Efficient Learning in Polyhedral Games via Best Response Oracles*

Darshan Chakrabarti DC3595@COLUMBIA.EDU
IEOR Department, Columbia University

Gabriele Farina GFARINA@MIT.EDU
EECS Department, MIT

Christian Kroer CK2945@COLUMBIA.EDU
IEOR Department, Columbia University

Abstract

We study online learning and equilibrium computation in games with polyhedral decision sets with only first-order oracle and best-response oracle access. Our approach achieves constant regret in zero-sum games and $O(T^{1/4})$ in general-sum games while using only $O(\log t)$ best-response queries at a given iteration t. This convergence occurs at a linear rate, though with a condition-number dependence. Our algorithm also achieves best-iterate convergence at a rate of $O(1/\sqrt{T})$ without such a dependence. Our algorithm uses a linearly convergent variant of Frank-Wolfe (`FW`) whose linear convergence depends on a condition number of the polytope known as the facial distance. We show two broad new results, characterizing the facial distance when the polyhedral sets satisfy a certain structure.

1. Introduction

Learning in games is a well-studied framework in which agents iteratively refine their strategies through repeated interactions with their environment. Best-responding is a natural way for agents to refine their strategies iteratively. This leads to the question: what are the best convergence guarantees that can be obtained for the computation of Nash equilibria in two-player zero-sum games or coarse correlated equilibria in multiplayer games when learning using a best-response oracle?

In the online learning community, methods based only on best-response oracles are special cases of methods based on a *linear minimization oracle* (LMO), which can be queried for points that minimize a linear objective over the feasible set. Such methods are known as *projection-free* methods because they may avoid potentially expensive projections onto the feasible set.

Projection-free online learning algorithms might perform multiple LMO calls per iteration, so our paper and related literature are concerned with both the number of iterations T of online learning and the total number of LMO calls, which we denote by N. Since LMOs for polyhedral decision sets essentially correspond to best-response oracles (BROs), we will use these two terms interchangeably.

The table below compares our algorithm with `FTPL` [36] and `OFTPL` [53] which have the best-known guarantees for the setting we consider. In the table, we also include a non-optimistic version of our algorithm; despite having worse theoretical guarantees than existing projection-free algorithms, it outperforms them in our numerical experiments.

The optimization community has also done substantial work on developing projection-free methods, spurred by the work of Frank and Wolfe [24]. Guarantees for the Frank-Wolfe algorithm

* The full version of this paper can be found at https://arxiv.org/abs/2312.03696.

typically assume the function f being optimized is smooth (has Lipschitz gradient) and convex, the domain $\mathcal{X}$ being optimized over is convex and compact, and that the algorithm has access to a first-order oracle for the function which returns gradients $\nabla f(\boldsymbol{x})$ at a queried point $\boldsymbol{x}$ and a LMO which returns solutions to minimization problems of the form $\operatorname{argmin}_{\boldsymbol{x}\in\mathcal{X}}\langle \boldsymbol{c}, \boldsymbol{x}\rangle$ for any choice of $\boldsymbol{c} \in \mathbb{R}^n$. Given an initial iterate $\boldsymbol{x}^{(0)}$, it produces new iterates given by the following update rule:

$$\boldsymbol{x}^{(t+1)} = \frac{t}{t+2}\boldsymbol{x}^{(t)} + \frac{2}{t+2}\operatorname*{argmin}_{\boldsymbol{x}'\in\mathcal{X}}\langle \nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{x}'\rangle$$

In recent years, there has been work on developing `FW`-based approaches to saddle-point computation (e.g., Gidel et al. [28], Lan and Zhou [43]). However, Gidel et al. [28] only has fast convergence guarantees for strongly convex-concave objectives, and Lan and Zhou [43] are only able to provide $O(1/\sqrt{N})$ convergence to saddle-points. On the other hand, our method is able to leverage a `FW` variant, away-step Frank-Wolfe (`AFW`), to achieve faster convergence rates.

An extended discussion of related work is given in Appendix A.1.

Contributions We present a projection-free online learning method, Approximate Reflected Online Mirror Descent, using away-step Frank-Wolfe (`AFW-ROMD`), for learning over compact and convex polyhedral decision sets. Using the linear convergence of `AFW` for polyhedral domains, we implement approximate steps of reflected online mirror descent (`ROMD`) using only a logarithmic number of `AFW` iterations. While using `FW`-based methods to approximate proximal steps has been previously studied, pioneered by work of Lan and Zhou [43], it is a surprising blind spot in the literature that the connection to regret guarantees for games has not previously been made.

As can be seen in Table 1, in a two-player zero-sum polyhedral game, when both players employ `AFW-ROMD`, it is possible to converge to a Nash equilibrium in a two-player zero-sum game at a rate of $O(\log N/N)$. More generally, we show that `AFW-ROMD` requires only $O(\log t)$ best-response queries at each self-play iteration t while guaranteeing constant social regret, as well as $O(T^{1/4})$ regret for each player after T total iterations of self-play. We go on to study the *last-iterate convergence* properties of `AFW-ROMD` in self-play in zero-sum settings. We show that the last iterate converges at a condition-number-dependent linear rate (up to error induced by the approximate proximal computation). To the best of our knowledge, these are both the first last-iterate convergence and the first linear-rate convergence results for self-play dynamics that purely rely on best-response oracles. We also show asymptotic last-iterate convergence at a condition-number-free rate of $O(1/\sqrt{T})$.

The linear convergence of `AFW` depends on the facial distance constant of the polytope in question (in addition to the strong convexity and smoothness constants). To that end, we show two novel lower bounds on the facial distance of a polytope. Our first result concerns polytopes that can be described in the form $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \mathbf{0}$ where $\boldsymbol{x} \in \mathbb{R}^n$. Let γ be the minimum value of a nonzero coordinate of a vertex in the polytope. Then, we show that the facial distance is at least $\gamma/\sqrt{n}$. The fact that the facial distance is only square-root power small in the dimension of the problem ensures that the convergence rate of linearly convergent `FW` variants over these polytopes does not scale poorly as the ambient dimension of the problem increases. Our second result concerns an *integral* polytope $\mathcal{P}$ given by $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \mathbf{C}\boldsymbol{x} \leq \boldsymbol{d}, \boldsymbol{x} \geq \mathbf{0}$ where $\boldsymbol{x} \in \mathbb{R}^n$, with $\mathbf{C} \geq \mathbf{0}$ a non-zero integral matrix, and $\boldsymbol{d} \geq \mathbf{0}$. In that case, we show that the facial distance is at least $1/(\|\mathbf{C}\|_\infty\sqrt{n})$.

Finally, we conduct experiments demonstrating competitive practical performance of our algorithm relative to other projection-free algorithms when computing Nash and coarse correlated equilibria in polyhedral games.

Algorithm	∇ computations at iteration t	LMO calls at iteration t	Social regret $\sum_i \text{Reg}_i^{(T)}$	Avg. social regret $\sum_i \frac{1}{T}\text{Reg}_i^{(T)}$
FTPL [36]	$O(1)$	$O(1)$	$O(\sqrt{T})$	$O(1/\sqrt{N})$
Optimistic FTPL (OFTPL) [53]	$O(1)$	$O(T)$	$O(1)$	$O(1/\sqrt{N})$
AFW-OMD [this paper]	$O(1)$	$O(\log t)$	$O(\sqrt{T})$	$O(\sqrt{\log N/N})$
AFW-ROMD [this paper]	$O(1)$	$O(\log t)$	$O(1)$	$O(\log N/N)$

Table 1: Number of gradient (∇) computations, number of LMO calls, cumulative regret (as a function of the total number of iterations T), and average regret (as a function of total LMO calls N) of various projection-free algorithms. In two-player zero-sum games, average social regret upper bounds the duality gap to Nash equilibrium for the averaged iterates.

2. New Results on Polyhedral Facial Distance

Frank-Wolfe (FW) is a projection-free algorithm that converges with rate $O(1/T)$ for smooth convex functions over convex compact sets. Away-Step Frank Wolfe (AFW) is a variant of Frank-Wolfe which achieves linear convergence for strongly convex objectives over polyhedral sets [30, 41, 60]. Pseudocode for AFW is provided in Appendix A.

The facial distance δ of a polytope is a relevant quantity when characterizing the convergence rate of several linearly convergent variants of FW, including AFW. It can be defined concisely using a theorem from [49]: $\delta(\mathcal{P}) = \min_{\substack{\mathcal{F}\in\text{faces}(\mathcal{P})\\ \emptyset\subsetneq\mathcal{F}\subsetneq\mathcal{P}}} \text{dist}(\mathcal{F}, \text{Conv}(\text{Vert}(\mathcal{P}) \setminus \mathcal{F}))$. Theorem 1 demonstrates the convergence rate dependence of AFW on the facial distance.

Theorem 1 (AFW for strongly convex functions over polyhedral sets [41]) *In order to compute an ϵ-optimal solution to a μ-strongly convex L-smooth function over a convex polytope that has diameter D and facial distance δ, AFW requires $O(\frac{LD^2}{\delta^2}\log\frac{LD}{\epsilon})$ LMO calls.*

In general, computing the facial distance is considered to be non-trivial for most polytopes besides hypercubes, unit ℓ_1 balls, and simplices [6, 10, 27], since the definition requires evaluating a combinatorial number of distances. We provide lower bounds on the facial distance in special cases where the constraints of the polytope can be written in a certain form. The proofs are deferred to Appendix C. For any $0/1$-polytope, $\mathcal{X}$, $\delta(\mathcal{X}) \geq \frac{1}{\sqrt{n}}$ since Theorem 2 holds with $\gamma = 1$ for $0/1$ polytopes.

Theorem 2 *Let $\mathcal{P}$ be a polytope given by $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \mathbf{0}$ where $\boldsymbol{x} \in \mathbb{R}^n$. Let γ be the minimum value of a nonzero coordinate of a vertex. Then $\delta(\mathcal{P}) \geq \frac{\gamma}{\sqrt{n}}$. Moreover, if the optimal solution lies in a face $\mathcal{F}$ such that k coordinates are zero, then $\delta(\mathcal{P}) \geq \frac{\gamma}{\sqrt{k}}$.*

Theorem 3 *Let $\mathcal{P}$ be an integral polytope given by $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \mathbf{C}\boldsymbol{x} \leq \boldsymbol{d}, \boldsymbol{x} \geq \mathbf{0}$ where $\boldsymbol{x} \in \mathbb{R}^n$, with $\mathbf{C} \geq \mathbf{0}$ a nonzero integral matrix, and $\boldsymbol{d} \geq \mathbf{0}$. Then $\delta(\mathcal{P}) \geq \frac{1}{\|\mathbf{C}\|_\infty\sqrt{n}}$.*

3. Approximate Reflected Online Mirror Descent using Away-Step Frank-Wolfe

In this section, we propose a framework of algorithms that uses approximate proximal updates instead of exact proximal updates. We show that this framework still retains several nice properties of ROMD, up to the error in the approximation oracle. Then, we propose the use of linearly convergent variants of FW for implementing the approximate proximal step, specifically when the regularizer is smooth and strongly convex (as is the case with the Euclidean regularizer) and the decision set is a convex polytope, which is the case in normal-form games (NFGs) and extensive-form games (EFGs).

We abstract away the concept of computing an approximate proximal update using what we call an approximate proximal oracle (APO).

Definition 1 (Approximate proximal oracle) *An* $\text{APO}_{\mathcal{X}}$*, given a choice of convex and compact set* $\mathcal{X}$*, takes as input a function* $f : \mathcal{X} \to \mathbb{R}$*, a L-smooth and 1-strongly convex regularizer* φ*, a prox center* $\boldsymbol{x}_c$*, and a desired accuracy* $\epsilon \geq 0$*, and returns* $\boldsymbol{x}' \in \mathcal{X}$ *such that*

$$f(\boldsymbol{x}') + \mathcal{D}_{\varphi}(\boldsymbol{x}' \,\|\, \boldsymbol{x}_c) \leq \min_{\boldsymbol{x} \in \mathcal{X}} \Big\{ f(\boldsymbol{x}) + \mathcal{D}_{\varphi}(\boldsymbol{x} \,\|\, \boldsymbol{x}_c) \Big\} + \epsilon. \tag{APO}$$

While our framework can be adapted to various online learning algorithms, we illustrate the framework using ROMD.

Given a convex and compact set $\mathcal{X} \subseteq \mathbb{R}^n$, a L-smooth, 1-strongly convex regularizer $\varphi : \mathcal{X} \to \mathbb{R}_{\geq 0}$, step-size $\eta > 0$, $\epsilon^{(t)}$ the desired accuracy of the prox call at iteration t, and an APO for $\mathcal{X}$, we consider the following update for the next strategy: $\boldsymbol{x}^{(t)} = \text{APO}_{\mathcal{X}}(-\eta\langle \boldsymbol{\ell}^{(t-1)} + \boldsymbol{m}^{(t)} - \boldsymbol{m}^{(t-1)}, \cdot\rangle, \varphi, \boldsymbol{x}^{(t-1)}, \epsilon^{(t)})$, where $\boldsymbol{\ell}^{(t)}$ is the loss received at iteration t, and $\boldsymbol{m}^{(t)}$ is the prediction of the loss to be used at iteration t. Proofs for results in this section are deferred to Appendix D.

When the above framework is instantiated with AFW (Algorithm 1) as the APO, we refer to it as AFW-ROMD. We are able to show ergodic convergence to equilibrium and characterize average regret in terms of LMO calls for AFW-ROMD.

Theorem 4 *An* ϵ'*-Nash equilibrium in any two-player zero-sum polyhedral game can be computed in* $O(1/\epsilon')$ *iterations of the above framework. This corresponds to* $O(\max_{i \in \{1,2\}} \frac{1}{\epsilon'} \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i}{\epsilon'}\right])$ *LMO calls when using* AFW-ROMD.

Theorem 5 *An* ϵ'*-CCE in any* N*-player general-sum polyhedral game can be computed in* $O(1/\epsilon'^{\frac{4}{3}})$ *iterations of the above framework. This corresponds to* $O(\max_{i \in [N]} \frac{1}{\epsilon'^{\frac{4}{3}}} \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i}{\epsilon'}\right])$ *LMO calls when using* AFW-ROMD.

Last Iterate Convergence We obtain asymptotic last-iterate convergence to an approximate Nash equilibrium, when $\sum_{i=1}^N \text{Reg}_i^{(t)} \geq 0$ for any $t \in \mathbb{N}$, adapting a result from Anagnostides et al. [3]. A wide class of games, including two-player NFGs and EFGs, polymatrix zero-sum games, constant-sum polymatrix games, strategically zero-sum games, and polymatrix strategically zero-sum games satisfy this condition on social regret [3]; thus, our result holds for this class of games as well.

Theorem 6 *For any* N*-player general-sum polyhedral game, given* $\epsilon \in (0, 1)$*, let Player* i *employ the above framework with* $\epsilon_i^{(t)} = \epsilon^2$ *and* $\boldsymbol{m}_i^{(t)} = \boldsymbol{\ell}_i^{(t-1)}$*. Let* $\eta_{\max} \leq \frac{1}{2\sqrt{2(N-1)}}$ *where* $\eta_{\max} = \max_{i \in [N]} \eta_i$ *and suppose* $\sum_{i=1}^N \text{Reg}_i^{(t)} \geq 0$ *for any* $t \in \mathbb{N}$*. Define* $\alpha_i = \left(\frac{1}{\eta_i} + \frac{2\Omega_i}{\eta_i}(L_i + N - 1) + 1\right)$*. Then, after* $T > \left\lceil \frac{8\eta_{\max}}{\epsilon^2} \sum_{i=1}^N \frac{(\Omega_i + 2)}{\eta_i} \right\rceil$ *iterations, there exists* $\boldsymbol{x}^{(t)}$ *with* $t \in [T]$ *which is an* $\epsilon\Big(\max_{i \in [N]} \sqrt{2\eta_i}(\frac{2L_i D_i}{\eta_i} + 3) + \alpha_i\Big)$*-approximate Nash equilibrium.* AFW-ROMD *will yield an iterate that is an* ϵ'*-approximate Nash equilibrium in* $O\left(\max_{j \in [N]} \left\{ \frac{\eta_{\max}\alpha_j^2}{\epsilon'^2} \sum_{i=1}^N (\frac{\Omega_i + 2}{\eta_i}) \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i \alpha_j}{\epsilon'}\right] \right\}\right)$ *LMO calls when* $\epsilon \leq \min_{i \in [N]} \frac{\epsilon'}{\alpha_i}$.

In the two-player zero-sum case, we also obtain last-iterate linear-rate convergence to ϵ'-equilibria when instantiated with the Euclidean regularizer, $\varphi_i(\boldsymbol{x}_i) = \frac{1}{2}\|\boldsymbol{x}_i\|_2^2$ for $i \in \{1, 2\}$, for any ϵ'.

