Online Submodular Optimization in Network Games (INCOMPLETE DRAFT)

Abstract

We present analysis for performance in social network submodular games using no-regret dynamics. In certain well known models of social influence, many multiplayer games have been studied, especially their corresponding best response algorithms. However, no results so far have been shown on the applicability of online convex optimization methods and no-regret learning for such games, both in the full-information setting and the bandit setting. In this paper, we will discuss the methods in which a party can adaptively change their strategy against competing sources of influence, and how in certain games, this may converge to approximate Nash equilibria.

Contents

1 Introduction

In games involving networks, it is usually the case that multiple rounds are played in a competitive setting, with changing strategies. For example, given a social network, a company may attempt to propagate its product in rumor-spreading fashion, among many other companies. In social cascades, this is becoming more and more of the case, as many different advertisements compete for maximal influence over many rounds. In this setting, a company may wish to adapt its strategies given the placement seeds of the other player. In some cases, only the numerical result (bandit) is established as feedback as well.

Furthermore, there is an inherent limitation to many of the established works in social network optimization, because they mainly focus on producing almost-deterministic strategies, mostly in the form of best response (randomness is mainly based on errors from computational feasibility rather than producing intentional random strategies). This may limit the effectiveness of gaining opinions, because an adversary may also adapt to a player's strategies. For instance, if an adversary knows that the player will use a best-response strategy, then it may easily predict the player's next response and act accordingly.

To address this issue, we consider the use of online submodular optimization methods, and general online convex optimization (OCO) for large scale games in graphs. One reason for this method is that it is mainly an optimization-type approach to graph games, rather than one based on graph structure. Best response strategies require the use of almost the entire graph to produce a strategy; the same computational resources can therefore be given to an online learning strategy by giving the full-information setting.

Multi-agent dynamics has been recently, widely popular. It has been shown that many variants of online learning algorithms allow for faster computation of correlated equilibrium in $O(1/T)$ time, as presented in [FLL$^+$16] in which the feedback is the expected utility, as well as stochastic feedback in [SALS15]. In some cases, no-regret learning algorithms have provided convergence to Nash equilibrium in certain forms of games, such as zero-sum games in [DDK15], as well as [EMN09]

In other games, however, there may exist cycles and other phenomena generated by no-regret algorithms, such as in the case of [PP16].

All of these have in common, however, that the assumptions that certain responses are easy to compute, and that the updating strategies from the online methods are also easy to compute. However, in many games, this is not the case, for example, as action spaces may be a subset of a ground set, and enumeration of all of the probabilities (as well as merely generating them) are computationally intractible. For instance, submodular games are a an example in which it is difficult to do so. This is especially inherent in social network games, which have a general intractibility to fully solving their tasks. In this setting, we attack the performance of no-regret algorithms in a particular set of submodular games, which are competitive influence maximization games.

2 Influence Maximization

In the seminal work for influence maximization in [KKT03], a directed graph $G = (V, E)$ is given, where each edge from u to v has a corresponding probability p_e, in which u may convince v of an opinion. As a stochastic process given a number of initial seed set $S \subseteq V$, one may define the the influence maximization problem as maximizing the expected number of people influenced, $f(S)$, given that S is an initial seed set. Constraints are given for choosing the set S, usually bounding its cardinality $|S| \leq k$ for some number k. As it is well known that this reduces to the max-coverage problem, with an $1 - 1/e$ approximation guarentee, it is computationally intractible to optimize this function fully.

In a long line of works in social influence optimization with its relationship in submodular and coverage optimization, for example in [BPR$^+$15], [RSS15], [BBCL14], it is well understood that achieving a $(1 - 1/e)$ performance can be done in nearly linear time in many variants of the social network problem, due to its nice approximation properties to certain submodular functions, such as coverage functions, by performing sampling methods.

In a different setting, many game theoretic models have been proposed for the social network game, including network blocking and security [TNT12], as well as competitive cases, such as [HK13]. In many security games, where there is a zero sum game between two players, one attacker and one defender, computation of certain equilibriums are efficient in spite of the large action set due to the use of double oracle strategies [BKLP14]. However, in a multi-party competitive game in which every player only wants to improve his influence gains, efficient algorithms have not been studied, even though the price of anarchy is shown to be quite small, with a value of 2 in [HK13], and a value of 4 in other non-submodular cases, such as [GK12].

