Ht} G \left[ \|x_t - x_{t+1}\| + \|y_t - y_{t+1}\| \right] \geq \|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2$$

$$\iff\frac{2G}{Ht} \geq \frac{\|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2}{\|x_t - x_{t+1}\| + \|y_t - y_{t+1}\|}.$$

Now, since $x^2$ is a convex function $\frac{a^2}{2} + \frac{b^2}{2} \geq \left(\frac{a+b}{2}\right)^2$ therefore $a^2 + b^2 \geq \frac{(a+b)^2}{2}$. This, together with the last implication, yields the result

$$\frac{4G}{Ht} \geq \|x_t - x_{t+1}\| + \|y_t - y_{t+1}\|.$$

Now we are ready to prove Theorem [1]
Proof. Proof of Theorem 1. We have
\[
\sum_{t=1}^{T} \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y)
\]
\[
\leq \sum_{t=1}^{T} \mathcal{L}_t(x_t, y_t) - \sum_{t=1}^{T} \mathcal{L}_t(x_{t+1}, y_{t+1}) + G \sum_{t=1}^{T} \|y_t - y_{t+1}\| \quad \text{by Lemma 1}
\]
\[
\leq G \sum_{t=1}^{T} \|x_t - x_{t+1}\| + \|y_t - y_{t+1}\| + G \sum_{t=1}^{T} \|y_t - y_{t+1}\| \quad \text{since } \mathcal{L}_t \text{ is } G-Lipschitz
\]
\[
\leq G \sum_{t=1}^{T} \frac{4G}{Ht} + G \sum_{t=1}^{T} \frac{4G}{Ht} \quad \text{by Lemma 2}
\]
\[
\leq \frac{8G^2}{H}(1 + \int_{1}^{T} \frac{1}{t} \, dt)
\]
\[
= \frac{8G^2}{H}(1 + \ln T).
\]

The other side of the inequality follows analogously by using the other inequality in Lemma 1. \qed

4.2 The General Convex-Concave Case

We now consider the general case where \( \{\mathcal{L}_t\}_{t=1}^{T} \) is convex-concave, but not necessarily strongly convex-concave. We define a new sequence of functions given by
\[
\bar{\mathcal{L}}_t(x, y) \triangleq \mathcal{L}_t(x, y) + H\|x\|^2 - H\|y\|^2. \tag{8}
\]
It can be easily verified that \( \bar{\mathcal{L}}_t(x, y) \) is \( H \)-strongly convex-concave. We will then run SP-FTL on this modified sequence of functions.

**Theorem 2.** Let \( \{\mathcal{L}_t(x, y)\}_{t=1}^{T} \) be an arbitrary sequence of convex-concave functions. Then, running SP-FTL on \( \bar{\mathcal{L}}_t(x, y) \) as defined in (8) with \( H = \frac{1}{\sqrt{T}} \) guarantees
\[
\text{SP-Regret}(T) = \left| \sum_{t=1}^{T} \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y) \right| \leq \tilde{O}(\sqrt{T}).
\]

Before diving into the proof of Theorem 2, we must establish some notation. Let \( D_X = \sup_{x \in X} \|x\| \), \( D_Y = \sup_{y \in Y} \|y\| \). Notice that by definition of \( \bar{\mathcal{L}}_t \triangleq \mathcal{L}_t(x, y) + H\|x\|^2 - H\|y\|^2 \), using the fact that \( \| \cdot \| \) is nonnegative everywhere, we have that for all \( x \in X, y \in Y \)
\[
-HD_Y^2 \leq \bar{\mathcal{L}}_t(x, y) - \mathcal{L}_t(x, y) \leq HD_X^2, \quad \forall t = 1, \ldots, T. \tag{9}
\]
Before we prove the theorem, we need the following lemma.

**Lemma 3.** It holds that
\[
-HD_Y^2 T \leq \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \bar{\mathcal{L}}_t(x, y) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_t(x, y) \leq HD_X^2 T.
\]
We are now ready to prove the theorem.

**Proof.** Proof. Let \( x'_{t+1} \in \text{arg min}_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) \), \( y'_{t+1} \in \text{arg max}_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) \). We have

\[
\min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) \\
\leq \sum_{t=1}^T \mathcal{L}_t(x'_{t+1}, y'_{t+1}) \\
= \sum_{t=1}^T \left[ \mathcal{L}_t(x'_{t+1}, y'_{t+1}) + H \|x'_{t+1}\|^2 - H\|y'_{t+1}\|^2 \right] \\
\leq \sum_{t=1}^T \left[ \mathcal{L}_t(x'_t, y'_{t+1}) + H \|x'_{t+1}\|^2 - H\|y'_{t+1}\|^2 \right] \\
\leq \sum_{t=1}^T \left[ \mathcal{L}_t(x'_t, y'_{t+1}) + H \|x'_{t+1}\|^2 \right] \\
\leq \sum_{t=1}^T \mathcal{L}_t(x'_t, y'_{t+1}) + H \mathcal{D}_X^2 T \\
\leq \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) + H \mathcal{D}_X^2 T.
\]

The other inequality follows from the same reasoning. \( \square \)

We are now ready to prove the theorem.

**Proof.** Proof of Theorem 2. Since \( \mathcal{L}_t \) is \( G \)-Lipschitz continuous, it holds that \( \mathcal{L}_t \) is \( (G + 2H(D_X + D_Y)) \)-Lipschitz continuous. Therefore,

\[
\sum_{t=1}^T \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) \\
\leq \sum_{t=1}^T \left[ \mathcal{L}_t(x_t, y_t) \right. \\
- \left. \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) + H \mathcal{D}_X^2 T \right] \\
\leq \sum_{t=1}^T \left[ \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) + H \mathcal{D}_X^2 T \right] \\
\leq \sum_{t=1}^T \left[ \mathcal{L}_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) + H \mathcal{T}(\mathcal{D}_X^2 + \mathcal{D}_Y^2) \right] \\
\leq \frac{8(G + 2H(D_X + D_Y))^2(1 + \ln(T))}{H} + HT(\mathcal{D}_X^2 + \mathcal{D}_Y^2). \quad \text{by Theorem 1}
\]

Similarly, we have

\[
\min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) - \sum_{t=1}^T \mathcal{L}_t(x_t, y_t) \\
\leq \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) - \sum_{t=1}^T \mathcal{L}_t(x_t, y_t) + H \mathcal{T} \mathcal{D}_Y^2 \\
\leq \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) - \sum_{t=1}^T \mathcal{L}_t(x_t, y_t) + H \mathcal{T} \mathcal{D}_Y^2 \\
\leq \frac{8(G + 2H(D_X + D_Y))^2(1 + \ln(T))}{H} + HT(\mathcal{D}_X^2 + \mathcal{D}_Y^2). \quad \text{by Theorem 1}
\]
Choosing \( H = \frac{1}{\sqrt{T}} \) yields the result. \( \square \)

We note that the rate in Theorem 3 is optimal with respect to \( T \), since when \( Y \) is a singleton, the problem reduces to the OCO problem with strongly convex loss functions. In that case, it is well known that no algorithm can achieve regret smaller than \( \Omega(\log(T)) \) [23]. Similarly, the \( O(\sqrt{T}) \) rate in Theorem 2 is optimal with respect to \( T \) up to a logarithmic factor according to the \( \Omega(\sqrt{T}) \) lower bound result for the general OCO problem [21].