Theorem 7 *In any two-player zero-sum polyhedral game, both players employing the approximate-proximal-step-based framework presented in Section 3 with $\boldsymbol{m}_i^{(t)} = \boldsymbol{\ell}_i^{(t-1)}$, $\epsilon_i^{(t)} = \epsilon$, $\varphi_i(\boldsymbol{x}_i) = \frac{1}{2}\|\boldsymbol{x}_i\|_2^2$, and $\eta_i = \eta \leq \frac{1}{4}$ yields linear last-iterate convergence to a $\frac{(16+C_1)\epsilon+32\max_{i\in\{1,2\}}\sqrt{2\eta\epsilon}(2L_iD_i+3\eta)}{C_2}$-approximate Nash equilibrium, where ν is a game-dependent constant associated with the SP-MS condition, $C_1 = 2(1+\frac{4\eta^2\nu^2}{25})$, and $C_2 = \min(\frac{1}{2}, \frac{\eta^2\nu^2}{25})$:*

$$\mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 \leq 2\left(1+\frac{C_2}{4}\right)^{-t}\mathrm{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2 + \frac{(16+C_1)\epsilon + 32\max_{i\in\{1,2\}}\sqrt{2\eta\epsilon}(2L_iD_i+3\eta)}{C_2}.$$

In the same setting ($\boldsymbol{m}_i^{(t)} = \boldsymbol{\ell}_i^{(t-1)}$, $\epsilon_i^{(t)} = \epsilon$, and $\eta_i = \eta \leq \frac{1}{4}$), if it is assumed that both players are applying `AFW-ROMD`*, then they can achieve linear last-iterate convergence to a $\frac{48+C_1}{C_2}\epsilon$-approximate Nash equilibrium, with the same definitions for ν, C_1, C_2.*

$$\mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 \leq 2\left(1+\frac{C_2}{4}\right)^{-t}\mathrm{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2 + \frac{48+C_1}{C_2}\epsilon.$$

`AFW-ROMD` *requires*

$$O\left(\max_{i\in\{1,2\}} \frac{\log\frac{2C_2+48+C_1}{C_2\epsilon'}}{\log\frac{4+C_2}{4}} \frac{L_iD_i^2}{\delta_i^2}\log\left[\frac{(2C_2+48+C_1)L_iD_i}{C_2\epsilon'}\right]\right).$$

LMO calls to compute an ϵ'-NE. Furthermore, the approximate solution it returns will have support of size $O\Big(\max_{i\in\{1,2\}}\frac{L_iD_i^2}{\delta_i^2}\log\Big[\frac{L_iD_i(2C_2+48+C_1)}{C_2\epsilon'}\Big]\Big)$.

4. Experimental Results and Discussion

We conduct experiments on standard EFG benchmarks to demonstrate the numerical performance of our algorithm relative to known algorithms from the literature. Details of games are provided in Appendix F. In addition to evaluating `AFW-ROMD`, we also consider its non-optimistic variant, `AFW-OMD`, which corresponds to using `AFW` as an APO for the prox computation in vanilla `OMD`. We use the Euclidean regularizer, $\varphi_i(\boldsymbol{x}_i) = \frac{1}{2}\|\boldsymbol{x}_i\|_2^2$ for $i \in [N]$.

We compare against `FTPL` and `OFTPL`, fictitious play (`FP`) [11] and best-response dynamics (`BR`), the latter two being unregularized variants of `FTRL`/`FTPL` and `OMD`, respectively. Finally, we also compare to *optimistic* versions of fictitious play (`OFP`) and best-response dynamics (`OBR`). We provide pseudocode for all of these algorithms in Appendix G; the pseudocode for these algorithms explicitly demonstrates that only one LMO call is required per iteration of these algorithms. Because `FP` and `BR` represent unregularized variants of `FTRL`/`FTPL` and `OMD`, they can be thought of as letting the stepsize be arbitrarily large for `FTRL` and `OMD` respectively (as the stepsize grows arbitrarily large or noise grows arbitrarily small).

For `AFW-OMD`, `AFW-ROMD`, `FTPL`, and `OFTPL`, we try $\eta \in 0.01 \cdot 2^{[14]}$, where η is the stepsize for our algorithms, while η is the noise used for `FTPL` and `OFTPL`. Additionally, we try uniform, linear, and quadratic iterate averaging for all algorithms, as well as last-iterate. Non-uniform averaging schemes are known to often outperform uniform averaging when solving BSPPs [25, 55]. Note that we demonstrate theoretical guarantees for last-iterate convergence of `AFW-ROMD`, whereas the

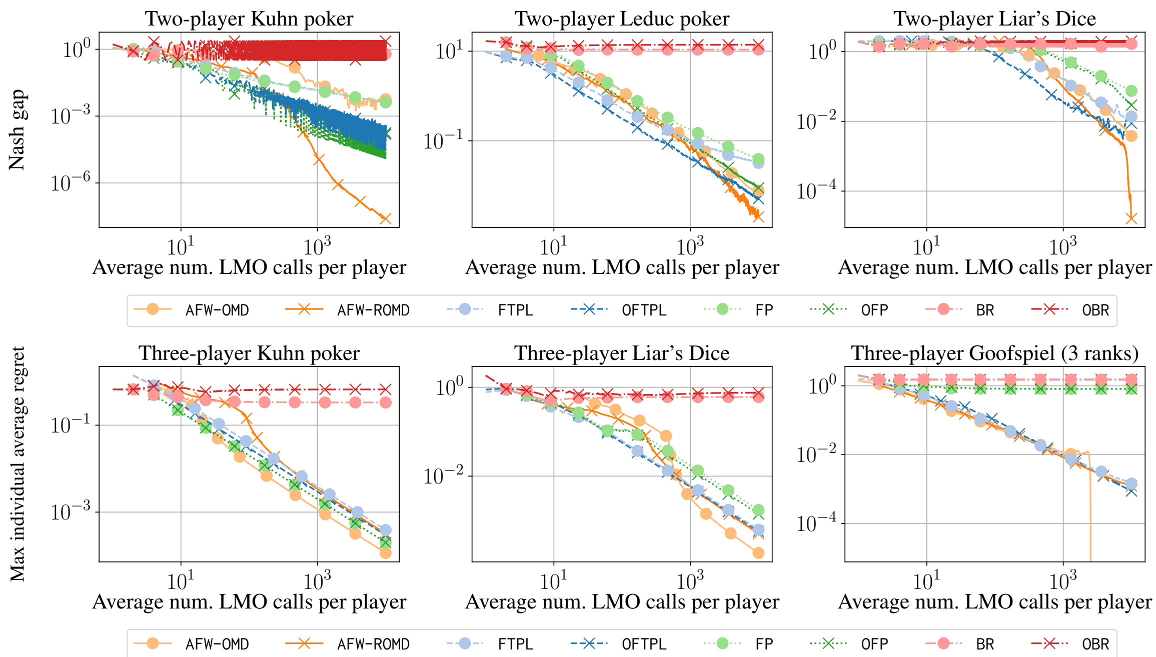

Figure 1: Convergence to equilibrium (Nash eq. and CCE) as a function of average LMO calls per player for AFW-OMD, AFW-ROMD, FTPL, OFTPL, FP, OFP, BR, and OBR.

other algorithms are not known to have such guarantees. Moreover, in the case of averaging, we examine the effects of applying *adaptive restarting* in Appendix H. Adaptive restarts are known to lead to linear convergence for polyhedral BSPPs for some algorithms, as they satisfy a *sharpness* property [5, 23, 29, 56].

For our algorithms and (O)FTPL, we restrict the number of LMO calls per iteration to be in $\{1, 2, 3, 4, 5, 10, 20, 100, 200\}$. We find that using the number of LMO calls as a termination criterion generally works best for our algorithms as well. Furthermore, we use warmstarting for our algorithm, which involves initializing the active set of AFW in the current iteration of our algorithms with the active set of AFW in the previous iteration. We provide complete pseudocode for adaptive restarting and various iterate averaging schemes in Appendix G. We conduct ablation on the averaging scheme, termination criteria, and restarting in Appendix H. For each of the six algorithms, we use the choice of step size, number of LMO calls, and averaging, which generally leads to the best performance for each game. We provide additional graphs in Appendix H demonstrating that the performance of our algorithms relative to the others generally holds irrespective of the choice of the averaging. All of our experiments are run until the average number of LMO calls for each player is 10^4.

We show the results of running our algorithms on two-player Kuhn poker, two-player Leduc poker, two-player Liar's Dice, three-player Kuhn poker, three-player Liar's Dice, and three-player Goofspiel (3 ranks) in Figure 1, seeking to compute Nash equilibria in the former three games and CCE in the latter three games. In the case of NE computation, AFW-ROMD outperforms existing algorithms in all three games (while (O)FTPL achieves low Nash gaps, it erratically jumps to high gaps in following iterations). In the case of CCE computation, we measure the maximum individual player's average regret since a bound of ϵ' on each player's average regret corresponds to an ϵ'-CCE. Again, in all of the games, our algorithms are competitive with existing algorithms.

References

[1] Jacob Abernethy, Chansoo Lee, and Ambuj Tewari. Perturbation techniques in online learning and optimization. *Perturbations, Optimization, and Statistics*, 233, 2016.

[2] Ioannis Anagnostides, Gabriele Farina, Christian Kroer, Chung-Wei Lee, Haipeng Luo, and Tuomas Sandholm. Uncoupled learning dynamics with $O(\log T)$ swap regret in multiplayer games. In *Neural Information Processing Systems (NeurIPS)*, 2022.

[3] Ioannis Anagnostides, Ioannis Panageas, Gabriele Farina, and Tuomas Sandholm. On last-iterate convergence beyond zero-sum games. In *International Conference on Machine Learning (ICML)*, 2022.

[4] Ioannis Anagnostides, Gabriele Farina, and Tuomas Sandholm. Near-optimal Φ-regret learning in extensive-form games. In *International Conference on Machine Learning (ICML)*, 2023.

[5] David Applegate, Oliver Hinder, Haihao Lu, and Miles Lubin. Faster first-order primal-dual methods for linear programming using restarts and sharpness. *Mathematical Programming*, 201(1-2):133–184, 2023.

[6] Mohammad Ali Bashiri and Xinhua Zhang. Decomposition-invariant conditional gradient for general polytopes with line search. In *Neural Information Processing Systems (NIPS)*, 2017.

[7] Amir Beck and Shimrit Shtern. Linearly convergent away-step conditional gradient for non-strongly convex functions. *Mathematical Programming*, 164:1–27, 2017.

[8] Immanuel M. Bomze, Francesco Rinaldi, and Damiano Zeffiro. Frank–Wolfe and friends: a journey into projection-free first-order optimization methods. *4OR*, 19:313–345, 2021.

[9] Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit hold'em poker is solved. *Science*, 347(6218):145–149, 2015.

[10] Gábor Braun, Alejandro Carderera, Cyrille W. Combettes, Hamed Hassani, Amin Karbasi, Aryan Mokhtari, and Sebastian Pokutta. Conditional gradient methods. *arXiv preprint arXiv:2211.14103*, 2022.

[11] George W. Brown. Iterative solution of games by fictitious play. *Activity Analysis of Production and Allocation*, 13(1):374, 1951.

[12] Noam Brown and Tuomas Sandholm. Superhuman AI for heads-up no-limit poker: Libratus beats top professionals. *Science*, 359(6374):418–424, 2018.

[13] Noam Brown and Tuomas Sandholm. Solving imperfect-information games via discounted regret minimization. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

[14] Noam Brown and Tuomas Sandholm. Superhuman AI for multiplayer poker. *Science*, 365 (6456):885–890, 2019.

[15] Darshan Chakrabarti, Jelena Diakonikolas, and Christian Kroer. Block-coordinate methods and restarting for solving extensive-form games. *arXiv preprint arXiv:2307.16754*, 2023.

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

[17] Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. Near-optimal no-regret learning in general games. In *Neural Information Processing Systems (NeurIPS)*, 2021.

[18] Gabriele Farina, Andrea Celli, Nicola Gatti, and Tuomas Sandholm. Ex ante coordination and collusion in zero-sum multi-player extensive-form games. In *Neural Information Processing Systems (NeurIPS)*, 2018.

[19] Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Better regularization for sequential decision spaces: Fast convergence rates for Nash, correlated, and team equilibria. In *ACM Conference on Economics and Computation (EC)*, 2021.

[20] Gabriele Farina, Christian Kroer, and Tuomas Sandholm. Faster game solving via predictive Blackwell approachability: Connecting regret matching and mirror descent. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[21] Gabriele Farina, Ioannis Anagnostides, Haipeng Luo, Chung-Wei Lee, Christian Kroer, and Tuomas Sandholm. Near-optimal no-regret learning dynamics for general convex games. In *Neural Information Processing Systems (NeurIPS)*, 2022.

[22] Gabriele Farina, Chung-Wei Lee, Haipeng Luo, and Christian Kroer. Kernelized multiplicative weights for 0/1-polyhedral games: Bridging the gap between learning in extensive-form and normal-form games. In *International Conference on Machine Learning (ICML)*, 2022.

[23] Olivier Fercoq. Quadratic error bound of the smoothed gap and the restarted averaged primal-dual hybrid gradient. *arXiv preprint arXiv:2206.03041*, 2023.

[24] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1-2):95–110, 1956.

[25] Yuan Gao, Christian Kroer, and Donald Goldfarb. Increasing iterate averaging for solving saddle-point problems. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

[26] Dan Garber and Elad Hazan. Faster rates for the Frank-Wolfe method over strongly-convex sets. In *International Conference on Machine Learning (ICML)*, 2015.

[27] Dan Garber and Ofer Meshi. Linear-memory and decomposition-invariant linearly convergent conditional gradient algorithm for structured polytopes. In *Neural Information Processing Systems (NIPS)*, 2016.

[28] Gauthier Gidel, Tony Jebara, and Simon Lacoste-Julien. Frank-Wolfe algorithms for saddle point problems. In *Artificial Intelligence and Statistics (AISTATS)*, 2017.

[29] Andrew Gilpin, Javier Peña, and Tuomas Sandholm. First-order algorithm with $\mathcal{O}(\ln(1/\epsilon))$ convergence for ϵ-equilibrium in two-person zero-sum games. *Mathematical Programming*, 133(1):279–298, 2012.

[30] Jacques Guélat and Patrice Marcotte. Some comments on Wolfe's 'away step'. *Mathematical Programming*, 35(1):110–119, 1986.

[31] Elad Hazan and Satyen Kale. Projection-free online learning. In *International Coference on Machine Learning (ICML)*, 2012.

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

[33] Samid Hoda, Andrew Gilpin, Javier Pena, and Tuomas Sandholm. Smoothing techniques for computing Nash equilibria of sequential games. *Mathematics of Operations Research*, 35(2): 494–512, 2010.

[34] Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning (ICML)*, 2013.

[35] Pooria Joulani, András György, and Csaba Szepesvári. A modular analysis of adaptive (non-)convex optimization: Optimism, composite objectives, and variational bounds. In *International Conference on Algorithmic Learning Theory (ALT)*, 2017.

[36] Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.

[37] Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, 2016.

[38] Daphne Koller, Nimrod Megiddo, and Bernhard von Stengel. Efficient computation of equilibria for extensive two-person games. *Games and Economic Behavior*, 14(2):247–259, 1996.

[39] Christian Kroer, Kevin Waugh, Fatma Kılınç-Karzan, and Tuomas Sandholm. Faster algorithms for extensive-form game solving via improved smoothing functions. *Mathematical Programming*, pages 1–33, 2020.

[40] Harold W. Kuhn. A simplified two-person poker. *Contributions to the Theory of Games*, 1: 97–103, 1950.

[41] Simon Lacoste-Julien and Martin Jaggi. On the global linear convergence of Frank-Wolfe optimization variants. In *Neural Information Processing Systems (NIPS)*, 2015.

[42] Guanghui Lan. Gradient sliding for composite optimization. *Mathematical Programming*, 159: 201–235, 2016.

[43] Guanghui Lan and Yi Zhou. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization*, 26(2):1379–1409, 2016.

[44] Marc Lanctot, Kevin Waugh, Martin Zinkevich, and Michael Bowling. Monte Carlo sampling for regret minimization in extensive games. In *Neural Information Processing Systems (NIPS)*, 2009.

[45] Chung-Wei Lee, Christian Kroer, and Haipeng Luo. Last-iterate convergence in extensive-form games. In *Neural Information Processing Systems (NeurIPS)*, 2021.

[46] Viliam Lisỳ, Marc Lanctot, and Michael Bowling. Online Monte Carlo counterfactual regret minimization for search in imperfect information games. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2015.

[47] Yu Malitsky. Projected reflected gradient methods for monotone variational inequalities. *SIAM Journal on Optimization*, 25(1):502–520, 2015.

[48] Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

[49] Javier Pena and Daniel Rodriguez. Polytope conditioning and linear convergence of the Frank–Wolfe algorithm. *Mathematics of Operations Research*, 44(1):1–18, 2019.

[50] I.V. Romanovskii. Reduction of a game with full memory to a matrix game. *Doklady Akademii Nauk SSSR*, 144(1):62–+, 1962.

[51] Sheldon M. Ross. Goofspiel — the game of pure strategy. *Journal of Applied Probability*, 8(3): 621–625, 1971.

[52] Finnegan Southey, Michael P. Bowling, Bryce Larson, Carmelo Piccione, Neil Burch, Darse Billings, and Chris Rayner. Bayes' bluff: Opponent modelling in poker. *arXiv preprint arXiv:1207.1411*, 2012.

[53] Arun Suggala and Praneeth Netrapalli. Follow the perturbed leader: Optimism and fast parallel algorithms for smooth minimax games. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[54] Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E Schapire. Fast convergence of regularized learning in games. In *Neural Information Processing Systems (NIPS)*, 2015.