3 Game Theory

In this section, we provide preliminaries on the definition and notation of a game. A game with N players is assigned an action set A_i to each player i, with the utility function for player i as a parameter of all the player's inputs at one time, $u_i(a_1, ..., a_n) = u_i(a_i, a_{-i})$ where a_{-i} is shorthanded for the vector of the other's players actions, and $a_i \in A_i$. For simplicity, let us assume that the player's action sets have the same cardinality $|A_i| = d$, although we may choose to relax this constraint for the social network case. A player i may choose to pick a mixed strategy $\mathrm{w}_i \in \triangle_d$ corresponding to a probability vector over all the actions A_i, from which player i will choose the j-th action with probability the j-th component of the vector w_i. Thus, we can define the expected utility as $u_i(\mathrm{w}_i, \mathrm{w}_{-i})$

A Nash equilibrium $\mathbf{w} = (\mathrm{w}_1, ..., \mathrm{w}_N)$ is a vector of mixed strategies in which every player i's mixed vector is the optimal given everyone else's strategies are fixed, i.e. $\forall i, u_i(\mathrm{w}_i, \mathrm{w}_{-i}) \geq u_i(\hat{\mathrm{w}}_i, \mathrm{w}_{-i})$ for any deviation $\hat{\mathrm{w}}_i$.

An important thing to notice here is that we may only compute a multiplicatively approximate $(1 - 1/e)$ equilibrium in the competitive social network case, i.e. finding a vector $\mathbf{w}$ such that $u_i(\mathrm{w}_i, \mathrm{w}_{-i}) \geq (1 - 1/e) u_i(\mathrm{w}_i, \mathrm{w}_{-i})$

Theorem 1. *It is NP-hard to construct better Nash equilibrium than* $(1 - 1/e)$ *for any player in the Network Cascade Game.*

The proof is simple, following a common reduction to the max-cover problem. We may generate an instance of a coverage problem, and allow all of the other players to have one node directing into all of the ground elements in the coverage problem, all with probability 1. The main player must solve this coverage problem, and hence his approximation ratio is at most $(1 - 1/e)$.

On the topic of computational efficiency, furthermore, as we cannot efficiently enumerate the entire $\binom{V}{k}$ set of actions with probabilities. Instead, we seek an efficient way to sample all such actions at the $(1 - 1/e)$ equilibrium.

The Price of Anarchy for a social welfare function is defined as $W(\mathbf{w})$, which is usually a sum over all the utilities of the players given $\mathbf{w}$, i.e. $W(\mathbf{w}) = \sum_{i=1}^{N} u_i(\mathbf{w})$ where we overload $u_i(\mathrm{w}_i, \mathrm{w}_{-i}) = u_i(\mathbf{w})$.

A best response a_i given w_{-i} is a pure strategy that maximizes i's utility.

3.1 Network Games

We provide examples examples of network games that produce the submodular case. In most cases, there have been mainly studies for efficient best-response strategy playing, but none on the usage of no-regret algorithms. We focus on three different types of games, each with different notions of maximization and minimization and player intent. **Zero-Sum** games are when a defender must utilize submodular minimization policies, **blocking** games are when a defender must use submodular maximization policies, and **network cascading** games are when all players are using submodular maximization policies as "attackers".

3.1.1 Zero-Sum Games

In this setting, an attacker A may try to maximize his influence spread, while a player B will try to minimize A's utility. More formally, A wishes to maximize $f(S_A, S_B)$, which represents the spread of A's nodes given B is competing against A, while B wishes to maximize the negative of A's utility, i.e. $-f(S_A, S_B)$. It is well known that f is a submodular function, and thus A is inherently attempting to maximize a submodular function, while B is attempting to minimize a submodular function. Zero-Sum structure is exploited in [TNT12] and [BKLP14], in which a Nash equilibrium is attempted to be computed by producing a double-oracle alorithm, or essentially appending new actions into consideration for both players until equilibrium is achieved. However, the methods are empirical, and do not prove any run-time guarantees.