Although the focus of our work is mainly concerned with showing sublinear rate of SP-Regret, it is worth discussing the computation complexity for each iteration of our algorithms. Notice that in each iteration we must solve a strongly convex strongly concave constrained saddle point problem. It is well known that by simultaneously playing two no Individual Regret algorithms for strongly convex functions (such as those in [22] which achieve Individual Regret \( O(\log(K)) \)), one can generate after \( K \) rounds a solution to the problem that is \( O(\log(K)/K) \) close to the Nash Equilibrium (in terms of the value of the game) (See Theorem 9 in [11]). Recently [11] showed that with additional smoothness assumptions it is possible to obtain linear convergence rates for some static saddle point problems. It is also possible to solve the subproblem for each iteration using the (Stochastic Approximation) Mirror Descent algorithm from [36]. All the previously discussed algorithms are variants of the seminal work of [7].

5 Relationship Between SP-regret and individual-regret

We have defined two regret metrics for the OSP problem, namely the \( \text{SP-regret} \) and the \( \text{individual-regret} \). In the previous subsection, we proposed an algorithm (SP-FTL) with sublinear SP-regret. We have also mentioned that any OCO algorithm (e.g., online gradient descent, online mirror descent, Follow-the-Leader) can achieve sublinear individual-regret. A natural question is whether there exists a single algorithm that has both sublinear SP-regret and individual-regret. Surprisingly, the answer is negative.

**Theorem 3.** If the sequence of convex-concave functions \( \{\mathcal{L}_t\}_{t=1}^T \) is chosen arbitrarily, for any online algorithm, either SP-Regret\((T)\) or Ind-Regret\((T)\) is \( \Omega(T) \).

To prove the negative result, we construct the following problem instance. The \( T \) iterations are evenly split into two halves. In the first \( \lfloor T/2 \rfloor \) iterations, the payoff function is \( \mathcal{L}_t(x, y) = x^2 + xy \). In the remaining iterations, Nature chooses two cases with equal probability: 1) \( \mathcal{L}_t(x, y) = 0 \), or 2) \( \mathcal{L}_t(x, y) = -(y - 1)^2 \). The feasible action sets are \( X = [-1, 1] \) \( Y = [-1, 1] \). Note that in case 1, the saddle point value for the aggregate payoff function over \( T \) iterations is \( \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) = 0 \). In case 2, the saddle point value for the aggregate payoff function is \( \min_{x \in X} \max_{y \in Y} \sum_{t=1}^T \mathcal{L}_t(x, y) = -T/5 \).

We then prove the result by contradiction. Suppose there exists an algorithm that has sublinear SP-regret and sublinear individual-regret for both cases. In case 1, in order to achieve sublinear SP-regret and individual-regret, it can be shown that the action sequence selected by the algorithm must satisfy \( \sum_{t=1}^{\lfloor T/2 \rfloor} x_t^2 = o(T) \). Since the algorithm cannot differentiate between case 1 and case 2 up to iteration \( \lfloor T/2 \rfloor \), the same equation holds in case 2, which implies the sum of SP-regret and player 2’s individual-regret in case 2 is at least \( T/5 - o(T) \). This contradicts with the assumption that the algorithm has sublinear SP-regret and individual-regret for case 2.

We note that despite the negative result in Theorem 3 it is possible to achieve both sublinear SP-regret and individual-regret with further assumptions on the payoff functions \( \{\mathcal{L}_t\}_{t=1}^T \).

One such example is where \( \mathcal{L}_t(x, y) \) is sampled i.i.d.; this case is discussed in [8]. However, in light of Theorem 3 in the general case where \( \{\mathcal{L}_t\}_{t=1}^T \) is an arbitrary sequence, the best one can hope for is achieve either SP-regret or individual-regret, but not both. In [7], we include a numerical example to further illustrate the relationship between SP-regret and individual-regret.

5.1 Proof of the Impossibility Result

We now present a formal proof of the impossibility result.
Proof. Proof of Theorem 3 We prove the result by contradiction. Consider the following example. There are $2T$ time periods. Let $X = [-1, 1], Y = [-1, 1]$. For $t = 1, \ldots, T$, let $L_t(x, y) = x^2 + xy$. For $t = T + 1, \ldots, 2T$, either $L_t = 0$ (Case 1) or $L_t(x, y) = -(y - 1)^2$ (Case 2). We have

$$
\max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y)
= \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \sum_{t=1}^{T} L_t(x, y) - \sum_{t=1}^{T} L_t(x, y) - \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y)
\leq \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \sum_{t=1}^{T} L_t(x, y) - \sum_{t=1}^{T} L_t(x, y) - \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y)
= \text{Ind-Regret}_y(2T) + \text{SP-Regret}(2T).
$$

If both saddle-point regret or individual regret are $o(T)$, then for both Case 1 and Case 2, it holds that

$$
\max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) \leq o(T).
$$

Case 1: We have

$$
\max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y)
= \sum_{t=1}^{T} x_t^2 + \max_{y \in Y} \left( \sum_{t=1}^{T} x_t \right) y - T \min_{x \in X} \max_{y \in Y} (x^2 + xy)
= \sum_{t=1}^{T} x_t^2 + \max_{y \in Y} \left( \sum_{t=1}^{T} x_t \right) y,
$$

which implies $\sum_{t=1}^{T} x_t^2 = o(T)$ and $\left| \sum_{t=1}^{T} x_t \right| = o(T)$. Note that $\min_{-1 \leq x \leq 1} \max_{-1 \leq x \leq 1} (x^2 + xy) = 0$; the unique saddle point is $x = 0, y = 0$.