[55] Oskari Tammelin, Neil Burch, Michael Johanson, and Michael Bowling. Solving heads-up limit Texas hold'em. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[56] Paul Tseng. On linear convergence of iterative methods for the variational inequality problem. *Journal of Computational and Applied Mathematics*, 60(1-2):237–252, 1995.

[57] Bernhard von Stengel. Efficient computation of behavior strategies. *Games and Economic Behavior*, 14(2):220–246, 1996.

[58] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. Linear last-iterate convergence in constrained saddle-point optimization. In *International Conference on Learning Representations (ICLR)*, 2021.

[59] Andre Wibisono, Molei Tao, and Georgios Piliouras. Alternating mirror descent for constrained min-max games. In *Neural Information Processing Systems (NeurIPS)*, 2022.

[60] Philip Wolfe. Convergence theory in nonlinear programming. *Integer and Nonlinear Programming*, pages 1–36, 1970.

[61] Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. In *Neural Information Processing Systems (NIPS)*, 2007.

Appendix A. Away-Step Frank-Wolfe Pseudocode

Algorithm 1: Away-Step Frank-Wolfe (AFW) [30]

Data: $\mathcal{P} \subseteq \mathbb{R}^n$: convex polytope
$\mathcal{A} \subseteq \mathbb{R}^n$: set of atoms, such that $\mathrm{Conv}(\mathcal{A}) = \mathcal{P}$
$\mathrm{LMO}_{\mathcal{A}}$: linear minimization oracle over $\mathcal{A}$
$f : \mathcal{P} \to \mathbb{R}$: L-smooth, convex function to be optimized
$\boldsymbol{x}^{(0)} \in \mathcal{P}$
ϵ: desired accuracy T: maximum number of LMO calls

$\mathcal{S}^{(0)} = \{\boldsymbol{x}^{(0)}\}$
$\alpha^{(0)}_{\boldsymbol{x}^{(0)}} = 1$
for $t = 0 \dots T-1$ **do**
  $\boldsymbol{s}^{(t)} = \mathrm{LMO}_{\mathcal{A}}(\nabla f(\boldsymbol{x}^{(t)}))$, $\boldsymbol{d}^{(t)}_{\mathrm{FW}} = \boldsymbol{s}^{(t)} - \boldsymbol{x}^{(t)}$
  $\boldsymbol{v}^{(t)}_{\mathrm{A}} \in \mathrm{argmax}_{\boldsymbol{v} \in \mathcal{S}^{(t)}} \langle \nabla f(\boldsymbol{x}^{(t)}),\, \boldsymbol{v} \rangle$, $\boldsymbol{d}^{(t)}_{\mathrm{A}} = \boldsymbol{x}^{(t)} - \boldsymbol{v}^{(t)}_{\mathrm{A}}$
  if $\boldsymbol{g}^{(t)}_{\mathrm{FW}} = \langle -\nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{d}^{(t)}_{\mathrm{FW}} \rangle \leq \epsilon$ **then**
    return $\boldsymbol{x}^{(t)}$
  if $\langle -\nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{d}^{(t)}_{\mathrm{FW}} \rangle \geq \langle -\nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{d}^{(t)}_{\mathrm{A}} \rangle$ **then**
    $\boldsymbol{d}^{(t)} = \boldsymbol{d}^{(t)}_{\mathrm{FW}}$, $\gamma^{(t)}_{\max} = 1$
  else
    $\boldsymbol{d}^{(t)} = \boldsymbol{d}^{(t)}_{\mathrm{A}}$, $\gamma^{(t)}_{\max} = \frac{\alpha_{\boldsymbol{v}^{(t)}_{\mathrm{A}}}}{1-\alpha_{\boldsymbol{v}^{(t)}_{\mathrm{A}}}}$
  $\gamma^{(t)} = \min\left(\frac{\langle -\nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{d}^{(t)} \rangle}{L \left\| \boldsymbol{d}^{(t)} \right\|^2}, \gamma^{(t)}_{\max} \right)$
  $\boldsymbol{x}^{(t+1)} = \boldsymbol{x}^{(t)} + \gamma^{(t)} \boldsymbol{d}^{(t)}$
  if $\langle -\nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{d}^{(t)}_{\mathrm{FW}} \rangle \geq \langle -\nabla f(\boldsymbol{x}^{(t)}), \boldsymbol{d}^{(t)}_{\mathrm{A}} \rangle$ **then**
    $\alpha^{(t+1)}_{\boldsymbol{v}} = (1-\gamma^{(t)})\alpha^{(t)}_{\boldsymbol{v}}$ for all $\boldsymbol{v} \in \mathcal{S}^{(t)} \setminus \{\boldsymbol{s}^{(t)}\}$
    $\alpha^{(t+1)}_{\boldsymbol{s}^{(t)}} = \begin{cases} \gamma^{(t)} + (1-\gamma^{(t)})\boldsymbol{s}^{(t)} & \text{if} \boldsymbol{s}^{(t)} \in \mathcal{S}^{(t)} \\ \gamma^{(t)} & \text{otherwise} \end{cases}$
    $\mathcal{S}^{(t+1)} = \begin{cases} \mathcal{S}^{(t)} \cup \{\boldsymbol{s}^{(t)}\} & \text{if} \gamma^{(t)} < 1 \\ \{\boldsymbol{s}^{(t)}\} & \text{if} \gamma^{(t)} = 1 \end{cases}$
  else
    $\alpha^{(t+1)}_{\boldsymbol{v}} = (1+\gamma)\alpha^{(t)}_{\boldsymbol{v}}$ for all $\boldsymbol{v} \in \mathcal{S}^{(t)} \setminus \{\boldsymbol{v}^{(t)}_{\mathrm{A}}\}$
    $\alpha^{(t+1)}_{\boldsymbol{s}^{(t)}} = \begin{cases} \gamma^{(t)} + (1-\gamma^{(t)})\boldsymbol{s}^{(t)} & \text{if} \boldsymbol{v}^{(t)}_{\mathrm{A}} \in \mathcal{S}^{(t)} \\ \gamma^{(t)} & \text{otherwise} \end{cases}$
    $\mathcal{S}^{(t+1)} = \begin{cases} \mathcal{S}^{(t)} & \text{if} \gamma^{(t)} < \gamma^{(t)}_{\max} \\ \mathcal{S}^{(t)} \setminus \{\boldsymbol{v}^{(t)}_{\mathrm{A}}\} & \text{if} \gamma^{(t)} = \gamma^{(t)}_{\max} \end{cases}$

In Algorithm 1, we present pseudocode for away-step Frank-Wolfe, based on the presentation by Guélat and Marcotte [30]. We assume that the polytope is expressed as the convex hull of a set of *atoms* and that the LMO for the polytope always returns an atom (since there always exists an atom that minimizes a given linear objective over the polytope). This is equivalent to assuming that we have an LMO over the set of atoms themselves.

A.1. Related Work

Frank-Wolfe algorithm. Frank and Wolfe [24] presented the original Frank-Wolfe (FW) algorithm (also known as conditional gradient descent), a projection-free first-order method for solving smooth constrained convex minimization problems. Vanilla FW provides a $O(\frac{1}{t})$ rate of convergence for smooth objectives, but has stronger guarantees for special classes of functions; in particular, for strongly convex functions, Garber and Hazan [26] showed a $O(\frac{1}{t^2})$ convergence rate. Many FW variants have been developed since Frank and Wolfe's original presentation of the algorithm, including the away-step variant [60]. Importantly for us, Lacoste-Julien and Jaggi [41] showed that AFW and several other variants achieve linear convergence for strongly convex and smooth objectives over polyhedral domains; see also Beck and Shtern [7], Pena and Rodriguez [49]. There are several excellent overviews of Frank-Wolfe algorithms, e.g., Bomze et al. [8], Braun et al. [10], Jaggi [34].

FW **for saddle-point problems.** There has been some work on extending FW to saddle-point problems, prominently by Gidel et al. [28]. However, they only provide fast convergence guarantees for strongly convex-concave objectives (and moreover, require a significant degree of strong convexity, thus rendering their results incompatible with smoothing techniques). They provide a convergence guarantee for the bilinear case when the feasible sets are polytopes, but it is extremely slow. Lan [42] introduced the idea of "gradient sliding," which involves computing approximate prox steps to save on the number of gradient computations required to optimize a composite function with a smooth and nonsmooth component. In the spirit of this work, Lan and Zhou [43] introduced "conditional gradient sliding," which involved using conditional gradient methods to compute these approximate prox steps. They combine this idea with Nesterov acceleration to achieve $O(1/\sqrt{\epsilon})$ gradient computations and $O(1/\epsilon)$ LMO calls for smooth functions. They present a smoothed version of their algorithm as well that can be applied to saddle-point problems, but the smoothing degrades the guarantee to $O(1/\epsilon)$ gradient computations and $O(1/\epsilon^2)$ LMO calls.

Projection-free online learning. The first projection-free online learning algorithm was FTPL, introduced by Kalai and Vempala [36]. The algorithm involves randomly perturbing the sum of the observed losses (which serves as a form of regularization, see, e.g., Abernethy et al. [1]) before computing the best response and achieves $O(1/\sqrt{T})$ average regret for linear loss functions. Suggala and Netrapalli [53] introduced OFTPL, which achieves $O(1/T)$ average regret for players in zero-sum games but requires doing $O(T)$ LMO calls at every iteration. Hazan and Kale [31] presented OFW which uses FW to achieve $O(1/T^{1/4})$ average regret for Lipschitz convex losses.

Algorithms for game solving. Much work has been done to develop efficient algorithms for game-solving. We only touch on the major trends related to discrete-time methods and sequential games. A line of research has focused on constructing no-regret algorithms with ergodic convergence to equilibrium. Out of these, we highlight two categories: methods based on the CFR regret-decomposition framework [13, 20, 44, 55, 61], and methods based on the OMD framework and more generally first-order methods [19, 22, 33, 39]. Some of these methods were key in solving large games, such as poker [9, 12, 14]. A recent trend has focused on establishing learning algorithms with (poly)logarithmic per-player regret when used in self-play, including Anagnostides et al. [2, 3, 4], Daskalakis et al. [16, 17], Farina et al. [21], Wibisono et al. [59]. These methods can compute equilibria in multiplayer games at the rate of $\widetilde{O}(1/T)$. Finally, another recent trend in the literature has focused on establishing learning algorithms [45, 58] and first-order methods [29, 56] with guarantees for last-iterate convergence.

Appendix B. Notation and Preliminaries

We will use $\|\cdot\|_p$ to denote the ℓ_p-norm and $\|\cdot\|$ without subscript to denote $\|\cdot\|_2$. Any norm-dependent quantity (e.g., diameter, facial distance, strong convexity, and smoothness) will be with respect to the Euclidean norm (which is self-dual) unless otherwise noted. Because we are principally concerned with using these algorithms for equilibrium computation in games, we will use subscripts to indicate a set or constant corresponding to a particular agent. We will use $[n]$ to denote the set $\{1, \dots, n\}$, and L-smoothness refers to Lipschitz continuity of the gradient, with modulus L.

B.1. Online Linear Optimization

In online learning, an agent i repeatedly interacts with an environment, aiming to minimize its *regret*. At each time t, the agent chooses a strategy $\boldsymbol{x}_i^{(t)}$ from a given feasible set $\mathcal{X}_i \subseteq \mathbb{R}^{n_i}$ and then receives a loss vector $\boldsymbol{\ell}_i^{(t)} \in \mathcal{X}_i \to \mathbb{R}$. The loss is allowed to depend adversarially on $\boldsymbol{x}_i^{(t)}$. The agent then pays a cost of $\langle \boldsymbol{\ell}_i^{(t)}, \boldsymbol{x}_i^{(t)} \rangle$. The (cumulative) regret $\mathrm{Reg}_i^{(T)}$ after T iterations is defined as $\max_{\boldsymbol{x}' \in \mathcal{X}_i} \sum_{t=1}^{T} \langle \boldsymbol{\ell}_i^{(t)}, \boldsymbol{x}_i^{(t)} \rangle - \langle \boldsymbol{\ell}_i^{(t)}, \boldsymbol{x}' \rangle$, and average regret is defined as regret divided by the number of iterations. We will assume that losses are bounded and normalized: $\|\boldsymbol{\ell}_i^{(t)}\| \leq 1$ for all $t \in [T]$.

To achieve desired regret guarantees, online learning algorithms typically require some form of regularization. While FTPL achieves this regularization through randomization, the framework of algorithms utilizing approximate prox calls that we present will require access to a regularizer $\varphi_i : \mathcal{X}_i \to \mathbb{R}$, which is 1-strongly convex and L_i smooth on $\mathcal{X}_i$. The Bregman divergence between $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ is denoted by $\mathcal{D}_{\varphi_i}(\boldsymbol{x} \,\|\, \boldsymbol{y})$. Furthermore, we define $\Omega_i := \sup_{\boldsymbol{x},\boldsymbol{y} \in \mathcal{X}_i} \mathcal{D}_{\varphi_i}(\boldsymbol{x} \,\|\, \boldsymbol{y})$ and $D_i := \sup_{\boldsymbol{x},\boldsymbol{y} \in \mathcal{X}_i} \|\boldsymbol{x} - \boldsymbol{y}\|$. $\delta(\mathcal{X}_i)$ will be used for the facial distance of $\mathcal{X}_i$. For a given set $\mathcal{X}$, and a point $\boldsymbol{x} \in \mathcal{X}$, we denote $\mathrm{dist}(\boldsymbol{x}, \mathcal{X}) := \inf_{\boldsymbol{x}' \in \mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}'\|$ and in the case that $\mathcal{X}$ is compact, define $\Pi_{\mathcal{X}}(\boldsymbol{x}) = \mathrm{argmin}_{\boldsymbol{x}' \in \mathcal{X}} \|\boldsymbol{x} - \boldsymbol{x}'\|$.

Online Mirror Descent (OMD) is an algorithm which performs a single proximal computation at every iteration of the algorithm, generating iterates as follows:

$$\boldsymbol{x}_i^{(t+1)} = \operatorname*{argmin}_{\boldsymbol{x}_i \in \mathcal{X}_i} \left\{ \langle \boldsymbol{\ell}_i^{(t)}, \boldsymbol{x}_i \rangle + \frac{1}{\eta} \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}_i \,\|\, \boldsymbol{x}_i^{(t)} \right) \right\}.$$

It enjoys $O(1/\sqrt{T})$ average regret (e.g., Hazan et al. [32], Orabona [48]).

Reflected Online Mirror Descent (ROMD) is an optimistic version of OMD which utilizes a prediction $\boldsymbol{m}_i^{(t+1)}$ of the next loss $\boldsymbol{\ell}_i^{(t+1)}$ to generate the iterate at time $t+1$.

$$\boldsymbol{x}_i^{(t+1)} = \operatorname*{argmin}_{\boldsymbol{x}_i \in \mathcal{X}_i} \left\{ \langle \boldsymbol{\ell}_i^{(t)} + \boldsymbol{m}_i^{(t+1)} - \boldsymbol{m}_i^{(t)}, \boldsymbol{x}_i \rangle + \frac{1}{\eta} \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}_i \,\|\, \boldsymbol{x}_i^{(t)} \right) \right\}.$$

It is common to use the last observed loss as the prediction for the next loss: set $\boldsymbol{m}_i^{(t+1)}$ equal to $\boldsymbol{\ell}_i^{(t)}$. In this case, ROMD achieves $O(1/T)$ average regret [35, 47] in self-play. Since $\boldsymbol{m}_i^{(t+1)}$ is the prediction of a loss $\boldsymbol{\ell}_i^{(t+1)}$ which is assumed to have norm bounded by 1, we will assume that $\|\boldsymbol{m}_i^{(t)}\| \leq 1$ for all $t \in [T]$.

Syrgkanis et al. [54] introduce the notion of Regret bounded by Variation in Utilities (RVU), recalled next, and demonstrate that algorithms with this property exhibit faster convergence to equilibria in games.

Definition 2 (RVU [54]) *A learning algorithm for Player i is said to satisfy the RVU property if for some $\alpha, \beta, \gamma > 0$ and all possible $\boldsymbol{\ell}_i^{(1)}, \dots, \boldsymbol{\ell}_i^{(T)}$,*

$$\mathrm{Reg}_i^{(T)} \leq \alpha + \beta \sum_{t=1}^{T} \|\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}\|^2 - \gamma \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t-1)}\|^2.$$

ROMD satisfies this inequality with $\alpha = \Omega/\eta$, $\beta = \eta$, $\gamma = 1/4\eta$; we are not aware of a reference for this, but it can be shown very similarly to known results for optimistic OMD. Later, we will show in Lemma 4 that our approximate ROMD framework still satisfies the RVU property.

B.2. Game-Theoretic Notions

Normal-form games (NFGs) model single-shot simultaneous interactions among a set of agents denoted by $[N]$. The agents each have a set of possible actions $\mathcal{A}_i$ and a normalized utility function $u_i : \prod_{i\in[N]} \mathcal{A}_i \to [-1, 1]$, the latter specifying their payoff for a given choice of actions by each of the agents. The game is said to be *zero-sum* if $\sum_{i\in[n]} u_i(\boldsymbol{a}) = 0$ for all $\boldsymbol{a} \in \prod_{i\in[N]} \mathcal{A}_i$. A mixed-strategy $\boldsymbol{x}_i$ for Player i, is a probability distribution over $\mathcal{A}_i$; $\boldsymbol{x}_i \in \Delta(\mathcal{A}_i)$. We can extend the domain of u_i to be over $\Delta(\mathcal{A}_i)$ by taking the expectation of the utility function over the distribution over $\mathcal{A}_i$ induced by $\boldsymbol{x}_i \in \Delta(\mathcal{A}_i)$.