3.1.2 Competitive Blocking

One game in the linear threshold model is the competitive blocking case, in [HSCJ12]. In this game, a player B is responsible for optimizing the number of blocked nodes, given the other player A fixes his nodes, i.e. $f(S_A, S_B) = \sigma(S_A) - \sigma(S_A, S_B)$ where $\sigma(S_A, S_B)$ represents the expected number of nodes propagated by A, given B had already fixed his nodes. Player A wishes to only maximize $\sigma(S_A, S_B)$. It is proved that f is a submodular function for B, given A fixed his action, and of course, $\sigma(S_A)$ is a submodular function as well. Note that this is not a zero-sum game for the blocker, because of the constant $\sigma(S_A)$. In a sense, both players attempt to maximize a submodular function even though the players have different goals.

3.1.3 Network Cascade

In the network cascade game, players wish to maximize their own influences instead. It is well known that from [MR10], [AFL15], one can generate a general network such that each player's utilities is submodular given all others' actions are fixed, as long as the local interactions are submodular as well.

It is established that in any network cascading game, the price of anarchy is bounded by 2, in [HK13]. The example of such a PoA is given by a directed star graph with N leaves, with directed probabilities as 1, and $N-1$ isolated nodes. The unique equilibrium here is when all players choose the star graph center, with expected utility $\frac{N+1}{N}$, which gives the expected social welfare as $N+1$. The maximal social utility however, is when all players use all of the given nodes, where one player chooses the star center, and the others choose their own isolated nodes, giving a sum of utilities as $2N$, producing a PoA of 2.

3.1.4 Best Response Oracles

A common naive strategy in game theory is the best response, which have been produced efficiently in many network games. In potential games, this is a common response dynamic to converge to Nash equilibria. In fact, a similar approximate equilibrium can be achieved for near-potential games, in the case of [COP13]. Unfortunately, social influence games are very far from potential, which may result in cyclic behaviors. A simple example is with 2 players and a graph $A \to B \to C \to D$ where the influence probabilities are 1; if Player 1 chooses A and Player 2 chooses B, there will be a cyclic behavior where Player 1 then chooses C, player 2 switches to A, player 1 switches to B, and a repeat of this dynamic returns to the same position. On average, this will be less efficient compared to a randomized strategy. In fact, such a dynamic will also produce a lower average social welfare.

3.2 No Regret Dynamics and Social Welfare

It is interesting to note that among the literature on social welfare optimization, these no-regret dynamics usually tend toward satisfactory welfare (usually a sum of utilities). This dynamic may imply a state in which players do not attempt to compete against each other in games such as social influence.

4 Online Convex Optimization and No-Regret Dynamics

In the online convex optimization setting, e.g. [Haz16], it is well-understood how to perform convex optimization in the online setting. Given a convex set $\mathcal{K}$, at every time t, a player chooses $x_t \in \mathcal{K}$, and nature provides a convex function f_t from some function set $\mathcal{F}$, in which the player's performance is $f(x_t)$. Both the bandit case (in which the player only receives the value of his performance $f_t(x_t)$, and the case in which the player receives the full f_t) have been studied intensely. In the minimization setting[1], a measure of performance, regret, is defined as the performance over all days from $t = 1$ to T, compared to having chosen a fixed action in hindsight given all of $f_1, ..., f_T$, i.e.

$$\text{Regret} = \sum_{t=1}^{T} f_t(x_t) - \min_{\hat{x} \in \mathcal{K}} \sum_{t=1}^{T} f_t(\hat{x}) \tag{1}$$

It is well known by regularization techniques, that the regret may be bounded by $O(GD\sqrt{T})$ where G is the Lipschitz constant for all $f \in \mathcal{F}$, i.e. $||f(x) - f(y)|| \leq G||x - y|| \ \forall x, y \in \mathcal{K}, f \in \mathcal{F}$ for some norm $|| \cdot ||$ and the diameter of $\mathcal{K}$ being D, i.e. $||x - y|| \leq D \ \forall x, y \in \mathcal{K}$

In the case of submodular functions, this is defined similarly, but there are more subtly different methods.

4.1 Online Learning in the Submodular Framework

In this section, we present background work on online submodular optimization. Generally, the long line of works attempt to exploit the submodularity by appending normal methods, such as experts and FTRL (Follow the Regularized Leader) methods. However, the emphasis and use for these algorithms are when we assume only bandit feedback. In bandit feedback, even if given the initial graph topology, it is not possible to infer what strategies others had played, and thus players must rely on the values of the function only for integral sets, rather than considering "fractional sets" in a relaxation.

In the case of full feedback, the following algorihms are not needed, because we can just perform concave extensions and proceed with normal convex optimization.