Case 2: We get

$$
\max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y)
= \sum_{t=1}^{T} x_t^2 + \max_{y \in Y} \left( \sum_{t=1}^{T} x_t y - T(y - 1)^2 \right) - T \min_{x \in X} \max_{y \in Y} (x^2 + xy - (y - 1)^2).
$$

With some calculation, one can show that

$$
\min_{-1 \leq x \leq 1} \max_{-1 \leq y \leq 1} (x^2 + xy - (y - 1)^2) = -\frac{1}{5},
$$

where the unique saddle point is $x = -\frac{2}{5}, y = \frac{4}{5}$.

Since the policy must choose the same sequence $\{x_t, t = 1, \ldots, T\}$ for Case 1 and Case 2, we have $\sum_{t=1}^{T} x_t^2 \leq o(T)$ and $\left| \sum_{t=1}^{T} x_t \right| \leq o(T)$. (This argument can be extended to randomized online algorithms as well by considering the expectation of these quantities.) Thus,

$$
\sum_{t=1}^{T} x_t^2 + \max_{y \in Y} \left( \sum_{t=1}^{T} x_t y - T(y - 1)^2 \right) \geq \sum_{t=1}^{T} x_t^2 + T x_t = -o(T) \quad \text{(by setting } y = 1).\n$$

Therefore, we get

$$
\max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{2T} L_t(x, y) = \frac{T}{5} - o(T),
$$

which contradicts with Equation (10).
6 Online Convex Optimization with Knapsacks

Next, we consider the online convex optimization with knapsacks (OCOWK) problem, motivated by various applications in dynamic pricing, online ad auctions, and crowdsourcing (see §1). The OCOWK model generalizes the standard OCO framework by having an additional set of resource constraints. We will show that OCOWK is closely related to the OSP problem studied in §4.

In this problem, the decision maker has a set of resources $i = 1, \ldots, m$ with given budgets $b_i = (b_{i,t})_{t \in [m]}$. There are $T$ time periods. At each time period, the decision maker chooses $x_t \in X \subset \mathbb{R}^n$. After the decision is chosen, Nature reveals two functions: a concave reward function $r_t : X \to \mathbb{R}$, and a convex resource consumption function $c_t : X \to \mathbb{R}^m_+$. The objective is to maximize cumulative reward while satisfying the budget constraints. In particular, we assume that if a decision $x_t$ violates any of the budget constraints, no reward is collected at period $t$. Therefore, the decision maker’s cumulative reward is given by

$$R(x_1, x_2, \cdots, x_T) = \sum_{t=1}^{T} \left( r_t(x_t) \mathbb{I}\left[ \sum_{\tau=1}^{t} c_\tau(x_\tau) \leq b_t \right] \right),$$

where $\mathbb{I}[\cdot]$ denotes the indicator function. In (11), if $b = +\infty$, the problem reduces to the standard OCO setting.

In order to guarantee that the budget constraint can always be satisfied, we assume there exists a “null action” that doesn’t consume any resource or generate any reward.

**Assumption 1.** There exists an action $x_0 \in X$ such that $r_t(x_0) \equiv 0$ and $c_t(x_0) \equiv 0$ for all $t = 1, \ldots, T$.

The “null action” assumption is often satisfied in different application domains of OCOWK. For example, in the dynamic pricing context, the “null action” is equivalent to charging an extremely high price so there is no customer demand; in the auction context, the “null action” corresponds to bidding at $0$.

If the reward and consumption functions are chosen arbitrarily, it can be shown that no algorithm can achieve sublinear regret for OCOWK. Intuitively, if the reward and consumption functions shift at $[T/2]$, no algorithm can recover the mistake made before $T/2$ in the remaining periods (which is similar to the case in §5). Therefore, we consider the setting where the reward and consumption functions are stochastic.

**Assumption 2.** For $t = 1, \ldots, T$, the reward function $r_t$ and consumption function $c_t$ are sampled i.i.d. from a fixed joint distribution.

Notice that even when the reward and consumption distribution is known, the optimal policy for the OCOWK problem is not a static decision, as the optimal decision depends on the remaining time and remaining budget. Therefore, defining the offline benchmark for OCOWK is not as straightforward as in the stochastic OCO setting. However, it has been shown in the literature that the following offline convex problem provides an upper bound of the expected reward of the optimal offline policy under Assumption 2 (see e.g. [8][11]):

$$r^* \triangleq \max_{x \in X} \left\{ \sum_{t=1}^{T} \mathbb{E}[r_t(x)], \text{ subject to } \sum_{t=1}^{T} \mathbb{E}[c_t(x)] \leq b \right\}$$

Thus, we define the expected regret for the OCOWK problem as

$$\text{Regret}(T) \triangleq r^* - \mathbb{E}[R(x_1, x_2, \cdots, x_T)],$$

where the expectation is taken with respect to the random realizations of functions $r_t$ and $c_t$.

6.1 Reduction to a Saddle Point Problem

We relate the OCOWK problem to the OSP problem studied in §4 by defining a function $L_t(x, y) \triangleq -r_t(x) - y^T(b/T - c_t(x))$, $\forall y \in \mathbb{R}^m_+$. Note that $L_t(x, y)$ is convex in $x$ and concave in $y$, so
we can treat $L_t(x, y)$ as a payoff function in the OSP problem. Here, $y$ can be viewed as the dual prices associated with the budget constraints in (12), and the function $L_t(x, y)$ penalizes the payoff if consumption at iteration $t$ exceeds the average budget per period.

We let constant $y^{i}_{\text{max}}$ be the maximum reward that can be gained by adding one unit of resource $i$, and define set $Y = \prod_{i=1}^{m} [0, y^{i}_{\text{max}}]$. For any sequence of decisions $x_1, \cdots, x_T$, we claim that the decision maker’s total reward is bounded by

$$R(x_1, x_2, \cdots, x_T) \geq \sum_{t=1}^{T} r_t(x_t) + \min_{y \in Y} \left\{ y^T \sum_{t=1}^{T} (b_t/c_t(x_t)) \right\} = -\max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y).$$  \hspace{1cm} (13)

To see this, consider a modified OCOwK problem where resource consumption is allowed to go over the budget, but the decision maker must pay $y^{i}_{\text{max}}$ for each additional unit of resource $i$ used over $b_i$. By the definition of $y^{i}_{\text{max}}$, the decision maker’s profit under the modified problem is given by the right-hand side of (13), which is a lower bound of the reward in the original problem.