Nash equilibrium is the de facto notion of equilibrium in NFGs, and the problem of computing a Nash equilibrium (NE) in two-player zero-sum games can be formulated as a bilinear saddle-point problem (BSPP):

$$\min_{\boldsymbol{x}\in\mathcal{X}} \max_{\boldsymbol{y}\in\mathcal{Y}} \langle \mathbf{A}\boldsymbol{x}, \boldsymbol{y} \rangle. \tag{BSPP}$$

In this case, $\mathcal{X}$ and $\mathcal{Y}$ are the space of mixed strategies for Player 1 and Player 2, respectively, and $\mathbf{A}$ encodes the utility of Player 2 for a given choice of strategies for both players. The duality gap ξ of $\bar{\boldsymbol{x}}$ and $\bar{\boldsymbol{y}}$ for (BSPP) can be defined as $\max_{\boldsymbol{y}\in\mathcal{Y}} \langle \mathbf{A}\bar{\boldsymbol{x}}, \boldsymbol{y} \rangle - \min_{\boldsymbol{x}\in\mathcal{X}} \langle \mathbf{A}\boldsymbol{x}, \bar{\boldsymbol{y}} \rangle$. This quantity is typically used to measure the quality of a solution; in the case of two-player zero-sum games, a duality gap of ϵ' corresponds to an ϵ'-NE (and thus is also known as Nash gap).

Definition 3 (ϵ'-coarse correlated equilibrium) *An ϵ'-coarse correlated equilibrium (ϵ'-CCE) is defined as $\boldsymbol{x} \in \Delta(\prod_{i\in[N]} \mathcal{A}_i)$ such that*

$$\mathbf{E}_{\boldsymbol{a}\sim\boldsymbol{x}}[u_i(\boldsymbol{a})] \geq \mathbf{E}_{\boldsymbol{a}_{-i}\sim\boldsymbol{x}}[u_i(\boldsymbol{a}'_i, \boldsymbol{a}_{-i})] - \epsilon'$$

for all players $i \in [N]$, for all $\boldsymbol{a}'_i \in \mathcal{A}_i$, for $\epsilon' \geq 0$; $\epsilon' = 0$ corresponds to an exact CCE.

Extensive-form games (EFGs) are a generalization of normal-form games, which also allow for modeling of sequential moves (and also private and/or imperfect information and stochasticity).

Extensive-form games can be represented using a game tree. Each of the internal nodes of the tree corresponds to points at which one of the players or nature (corresponding to stochastic outcomes independent of the players' choices) takes an action. The leaves of the game tree correspond to termination of the game and are associated with utilities for each player. Information sets correspond to partitions of the nodes, such that all the nodes in an information set correspond to a single player, and the actions available at each node of the information set are the same; players cannot distinguish between different nodes in an information set, so the actions must be the same. The set of information sets for a Player i is denoted $\mathcal{I}_i$.

We are concerned with the perfect recall setting, in which, when Player i is asked to make a decision at a given information set $I \in \mathcal{I}_i$, he or she remembers the entire history of information sets visited and actions taken at those information sets; each information set has a sequence of previously visited information sets and previously taken actions associated with it.

A strategy for a player is defined as a distribution over the choices of actions at each information set in the tree belonging to the player. Similar to NFGs, a utility function can be defined over the space of strategies for the players by taking an expectation with respect to the joint distribution over the leaves induced by the players' individual strategies.

The sequence-form representation allows for a compact representation of the strategies of a player in an EFG, and allows for formulation of the utility function as a linear function of the players' strategies [38, 50, 57]. A sequence for a player is defined as a choice of action and information set for that player. For a given strategy, the value associated with a given sequence in the corresponding sequence-form strategy is the probability that the player reaches that information set and plays that action given that nature, and the other plays play in such a way that allows the player to reach that sequence. The set of all sequences for a player is denoted Σ_i.

The space of all sequence-form strategies is a convex polytope: $Q_i = \{\boldsymbol{y} \in \mathbb{R}^{|\Sigma_i|} : \mathbf{F}_i \boldsymbol{y} = \boldsymbol{f}_i, \boldsymbol{y} \geq \mathbf{0}\}$, where $\mathbf{F}_i$ is a sparse $|\mathcal{I}_i| \times |\Sigma_i|$ matrix with entries in $\{0, 1, -1\}$, and $\boldsymbol{f}_i$ is a vector with entries in $\{0, 1\}$. Each row of the matrix $\mathbf{F}_i$ corresponds to a probability flow constraint which ensures that the sum of the probability of all the sequences associated with an information set sum up to the probability associated with the parent sequence. Note that the vertices of this polytope correspond to making deterministic choices at each information set, which means that the vertices have binary coordinates, and thus, the minimum non-zero entry in a vertex is 1.

Equilibrium computation for two-player zero-sum EFGs can also be formulated as (BSPP) by letting $\mathcal{X}$ and $\mathcal{Y}$ be sequence-form polytopes.

When we are analyzing multiplayer games involving N agents, we will let $\mathcal{X}_i \subset \mathbb{R}^{n_i}$ denote the convex and compact set of strategies for the i^{th} player, where $i \in [N]$, and let $\boldsymbol{x}_i \in \mathcal{X}_i$ represent their strategy. For NFGs, $\mathcal{X}_i$ is $\Delta(\mathcal{A}_i)$, the set of mixed strategies for i, while for EFGs, $\mathcal{X}_i$ is the sequence-form polytope for Player i. In the case of two-player zero-sum NFGs or EFGs, in which case Nash equilibrium computation corresponds to (BSPP), we will let $\mathcal{X} = \mathcal{X}_1$, $\mathcal{Y} = \mathcal{X}_2$, and $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. In this case, we will use $\mathcal{Z}^*$ to denote the set of solutions to the BSPP. We denote $\operatorname{dist}(\boldsymbol{z}, \mathcal{Z}) := \inf_{\boldsymbol{z}' \in \mathcal{Z}} \|\boldsymbol{z} - \boldsymbol{z}'\|$. We define the vector field $\boldsymbol{F}(\boldsymbol{z}) = (\mathbf{A}^T \boldsymbol{y}, -\mathbf{A}\boldsymbol{x})$ for $\boldsymbol{z} \in \mathcal{Z}$. Without loss of generality, we assume that $\boldsymbol{F}$ is smooth with constant 1 (the payoff matrix $\mathbf{A}$ can be scaled to ensure this is the case).

B.3. Saddle-Point Metric-Subregularity

Problems of the form (BSPP) satisfy a condition known as *Saddle-Point Metric Subregularity* [58] as long as $\mathcal{X}$ and $\mathcal{Y}$ are convex polytopes (this is the case for NFGs and EFGs).

Definition 4 (Saddle-Point Metric Subregularity) *The SP-MS condition is satisfied if for any $\boldsymbol{z} \in \mathcal{Z} \setminus \mathcal{Z}^*$ with $\boldsymbol{z}^* = \Pi_{\mathcal{Z}}(\boldsymbol{z})$ for some $\beta \geq 0$ and $\nu > 0$,*

$$\sup_{\boldsymbol{z}' \in \mathcal{Z}} \frac{\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{z} - \boldsymbol{z}' \rangle}{\|\boldsymbol{z} - \boldsymbol{z}'\|} \geq \nu \|\boldsymbol{z} - \boldsymbol{z}^*\|^{\beta+1}. \tag{SP-MS}$$

For any given NFG or EFG, there exists $\nu \geq 0$ so that Inequality (SP-MS) holds with $\beta = 0$; given a choice of a game, we will use ν to refer to this problem-dependent constant. Wei et al. [58] use this condition to demonstrate linear last-iterate convergence of certain online learning algorithms. Earlier works [29, 56] showed linear last-iterate convergence using error bounds, and Wei et al. [58] note that there is a close correspondence between the SP-MS condition and error bound techniques for bilinear polyhedral settings.

Appendix C. Facial Distance of Polytopes Proofs

Theorem 2 *Let $\mathcal{P}$ be a polytope given by $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \geq \mathbf{0}$ where $\boldsymbol{x} \in \mathbb{R}^n$. Let γ be the minimum value of a nonzero coordinate of a vertex. Then $\delta(\mathcal{P}) \geq \frac{\gamma}{\sqrt{n}}$. Moreover, if the optimal solution lies in a face $\mathcal{F}$ such that k coordinates are zero, then $\delta(\mathcal{P}) \geq \frac{\gamma}{\sqrt{k}}$.*

Proof Consider a face $\mathcal{F}$ of the polytope $\mathcal{P}$. The face is generated by making a subset of the inequalities that define $\mathcal{P}$ tight, that is, setting $x_i = 0$ for a subset $S \subseteq [n]$ of indices. Now consider the complement polytope

$$\mathcal{P}' := \mathrm{Conv}(\mathrm{Vert}(\mathcal{P}) \setminus \mathcal{F}).$$

Necessarily, the sum of the coordinates corresponding to the indices S of any vertex $\boldsymbol{x}' \in \mathrm{Vert}(\mathcal{P}) \setminus \mathcal{F}$, that is, $\sum_{i \in S} \boldsymbol{x}'_i$, must be at least γ since at least one of the k coordinates of a vertex in the complement polytope must be nonzero (otherwise, it would be in $\mathcal{F}$). Hence, any convex combination of points in $\mathrm{Vert}(\mathcal{P}) \setminus \mathcal{F}$ must also put total mass at least γ on coordinates S, implying that $\sum_{i \in S} \boldsymbol{x}'_i \geq \gamma$ for any $\boldsymbol{x}' \in \mathcal{P}'$.

On the other hand, by construction, any point on the chosen face $\mathcal{F}$ satisfies $x_i = 0$ for all $i \in S$. Lower bounding distances by focusing only on the coordinates in S means that the distance between any point on the chosen face $\mathcal{F}$ and any point in the complement polytope $\mathcal{P}'$ is at least

$$\min_{\substack{\boldsymbol{x}' \in \mathbb{R}^n \\ \sum_{i \in S} \boldsymbol{x}'_i \geq \gamma}} \sqrt{\sum_{i \in S} (\boldsymbol{x}'_i - 0)^2} = \sqrt{|S| \cdot \left(\frac{\gamma}{|S|}\right)^2} = \frac{\gamma}{\sqrt{|S|}},$$

where the minimum of the objective was obtained by setting all $|S| = k$ coordinates to be equal. ■

Theorem 3 *Let $\mathcal{P}$ be an integral polytope given by $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}, \mathbf{C}\boldsymbol{x} \leq \boldsymbol{d}, \boldsymbol{x} \geq \mathbf{0}$ where $\boldsymbol{x} \in \mathbb{R}^n$, with $\mathbf{C} \geq \mathbf{0}$ a nonzero integral matrix, and $\boldsymbol{d} \geq \mathbf{0}$. Then $\delta(\mathcal{P}) \geq \frac{1}{\|\mathbf{C}\|_\infty \sqrt{n}}$.*

Proof Consider a face $\mathcal{F}$ of the polytope $\mathcal{P}$. The face is generated by tightening some subset of the inequalities, that is, setting $x_i = 0$ for a subset $S \subseteq [n]$ of indices, and setting $\boldsymbol{c}_j^\top \boldsymbol{x} = \boldsymbol{d}_j$ for a subset $T \subseteq [n]$ of indices. We can call the submatrices obtained from $\mathbf{C}$ and $\boldsymbol{d}$ by collecting the rows whose indices are in T as $\mathbf{C}'$ and $\boldsymbol{d}'$, respectively.

Consider the complement polytope

$$\mathcal{P}' := \mathrm{Conv}(\mathrm{Vert}(\mathcal{P}) \setminus \mathcal{F})$$

and let $\boldsymbol{v}$ be a vertex of $\mathcal{P}'$. Now note that since $\boldsymbol{v}$ does not lie on $\mathcal{F}$, it must be the case that there exists an index $i \in S \cup T$ such that the corresponding inequality is not tight for $\boldsymbol{v}$.

Suppose that $i \in S$. Then, by the argument from the proof of Theorem 2, we immediately can argue that distance between the face and the complement polytope is at least $\frac{1}{\sqrt{n}}$, and thus at least $\frac{1}{\|\mathbf{C}\|_\infty \sqrt{n}}$.

Suppose that $i \in T$, and consider any point $\boldsymbol{x} \in \mathcal{F}$; necessarily $\mathbf{C}'\boldsymbol{x} = \boldsymbol{d}'$. Necessarily, we also have that $\left\|\mathbf{C}'\mathbf{v}\right\|_1 < \left\|\boldsymbol{d}'\right\|_1$, by the nonnegativity of $\mathbf{C}$, $\boldsymbol{d}$, and the polytope, and the fact that $\boldsymbol{v}$ doesn't lie on the face we are considering. Furthermore, by integrality of the polytope of $\mathbf{C}$, we must have that $\left\|\mathbf{C}'\mathbf{v}\right\|_1 \leq \left\|\boldsymbol{d}'\right\|_1 - 1$.

It follows that any convex combination of vertices on the complement polytope, $\boldsymbol{y}$, would also satisfy this inequality: $\left\|\mathbf{C}'\boldsymbol{y}\right\|_1 \leq \left\|\boldsymbol{d}'\right\|_1 - 1$. Noting again that the polytope lies in the nonnegative orthant, we can subtract the two inequalities and apply the Cauchy-Schwarz inequality to obtain:

$$1 \leq \left\|\mathbf{C}'(\boldsymbol{x} - \boldsymbol{y})\right\|_1 \leq \left\|\mathbf{C}'\right\|_\infty \left\|\boldsymbol{x} - \boldsymbol{y}\right\|_1 \leq \sqrt{n}\left\|\mathbf{C}'\right\|_\infty \left\|\boldsymbol{x} - \boldsymbol{y}\right\|_2.$$

Thus, this means in either case that the distance between the chosen face and the complement polytope is at least $\frac{1}{\left\|\mathbf{C}\right\|_\infty \sqrt{n}}$.

Since this bound holds for any chosen face, this means the facial distance is bounded below by $\frac{1}{\left\|\mathbf{C}\right\|_\infty \sqrt{n}}$ as well. ■

Appendix D. Approximate ROMD Proofs

First, we show the following crucial inequality, which we will repeatedly use to bound regret when approximate prox calls are used.

Lemma 1 *Let $\boldsymbol{x}^* = \mathrm{argmin}_{\boldsymbol{x} \in \mathcal{X}} \langle \boldsymbol{g}, \boldsymbol{x} \rangle + \frac{1}{\eta}\mathcal{D}_\varphi(\boldsymbol{x} \,\|\, \boldsymbol{c})$ and $\hat{\boldsymbol{x}}$ be such that $\langle \boldsymbol{g}, \hat{\boldsymbol{x}} \rangle + \frac{1}{\eta}\mathcal{D}_\varphi(\hat{\boldsymbol{x}} \,\|\, \boldsymbol{c}) \leq \langle \boldsymbol{g}, \boldsymbol{x}^* \rangle + \frac{1}{\eta}\mathcal{D}_\varphi(\boldsymbol{x}^* \,\|\, \boldsymbol{c}) + \epsilon$; $\hat{\boldsymbol{x}}$ is an ϵ argmin to the prox computation.*

Then we have for any $\mathbf{d} \in \mathcal{X}$:

$$\eta\langle \boldsymbol{g}, \hat{\boldsymbol{x}} - \boldsymbol{d} \rangle \leq \mathcal{D}_\varphi(\boldsymbol{d} \,\|\, \boldsymbol{c}) - \mathcal{D}_\varphi(\boldsymbol{d} \,\|\, \hat{\boldsymbol{x}}) - \mathcal{D}_\varphi(\hat{\boldsymbol{x}} \,\|\, \boldsymbol{c}) + \epsilon.$$

Furthermore, $\frac{1}{2\eta}\left\|\boldsymbol{x}^* - \hat{\boldsymbol{x}}\right\|^2 \leq \epsilon$.

Proof By definition of $\hat{\boldsymbol{x}}$, we have

$$\eta\langle \boldsymbol{g}, \hat{\boldsymbol{x}} \rangle + \varphi(\hat{\boldsymbol{x}}) - \varphi(\boldsymbol{c}) - \langle \nabla\varphi(\boldsymbol{c}), \hat{\boldsymbol{x}} - \boldsymbol{c} \rangle \leq \eta\langle \boldsymbol{g}, \boldsymbol{d} \rangle + \varphi(\boldsymbol{d}) - \varphi(\boldsymbol{c}) - \langle \nabla\varphi(\boldsymbol{c}), \boldsymbol{d} - \boldsymbol{c} \rangle + \epsilon.$$

Subtracting these inequalities, we have

$$\begin{aligned}\eta\langle \boldsymbol{g}, \hat{\boldsymbol{x}} - \boldsymbol{d}\rangle &\leq \varphi(\boldsymbol{d}) - \varphi(\hat{\boldsymbol{x}}) - \langle \nabla\varphi(\boldsymbol{c}), \boldsymbol{d} - \hat{\boldsymbol{x}}\rangle + \epsilon \\ &\leq \langle \nabla\varphi(\boldsymbol{c}) - \nabla\varphi(\hat{\boldsymbol{x}}), \hat{\boldsymbol{x}} - \boldsymbol{d}\rangle + \epsilon \\ &= \mathcal{D}_\varphi(\boldsymbol{d} \,\|\, \boldsymbol{c}) - \mathcal{D}_\varphi(\boldsymbol{d} \,\|\, \hat{\boldsymbol{x}}) - \mathcal{D}_\varphi(\hat{\boldsymbol{x}} \,\|\, \boldsymbol{c}) + \epsilon.\end{aligned}$$

The second inequality follows from the convexity of φ, and the equality follows from the three-point lemma.