4.1.1 Online Submodular Maximization

[SG08] present a method for submodular approximation. In this setting, regret for the maximization problem is defined as

$$R_{max} = (1 - 1/e) \max_{S \in \mathcal{S}} \frac{1}{T} \sum_{t=1}^{T} f_t(S) - \frac{1}{T} \sum_{t=1}^{T} f_t(S_i) \tag{2}$$

Their regret bound becomes $O(k\sqrt{TV \log V})$, due to associating k experts $A_1, ..., A_k$ where expert A_i chooses element $e_i \in V$ and receives the marginal benefit, $f_t(\{e_1, ..., e_i\} - f_t(\{e_1, ..., e_{i-1}\})$ as reward feedback. More specifically, if expert A_i has regret ϵ_i, then it is shown that $f(S_k) \geq (1 - 1/e) \max_{|S| \leq k} f(S) - \sum_{i=1}^{k} \epsilon_i$.

One limitation of this method is that this does not allow the actions $S_1, ..., S_T$ to interact. There is no "averaging" of the actions in this case, for the sake of finding equilibria in games. To convert a boolean function f into one that admits inputs from $\mathbb{R}^V$, we are required to use a form of relaxation.

There are two well known considerations for such a relaxation. Note that applying a relaxation implies requiring almost full-information access to previous functions in the online setting in order to calculate the extension value for fractional inputs. We first provide an example of online submodular minimization, which uses well known submodular extensions.

In order to exploit the graph structure of the problem, we are required to use more specific extensions.

[1] Concave maximization is the same as convex minimization

4.1.2 Online Submodular Minimization

In the online submodular minimization in [HK12], similar methods to FTPL (follow the peturbed leader) are used in the full-information setting. Primarily, a randomized regularizer $R(S)$ is introduced, which is based on randomly selected values, and a parameter balancing parameter η, and $M = \max_{f \in \mathcal{F}} \max_{|S|=k} f(S)$ In other words, $R(S) = \sum_{i \in S} r_i$, where r_i's are i.i.d. sampled uniformly from $[-M/\eta, M/\eta]$. Then, the learner picks the set $S_t = \arg\min_{|S|=k} \sum_{\tau=1}^{t-1} f_\tau(S) + R(S)$ It is known that this achieves a regret bound of $O(M\sqrt{nT})$.

However, in the bandit case, the algorithm uses the Lovasz extension. The Lovasz extension is, for a given submodular f, $\mathbb{E}_{\lambda \in [0,1]}[f(\{i | x_i > \lambda\})]$. In the bandit setting, the regret is given by $O(MnT^{2/3})$

5 Relaxations of Submodular Graph

5.1 Network Cascade Relaxations

In general for cascading graph processes, there is a pattern of enumerating every realization G_r of the stochastic process. In [BPR+15], in the triggering model, for every realization G_r, they define $f_{G_r}(S)$ as the number of nodes reachable from S, with $f(S) = \sum_{G_r} p(G_r) f_{G_r}(S)$, over all possible realized graphs G_r, where $p(G_r)$ is the probability of obtaining G_r. This is a very well known technique used in a variety of approximation algorithms for influence maximization. Then they define the multilinear relaxation of f_{G_r}, bounded by another concave relaxation, where if $C_r(u)$ is the set of nodes that have a path to u, i.e.

$$\hat{f}_{G_r}(y) = \sum_{S \subseteq V} \prod_{j \in S} y_j \prod_{j \notin S} (1 - y_j) f_{G_r}(S) = \sum_{u \in V} 1 - \prod_{j \in C_r(u)} (1 - y_j) \ \forall y \in [0,1]^V \tag{3}$$

The RHS can be interpreted for each $u \in V$ as the probability that at least one node is selected in the reachability graph to u.

Since the multi-linear extension is generally not concave for the purposes of optimization, they bound the extension with $L_{G_r}(y) \geq \hat{f}_{G_r}(z) \geq (1 - 1/e) L_{G_r}(y)$, where

$$L_{G_r}(y) = \sum_{u \in V} \min\{1, \sum_{j \in C_r(u)} y_j\} \tag{4}$$

from making a first order approximation on the polynomial $\prod_{j \in C_i(u)} (1 - y_j) \approx 1 - (\sum_{j \in C_r(u)} y_j)$ Thus, it is easy to define a weighted total sum, $L(y) = \sum_{G_r} p(G_r) L_{G_r}(y)$ and L is concave because it is a weighted sum of concave functions.