We now consider the benchmark (12). By Assumption 1, the Slater condition holds for the convex optimization problem (12), so by using strong duality, we have

$$r^* = -\min_{x \in X} \max_{y \in Y} \left\{ -\sum_{t=1}^{T} E[r_t(x)] - y^T \sum_{t=1}^{T} (b_t/c_t(x)) \right\} = -\min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} L_t(x, y).$$

Therefore, the expected regret for OCOwK is bounded by

$$\text{Regret}(T) = r^* - E[R(x_1, x_2, \cdots, x_T)] \leq E\left[ \max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) \right] - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} L_t(x, y)$$

$$= E\left[ \max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) - \sum_{t=1}^{T} L_t(x_t, y_t) \right] + E\left[ \sum_{t=1}^{T} L_t(x_t, y_t) \right] - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} L_t(x, y)$$  \hspace{1cm} (14)

We have bounded the regret of the OCOwK problem by two quantities in a related OSP problem. In particular, the term $(\dagger)$ is equal to the expectation of player 2’s individual-regret (see Eq (29)), and the second term is related to the SP-regret.

### 6.2 Algorithms for OCOwK

Motivated by its connection to the OSP problem, we propose two algorithms for the OCOwK problem.

First, we consider SP-FTL defined in Algorithm 1. In view of Eq (14), we can bound the regret for OCOwK by the sum of an individual-regret and the SP-regret. Theorem 1 has already provided a SP-regret bound for SP-FTL, so we just need to prove a sublinear individual-regret bound. In general, this is impossible due to the negative result in Theorem 3. However, since we made the additional assumption that $r_t$ and $c_t$ are sampled i.i.d., we are able to get a sublinear individual-regret for SP-FTL in the OCOwK problem.

We start by establishing a high probability bound on the individual-regret for the general OSP problem when the payoff function $L_t(x, y)$ is strongly convex-concave.

**Lemma 4.** Suppose $L_t$ is i.i.d., $H$-strongly convex-concave and $G$-Lipschitz. Let $d$ be the dimension of $X \times Y$, and $D_{XY}$ be its diameter. Then, with probability at least $1 - 1/T$, SP-FTL guarantees

$$\text{Ind-Regret}_y(T) \leq \frac{8G^2}{H} (1 + \ln(T)) + O\left( \frac{G^{3/2} D_{XY}^{1/2} (d \ln(T) \ln(dT))^{1/4} T^{3/4}}{H^{1/2}} \right).$$

The proof of Lemma 4 uses a concentration inequality for Lipschitz functions by Shalev-Shwartz et al. (39). The key step in the proof is to show that the solution of the sample average approximation at step $t$ i.e. $x_t$ is close to $x^*$, the saddle point of the expected game.
However, we cannot directly use Lemma 4 to bound the individual-regret term \((\dagger)\) in (14) because the function \(L_t(x, y)\) is linear in \(y\) and thus not strongly convex-concave. We add a regularization term to \(L_t(x, y)\) to make it \(H\)-strongly convex-concave. Notice our choice of the regularization term here is not the same as in Theorem 2, which leads to a \(\tilde{O}(T^{5/6})\) bound in the following theorem.

**Theorem 4.** Define \(\hat{L}_t(x, y) \triangleq L_t(x, y) + H\|x\|^2 - H\|y\|^2\), where \(H \triangleq T^{-1/6}\). Applying the SP-FTL algorithm on functions \(\{\hat{L}_t\}_{t=1}^T\) guarantees that the following inequality holds with probability at least \(1 - 1/T\):

\[
\max_{y \in Y} \sum_{t=1}^T L_t(x, y) - \sum_{t=1}^T \hat{L}_t(x, y_t) \leq O\left((d \log(T) \log(dT))^{1/4} T^{5/6}\right). \tag{15}
\]

This implies the expected regret for the OCOwK problem using SP-FTL is \(\tilde{O}(T^{5/6})\).

Next, we present an algorithm for OCOwK that improves the regret bound in Theorem 4. The key idea of this algorithm is to update primal variable \(x\) and dual variable \(y\) of \(L_t(x, y)\) in parallel. We call this algorithm Primal-Dual Follow-the-Leader (PD-FTL) (see Algorithm 2).

**Algorithm 2** Primal-Dual Regularized Follow-the-Leader (PD-RFTL)

**input:** \(x_1 \in X, y_1 \in Y\)

**for** \(t = 1, \ldots, T\) **do**

Play \((x_t, y_t)\)

Observe \(L_t\); define \(f_t(x) \triangleq L_t(x, y_t)\) and \(g_t(y) \triangleq L_t(x_t, y)\)

Set \(x_{t+1} = \arg \min_{x \in X} \sum_{\tau=1}^t \left[f_{\tau}(x) + \frac{1}{\sqrt{T}} \left(\|x - x_{\tau}\|^2\right)\right]\)

Set \(y_{t+1} = \arg \min_{y \geq 0} \sum_{\tau=1}^t \left[g_{\tau}(y) - \frac{1}{\sqrt{T}} \left(\|y - y_{\tau}\|^2\right)\right]\)

**end for**

We again bound the regret of PD-RFTL using Eq (14). Recall that for SP-FTL, it was more challenging to bound the first term \((\dagger)\) and relatively easy to bound the term \((\ddagger)\). For PD-RFTL, it is quite the opposite. By defining \(g_t(y) \triangleq L_t(x_t, y)\), the first term \((\dagger)\) can be written as \(E[\max_{y \in Y} \sum_{t=1}^T g_t(y) - \sum_{t=1}^T g_t(y_t)]\), so we immediately have \((\dagger) = O(\sqrt{T})\) using the regret bound for Regularized Follow-the-Leader in the OCO setting. To bound the second term \((\ddagger)\), we have the following result.

**Theorem 5.** For PD-RFTL, it holds that

\[
(\ddagger) \triangleq E \left[\max_y \sum_{t=1}^T L_t(x_t, y) - \min_x \max_{y \geq 0} \sum_{t=1}^T L_t(x, y)\right] \leq O(\sqrt{T}),
\]

which implies that the expected regret for the OCOwK problem using PD-FTL is \(O(\sqrt{T})\).

Compared to other algorithms for OCOwK, including the UCB-based algorithm in [8][3] and Thompson sampling-based algorithm in [18], the proof for Theorem 5 is surprisingly simple, as we are able to exploit the connection between OCOwK and the OSP problem.

The \(O(\sqrt{T})\) regret bound in Theorem 5 also gives the best possible rate in \(T\), since OCO is a special case of OCOwK, and it is well-known that any algorithm must have \(\Omega(\sqrt{T})\) regret for the general OCO problem. In Section 7.2, we compare the performance of SP-FTL and PD-FTL in a numerical experiment.