In the case that φ is 1-strongly convex with respect to the Euclidean norm, we have that

$$\frac{1}{2\eta}\|\boldsymbol{x}^* - \hat{\boldsymbol{x}}\|^2 \leq \langle \boldsymbol{g}, \hat{\boldsymbol{x}}\rangle + \frac{1}{\eta}\mathcal{D}_\varphi(\hat{\boldsymbol{x}} \,\|\, \boldsymbol{c}) - \langle \boldsymbol{g}, \boldsymbol{x}^*\rangle - \frac{1}{\eta}\mathcal{D}_\varphi(\boldsymbol{x}^* \,\|\, \boldsymbol{c}) \leq \epsilon.$$

where the first inequality follows from strong convexity of a function implying quadratic growth of the function [37], and the second inequality by assumption on $\hat{\boldsymbol{x}}$. ■

Next, we show the following lemma characterizing an approximate first-order condition for the approximate-proximal-step-based framework presented in Section 3, which will be useful in analyzing the convergence rates of our algorithms.

Lemma 2 *The approximate-proximal-step-based framework presented in Section 3 satisfies the following inequality:*

$$\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}), \boldsymbol{x}^{(t)} - \boldsymbol{x}\rangle \leq \sqrt{2\eta\epsilon^{(t)}}(2LD + 3\eta).$$

In fact, when the approximate-proximal-step-based framework presented in Section 3 is instantiated with a `FW` *variant that uses the Wolfe gap as a termination criterion, as is the case for* `AFW-ROMD`*, we have the following approximate first-order optimality condition:*

$$\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}), \boldsymbol{x}^{(t)} - \boldsymbol{x}\rangle \leq \epsilon^{(t)}.$$

Proof We define $\boldsymbol{x}_*^{(t)}$ to be the result of using an exact `ROMD` update instead of using the approximate-proximal-step-based framework presented in Section 3 update at the t^{th} iteration for Player i; this corresponds to assuming that we have an exact prox oracle instead of an approximate prox oracle.

Using the first-order optimality condition, we have that for any $\boldsymbol{x} \in \mathcal{X}$

$$\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}_*^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}), \boldsymbol{x}_*^{(t)} - \boldsymbol{x}\rangle \leq 0 \tag{1}$$

.

We now have the following:

$$
\begin{aligned}
&\left\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}), \boldsymbol{x}^{(t)} - \boldsymbol{x} \right\rangle \\
&\qquad = \left\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}_*^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}), \boldsymbol{x}_*^{(t)} - \boldsymbol{x} \right\rangle \\
&\qquad\qquad + \left\langle \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}_*^{(t)}), \boldsymbol{x}_*^{(t)} - \boldsymbol{x}) \right\rangle \\
&\qquad\qquad + \left\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}), \boldsymbol{x}^{(t)} - \boldsymbol{x}_*^{(t)} \right\rangle \\
&\qquad \le 0 + \left\| \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}_*^{(t)}) \right\| \left\| \boldsymbol{x}_*^{(t)} - \boldsymbol{x} \right\| \qquad (2)\\
&\qquad\qquad + \left\| \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}) \right\| \left\| \boldsymbol{x}^{(t)} - \boldsymbol{x}_*^{(t)} \right\| \\
&\qquad \le L \left\| \boldsymbol{x}^{(t)} - \boldsymbol{x}_*^{(t)} \right\| \left\| \boldsymbol{x}_*^{(t)} - \boldsymbol{x} \right\| \qquad (3)\\
&\qquad\qquad + \left(\eta \left\| 2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)} \right\| + \left\| \nabla\varphi(\boldsymbol{x}^{(t)}) - \nabla\varphi(\boldsymbol{x}^{(t-1)}) \right\| \right) \left\| \boldsymbol{x}^{(t)} - \boldsymbol{x}_*^{(t)} \right\| \\
&\qquad \le \sqrt{2\eta\epsilon^{(t)}} LD + \sqrt{2\eta\epsilon^{(t)}} \left(\eta(2\left\| \boldsymbol{\ell}^{(t-1)} \right\| + \left\| \boldsymbol{\ell}^{(t-2)} \right\|) + L \left\| \boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)} \right\| \right) \qquad (4)\\
&\qquad \le \sqrt{2\eta\epsilon^{(t)}} (2LD + 3\eta). \qquad (5)
\end{aligned}
$$

We use Inequality (1) and the Cauchy-Schwarz inequality in Inequality (2), the triangle inequality and smoothness of φ in Inequality (3), Lemma 1 in Inequality (4), and bounded losses in Inequality (5).

Suppose the Wolfe gap is used as a termination criterion. In that case, the stated approximate first-order optimality condition immediately follows because the left-hand side of the stated inequality is precisely the Wolfe gap. ■

Lemma 3 *Let* $\boldsymbol{x}^* = \operatorname{argmax}_{\boldsymbol{x}' \in \mathcal{X}} \sum_{t=1}^{T} \langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t)} \rangle - \langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}' \rangle$. *the approximate-proximal-step-based framework presented in Section 3 yields*

$$
\operatorname{Reg}^{(T)} \le \sum_{t=1}^{T} \| \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)} \| \cdot \| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \| - \frac{1}{2\eta} \sum_{t=1}^{T} \| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \|^2 + \frac{1}{\eta} \mathcal{D}_{\varphi}\left(\boldsymbol{x}^* \,\|\, \boldsymbol{x}^{(0)} \right) + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} - \boldsymbol{x}^* \right\rangle + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta}
$$

Proof Let $\boldsymbol{x}' \in \mathcal{X}$. If we apply Lemma 1, then we obtain:

$$
\eta \left\langle \boldsymbol{\ell}^{(t)} + \boldsymbol{m}^{(t+1)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}' \right\rangle \le \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}' \,\|\, \boldsymbol{x}^{(t)} \right) - \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}' \,\|\, \boldsymbol{x}^{(t+1)} \right) - \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{(t+1)} \,\|\, \boldsymbol{x}^{(t)} \right) + \epsilon^{(t)}. \qquad (6)
$$

Summing the left side over $t = 1, \dots, T$, and noting that we can let $\boldsymbol{m}^{(T+1)} = 0$ without affecting losses at timesteps before $T+1$, we have:

$$\begin{aligned}
&\sum_{t=1}^{T} \left\langle \boldsymbol{\ell}^{(t)} + \boldsymbol{m}^{(t+1)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}' \right\rangle \\
&\quad = \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}' \right\rangle + \left\langle \boldsymbol{m}^{(t+1)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}' \right\rangle \right] \\
&\quad = \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}' \right\rangle + \left\langle \boldsymbol{m}^{(t+1)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} \right\rangle \right] + \left\langle \boldsymbol{m}_i^{(1)} - \boldsymbol{m}^{(T+1)}, \boldsymbol{x}' \right\rangle \\
&\quad = \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}' \right\rangle + \left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\rangle + \left\langle \boldsymbol{m}^{(t+1)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} \right\rangle \right] + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}' \right\rangle \\
&\quad = \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}' \right\rangle + \left\langle \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\rangle \right] + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}' \right\rangle - \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} \right\rangle \\
&\qquad\qquad + \left\langle \boldsymbol{m}^{(T+1)}, \boldsymbol{x}^{(T+1)} \right\rangle \\
&\quad = \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}' \right\rangle + \left\langle \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\rangle \right] + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}' - \boldsymbol{x}^{(2)} \right\rangle. \qquad (7)
\end{aligned}$$

This allows us to decompose the regret into two terms and apply Inequality (6) to one of these terms:

$$\begin{aligned}
\mathrm{Reg}^{(T)} &= \max_{\boldsymbol{x}' \in \mathcal{X}} \sum_{t=1}^{T} \left\langle \boldsymbol{\ell}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}' \right\rangle \\
&= \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t+1)} \right\rangle + \left\langle \boldsymbol{\ell}^{(t)} + \boldsymbol{m}^{(t+1)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{*} \right\rangle \right] \\
&\qquad + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} - \boldsymbol{x}^{*} \right\rangle \qquad (8) \\
&\leq \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t+1)} \right\rangle \right] + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} - \boldsymbol{x}^{*} \right\rangle + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta} \\
&\qquad + \sum_{t=1}^{T} \frac{1}{\eta} \left(\mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{*} \,\|\, \boldsymbol{x}^{(t)} \right) - \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{*} \,\|\, \boldsymbol{x}^{(t+1)} \right) - \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{(t+1)} \,\|\, \boldsymbol{x}^{(t)} \right) \right) \qquad (9) \\
&\leq \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t+1)} \right\rangle - \frac{1}{\eta} \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{(t+1)} \,\|\, \boldsymbol{x}^{(t)} \right) \right] + \frac{1}{\eta} \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{*} \,\|\, \boldsymbol{x}^{(0)} \right) \\
&\qquad + \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} - \boldsymbol{x}^{*} \right\rangle + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta} \qquad (10) \\
&\leq \sum_{t=1}^{T} \left[\left\langle \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)}, \boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t+1)} \right\rangle - \frac{1}{2\eta} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^{2} \right] + \frac{1}{\eta} \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^{*} \,\|\, \boldsymbol{x}^{(0)} \right)
\end{aligned}$$

$$+ \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} - \boldsymbol{x}^* \right\rangle + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta} \tag{11}$$

$$\leq \sum_{t=1}^{T} \left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{m}^{(t)} \right\| \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\| - \frac{1}{2\eta} \sum_{t=1}^{T} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2 + \frac{1}{\eta} \mathcal{D}_{\varphi_i}\left(\boldsymbol{x}^* \parallel \boldsymbol{x}^{(0)} \right)$$

$$+ \left\langle \boldsymbol{m}^{(1)}, \boldsymbol{x}^{(2)} - \boldsymbol{x}^* \right\rangle + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta}. \tag{12}$$

■

We apply Equation (7) to obtain Equation (8), Inequality (6) to obtain Inequality (9), drop a negative term to obtain Inequality (10), apply strong convexity of φ to obtain Inequality (11), and finally apply the Cauchy-Schwarz inequality to obtain Inequality (12).

Lemma 4 *The approximate proximal step based framework presented in Section 3 with $\epsilon^{(t)} = \frac{1}{t^2}$-optimal prox computations at each time step and using $\boldsymbol{m}^{(t)} = \boldsymbol{\ell}^{(t-1)}$ yields*

$$\mathrm{Reg}^{(T)} \leq \frac{\Omega + 2}{\eta} + \eta \sum_{t=1}^{T} \|\boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)}\|^2 - \frac{1}{4\eta} \sum_{t=1}^{T} \|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\|^2$$

In particular, this satisfies the RVU property with $\alpha = \frac{\Omega+2}{\eta}, \beta = \eta, \gamma = \frac{1}{4\eta}$.

Proof Instantiating Lemma 3 with $\boldsymbol{m}^{(t)} = \boldsymbol{\ell}^{(t-1)}$, noting that $\boldsymbol{\ell}^{(0)} = \mathbf{0}$ and using the definition of Ω:

$$\mathrm{Reg}^{(T)} \leq \sum_{t=1}^{T} \left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)} \right\| \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\| - \frac{1}{2\eta} \sum_{t=1}^{T} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2 + \frac{\Omega}{\eta} + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta}.$$

Applying Young's inequality:

$$\left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)} \right\| \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\| \leq \frac{1}{2} \left(2\eta \left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)} \right\|^2 + \frac{1}{2\eta} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2 \right).$$

Furthermore, instantiating $\epsilon^{(t)} = \frac{1}{t^2}$, we also have that:

$$\sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta} = \frac{1}{\eta} \sum_{t=1}^{T} \frac{1}{t^2} \leq \frac{2}{\eta}.$$

Combining the above, we have:

$$\mathrm{Reg}^{(T)} \leq \sum_{t=1}^{T} \frac{1}{2} (2\eta \left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)} \right\|^2 + \frac{1}{2\eta} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2) - \frac{1}{2\eta} \sum_{t=1}^{T} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2 + \frac{\Omega}{\eta} + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta}$$

$$= \eta \sum_{t=1}^{T} \left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)} \right\|^2 - \frac{1}{4\eta} \sum_{t=1}^{T} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2 + \frac{\Omega}{\eta} + \sum_{t=1}^{T} \frac{\epsilon^{(t)}}{\eta}$$

$$\leq \frac{\Omega + 2}{\eta} + \eta \sum_{t=1}^{T} \left\| \boldsymbol{\ell}^{(t)} - \boldsymbol{\ell}^{(t-1)} \right\|^2 - \frac{1}{4\eta} \sum_{t=1}^{T} \left\| \boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)} \right\|^2.$$

■

D.0.1. SELF-PLAY IN GAMES

In order to compute equilibria in games, we will assume that Player i receives as its loss $\boldsymbol{\ell}_i^{(t)} = -\nabla_{\boldsymbol{x}_i} u_i(\boldsymbol{x}_1^{(t)}, \ldots, \boldsymbol{x}_n^{(t)})$.

Assumption 1 *We assume that our games satisfy a smoothness condition:*

$$\|\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}\| \le \sum_{j \neq i} \|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\|.$$

This can always be satisfied by rescaling the utility function for any game where the utility function is multilinear in the players' strategies (as is the case in NFGs and EFGs).

Next, we restate Theorem 4 from Syrgkanis et al. [54], which characterizes the stepsize required to achieve constant regret.

Theorem 8 *[54] If each player employs an algorithm satisfying the RVU property with parameters α, β, and γ, such that $\beta \le \gamma/(N-1)^2$, then $\sum_{i \in [N]} \mathrm{Reg}_i^{(T)} \le \alpha N$.*

Proof By Assumption 1 and Jensen's inequality we have:

$$\begin{aligned}\|\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}\|^2 &\le \left(\sum_{j \neq i} \|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\|\right)^2 \\ &\le (N-1) \sum_{j \neq i} \|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\|^2.\end{aligned}$$

Summing up the terms for all the players, we have that

$$\sum_{i \in [N]} \|\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}\|^2 \le (N-1)^2 \sum_{i \in [N]} \|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\|^2.$$

The theorem immediately follows by noting that the assumption on β and γ ensures that the latter two terms in the RVU bound can be dropped and the inequality will still hold. ■

Lemma 5 *When running T iterations of* `AFW-ROMD` *using $\boldsymbol{m}^{(t)} = \boldsymbol{\ell}^{(t-1)}$:*

1. $\epsilon^{(t)} = \frac{1}{t^2}$-optimal prox computations at each time step requires $O(TL\frac{D^2}{\delta^2} \log[LDT])$ LMO calls.
2. $\epsilon^{(t)} = \epsilon$-optimal prox computations at each time step requires $O(TL\frac{D^2}{\delta^2} \log\left[\frac{LD}{\epsilon}\right])$ LMO calls.

Proof In the first case, note that `AFW` can achieve a $\epsilon^{(t)}$ optimal solution with $O\big(\frac{LD^2}{\delta^2} \log\left[\frac{LD}{\epsilon^{(t)}}\right]\big)$ LMO calls, which means that `AFW-ROMD` requires $O\big(TL\frac{D^2}{\delta^2} \log[LDT]\big)$ LMO calls in order to achieve constant cumulative regret, since we can lower bound $\epsilon^{(t)}$ by $\frac{1}{T^2}$.

In the second case, note that `AFW` can achieve an ϵ optimal solution with $O\big(\frac{LD^2}{\delta^2} \log\left[\frac{LD}{\epsilon^{(t)}}\right]\big)$ LMO calls, which means that `AFW-ROMD` requires $O\big(TL\frac{D^2}{\delta^2} \log\left[\frac{LD}{\epsilon}\right]\big)$ LMO calls in order to achieve constant cumulative regret.

Additionally, note that at least 1 LMO call is required by `AFW` at each iteration to check whether `AFW` has reached an $\epsilon^{(t)}$-optimal solution. It follows that N is also $O(T \log N)$, since T is $O(N)$. It follows then that $\frac{1}{T}$ is in $O(\frac{\log N}{N})$, and thus with N LMO calls we can achieve $O(\frac{\log N}{N})$ average regret. ■

Theorem 4 *An ϵ'-Nash equilibrium in any two-player zero-sum polyhedral game can be computed in $O(1/\epsilon')$ iterations of the above framework. This corresponds to $O(\max_{i \in \{1,2\}} \frac{1}{\epsilon'} \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i}{\epsilon'}\right])$ LMO calls when using* `AFW-ROMD`.

Proof For `AFW-ROMD`, we know that the RVU property is satisfied with $\alpha_i = \frac{\Omega_i + 2}{\eta_i}$, $\beta_i = \eta_i$, $\gamma_i = \frac{1}{4\eta_i}$ for Player i. By Theorem 8, if we take $\eta_i \leq \frac{1}{2(N-1)}$, then we have that $\sum_{i \in [N]} \text{Reg}_i^{(T)} \leq \alpha(N-1)$, where $\alpha = \max_{i \in [N]} \alpha_i$.