This can be extended to a multi-player case, where we may define $\hat{f}_{G_r}(z_1, ..., z_n) = \hat{f}_{G_r}(\mathbf{z})$, $\forall z_1, ..., z_n \in [0,1]^V$ as a vector, with the i-th entry $\hat{f}^{(i)}_{G_r}(\mathbf{z})$ being the multi-linear extension based on player i, where we replace $f_{G_r}(S)$ with a vectorized version $f_{G_r}(S_1, ..., S_n) = f_{G_r}(\mathbf{S})$, that outputs a vector corresponding to each of the player's gains for that realization. We may assume that $f_{G_r}(\mathbf{S})$ may output fractional values as well, depending on the de-randomization G_r of the graph function. We also assume that such a de-randomization can be given from an oracle sampler. In some competitive cases this may be a flawed assumption, but many nice and interesting competitive cases in fact, do admit a good sampler. Writing this out, we have

$$\hat{f}_{G_r}(z_1, ..., z_n) = \sum_{S_1, ..., S_n \subseteq V} \left(\prod_{1 \leq i \leq n} \prod_{j \in S_i} z_{i,j} \prod_{j \notin S} (1 - z_{i,j}) \right) f_{G_r}(S_1, ..., S_n) \tag{5}$$

The above is a multivariate polynomial in the $z_{i,j}$ terms. If we treat one player i's as variables, while the other players as held fixed, then we have a similar structure, admitting to a concave relaxation due to first order approximations. Rearranging terms and only looking at the i-th component will give

$$\hat{f}_{G_r}(z_1, ..., z_n)_i = \sum_{u \in V} \sum_{C_r(u)} Pr(C_r(u)) \left(1 - \prod_{j \in C_r(u)} (1 - z_{i,j}) \right) \tag{6}$$

with different coefficients $Pr(C_r(u))$ representing the probability of obtaining reachability set $C_r(u)$ after the other players had probabilistically played their strategies from z_{-i} (the realization G_r however, is fixed as a directed graph). Then we may have a similar relaxation,

i.e.

$$L_{G_r}(\mathbf{z})_i = \sum_{u \in V} \sum_{C_r(u)} Pr(C_r(u)) \min\{1, \sum_{j \in C_r(u)} z_{i,j}\} \tag{7}$$

which is also concave, and gives the same $(1 - 1/e)$ bounding because the only part relaxed was from $1 - \prod_{x_i}(1 - x_i)$ to $\min\{1, \sum_{x_i} x_i\}$, which is well known. We can give the similar definition of general $L(\mathbf{z})$, as

$$L(\mathbf{z}) = \sum_{G_r} p(G_r) L_{G_r}(\mathbf{z}) \tag{8}$$

Calculating L precisely is difficult due to the exponential size of the instances. Instead, one may approximate L by sampling many instances G_r.

There are certain nice properties for L which makes it useful for game theory. We invoke the conditions found in [EMN09] for socially concave games, in which the game satisfies two properties:

1. There exists a positive convex combination, $g(z) = \sum_{i \in n} \lambda_i u_i(\mathbf{z}) = \sum_{i \in n} \lambda_i L(\mathbf{z})_i$, where $\lambda_i > 0 \ \forall i = 1, ..., n$ of each player's utilities, which is a concave function.
2. Every player's utility function is convex with respect to the other players' actions, i.e. $u_i(z, z_{-i}) = L(z_i, z_{-i})_i$ is convex in z_{-i}.

Furthermore, an additive ϵ_t-Nash equilibrium may be obtained by averaging the previous z-vectors used, where the difference is $\epsilon_t = \frac{1}{\lambda_{min}} \sum_{i \in n} \frac{\lambda_i R_i(t)}{t}$, given that $R_i(t)$ corresponds to the regret incurred by player i at time t.

Theorem 2. *The game induced by L is a socially concave game.*

Condition 1 is a form of the welfare of the players;

and condition 2 is satisfied because the marginal decrease slows down as the other players seed more (NEED TO PROVE).

The work in [EMN09] states that given these two conditions, it is possible to use no-regret dynamics to converge towards a Nash equilibrium, where the equilibrium is the limit of the averages of the actions.