**Remark 1.** Our proof for Theorem 5 allows the RFTL subroutine in Algorithm 2 being replaced with other OCO algorithms with \(O(\sqrt{T})\) regret (e.g. online gradient descent, online mirror descent). In particular, we can extend Algorithm 2 to the bandit setting of OCOwK, where we only observe the values \(r_t(x_t)\) and \(c_t(x_t)\) after \(x_t\) is chosen. By replacing the RFTL subroutine with any Bandit Convex Optimization (BCO) algorithm, we can also establish sublinear regret bounds for OCOwK in the bandit setting.
6.3 Proof of SP-FTL for OCOwK

In this section we present the analysis of SP-FTL, applied to the OCOwK problem. The following result from Shalev-Shwartz et al. [39] (Theorem 5) will be useful.

**Theorem 6** ([39]). Let \( f(w, \xi) : W \times \Xi \to \mathbb{R} \) be \( G \)-Lipschitz with respect to \( w \), where \( W \subset \mathbb{R}^d \) is bounded with diameter \( D_W \). Then with probability at least \( 1 - \delta \), for all \( w \in W \), it holds that

\[
\left| \sum_{t=1}^{T} f(w, \xi_t) - T \mathbb{E}_\xi[f(x, \xi)] \right| \leq O(GD_W \sqrt{d \ln(T) \ln(d/\delta)T}).
\]

First, we prove the following lemma.

**Lemma 5.** Let \((x_t, y_t)\) be the iterates of SPFTL. With probability at least \( 1 - \delta \), for any \( t = 1, \ldots, T \) it holds that

\[
\|x_{t+1} - x^*\| \leq O\left( \frac{G^{1/2}D^{1/2}_{XY} (d \ln(t) \ln(d/\delta))^{1/4}}{H^{1/2} t^{1/4}} \right). \quad (16)
\]

**Proof.** Define the concentration error at time \( t \) as

\[
CE_t \triangleq O\left( GD_{XY} \sqrt{d \ln(t) \ln(d/\delta)}t \right). \quad (17)
\]

Notice that \( L_t \) satisfies all the assumptions of Theorem 6 so with probability at least \( 1 - \delta \), for all \( x \in X, y \in Y \) we have

\[
\left| \sum_{\tau=1}^{t} L_\tau(x, y) - t \bar{L}(x, y) \right| \leq CE_t. \quad (18)
\]

We now derive some consequences of this fact. With probability at least \( 1 - \delta \),

\[
t \bar{L}(x^*, y^*) \leq t \bar{L}(x_{t+1}, \bar{y}) \leq \sum_{\tau=1}^{t} L_\tau(x_{t+1}, y^*) + CE_t \quad \text{by definition of } x^*
\]

\[
\leq \sum_{\tau=1}^{t} L_\tau(x_{t+1}, y_{t+1}) + CE_t. \quad \text{by definition of } y_{t+1}
\]

This implies

\[
t \bar{L}(x^*, y^*) - \sum_{\tau=1}^{t} L_\tau(x_{t+1}, y_{t+1}) \leq CE_t. \quad (19)
\]

We now show that

\[
\sum_{\tau=1}^{t} L_\tau(x^*, y_{t+1}) - t \bar{L}(x^*, y^*) \leq CE_t. \quad (20)
\]

Indeed, it holds that

\[
\sum_{\tau=1}^{t} L_\tau(x^*, y_{t+1}) \leq t \bar{L}(x^*, y_{t+1}) + CE_t \quad \text{by Equation } (18)
\]

\[
\leq t \bar{L}(x^*, y^*) + CE_t. \quad \text{by definition of } y^*
\]
Now, using the fact that \( x_{t+1} \) is the saddle point of \( \sum_{\tau=1}^{t} \mathcal{L}_{\tau}(x, y) \), which is \((Ht)\)-strongly convex, we have that

\[
\frac{Ht}{2} \|x_{t+1} - x^*\|^2 \leq \sum_{\tau=1}^{t} \mathcal{L}_{\tau}(x^*, y_{t+1}) - \sum_{\tau=1}^{t} \mathcal{L}_{\tau}(x_{t+1}, y_{t+1})
\]

\[
= \sum_{\tau=1}^{t} \mathcal{L}_{\tau}(x^*, y_{t+1}) - t \tilde{\mathcal{L}}(x^*, y^*) + t \tilde{\mathcal{L}}(x^*, y^*) - \sum_{\tau=1}^{t} \mathcal{L}_{\tau}(x_{t+1}, y_{t+1})
\]

\[
\leq 2CE_t \quad \text{by Equations (19) and (20).}
\]

It follows that

\[
\|x_{t+1} - x^*\| \leq O\left(\frac{G^{1/2}D_{XY}^{1/2}(d \ln(t) \ln(D^2))^{1/4}}{H^{1/2}t^{1/4}}\right).
\]

We now prove Lemma 4 in Section 6.2.

**Proof.** Proof of Lemma 4. For all \( y \in Y \), it holds that

\[
\sum_{t=1}^{T} \mathcal{L}_{t}(x_t, y) \leq \sum_{t=1}^{T-1} \mathcal{L}_{t}(x_t, y) + \mathcal{L}_{T}(x^*, y) + G\|x_T - x^*\| \quad \text{since } \mathcal{L}_{T} \text{ is } G\text{-Lipschitz}
\]

\[
\leq \sum_{t=1}^{T} \mathcal{L}_{t}(x^*, y) + G \sum_{t=1}^{T} \|x_t - x^*\| \quad \text{since each } \mathcal{L}_{t} \text{ is } G\text{-Lipschitz}
\]

\[
\leq \sum_{t=1}^{T} \mathcal{L}_{t}(x_{T+1}, y) + GT\|x^* - x_{T+1}\| + G \sum_{t=1}^{T} \|x_t - x^*\|.
\]

It follows that

\[
\max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_{t}(x_t, y) \leq \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_{t}(x_{T+1}, y) + GT\|x^* - x_{T+1}\| + G \sum_{t=1}^{T} \|x_t - x^*\|
\]

\[
= \sum_{t=1}^{T} \mathcal{L}_{t}(x_{T+1}, y_{T+1}) + GT\|x^* - x_{T+1}\| + G \sum_{t=1}^{T} \|x_t - x^*\|
\]