It follows that we if we take ξ as the Nash gap corresponding to the average strategies of the two players $\bar{\boldsymbol{x}}_1 = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{x}_1^{(T)}$ and $\bar{\boldsymbol{x}}_2 = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{x}_2^{(T)}$, then we have:

$$\xi = \max_{\boldsymbol{x}_2 \in \mathcal{X}_2} \langle \mathbf{A}\bar{\boldsymbol{x}}_1, \boldsymbol{x}_2 \rangle - \min_{\boldsymbol{x}_1 \in \mathcal{X}_1} \langle \mathbf{A}\boldsymbol{x}_1, \bar{\boldsymbol{x}}_2 \rangle = \frac{1}{T}\left(\text{Reg}_2^{(T)} + \text{Reg}_1^{(T)}\right) \leq \frac{\alpha}{T}$$

since for any given $\boldsymbol{x}_2 \in \mathcal{X}_2$ we have $\langle \mathbf{A}\bar{\boldsymbol{x}}_1, \boldsymbol{x}_2 \rangle = \sum_{t=1}^{T} \langle \mathbf{A}\boldsymbol{x}_1^{(t)}, \boldsymbol{x}_2 \rangle$ and for any given $\boldsymbol{x}_1 \in \mathcal{X}_1$ we have $\langle \mathbf{A}\boldsymbol{x}_1, \bar{\boldsymbol{x}}_2 \rangle = \sum_{t=1}^{T} \langle \mathbf{A}\boldsymbol{x}_1, \boldsymbol{x}_2^{(t)} \rangle$.

This demonstrates that we can achieve an ϵ'-NE in a two-player zero-sum game in $O(1/\epsilon')$ iterations. By Lemma 5, if we use $\epsilon_i^{(t)} = \frac{1}{t^2}$, since $T = O(1/\epsilon')$, we require $O(\max_{i \in [N]} \frac{1}{\epsilon'} \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i}{\epsilon'}\right])$ LMO calls to achieve a ϵ'-NE. ■

Theorem 5 *An ϵ'-CCE in any N-player general-sum polyhedral game can be computed in $O(1/\epsilon'^{\frac{4}{3}})$ iterations of the above framework. This corresponds to $O(\max_{i \in [N]} \frac{1}{\epsilon'^{\frac{4}{3}}} \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i}{\epsilon'}\right])$ LMO calls when using* `AFW-ROMD`.

Proof We claim that letting $\eta_i = \frac{1}{T^{1/4}}$ allows for this result to hold.

To prove this statement, we first prove a lemma about the *stability* of our iterates.

Lemma 6 *The approximate proximal step based framework presented in Section 3 with $\boldsymbol{m}^{(t)} = \boldsymbol{\ell}^{(t-1)}$ has the following property:*

$$\left\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^{(t)}\right\| \leq 3\eta + \sqrt{2\eta\epsilon^{(t)}}.$$

Proof As in the proof of Lemma 2, we define $\boldsymbol{x}_*^{(t)}$ as the result of using an exact `ROMD` update instead of using the approximate proximal step based updated presented in Section 3 at the t^{th} iteration for Player i; this corresponds to assuming that we have an exact prox oracle instead of an approximate prox oracle.

Using first-order optimality, we have that for any $\boldsymbol{x} \in \mathcal{X}$

$$\left\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}) + \nabla\varphi(\boldsymbol{x}_*^{(t)}) - \nabla\varphi(\boldsymbol{x}), \boldsymbol{x}_*^{(t)} - \boldsymbol{x} \right\rangle \leq 0. \tag{13}$$

Using this inequality, we have:

$$\begin{aligned}
\left\|\boldsymbol{x}_*^{(t)} - \boldsymbol{x}\right\|^2 &\leq \left\langle \nabla\varphi(\boldsymbol{x}_*^{(t)}) - \nabla\varphi(\boldsymbol{x}), \boldsymbol{x}_*^{(t)} - \boldsymbol{x} \right\rangle && (14)\\
&\leq -\left\langle \eta(2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}), \boldsymbol{x}_*^{(t)} - \boldsymbol{x} \right\rangle && (15)\\
&\leq \eta\left\|2\boldsymbol{\ell}^{(t-1)} - \boldsymbol{\ell}^{(t-2)}\right\|\left\|\boldsymbol{x}_*^{(t)} - \boldsymbol{x}\right\| && (16)\\
&\leq \eta\Big(2\left\|\boldsymbol{\ell}^{(t-1)}\right\| + \left\|\boldsymbol{\ell}^{(t-2)}\right\|\Big)\left\|\boldsymbol{x}_*^{(t)} - \boldsymbol{x}\right\| && (17)\\
&\leq 3\eta\left\|\boldsymbol{x}_*^{(t)} - \boldsymbol{x}\right\|. && (18)
\end{aligned}$$

In Inequality (14), we apply the strong convexity of φ, in Inequality (15) we apply Inequality (13), in Inequality (16) we apply the Cauchy-Schwarz inequality, in Inequality (17) we apply the triangle inequality, and finally in Inequality (18) we apply the assumption on the norms of the losses. Dividing by $\left\|\boldsymbol{x}_*^{(t)} - \boldsymbol{x}^{(t-1)}\right\|$ (assuming it is non-zero; otherwise, the below inequality trivially holds), we have that:

$$\left\|\boldsymbol{x}_*^{(t)} - \boldsymbol{x}^{(t-1)}\right\| \leq 3\eta.$$

Now using the triangle inequality and Lemma 1, we have that

$$\left\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^{(t-1)}\right\| \leq 3\eta + \sqrt{2\eta\epsilon^{(t)}}.$$

■

By Assumption 1, Jensen's inequality, and the above, we have:

$$\left\|\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}\right\|^2 \leq (N-1)\sum_{j \neq i}\left\|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\right\|^2 \leq (N-1)^2 \max_{j\in[N]}\left(18\eta_j^2 + 4\eta_j\epsilon_j^{(t)}\right). \tag{19}$$

Now we can use the refined RVU bound from Lemma 4:

$$\begin{aligned}
\mathrm{Reg}_i^{(T)} &\leq \frac{\Omega_i + 2}{\eta_i} + \eta_i(N-1)^2 \max_{j\in[N]}\sum_{t=1}^{T}\left(18\eta_j^2 + 4\eta_j\epsilon_j^{(t)}\right) - \frac{1}{4\eta_i}\sum_{t=1}^{T}\left\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\right\|^2 && (20)\\
&\leq \frac{\Omega_i + 2}{\eta_i} + \eta_i(N-1)^2 \max_{j\in[N]}\sum_{t=1}^{T}\left(18\eta_j^2 + 4\eta_j\epsilon_j^{(t)}\right)\\
&= \frac{\Omega_i + 2}{\eta_i} + \eta_i(N-1)^2 \max_{j\in[N]}\left(18T\eta_j^2 + 8\eta_j\right). && (21)
\end{aligned}$$

To obtain Inequality (20), we plug Inequality (19) into Lemma 4, and for Equation (21), we use the fact that $\epsilon_i^{(t)} = \frac{1}{t^2}$ so $\sum_{i=1}^{T} \epsilon_i^{(t)} \leq 2$.

Now if we let $\eta_i = \frac{1}{T^{1/4}}$, we get that $\text{Reg}_i^{(T)}$ is in $O(T^{1/4})$, showing that the average joint strategy of the players converges to a CCE, at a rate $O(T^{-3/4})$. Equivalently, to reach a ϵ'-CCE, we require $O(\epsilon'^{-4/3})$ iterations of the approximate-proximal-step-based framework presented in Section 3. By Lemma 5, when using AFW-ROMD, if we use $\epsilon_i^{(t)} = \frac{1}{t^2}$, since $T = O(1/\epsilon'^{4/3})$, we require $O(\max_{i\in[N]} \frac{1}{\epsilon'^{4/3}} \frac{L_i D_i^2}{\delta_i^2} \log \left[\frac{L_i D_i}{\epsilon'}\right])$ LMO calls to achieve a ϵ'-CCE.

■

Appendix E. Last-Iterate Results

Theorem 6 *For any N-player general-sum polyhedral game, given $\epsilon \in (0,1)$, let Player i employ the above framework with $\epsilon_i^{(t)} = \epsilon^2$ and $\boldsymbol{m}_i^{(t)} = \boldsymbol{\ell}_i^{(t-1)}$. Let $\eta_{\max} \leq \frac{1}{2\sqrt{2}(N-1)}$ where $\eta_{\max} = \max_{i\in[N]} \eta_i$ and suppose $\sum_{i=1}^{N} \text{Reg}_i^{(t)} \geq 0$ for any $t \in \mathbb{N}$. Define $\alpha_i = \left(\frac{1}{\eta_i} + \frac{2\Omega_i}{\eta_i}(L_i + N - 1) + 1\right)$. Then, after $T > \left\lceil \frac{8\eta_{\max}}{\epsilon^2} \sum_{i=1}^{N} \frac{(\Omega_i+2)}{\eta_i} \right\rceil$ iterations, there exists $\boldsymbol{x}^{(t)}$ with $t \in [T]$ which is an $\epsilon\left(\max_{i\in[N]} \sqrt{2\eta_i}(\frac{2L_iD_i}{\eta_i} + 3) + \alpha_i\right)$-approximate Nash equilibrium. AFW-ROMD will yield an iterate that is an ϵ'-approximate Nash equilibrium in $O\left(\max_{j\in[N]} \left\{\frac{\eta_{\max}\alpha_j^2}{\epsilon'^2} \sum_{i=1}^{N} (\frac{\Omega_i+2}{\eta_i}) \frac{L_i D_i^2}{\delta_i^2} \log\left[\frac{L_i D_i \alpha_j}{\epsilon'}\right]\right\}\right)$ LMO calls when $\epsilon \leq \min_{i\in[N]} \frac{\epsilon'}{\alpha_i}$.*

Proof We follow the proof of Theorem A.12 from Anagnostides et al. [3].

We rewrite the regret bound given by Lemma 4 for Player i:

$$\text{Reg}_i^{(T)} \leq \frac{\Omega_i + 2}{\eta_i} + \eta_i \sum_{t=1}^{T} \|\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}\|^2 - \frac{1}{8\eta_i} \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\|^2 - \frac{1}{8\eta_i} \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\|^2.$$

Applying Assumption 1 and Jensen's inequality, we have:

$$\begin{aligned}\text{Reg}_i^{(T)} \leq \frac{\Omega_i + 2}{\eta_i} &+ (N-1)\eta_i \sum_{j\neq i} \sum_{t=1}^{T} \|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\|^2 - \frac{1}{8\eta_i} \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\|^2 \\ &- \frac{1}{8\eta_i} \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\|^2.\end{aligned}$$

Using the fact that $\eta_{\max} = \max_{i\in[N]} \eta_i$, we can rewrite the above as:

$$\begin{aligned}\text{Reg}_i^{(T)} \leq \frac{\Omega_i + 2}{\eta_i} &+ (N-1)\eta_{\max} \sum_{j\neq i} \sum_{t=1}^{T} \|\boldsymbol{x}_j^{(t)} - \boldsymbol{x}_j^{(t-1)}\|^2 - \frac{1}{8\eta_{\max}} \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\|^2 \\ &- \frac{1}{8\eta_{\max}} \sum_{t=1}^{T} \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\|^2.\end{aligned}$$

Summing these terms over all the players yields:

$$\sum_{i=1}^{N}\mathrm{Reg}_i^{(T)} \le \sum_{i=1}^{N}\frac{\Omega_i+2}{\eta_i} + \Big((N-1)^2\eta_{\max} - \frac{1}{8\eta_{\max}}\Big)\sum_{i=1}^{N}\sum_{t=1}^{T}\big\|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t-1)}\big\|^2 \\ - \frac{1}{8\eta_{\max}}\sum_{i=1}^{N}\sum_{t=1}^{T}\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\big\|^2.$$

Using the assumptions $\eta_{\max} \le \frac{1}{2\sqrt{2}(N-1)}$ and $\sum_{i=1}^{N}\mathrm{Reg}_i^{(T)} \ge 0$, we can write

$$0 \le \sum_{i=1}^{N}\mathrm{Reg}_i^{(T)} \le \sum_{i=1}^{N}\Big[\frac{\Omega_i+2}{\eta_i} - \frac{1}{8\eta_{\max}}\sum_{t=1}^{T}\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\big\|^2\Big].$$

Hence,

$$\sum_{t=1}^{T}\sum_{i=1}^{N}\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\big\|^2 \le 8\eta_{\max}\sum_{i=1}^{N}\frac{\Omega_i+2}{\eta_i}.$$

Assuming for all $t \in [T]$ that $\sum_{i=1}^{N}\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\big\|^2 \ge \epsilon^2$, we must have that $T \le \frac{8\eta_{\max}}{\epsilon^2}\sum_{i=1}^{N}\frac{\Omega_i+2}{\eta_i}$. Thus, as long as $T > \Big\lceil\frac{8\eta_{\max}}{\epsilon^2}\sum_{i=1}^{N}\frac{\Omega_i+2}{\eta_i}\Big\rceil$, there must exist $t \in [T]$ such that

$$\frac{1}{2}\Bigg(\sum_{i=1}^{N}\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\big\|^2 + \sum_{i=1}^{N}\big\|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t-1)}\big\|^2\Bigg) \le \epsilon^2.$$

Next, we show that $\frac{1}{2}\Big(\sum_{i=1}^{N}\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\big\|^2 + \sum_{i=1}^{N}\big\|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t-1)}\big\|^2\Big) \le \epsilon^2$ implies that we are at an approximate Nash equilibrium.

Using Lemma 2, we have for any $\boldsymbol{x}_i \in \mathcal{X}_i$:

$$\big\langle \eta_i(2\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}) + \nabla\varphi_i(\boldsymbol{x}_i^{(t+1)}) - \nabla\varphi_i(\boldsymbol{x}_i^{(t)}), \boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i\big\rangle \le \sqrt{2\eta_i\epsilon_i^{(t+1)}}(2L_iD_i + 3\eta_i).$$

Rearranging, we have:

$$\begin{aligned}
\eta_i\big\langle\boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i\big\rangle &\le \sqrt{2\eta_i\epsilon_i^{(t+1)}}(2L_iD_i + 3\eta_i) \\
&\qquad + \big\langle \eta_i(-\boldsymbol{\ell}_i^{(t)} + \boldsymbol{\ell}_i^{(t-1)}) - \nabla\varphi_i(\boldsymbol{x}_i^{(t+1)}) + \nabla\varphi_i(\boldsymbol{x}_i^{(t)}), \boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i\big\rangle \\
&\le \sqrt{2\eta_i\epsilon_i^{(t+1)}}(2L_iD_i + 3\eta_i) \\
&\qquad + \Big(\big\|\boldsymbol{\ell}_i^{(t-1)} - \boldsymbol{\ell}_i^{(t)}\big\| + \big\|\nabla\varphi_i(\boldsymbol{x}_i^{(t)}) - \nabla\varphi_i(\boldsymbol{x}_i^{(t+1)})\big\|\Big)\Big(\big\|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i\big\|\Big) \qquad (22) \\
&\le \sqrt{2\eta_i\epsilon_i^{(t+1)}}(2L_iD_i + 3\eta_i) + \Bigg(\sum_{j\ne i}\big\|\boldsymbol{x}_j^{(t-1)} - \boldsymbol{x}_j^{(t)}\big\| + L_i\big\|\boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t+1)}\big\|\Bigg)\Omega_i \qquad (23) \\
&\le \sqrt{2\eta_i\epsilon_i^{(t+1)}}(2L_iD_i + 3\eta_i) + 2\Omega_i\epsilon(L_i + N - 1). \qquad (24)
\end{aligned}$$

We applied the Cauchy-Schwarz inequality in Inequality (22), Assumption 1 and smoothness of φ_i in Inequality (23), and the bound on the second-order path lengths in Inequality (24).