Note that this does not imply any results for general submodular games, because it is not necessarily always the case that submodular games admit concave relaxations. In certain games in which other players' actions may marginally help a fixed player, these do not correspond to socially concave games.

Theorem 3. *A ϵ- Nash equilibrium for the relaxation game L implies a $(1-1/e-o(1))$ approximate Nash equilibrium for the original game.*

Given the convergent vectors of each player, we can then apply Pipage Rounding 5.2 to give an expectation guarentee. (THE PROOF MUST INVOLVE CONVERTING ROUNDING GUARENTEES FOR FIXED ADVERSARIIES TO MOVING ADVERSARIES)

Suppose we found an ϵ-Nash Equilibrium vector $\mathbf{z} = (z_1, ..., z_n)$ for L; i.e. for all alternative strategies $\hat{z}_i$,

$$L(z_i, z_{-i})_i \geq L(\hat{z}_i, z_{-i})_i - \epsilon \tag{9}$$

which implies that

$$\hat{f}(z_i, z_{-i})_i \geq (1-1/e)L(z_i, z_{-i})_i \geq (1-1/e)\left(L(z_i, z_{-i})_i - \epsilon\right) \geq (1-1/e)\left(\hat{f}(z_i, z_{-i})_i - \epsilon\right) \quad (10)$$

Note that we are unable to state that the blocking and zero-sum games are socially concave, however.

5.2 Randomized Roundings from Continuous to Discrete

One important restriction that we must handle is the conversion from a continuous vector z to a discrete set S at each turn. There is a pipage rounding scheme found in [AS04] and [CCPV07] that gives a set $|S| = k$ with high probability, satisfying $(1+o(1))\hat{f}(z) \geq f(S) \geq (1-o(1))\hat{f}(z)$. A reason for concern is the robustness of such algorithms, as mentioned in [FLL+16]. It could be possible that a randomized strategy may give worst regret rates, and the dynamics may lead toward suboptimal positions with some probability.

The pipage rounding scheme, treated as a black box, will always output a random $|S| = k$ in polynomial time given z, such that $(1+\delta)\hat{f}(z) \geq \mathbb{E}_S[f(S)] \geq \hat{f}(z)$. Note that these bounds then satisfy $(1+\delta)L(z) \geq \mathbb{E}_S[f(S)] \geq (1-1/e)L(z)$

5.2.1 No-regret algorithms for the concave relaxation

Due to the concave relaxation (and for practical purposes, the sampled $\hat{L}(\cdot, z_{-i}^{(\tau)})$, we wish to produce an online-convex optimization approach to minimizing our regret). In [Haz16], unfortunately we cannot perform normal approaches, such as FTRL (follow the regularized leader), because L is not globally differentiable due to the min function. However, the gradient of L exists near the origin if we remove the min functions (which will be an upper bound index-wise for all parts of the domain for which L does have a derivative); for each realization G_r,

$$\frac{\partial}{\partial (z_i)_j} L_{G_i}(z_i, z_{-i})_i \leq \sum_{u \in V} w_j^i \mathbf{1}[j \in C_r(u)] \quad (11)$$

where $\mathbf{1}[j \in C_r(u)]$ is the indicator function for if j can reach u in the realization G_r. In the base case for which the weights $w_j^i = 1$ in the normal cascading case, assuming no other players have placed their seed sets (z_{-i} is the all-zero vector), the above expression on the right will be the number of points reachable from vertex j. In total, therefore, the general function L's gradient will have the component wise bound,

$$\frac{\partial}{\partial (z_i)_j} L(z_i, z_{-i})_i \leq \sum_{G_r} p(G_r) \left(\sum_{u \in V} \mathbf{1}[j \in C_r(u)] \right) \quad (12)$$

The RHS is really just the expected influence of only putting node j in the graph for a player (which is expected due to the submodularity of f). This implies that the Lipschitz constant in the ℓ_2-norm for $(L)_i$ is at most $G = \max_{j \in V} f(\{j\})_i$. This constant is generally quite small in real world graphs, but it encapsulates the structure of the graph as well. In case this constant is too large, we may lower it by considering the opponents' actions as well.

An alternative for online convex optimization, without the gradient involve non-differentiable convex optimization methods, such as online subgradient descent. We propose the following strategy in this game, which will give us $O(kG\sqrt{T})$ regret.