(by definition of \( y_{T+1} \))

\[
= \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} \mathcal{L}_{t}(x, y) + GT\|x^* - x_{T+1}\| + G \sum_{t=1}^{T} \|x_t - x^*\|.
\]
Subtracting in both sides $\sum_{t=1}^{T} L_t(x_t, y_t)$, we get
\[
\max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) - \sum_{t=1}^{T} \hat{L}_t(x_t, y_t)
\leq \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} L_t(x, y) - \sum_{t=1}^{T} \hat{L}_t(x_t, y_t) + GT\|x^* - x_{T+1}\| + G \sum_{t=1}^{T} \|x_t - x^*\|
\leq \frac{8G^2}{H} (1 + \ln(T)) + GT\|x^* - x_{T+1}\| + G \sum_{t=1}^{T} \|x_t - x^*\| \quad \text{by Theorem}[4]
\leq \frac{8G^2}{H} (1 + \ln(T)) + \frac{G^{3/2} D_{XY}^{1/2} (d \ln(T) \ln(\frac{4}{\delta}))^{1/4} T^{3/4}}{H^{1/2}} + \sum_{t=1}^{T} \ln^{1/4}(t) \|x_t - x^*\| \quad \text{by Lemma}[5]
\leq \frac{8G^2}{H} (1 + \ln(T)) + \frac{G^{3/2} D_{XY}^{1/2} (d \ln(T) \ln(\frac{4}{\delta}))^{1/4} T^{3/4}}{H^{1/2}} + \int_{1}^{T} \ln^{1/4}(t) \frac{dt}{t^{1/4}}
= \frac{8G^2}{H} (1 + \ln(T)) + O\left(\frac{G^{3/2} D_{XY}^{1/2} (d \ln(T) \ln(\frac{4}{\delta}))^{1/4} T^{3/4}}{H^{1/2}}\right)
\]

with probability at least $1 - \delta T$ (by using a union bound in the second to last inequality). Setting $\delta = 1/T^2$ yields the result.

We are ready to prove Theorem[4].

**Proof.** Proof of Theorem[4] For any $y \in Y$, by adding up Equation [9], we have that
\[
\sum_{t=1}^{T} L_t(x_t, y) \leq \sum_{t=1}^{T} \hat{L}_t(x_t, y) + H D_Y^2 T.
\]
This implies
\[
\max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) \leq \max_{y \in Y} \sum_{t=1}^{T} \hat{L}_t(x_t, y) + H D_Y^2 T.
\]

Therefore, we have
\[
\max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) - \sum_{t=1}^{T} \hat{L}_t(x_t, y_t)
\leq \max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) - \sum_{t=1}^{T} \hat{L}_t(x_t, y_t) + T H D_X^2 \quad \text{by Equation}[9]
\leq \max_{y \in Y} \sum_{t=1}^{T} \hat{L}_t(x_t, y_t) - \sum_{t=1}^{T} \hat{L}_t(x_t, y_t) + T H D_X^2 + T H D_Y^2
\leq \frac{8(G + 2H(D_X + D_Y))^2}{H} (1 + \ln(T)) + O\left(\frac{(G + 2H(D_X + D_Y))^{3/2} D_{XY}^{1/2} (d \ln(T) \ln(dT))^{1/4} T^{3/4}}{H^{1/2}}\right) + T H (D_X^2 + D_Y^2)
\]

(with probability at least $1 - \frac{1}{T}$, by Lemma[4]).

Since
\[
(G + 2H(D_X + D_Y))^{3/2} = (G^3 + 6G^2H(D_X + D_Y) + 12GH^2(D_X + D_Y)^2 + 8H^3(D_X + D_Y) + 3)^{1/2}
\leq G^{3/2} + (6G^2H(D_X + D_Y))^{1/2} + (12GH^2(D_X + D_Y)^2)^{1/2} + (8H^3(D_X + D_Y)^3)^{1/2},
\]

18
choosing \( H = T^{-1/6} \) yields the result in Eq (15). In addition, the high probability bound implies that

\[
E \left[ \max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) - \sum_{t=1}^{T} L_t(x_t, y_t) \right] = \tilde{O}(T^{5/6}).
\]

Now, using Eq (14), we have

\[
\text{Regret}(T) \leq E \left[ \max_{y \in Y} \sum_{t=1}^{T} L_t(x_t, y) - \sum_{t=1}^{T} L_t(x_t, y_t) \right]
+ E \left[ \sum_{t=1}^{T} L_t(x_t, y_t) \right] - \min_{x \in X} \max_{y \in Y} E \left[ \sum_{t=1}^{T} L_t(x, y) \right]
\leq \tilde{O}(T^{5/6}) + \tilde{O}(T^{5/6}) = \tilde{O}(T^{5/6}),
\]

where the first term is bounded by the equation above and the second term is bounded by the SP-regret of the SP-FTL algorithm from Theorem 2 using the regularization factor \( H = T^{-1/6} \).

\[
\square
\]

6.4 Proof of PD-RFTL for OCOwK

In this section we present the analysis of PD-RFTL, applied to the OCOwK problem.

**Proof.** Proof of Theorem 5 Using the \( O(\sqrt{T}) \) regret bound for the Follow-The-Leader (FTL) algorithm in the online convex optimization setting, we have

\[
\sum_{t=1}^{T} L_t(x_t, y_t) = \sum_{t=1}^{T} f_t(x_t) \leq \min_{x} \sum_{t=1}^{T} f_t(x) + O(\sqrt{T}) = \min_{x} \sum_{t=1}^{T} L_t(x, y_t) + O(\sqrt{T}), \quad (21)
\]

and

\[
\sum_{t=1}^{T} L_t(x_t, y_t) = \sum_{t=1}^{T} g_t(y_t) \geq \max_{y} \sum_{t=1}^{T} g_t(y) - O(\sqrt{T}) = \max_{y} \sum_{t=1}^{T} L_t(x_t, y) - O(\sqrt{T}). \quad (22)
\]

Let \( \bar{L}(x, y) = E[L_t(x, y)] \) for any \( x \in X, y \in Y \). Let \((x^*, y^*)\) be the saddle point of \( \bar{L} \), satisfying

\[
\bar{L}(x^*, y^*) = \max_{y \in Y} \bar{L}(x^*, y) = \min_{x \in X} \bar{L}(x, y^*). \quad (23)
\]

We have

\[
E \left[ \max_{y} \sum_{t=1}^{T} L_t(x_t, y) \right] - \min_{x} \max_{y \geq 0} E \left[ \sum_{t=1}^{T} L_t(x_t, y) \right]
\leq E \left[ \min_{x} \sum_{t=1}^{T} L_t(x_t, y_t) \right] - \min_{x} \max_{y \geq 0} E \left[ \sum_{t=1}^{T} L_t(x_t, y) \right] + O(\sqrt{T}) \quad \text{by Equations (21), (22)}
\]

\[
= E \left[ \min_{x} \sum_{t=1}^{T} L_t(x_t, y_t) \right] - \sum_{t=1}^{T} \bar{L}(x^*, y^*) + O(\sqrt{T}) \quad \text{by Equation (23)}
\]

\[
\leq E \left[ \sum_{t=1}^{T} L_t(x^*, y_t) \right] - \sum_{t=1}^{T} \bar{L}(x^*, y^*) + O(\sqrt{T}) \quad \text{because} \ \min_{x} \sum_{t=1}^{T} L_t(x, y_t) \leq \sum_{t=1}^{T} L_t(x^*, y_t)
\]

\[
= E \left[ \sum_{t=1}^{T} (T(x^*, y_t) - T(x^*, y^*)) \right] + O(\sqrt{T})
\leq 0 + O(\sqrt{T}).
\]