Furthermore, note that $\langle \boldsymbol{\ell}_i^{(t)}, \boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t+1)} \rangle \leq \|\boldsymbol{\ell}_i^{(t)}\| \|\boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i^{(t)}\| \leq \epsilon$ due to the assumptions on the boundedness of the losses and the second-order path lengths. Thus, we have that:

$$\begin{aligned} \langle \boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i \rangle &= \langle \boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i \rangle + \langle \boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t+1)} \rangle \\ &\leq \sqrt{2\eta_i \epsilon_i^{(t+1)}} \Big(\frac{2L_i D_i}{\eta_i} + 3 \Big) + \frac{2\Omega_i}{\eta_i} \epsilon (L_i + N - 1) + \epsilon \\ &= \sqrt{2\eta_i \epsilon^2} \Big(\frac{2L_i D_i}{\eta_i} + 3 \Big) + \frac{2\Omega_i}{\eta_i} \epsilon (L_i + N - 1) + \epsilon \\ &\leq \epsilon \bigg(\sqrt{2\eta_i} \Big(\frac{2L_i D_i}{\eta_i} + 3 \Big) + \alpha_i \bigg). \end{aligned}$$

The result follows by noting the definition of approximate Nash equilibrium. In the case that the players are employing AFW-ROMD, by Lemma 2, we have instead that for any $\boldsymbol{x}_i \in \mathcal{X}_i$:

$$\langle \eta_i (2\boldsymbol{\ell}_i^{(t)} - \boldsymbol{\ell}_i^{(t-1)}) + \nabla\varphi_i(\boldsymbol{x}_i^{(t+1)}) - \nabla\varphi_i(\boldsymbol{x}_i^{(t)}), \boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i \rangle \leq \epsilon_i^{(t+1)}.$$

Using the same analysis as above and noting that $\epsilon^2 < \epsilon$ we have that

$$\begin{aligned} \langle \boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i \rangle &= \langle \boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t+1)} - \boldsymbol{x}_i \rangle + \langle \boldsymbol{\ell}_i^t, \boldsymbol{x}_i^{(t)} - \boldsymbol{x}_i^{(t+1)} \rangle \\ &\leq \frac{\epsilon^2}{\eta_i} + \frac{2\Omega_i}{\eta_i} \epsilon (L_i + N - 1) + \epsilon \\ &\leq \epsilon \bigg(\frac{1}{\eta_i} + \frac{2\Omega_i}{\eta_i} (L_i + N - 1) + 1 \bigg) \\ &= \alpha_i \epsilon. \end{aligned}$$

Thus, it is sufficient to let $\epsilon \leq \min_{i \in [N]} \frac{\epsilon'}{\alpha_i}$ to ensure that the iterate corresponds to a ϵ'-NE. The number of required LMO calls follows immediately from Lemma 5. ■

Theorem 7 *In any two-player zero-sum polyhedral game, both players employing the approximate-proximal-step-based framework presented in Section 3 with* $\boldsymbol{m}_i^{(t)} = \boldsymbol{\ell}_i^{(t-1)}$, $\epsilon_i^{(t)} = \epsilon$, $\varphi_i(\boldsymbol{x}_i) = \frac{1}{2}\|\boldsymbol{x}_i\|_2^2$, *and* $\eta_i = \eta \leq \frac{1}{4}$ *yields linear last-iterate convergence to a* $\frac{(16+C_1)\epsilon + 32 \max_{i\in\{1,2\}} \sqrt{2\eta\epsilon}(2L_iD_i+3\eta)}{C_2}$*-approximate Nash equilibrium, where* ν *is a game-dependent constant associated with the SP-MS condition,* $C_1 = 2(1 + \frac{4\eta^2\nu^2}{25})$, *and* $C_2 = \min(\frac{1}{2}, \frac{\eta^2\nu^2}{25})$:

$$\operatorname{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 \leq 2\bigg(1 + \frac{C_2}{4}\bigg)^{-t} \operatorname{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2 + \frac{(16 + C_1)\epsilon + 32 \max_{i\in\{1,2\}} \sqrt{2\eta\epsilon}(2L_iD_i + 3\eta)}{C_2}.$$

In the same setting ($\boldsymbol{m}_i^{(t)} = \boldsymbol{\ell}_i^{(t-1)}$, $\epsilon_i^{(t)} = \epsilon$, *and* $\eta_i = \eta \leq \frac{1}{4}$*), if it is assumed that both players are applying* AFW-ROMD*, then they can achieve linear last-iterate convergence to a* $\frac{48+C_1}{C_2}\epsilon$*-approximate Nash equilibrium, with the same definitions for* ν, C_1, C_2.

$$\operatorname{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 \leq 2\bigg(1 + \frac{C_2}{4}\bigg)^{-t} \operatorname{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2 + \frac{48 + C_1}{C_2}\epsilon.$$

`AFW-ROMD` *requires*

$$O\left(\max_{i\in\{1,2\}} \frac{\log \frac{2C_2+48+C_1}{C_2\epsilon'}}{\log \frac{4+C_2}{4}} \frac{L_i D_i^2}{\delta_i^2} \log\left[\frac{(2C_2+48+C_1)L_i D_i}{C_2\epsilon'}\right]\right).$$

LMO calls to compute an ϵ'-NE. Furthermore, the approximate solution it returns will have support of size $O\Big(\max_{i\in\{1,2\}} \frac{L_i D_i^2}{\delta_i^2}\log\left[\frac{L_i D_i(2C_2+48+C_1)}{C_2\epsilon'}\right]\Big)$.

We adapt arguments from Wei et al. [58] and Malitsky [47].

Proof In this proof, for convenience, we define the following $\boldsymbol{\ell}^{(t)} := (\boldsymbol{\ell}_{\boldsymbol{x}}^{(t)}, \boldsymbol{\ell}_{\boldsymbol{y}}^{(t)})$, $\boldsymbol{m}^{(t)} := (\boldsymbol{m}_{\boldsymbol{x}}^{(t)}, \boldsymbol{m}_{\boldsymbol{y}}^{(t)})$, $\varphi(\boldsymbol{z}) := \varphi_{\boldsymbol{x}}(\boldsymbol{x}) + \varphi_{\boldsymbol{y}}(\boldsymbol{y})$. We take $\epsilon_{\boldsymbol{x}}^{(t)} := \epsilon_{\boldsymbol{y}}^{(t)} := \epsilon$, and $\epsilon_{\boldsymbol{z}}^{(t)} := \epsilon_{\boldsymbol{x}}^{(t)} + \epsilon_{\boldsymbol{y}}^{(t)} := 2\epsilon$. Additionally, we let $\boldsymbol{w}^{(t)} := 2\boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t-1)}$.

The calls to their respective APOs for $\boldsymbol{x}$ and $\boldsymbol{y}$ in a single iteration of the approximate-proximal-step-based framework presented in Section 3 can be written as

$$\begin{aligned}\boldsymbol{x}^{(t+1)} &= \mathrm{APO}_{\mathcal{X}}\big(-\eta\langle \boldsymbol{\ell}_{\boldsymbol{x}}^{(t)} + \boldsymbol{m}_{\boldsymbol{x}}^{(t)} - \boldsymbol{m}_{\boldsymbol{x}}^{(t-1)}, \cdot\rangle, \varphi, \boldsymbol{x}^{(t)}, \epsilon\big),\\ \boldsymbol{y}^{(t+1)} &= \mathrm{APO}_{\mathcal{Y}}\big(-\eta\langle \boldsymbol{\ell}_{\boldsymbol{y}}^{(t)} + \boldsymbol{m}_{\boldsymbol{y}}^{(t)} - \boldsymbol{m}_{\boldsymbol{y}}^{(t-1)}, \cdot\rangle, \varphi, \boldsymbol{y}^{(t)}, \epsilon\big).\end{aligned} \tag{25}$$

This can be written as a single prox call for $\boldsymbol{z}$ as follows.

$$\boldsymbol{z}^{(t+1)} = \mathrm{APO}_{\mathcal{Z}}\big(-\eta\langle \boldsymbol{\ell}^{(t)} + \boldsymbol{m}^{(t)} - \boldsymbol{m}^{(t-1)}, \cdot\rangle, \varphi, \boldsymbol{z}^{(t)}, 2\epsilon\big). \tag{26}$$

We define $\boldsymbol{z}_*^{(t+1)}$ as true solution to the prox call in Equation (26).

First, we prove a version of Lemma 1 from Wei et al. [58] and Lemma 3.1 from Malitsky [47]; this inequality will allow us to characterize the bound the current distance to optimality in terms of the distance to optimality at the previous iterate. We define $g(\epsilon) = \max_{i\in\{1,2\}} \sqrt{2\eta_i\epsilon}(2L_iD_i + 3\eta_i)$ when the players are assumed to be employing the approximate-proximal-step-based framework presented in Section 3, and $g(\epsilon) = \epsilon$ when the players are assumed to be employing `AFW-ROMD`; the reason we make this definition is because Lemma 2 yields a simpler bound when the Wolfe gap is used as a stopping criterion for the approximate proximal computation.

Lemma 7 *Under the same assumptions as Theorem 7,*

$$\begin{aligned}\mathrm{dist}(\boldsymbol{z}^{(t+1)}, \mathcal{Z}^*)^2 + \frac{1}{2}\big\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\big\|^2 \le \mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 + \frac{1}{2}\big\|\boldsymbol{w}^{(t-1)} - \boldsymbol{z}^{(t)}\big\|^2 - \frac{1}{4}\big\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\big\|^2\\ - \frac{1}{4}\big\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\big\|^2 + 4\epsilon + 8g(\epsilon).\end{aligned}$$

Corollary 1 *Let $\Theta^{(t)} := \mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 + \frac{1}{2}\big\|\boldsymbol{w}^{(t-1)} - \boldsymbol{z}^{(t)}\big\|^2$. and $\zeta^{(t)} := \big\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\big\|^2 + \big\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\big\|^2$.*

Under the same assumptions as Theorem 7,

$$\Theta^{(t+1)} \le \Theta^{(t)} - \frac{1}{4}\zeta^{(t)} + 4\epsilon + 8g(\epsilon).$$

Proof We use Lemma 1 to first note the following for any $\boldsymbol{z} \in \mathcal{Z}$:

$$\eta\left\langle \boldsymbol{\ell}^{(t)} + \boldsymbol{m}^{(t)} - \boldsymbol{m}^{(t+1)}, \boldsymbol{z}^{(t+1)} - \boldsymbol{z} \right\rangle \leq \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}^{(t)}\right) + 2\epsilon.$$

Note that by definition, $\boldsymbol{\ell}^{(t)} = \boldsymbol{F}(\boldsymbol{z}^{(t)})$, and additionally by assumption $\boldsymbol{m}^{(t)} = \boldsymbol{\ell}^{(t-1)} = \boldsymbol{F}(\boldsymbol{z}^{(t-1)})$, so we can rewrite Equation (26):

$$\eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}), \boldsymbol{z}^{(t+1)} - \boldsymbol{z} \right\rangle \leq \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}^{(t)}\right) + 2\epsilon. \tag{27}$$

Now note that since $\langle \mathbf{A}\boldsymbol{x}, \boldsymbol{y} \rangle$ is convex with respect to $\boldsymbol{x}$ and concave with respect to $\boldsymbol{y}$, we have that $\eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}) - \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle \geq 0$ so this term can be added to the right side of Inequality (27) to yield the following:

$$\begin{aligned}
\mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}\right) &\leq \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}^{(t)}\right) - \eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}), \boldsymbol{z}^{(t+1)} - \boldsymbol{z} \right\rangle \\
&\qquad + \eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}) - \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle + 2\epsilon \\
&= \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}^{(t)}\right) + \eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}), \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle \\
&\qquad - \eta\left\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle + 2\epsilon \\
&= \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \,\|\, \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \,\|\, \boldsymbol{z}^{(t)}\right) + \eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}) - \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle \\
&\qquad + \eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle - \eta\left\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle + 2\epsilon.
\end{aligned} \tag{28}$$

First, we attempt to bound the first inner product that appears on the right-hand side of Inequality (28).

$$\begin{aligned}
\eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t)}) - \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle &\leq \eta\left\|\boldsymbol{F}(\boldsymbol{w}^{(t)}) - \boldsymbol{F}(\boldsymbol{w}^{(t-1)})\right\|\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\| && (29)\\
&\leq \eta\left\|\boldsymbol{w}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\| && (30)\\
&\leq \frac{1}{2}\eta\left(\left\|\boldsymbol{w}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2\right) && (31)\\
&\leq \frac{1}{2}\eta\left(2\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + 2\left\|\boldsymbol{z}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2\right). && (32)
\end{aligned}$$

Here we have used the Cauchy-Schwarz inequality in (29), smoothness of $\boldsymbol{F}$ with modulus 1 in (30), Young's inequality in (31), and the triangle inequality and again Young's inequality in Inequality (32).

Next, we try to bound the second inner product that appears on the right-hand side of Inequality (28).

We apply Lemma 2 to note that we have:

$$\begin{aligned}
\left\langle \boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t-1)} + \eta \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle &\leq 2g(\epsilon), \\
\left\langle \boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t-1)} + \eta \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t-1)} \right\rangle &\leq 2g(\epsilon).
\end{aligned}$$

Adding these two inequalities together we have:

$$\left\langle \boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t-1)} + \eta \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle \leq 4g(\epsilon).$$

It follows that

$$\begin{aligned}
\eta\left\langle \boldsymbol{F}(\boldsymbol{w}^{(t-1)}), \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)} \right\rangle &\leq \left\langle \boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t-1)}, \boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)} \right\rangle + 4g(\epsilon) \\
&= \left\langle \boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}, \boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)} \right\rangle + 4g(\epsilon) \\
&= \frac{1}{2}\left(\left\|\boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 - \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 - \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 \right) + 4g(\epsilon).
\end{aligned} \tag{33}$$

Combining Inequality (32) and Inequality (33) with Inequality (28), and noting that $\varphi(\boldsymbol{z}) = \frac{1}{2}\left\|\boldsymbol{z}\right\|^2$. we have

$$\begin{aligned}
\mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \parallel \boldsymbol{z}\right) &\leq \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \parallel \boldsymbol{z}\right) - \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t+1)} \parallel \boldsymbol{z}^{(t)}\right) - \eta\left\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle + 2\epsilon \\
&\quad + \frac{1}{2}\left(\left\|\boldsymbol{z}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 - \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 - \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 \right) + 4g(\epsilon) \\
&\quad + \frac{1}{2}\eta\left(2\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + 2\left\|\boldsymbol{z}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 \right) \\
&\leq \mathcal{D}_\varphi\left(\boldsymbol{z}^{(t)} \parallel \boldsymbol{z}\right) - \left(\frac{1}{2} - \eta\right)\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 - \left(\frac{1}{2} - \frac{1}{2}\eta\right)\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 \\
&\quad + \eta\left\|\boldsymbol{z}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2 - \eta\left\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle + 2\epsilon + g(4\epsilon) \\
&\leq \frac{1}{2}\left\|\boldsymbol{z}^{(t)} - \boldsymbol{z}\right\|^2 - \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 - \frac{3}{8}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 + \frac{1}{4}\left\|\boldsymbol{z}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2 \\
&\quad - \eta\left\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle + 2\epsilon + 4g(\epsilon).
\end{aligned}$$

Now if we set $\boldsymbol{z} = \Pi_{\mathcal{Z}^*}(\boldsymbol{z}^{(t)})$ above note that $-\eta\left\langle \boldsymbol{F}(\boldsymbol{z}), \boldsymbol{w}^{(t)} - \boldsymbol{z} \right\rangle \leq 0$ by convexity-concavity of $\left\langle \mathbf{A}\boldsymbol{x}, \boldsymbol{y} \right\rangle$ with respect to $\boldsymbol{x}$ and $\boldsymbol{y}$, and optimality of $\boldsymbol{z}$ ($\boldsymbol{z} \in \mathcal{Z}^*$). Additionally, we have that $\operatorname{dist}(\boldsymbol{z}^{(t+1)}, \mathcal{Z}^*)^2 \leq \operatorname{dist}(\boldsymbol{z}^{(t+1)}, \Pi_{\mathcal{Z}^*}(\boldsymbol{z}^{(t)}))^2$. Using these observations as well as multiplying both sides by 2 and adding $\frac{1}{2}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2$ to both sides:

$$\begin{aligned}
\operatorname{dist}(\boldsymbol{z}^{(t+1)}, \mathcal{Z}^*)^2 + \frac{1}{2}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 &\leq \left\|\boldsymbol{z}^{(t)} - \Pi_{\mathcal{Z}^*}(\boldsymbol{z})\right\|^2 - \frac{1}{2}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 - \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 \\
&\quad + \frac{1}{2}\left\|\boldsymbol{z}^{(t)} - \boldsymbol{w}^{(t-1)}\right\|^2 + 4\epsilon + 8g(\epsilon) \\
&= \operatorname{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 + \frac{1}{2}\left\|\boldsymbol{w}^{(t-1)} - \boldsymbol{z}^{(t)}\right\|^2 - \frac{1}{2}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 \\
&\quad - \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 + 4\epsilon + 8g(\epsilon) \\
&\leq \operatorname{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 + \frac{1}{2}\left\|\boldsymbol{w}^{(t-1)} - \boldsymbol{z}^{(t)}\right\|^2 - \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 \\
&\quad - \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 + 4\epsilon + 8g(\epsilon)
\end{aligned}$$

Next, we define $\Theta^{(t)} := \mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 + \frac{1}{2}\left\|\boldsymbol{w}^{(t-1)} - \boldsymbol{z}^{(t)}\right\|^2$. and $\zeta^{(t)} := \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2$. This allows us to write the above as

$$\Theta^{(t+1)} \le \Theta^{(t)} - \frac{1}{4}\zeta^{(t)} + 4\epsilon + 8g(\epsilon) \tag{34}$$

as desired.