5.3 Coupling for faster Regret Rates

In this section, we

Algorithm 1 Game Playing Algorithm for Full Bandit Case

$z_i^{(1)}$ to be some arbitrary vector from $\triangle_k$
for $t = 1...T$ **do**
  player plays $z_i^{(t)}$
  opponents play $z_{-i}^{(t)}$
  compute L by sampling
  $\tilde{z}_i^{(t+1)} = z_i^{(t)} - \eta_t g_t$ where g is a subgradient of L
  Projection: $z_i^{(t+1)} = \arg\min_{z \in \triangle_k} ||z - \tilde{z}_i^{(t+1)}||_2^2$
  Play $S_t =$ Pipage Round$(z_i^{(t+1)})$

5.4 Zero-Sum and Blocking Relaxations

In the zero-sum game case, we can just use the exact same relaxation methods proposed above, for the maximizing player.

In the blocking case,

6 Usage on Submodular Graph Games

6.1 Zero-Sum

In this setting, over many rounds, the attacking player will attempt to use online submodular maximization, while the defending player will use online submodular minimization. We examine the dynamics of this in the full-information case, as well as the bandit case. While it is known that no-regret dynamics converge quickly to Nash equilibrium in a zero sum game, as seen in [DDK15] (which uses a version of Nesterov's descent technique) and [RS13] (which uses a modified version of normal MWU), it is unclear at first if one player must use an approximation algorithm.

In the maximizer (attacker)'s case, he is bound by the $(1 - 1/e)$ approximation hardness for submodular maximization, while the minimizer (defender) actually does not have a hardness of approximation ratio, because submodular minimization is polynomial time-solvable [HK12].

This supports the instance that generally, "diligent attackers" are as powerful as "lazy defenders", a notion used from [BHP14] in security games.

(MUST ANALYZE)

6.1.1 Bandit Setting

From the standpoint of the bandit case, we seek the performance of the players oblivious to each other's actions. The bandit setting does not allow for relaxations of the function, and hence this setting implies that we will be approaching the dynamics of zero-sum submodular games, for general submodular functions. In [DDK15], it is known that in the bandit setting, there exists no-regret dynamics for 2-players in zero sum games that converge to exact Nash equilibrium in $O(1/T)$ time.

7 Appendix

7.1 Examples of De-Randomized Cascading Games

7.1.1 Tie-Breaking Functions

In most competitive games involving cascades, there exists a tiebreaking function for each node v (once initially activated), $h_v : N_v^{in}(S_1, ..., S_n) \to \mathbb{R}^n$ where N_v^{in} is the set of in-neighbors of v, and the output is a probability vector for choosing one of the players. Simple tie-breaking rules may include preference from an already established permutation π, where every node will choose among his active neighbors, the highest ranking node in the permutation. For a fixed opponent strategy, this still corresponds to the reachability-functions mentioned in 5.1. For instance, $f_{G_r}(S_1, ..., S_n)$ for a realization G_r would output a vector whose i-th component would be the number of nodes influenced by S_i in the game dynamics.

Another rule is proportional tiebreaking, in which $h_v(S_1, ..., S_n) = \left[\frac{|S_1|}{\sum |S_i|}, ..., \frac{|S_n|}{\sum |S_i|}\right]$ where each player's probability of being chosen for v is proportional to the number of his neighbors for v. The de-randomization of this is choosing a random permutation for each vertex, and following the fixed permutation case. Using the above case, then $f_{G_r}(S_1, ..., S_n)$ would be, after having sampled all the permutations for each vertex, outputs the vector of reached nodes by each of S_i. The sampling method is oblivious to support size, and hence the query complexity is still the same.

References

[AFL15] Aris Anagnostopoulos, Diodato Ferraioli, and Stefano Leonardi. Competitive influence in social networks: Convergence, submodularity, and competition effects. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2015, Istanbul, Turkey, May 4-8, 2015*, pages 1767–1768, 2015.

[AS04] Alexander A. Ageev and Maxim Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3):307–328, 2004.

[BBCL14] Christian Borgs, Michael Brautbar, Jennifer T. Chayes, and Brendan Lucier. Maximizing social influence in nearly optimal time. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 946–957, 2014.

[BHP14] Avrim Blum, Nika Haghtalab, and Ariel D. Procaccia. Lazy defenders are almost optimal against diligent attackers. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 573–579, 2014.