The second to last step uses the fact that \( L_t \) is drawn i.i.d., \( y_t \) only depends on \( L_1, \ldots, L_{t-1} \), so we can replace \( L_t(\cdot) \) by \( L(\cdot) \). The last step uses Equation (23) again. \( \square \)
7 Numerical Experiments

7.1 Individual-Regret and SP-Regret

To further illustrate the relationship between SP-regret and individual-regret and the impossibility result of Theorem 3, we compare the performance of two online algorithms numerically. The first algorithm is \textit{SP-FTL} defined in Algorithm 1. In the second algorithm, which we call \textit{OGDA}, player 1 applies online gradient descent to function \( L_t(\cdot, y_t) \) and player 2 applies online gradient ascent to function \( L_t(x_t, \cdot) \).

We generated two different instances. In both instances, we assume \( X = Y = [-10, 10] \). The payoff functions in both instances are the same for \( t = 1, \ldots, \lfloor T/3 \rfloor \), given by \( L_t(x, y) = xy + \frac{1}{2}||x-2||^2 - \frac{1}{2}||x+1||^2 \). In Instance 1, for \( t = \lfloor T/3 \rfloor + 1, \ldots, T \), we define \( L_t(x, y) = xy + \frac{1}{2}||x+1||^2 - \frac{1}{2}||x+2||^2 \). In Instance 2 for \( t = \lfloor T/3 \rfloor + 1, \ldots, T \), we define \( L_t(x, y) = xy + \frac{1}{2}||x+2||^2 - \frac{1}{2}||x-3||^2 \). Since these functions are strongly convex-concave, when players use OGDA with step size \( O(\frac{1}{t}) \), they are both guaranteed logarithmic individual-regret.

Figure 1: SP-regret on instance 1 (left) and instance 2 (right) of SP-FTL and OGDA.

\textit{Note.} Here we define the SP Regret of Player 1 (SP Reg. P1) as \( \sum_{t=1}^{T} L_t(x_t, y_t) - \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} L_t(x, y) \) and the SP Regret of Player 2 (SP Reg. P2) as \( \min_{x \in X} \max_{y \in Y} \sum_{t=1}^{T} L_t(x, y) - \sum_{t=1}^{T} L_t(x_t, y_t) \). According to this definition, we have \( \text{SP Reg. P1} = -\text{SP Reg. P2} \), and SP-regret of OGDA is equal to \( |\text{SP Reg. P1}| = |\text{SP Reg. P2}| \).

In Figure 1 we plot the SP-regret of the two instances. On the left, it can be seen that the SP-regret of OGDA increases significantly after the payoff function switches at \( \lfloor T/3 \rfloor \), while the SP-regret of SP-FTL remains small throughout the entire horizon.

Figure 2: Individual Regrets on Instance 1 (left) and Instance 2 (right) of Algorithms: SP-FTL and OGDA.

\textit{Note.} The Individual Regret (Indiv. Reg.) can be negative as we compare a sequence of dynamic decisions against the best fixed decision in hindsight.
From Figure 2, we can observe when both players use the OGDA algorithm, their individual-regrets are small. However, when they use SP-FTL, at least one player suffers from high individual-regret. Figures 1 and 2 verify Theorem 3, which states that no algorithm can achieve both sublinear SP-regret and sublinear individual-regret.

7.2 SP-FTL and OGDA for the OCOwK problem

In this section, we compare the numerical performance of SP-FTL and OGDA (Online Gradient Descent/Ascent) for solving a OCOwK problem. In OGDA, player 1 applies online gradient descent to function \( L_t(\cdot, y_t) \) and player 2 applies online gradient ascent to function \( L_t(x_t, \cdot) \). The proof for Theorem 5 can also show that OGDA has a regret of \( O(\sqrt{T}) \) (see Remark 1).

We construct a numerical example where for each iteration \( t = 1, \ldots, T \), the decision maker chooses an action \( x_t \in X = [0, 20] \). The reward function is \( r_t = -x^2 + b_t x \) where \( b_t \sim U[0, 20] \). There are two types of resources with budgets \( B_1 \) and \( B_2 \). The consumption function for the first resource is given by \( c_{t,1} = (a_t x)^2 + 50x \) where \( a_t \sim U[0, 3] \), and the consumption function for the is \( c_{t,2} = x \). We assume the budgets are some linear functions of \( T \), \( B_1(T) \) and \( B_2(T) \) respectively. In our simulations \( B_1 \) and \( B_2 \) are chosen so that playing the optimal solution to the problem without budgets is no longer optimal.

![Figure 3: Performance of SP-FTL and OGDA in the OCOwK problem.](image)

Figure 3 compares the performance of SP-FTL vs OGDA on the OCOwK instance defined above. Performance is measured as the ratio of total reward incurred by the algorithm and the solution to Equation (12) across 25 simulation runs. It can be observed that both algorithms indeed improve their performance as \( T \) increases. Moreover, it can be observed that while OGDA has worse performance for small values of \( T \), the rate at which performance improves is greater than that for SP-FTL, which is consistent with our theoretical results that SP-FTL has \( \tilde{O}(T^{5/6}) \) regret and OGDA (or PD-FTL) has \( O(\sqrt{T}) \) regret.

8 Conclusion

In this paper we introduced the Online Saddle Point problem. In this problem, we consider two players that jointly play an arbitrary sequence of convex-concave games against Nature. This problem is a generalization of the classical Online Convex Optimization problem, which focuses on a single player. The objective is to minimize the saddle-point regret (SP-regret), defined as the absolute difference between the cumulative payoffs and the saddle point value of the game in hindsight.

We proposed an algorithm SP-FTL for the Online Saddle Point problem and showed that it achieves \( \tilde{O}(\sqrt{T}) \) SP-regret for a game with \( T \) periods. In the special case where the payoff functions are strongly convex-concave, we showed that the algorithm attains \( O(\log T) \) SP-regret. Furthermore, we proved that if the sequence of payoff functions are chosen arbitrarily, any algorithm with \( o(T) \) regret for the Online Convex Optimization problem may incur \( \Omega(T) \) SP-regret in the worst case. This
implies that all existing algorithms for the Online Convex Optimization problem cannot be applied to the Online Saddle Point problem. Moreover, we showed how our algorithm can be applied to solve the problem of Stochastic Online Convex Optimization with Knapsacks. Finally, we performed some numerical simulations to validate our results.

References

[1] J. Abernethy, K. A. Lai, K. Y. Levy, and J.-K. Wang. Faster rates for convex-concave games. arXiv preprint arXiv:1805.06792, 2018.

[2] J. D. Abernethy and J.-K. Wang. On Frank-Wolfe and equilibrium computation. In Advances in Neural Information Processing Systems, pages 6587–6596, 2017.

[3] S. Agrawal and N. R. Devanur. Bandits with concave rewards and convex knapsacks. In Proceedings of the fifteenth ACM conference on Economics and computation, pages 989–1006. ACM, 2014.

[4] S. Agrawal and N. R. Devanur. Fast algorithms for online stochastic convex programming. In Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms, pages 1405–1424. SIAM, 2014.

[5] S. Agrawal, Z. Wang, and Y. Ye. A dynamic near-optimal algorithm for online linear programming. Operations Research, 62(4):876–890, 2014.

[6] A. Ahmadinejad, S. Dehghani, M. Hajiaghayi, B. Lucier, H. Mahini, and S. Seddighin. From duels to battlefields: Computing equilibria of Blotto and other games. In AAAI, pages 376–382, 2016.

[7] K. J. Arrow, L. Hurwicz, and H. Uzawa, editors. Studies in linear and non-linear programming. Stanford University Press, 1958.

[8] A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. Journal of the ACM (JACM), 65(3):13, 2018.

[9] S. Balseiro and Y. Gur. Learning in repeated auctions with budgets: Regret minimization and equilibrium. working paper, 2017.

[10] A. Bernstein, S. Mannor, and N. Shimkin. Online classification with specificity constraints. In Advances in Neural Information Processing Systems, pages 190–198, 2010.

[11] O. Besbes and A. Zeevi. Blind network revenue management. Operations research, 60(6):1537–1550, 2012.

[12] S. Boyd and L. Vandenberghe. Convex optimization. Cambridge University Press, 2004.

[13] N. Buchbinder and J. Naor. Online primal-dual algorithms for covering and packing. Mathematics of Operations Research, 34(2):270–286, 2009.

[14] N. Cesa-Bianchi and G. Lugosi. Prediction, learning, and games. Cambridge university press, 2006.

[15] T. Chen and G. B. Giannakis. Harnessing bandit online learning to low-latency fog computing. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 6418–6422. IEEE, 2018.

[16] S. M. Chowdhury, D. Kovenock, and R. M. Sheremeta. An experimental investigation of colonel blotto games. Economic Theory, 52(3):833–861, 2013.

[17] B. Cox, A. Juditsky, and A. Nemirovski. Decomposition techniques for bilinear saddle point problems and variational inequalities with affine monotone operators. Journal of Optimization Theory and Applications, 172(2):402–435, 2017.

[18] K. Ferreira, D. Simchi-Levi, and H. Wang. Online network revenue management using thompson sampling. Operations Research, 2018. (forthcoming).
[19] A. D. Flaxman, A. T. Kalai, and H. B. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394. Society for Industrial and Applied Mathematics, 2005.

[20] A. Gupta and M. Molinaro. How the experts algorithm can help solve LPs online. *Mathematics of Operations Research*, 41(4):1404–1431, 2016.

[21] E. Hazan. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

[22] E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

[23] E. Hazan and S. Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

[24] N. Ho-Nguyen and F. Kilınç-Karzan. The role of flexibility in structure-based acceleration for online convex optimization. Technical report, Carnegie Mellon University, 2016. Technical report, http://www.optimization-online.org/DB_HTML/2016/08/5571.html.

[25] R. Jenatton, J. Huang, and C. Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. *arXiv preprint arXiv:1512.07422*, 2015.

[26] A. Kalai and S. Vempala. Efficient algorithms for universal portfolios. *Journal of Machine Learning Research*, 3(Nov):423–440, 2002.

[27] D. Kovenock and B. Roberson. Coalitional colonel blotto games with application to the economics of alliances. *Journal of Public Economic Theory*, 14(4):653–676, 2012.

[28] J.-F. Laslier and N. Picard. Distributive politics and electoral competition. *Journal of Economic Theory*, 103(1):106–130, 2002.

[29] Z. Lu, A. Nemirovski, and R. D. Monteiro. Large-scale semidefinite programming via a saddle point mirror-prox algorithm. *Mathematical Programming*, 109(2-3):211–237, 2007.

[30] M. Mahdavi, R. Jin, and T. Yang. Trading regret for efficiency: online convex optimization with long term constraints. *Journal of Machine Learning Research*, 13(Sep):2503–2528, 2012.

[31] M. Mahdavi, T. Yang, and R. Jin. Online decision making under stochastic constraints. In *NIPS workshop on Discrete Optimization in Machine Learning*, 2012.

[32] S. Mannor, J. N. Tsitsiklis, and J. Y. Yu. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(Mar):569–590, 2009.

[33] R. B. Myerson. Incentives to cultivate favored minorities under alternative electoral systems. *American Political Science Review*, 87(4):856–869, 1993.

[34] M. J. Neely and H. Yu. Online convex optimization with time-varying constraints. *arXiv preprint arXiv:1702.04783*, 2017.

[35] A. Nemirovski. Deterministic and randomized first order saddle point methods for large-scale convex optimization, 2010.

[36] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[37] S. Paternain and A. Ribeiro. Online learning of feasible strategies in unknown environments. In *American Control Conference (ACC), 2015*, pages 4231–4238. IEEE, 2015.

[38] S. Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

[39] S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Stochastic convex optimization. *Proceedings of the 22nd Annual Conference on Computational Learning Theory*, 2009.
[40] J.-K. Wang and J. Abernethy. Acceleration through optimistic no-regret dynamics. arXiv preprint arXiv:1807.10455, 2018.

[41] H. Wu, R. Srikant, X. Liu, and C. Jiang. Algorithms with logarithmic or sublinear regret for constrained contextual bandits. In Advances in Neural Information Processing Systems, pages 433–441, 2015.

[42] H. Yu, M. Neely, and X. Wei. Online convex optimization with stochastic constraints. In Advances in Neural Information Processing Systems, pages 1427–1437, 2017.

[43] H. Yu and M. J. Neely. A low complexity algorithm with $o(\sqrt{T})$ regret and finite constraint violations for online convex optimization with long term constraints. arXiv preprint arXiv:1604.02218, 2016.

[44] J. Yuan and A. Lamperski. Online convex optimization for cumulative constraints. arXiv preprint arXiv:1802.06472, 2018.