[BKLP14] Branislav Bosanský, Christopher Kiekintveld, Viliam Lisý, and Michal Pechoucek. An exact double-oracle algorithm for zero-sum extensive-form games with imperfect information. *J. Artif. Intell. Res. (JAIR)*, 51:829–866, 2014.

[BPR+15] Ashwinkumar Badanidiyuru, Christos H. Papadimitriou, Aviad Rubinstein, Lior Seeman, and Yaron Singer. Locally adaptive optimization: Adaptive seeding for monotone submodular functions. *CoRR*, abs/1507.02351, 2015.

[CCPV07] Gruia Călinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a submodular set function subject to a matroid constraint (extended abstract). In *Integer Programming and Combinatorial Optimization, 12th International IPCO Conference, Ithaca, NY, USA, June 25-27, 2007, Proceedings*, pages 182–196, 2007.

[COP13] Ozan Candogan, Asuman E. Ozdaglar, and Pablo A. Parrilo. Dynamics in near-potential games. *Games and Economic Behavior*, 82:66–90, 2013.

[DDK15] Constantinos Daskalakis, Alan Deckelbaum, and Anthony Kim. Near-optimal no-regret algorithms for zero-sum games. *Games and Economic Behavior*, 92:327–348, 2015.

[EMN09] Eyal Even-Dar, Yishay Mansour, and Uri Nadav. On the convergence of regret minimization dynamics in concave games. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 523–532, 2009.

[FLL+16] Dylan J. Foster, Zhiyuan Li, Thodoris Lykouris, Karthik Sridharan, and Éva Tardos. Learning in games: Robustness of fast convergence. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4727–4735, 2016.

[GK12] Sanjeev Goyal and Michael Kearns. Competitive contagion in networks. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 759–774, 2012.

[Haz16] Elad Hazan. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2(3-4):157–325, 2016.

[HK12] Elad Hazan and Satyen Kale. Online submodular minimization. *Journal of Machine Learning Research*, 13:2903–2922, 2012.

[HK13] Xinran He and David Kempe. Price of anarchy for the n-player competitive cascade game with submodular activation functions. In *Web and Internet Economics - 9th International Conference, WINE 2013, Cambridge, MA, USA, December 11-14, 2013, Proceedings*, pages 232–248, 2013.

[HK16] Elad Hazan and Tomer Koren. The computational power of optimization in online learning. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 128–141, 2016.

[HSCJ12] Xinran He, Guojie Song, Wei Chen, and Qingye Jiang. Influence blocking maximization in social networks under the competitive linear threshold model. In *Proceedings of the Twelfth SIAM International Conference on Data Mining, Anaheim, California, USA, April 26-28, 2012.*, pages 463–474, 2012.

[HST15] Jason D. Hartline, Vasilis Syrgkanis, and Éva Tardos. No-regret learning in bayesian games. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 3061–3069, 2015.

[JB11] Stefanie Jegelka and Jeff A. Bilmes. Online submodular minimization for combinatorial structures. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 345–352, 2011.

[KKT03] David Kempe, Jon M. Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003*, pages 137–146, 2003.

[KV05] Adam Tauman Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *J. Comput. Syst. Sci.*, 71(3):291–307, 2005.

[MR10] Elchanan Mossel and Sébastien Roch. Submodularity of influence in social networks: From local to global. *SIAM J. Comput.*, 39(6):2176–2188, 2010.

[PP16] Christos H. Papadimitriou and Georgios Piliouras. From nash equilibria to chain recurrent sets: Solution concepts and topology. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science, Cambridge, MA, USA, January 14-16, 2016*, pages 227–235, 2016.

[Rou16] Tim Roughgarden. Intrinsic robustness of the price of anarchy: Abstract of the kalai prize talk. In *Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016*, page 457, 2016.

[RS13] Alexander Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3066–3074, 2013.

[RSS15] Aviad Rubinstein, Lior Seeman, and Yaron Singer. Approximability of adaptive seeding under knapsack constraints. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation, EC '15, Portland, OR, USA, June 15-19, 2015*, pages 797–814, 2015.

[SALS15] Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E. Schapire. Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 2989–2997, 2015.

[SG08] Matthew J. Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. In *Advances in Neural Information Processing Systems 21, Proceedings* *of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008*, pages 1577–1584, 2008.

[TNT12] Jason Tsai, Thanh Hong Nguyen, and Milind Tambe. Security games for controlling contagion. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, 2012.