■

Next, we prove a version of Lemma 4 of Wei et al. [58]:

Lemma 8 *Under the same assumptions as Theorem 7, for any $\boldsymbol{z}' \in \mathcal{Z}$ such that $\boldsymbol{z}' \neq \boldsymbol{z}_*^{(t+1)}$*

$$\frac{8}{25}\eta^2 \frac{\left[\left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}'\right\rangle\right]_+^2}{\left\|\boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)}\right\|^2} \le \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + 2\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 + 8\epsilon.$$

Proof By first-order optimality of $\boldsymbol{z}_*^{(t+1)}$ we have:

$$\left\langle \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t)} + \eta \boldsymbol{F}(\boldsymbol{w}^{(t)}), \boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)} \right\rangle \ge 0. \tag{35}$$

Rearranging Inequality (35), we have:

$$\begin{aligned}
\left\langle \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t)}, \boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)} \right\rangle &\ge \langle \eta \boldsymbol{F}(\boldsymbol{w}^{(t)}), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \rangle \\
&= \eta \left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\rangle - \eta \left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right) - \boldsymbol{F}(\boldsymbol{w}^{(t)}), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\rangle \\
&\ge \eta \left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\rangle - \eta \left\| \boldsymbol{F}(\boldsymbol{w}^{(t)}) - \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right) \right\| \left\| \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\| \quad (36) \\
&\ge \eta \left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\rangle - \eta \left\| \boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)} \right\| \left\| \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\| \quad (37) \\
&\ge \eta \left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\rangle - \frac{1}{4} \left\| \boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)} \right\| \left\| \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\|. \quad (38)
\end{aligned}$$

Here we have applied Cauchy-Schwarz in Inequality (36), smoothness of $\boldsymbol{F}$ with modulus 1 in Inequality (37), and the condition on η in Inequality (38).

Next, we apply Cauchy-Schwarz to upper bound the left-hand side and rearrange to obtain the following:

$$\left\| \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\| \left(\left\| \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t)} \right\| + \frac{1}{4} \left\| \boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)} \right\| \right) \ge \eta \left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}' \right\rangle.$$

Then squaring (and taking care in case the right-hand side is negative), we have:

$$\left(\left\|\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t)}\right\| + \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)}\right\|\right)^2 \geq \eta^2 \frac{\left[\left\langle \boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right), \boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}'\right\rangle\right]_+^2}{\left\|\boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)}\right\|^2}.$$

Now, note that

$$\begin{aligned}
\left(\left\|\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t)}\right\| + \frac{1}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)}\right\|\right)^2 &\leq \left(\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\| + \frac{5}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)}\right\|\right)^2 && (39)\\
&\leq \left(\frac{5}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\| + \frac{5}{4}\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)}\right\|\right)^2 \\
&\leq \frac{25}{8}\left(\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)}\right\|^2\right) && (40)
\end{aligned}$$

and so combining with the above, we have

$$\begin{aligned}
\frac{8}{25}\eta^2 \frac{\left[F\left(\boldsymbol{z}_*^{(t+1)}\right)^\top \left(\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}'\right)\right]_+^2}{\left\|\boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)}\right\|^2} &\leq \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}_*^{(t+1)}\right\|^2 \\
&\leq \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + 2\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 + 2\left\|\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t+1)}\right\|^2 && (41)\\
&\leq \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2 + 2\left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t+1)}\right\|^2 + 8\epsilon. && (42)
\end{aligned}$$

In Inequality (39), we use the triangle inequality, in Inequality (40), we use Young's inequality, in Inequality (41), we use the triangle inequality and Young's inequality, and in Inequality (42) we use Lemma 1. ■

Finally, we follow the Proof of Theorem 8 of Wei et al. [58] to prove our main result.

$$
\begin{aligned}
\zeta^{(t)} &\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{1}{4}\left(2\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \left\|\boldsymbol{w}^{(t)} - \boldsymbol{z}^{(t)}\right\|^2\right) \\
&\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{1}{4}\left(\sup_{z' \in \mathcal{Z}} \frac{8}{25}\eta^2 \frac{\left[\boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right)^\top \left(\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}'\right)\right]_+^2}{\left\|\boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)}\right\|^2} - 8\epsilon\right) && (43) \\
&= \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \sup_{z' \in \mathcal{Z}} \frac{2}{25}\eta^2 \frac{\left[\boldsymbol{F}\left(\boldsymbol{z}_*^{(t+1)}\right)^\top \left(\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}'\right)\right]_+^2}{\left\|\boldsymbol{z}' - \boldsymbol{z}_*^{(t+1)}\right\|^2} - 2\epsilon \\
&\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{2\eta^2\nu^2}{25}\left\|\boldsymbol{z}_*^{(t+1)} - \Pi_{\mathcal{Z}^*}\left(\boldsymbol{z}_*^{(t+1)}\right)\right\|^2 - 2\epsilon && (44) \\
&\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{2\eta^2\nu^2}{25}\left(\left\|\boldsymbol{z}^{(t+1)} - \Pi_{\mathcal{Z}^*}\left(\boldsymbol{z}_*^{(t+1)}\right)\right\| - \left\|\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t+1)}\right\|\right)^2 - 2\epsilon && (45) \\
&\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{2\eta^2\nu^2}{25}\left(\left\|\boldsymbol{z}^{(t+1)} - \Pi_{\mathcal{Z}^*}\left(\boldsymbol{z}^{(t+1)}\right)\right\| - \left\|\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t+1)}\right\|\right)^2 - 2\epsilon && (46) \\
&\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{\eta^2\nu^2}{25}\left(\left\|\boldsymbol{z}^{(t+1)} - \Pi_{\mathcal{Z}^*}\left(\boldsymbol{z}^{(t+1)}\right)\right\|^2 - 2\left\|\boldsymbol{z}_*^{(t+1)} - \boldsymbol{z}^{(t+1)}\right\|^2\right) - 2\epsilon && (47) \\
&\geq \frac{1}{2}\left\|\boldsymbol{z}^{(t+1)} - \boldsymbol{w}^{(t)}\right\|^2 + \frac{\eta^2\nu^2}{25}\left(\left\|\boldsymbol{z}^{(t+1)} - \Pi_{\mathcal{Z}^*}\left(\boldsymbol{z}^{(t+1)}\right)\right\|^2 - 8\epsilon\right) - 2\epsilon \\
&\geq C_2\Theta^{(t+1)} - C_1\epsilon.
\end{aligned}
$$

where $C_2 = \min(\frac{1}{2}, \frac{\eta^2\nu^2}{25})$ and $C_1 = 2\left(1 + \frac{4\eta^2\nu^2}{25}\right)$.

We use Lemma 8 in Inequality (43), the SP-MS condition in Inequality (44), the triangle inequality in Inequality (45), the definition of the projection operator in Inequality (46), and Young's inequality in Inequality (47).

By Corollary 1:

$$
\begin{aligned}
\Theta^{(t+1)} &\leq \Theta^{(t)} - \frac{1}{4}\zeta^{(t)} + 4\epsilon + 8g(\epsilon) \\
&\leq \Theta^{(t)} - \frac{1}{4}C_2\Theta^{(t+1)} + \left(\frac{1}{4}C_1 + 4\right)\epsilon + 8g(\epsilon).
\end{aligned}
$$

Rearranging, we have that

$$
\Theta^{(t+1)}\left(1 + \frac{1}{4}C_2\right) \leq \Theta^{(t)} + \left(\frac{1}{4}C_1 + 4\right)\epsilon + 8g(\epsilon).
$$

Then define $\Xi^{(t)} = \Theta^{(t)} - \frac{(16+C_1)\epsilon + 32g(\epsilon)}{C_2}$. We can then write the above as

$$\left(\Xi^{(t+1)} + \frac{(16+C_1)\epsilon + 32g(\epsilon)}{C_2}\right)\left(1+\frac{1}{4}C_2\right) \leq \Xi^{(t)} + \frac{(16+C_1)\epsilon + 32g(\epsilon)}{C_2} + \left(\frac{1}{4}C_1 + 4\right)\epsilon + 8g(\epsilon).$$

Rearranging we have

$$\left(1+\frac{C_2}{4}\right)\Xi^{(t+1)} \leq \Xi^{(t)}.$$

Thus we have

$$\Xi^{(t)} \leq \left(1+\frac{C_2}{4}\right)^{-t+1}\Xi^{(1)} \leq 2(1+\frac{C_2}{4})^{-t}\Xi^{(1)}.$$

Note by construction $\Xi^{(1)} \leq \Theta^{(1)} = \mathrm{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2$ and $\Xi^{(t)} \geq \mathrm{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2 - \frac{(16+C_1)\epsilon+32g(\epsilon)}{C_2}$ so from the above we have that

$$\begin{aligned}
\mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*)^2 - \frac{(16+C_1)\epsilon + 32g(\epsilon)}{C_2} &\leq \Xi^{(t)} \\
&\leq 2\left(1+\frac{C_2}{4}\right)^{-t}\Xi^{(1)} \\
&\leq 2\left(1+\frac{C_2}{4}\right)^{-t}\Theta^{(1)} \\
&= 2\left(1+\frac{C_2}{4}\right)^{-t}\mathrm{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z})^2
\end{aligned}$$

so that

$$\mathrm{dist}(\boldsymbol{z}^{(t)}, \mathcal{Z}^*) \leq 2\left(1+\frac{C_2}{4}\right)^{-t}\mathrm{dist}(\boldsymbol{z}^{(1)}, \mathcal{Z}^*)^2 + \frac{(16+C_1)\epsilon + 32g(\epsilon)}{C_2}$$

as desired (using the appropriate g based on whether the players are assumed to be employing the approximate-proximal-step-based framework presented in Section 3 or AFW-ROMD).

Next, we argue why the runtime is as stated when using AFW-ROMD. Note, that AFW for Player i can achieve a ϵ optimal solution with $O\left(L_i\frac{D_i^2}{\delta_i^2}\log\frac{L_iD_i}{\epsilon}\right)$ LMO calls. Given the convergence rate bound, note that we can achieve a $\frac{2C_2+48+C_1}{C_2}\epsilon$ solution we need at most $\frac{\log\frac{1}{\epsilon}}{\log\frac{4+C_2}{4}}$ iterations of the approximate-proximal-step-based framework presented in Section 3. It follows that we need $O\left(\frac{\log\frac{1}{\epsilon}}{\log\frac{4+C_2}{4}}L_i\frac{D_i^2}{\delta_i^2}\log\frac{L_iD_i}{\epsilon}\right)$ calls for Player i. If we let $\epsilon' = \frac{2C_2+48+C_1}{C_2}\epsilon$, then $\epsilon = \frac{C_2}{2C_2+48+C_1}\epsilon'$ so we can compute a ϵ'-NE in $O\left(\max_{i\in\{1,2\}}\frac{\log\frac{2C_2+48+C_1}{C_2\epsilon'}}{\log\frac{4+C_2}{4}}\frac{L_iD_i^2}{\delta_i^2}\log\frac{(2C_2+48+C_1)L_iD_i}{C_2\epsilon'}\right)$ calls.

Finally, the size of the support follows from the fact that AFW adds at most one pure strategy at every iteration, which means that at every iteration of our approximate method, AFW will return a strategy with $O\left(\max_{i\in\{1,2\}}L_i\frac{D_i^2}{\delta_i^2}\log\frac{L_iD_i}{\epsilon}\right) = O\left(\max_{i\in\{1,2\}}\frac{L_iD_i^2}{\delta_i^2}\log\frac{L_iD_i(2C_2+48+C_1)}{C_2\epsilon'}\right)$ in the support, and in particular this is yes for the last strategy returned by AFW-ROMD.

■

Appendix F. Game Descriptions

Two- and Three-Player Kuhn Poker Two-player Kuhn poker was originally proposed by Kuhn [40]. We employ the three-player variation described in Farina et al. [18]. In a three-player Kuhn poker game with rank r there are r possible cards. At the beginning of the game, each player pays one chip to the pot, and each player is dealt a single private card. The first player can check or bet, i.e., putting an additional chip in the pot. Then, the second player can check or bet after a first player's check or fold/call the first player's bet. If no bet was previously made, the third player can either check or bet. Otherwise, the player has to fold or call. After a bet from the second player (resp., third player), the first player (resp., the first and the second players) still has to decide whether to fold or call the bet. At the showdown, the player with the highest card who has not folded wins all the chips in the pot.

Two- and Three-Player Liar's Dice Liar's dice is another standard benchmark introduced by Lisỳ et al. [46]. At the beginning of the game, each of the players privately rolls an unbiased k-face die. Then, the players alternate in making (potentially no) claims about their toss. The first player begins bidding, announcing any face value up to k and the minimum number of dice that the player believes are showing that value among the dice of all the players. Then, each player has two choices during their turn: to make a higher bid or to challenge the previous bid by declaring the previous bidder a "liar". A bid is higher than the previous one if the face value is higher, or the number of dice is higher. If the current player challenges the previous bid, all dice are revealed. If the bid is valid, the last bidder wins and obtains a reward of $+1$ while the challenger obtains a negative payoff of -1. Otherwise, the challenger wins and gets reward $+1$, and the last bidder obtains reward of -1. All the other players obtain reward 0. We use parameter $k = 6$ in the two-player version (this is a standard value used in several papers) and $k = 3$ in the three-player version.

Two-Player Leduc Poker Leduc poker is another classic two-player benchmark game introduced by Southey et al. [52]. We employ game instances of rank three, in which the deck consists of three suits with three cards each. The maximum number of raises per betting round can be 1 or 2. As the game starts, players pay one chip to the pot. There are two betting rounds. In the first one, a single private card is dealt to each player, while in the second round, a single board card is revealed. The raise amount is set to 2 and 4 in the first and second rounds, respectively.

Three-Player Goofspiel This bidding game was originally introduced by Ross [51]. We use a 3-rank variant; that is, each player has a hand of cards with values $-1, 0, 1$. A third stack of cards with values $-1, 0, 1$ is shuffled and placed on the table. At each turn, a prize card is revealed, and each player privately chooses one of his or her cards to bid, with the highest card winning the current prize. In case of a tie, the prize is split evenly among the winners. After three turns, all the prizes have been dealt out, and the payoff of each player is computed as follows: each prize card's value is equal to its face value, and the players' scores are computed as the sum of the values of the prize cards they have won. For our experiments, we used the limited information variant [44]. In this variant, instead of the players revealing the cards they have chosen to play, the bid cards are submitted to a referee (who is fair and trusted by all the players), who simulates the gameplay as before (the highest card wins the prize, and in the case of a tie the prize is split evenly among the winners).

Appendix G. Pseudocode of Algorithms used in Experiments

In this section, we describe `AFW-OMD` and the algorithms we compare against in our experiments. In all our experiments, optimistic variants use the previous loss as the prediction: $\boldsymbol{m}^{(t)} = \boldsymbol{\ell}^{(t-1)}$. We drop the subscript i throughout most of this section because we take the point of view of a generic agent applying the algorithm, except for the description of averaging and restarting in Appendix G.5.

G.1. `AFW-OMD`

In Algorithm 2, we present a non-optimistic version of the approximate-proximal-step-based framework presented in Section 3. `AFW-OMD` is Algorithm 2 instantiated with `AFW` (Algorithm 1) as the APO.

Algorithm 2: `OMD` with Approximate Proximal Computations

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\varphi : \mathcal{X} \to \mathbb{R}_{\geq 0}$: L-smooth, 1-strongly convex
$\eta > 0$: step-size parameter
$\epsilon^{(t)}$: desired accuracy of prox call at each t
$\text{APO}_{\mathcal{X}}$: an APO for $\mathcal{X}$
$\boldsymbol{x}^{(0)} \in \mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \mathbf{0}$

function NEXTSTRATEGY()
  return $\text{APO}_{\mathcal{X}}(-\eta\langle \boldsymbol{\ell}^{(t-1)}, \cdot \rangle, \varphi, \boldsymbol{x}^{(t-1)}, \epsilon)$

G.2. `(O)FTPL`

In Algorithms 3 and 4, we present `FTPL` and `OFTPL`, as we implemented for our experiments. We used the Gumbel distribution to generate noise, with location 0, and scale η, since this corresponds to multiplicative weights using a stepsize of $\frac{1}{\eta}$ [1, 53].

Algorithm 3: `FTPL` [36]

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\text{LMO}_{\mathcal{X}}$: LMO for $\mathcal{X}$
$\eta > 0$: noise parameter
m: number of samples
$\boldsymbol{x}^{(0)} \in \mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \mathbf{0}$

function NEXTSTRATEGY()
  return $\text{LMO}_{\mathcal{X}}(\sum_{i=0}^{t-1} \boldsymbol{\ell}^{(i)} - \text{Gumbel}(0, \eta))$

Algorithm 4: Optimistic `FTPL` (`OFTPL`) [53]

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\mathrm{LMO}_{\mathcal{X}}$: LMO for $\mathcal{X}$
$\eta > 0$: noise parameter
m: number of samples
$\boldsymbol{x}^{(0)} \in \mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \boldsymbol{m}^{(0)} = \mathbf{0}$
function NEXTSTRATEGY($\boldsymbol{m}^{(t)}$)
  return $\mathrm{LMO}_{\mathcal{X}}(\sum_{i=0}^{t-1} \boldsymbol{\ell}^{(i)} + \boldsymbol{m}^{(t)} - \mathrm{Gumbel}(0, \eta))$

G.3. (O)FP

Next, in Algorithms 5 and 6, we present `FP` and `OFP`, as we implemented for our experiments. `OFP`, is an optimistic generalization of `FP`, based on the same idea used to generalize `FTPL` and `FTRL` to `OFTPL` and `OFTRL`. `FP` can be thought of as letting the regularization/perburbation term go to 0 in `FTRL` and `FTPL` (letting the stepsize go to infinity and the noise go to 0, respectively), and similarly with `OFP` and `OFTRL`/`OFTPL`.

Algorithm 5: Fictitious Play (`FP`) [11]

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\mathrm{LMO}_{\mathcal{X}}$: LMO for $\mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \mathbf{0}$
function NEXTSTRATEGY()
  return $\mathrm{LMO}_{\mathcal{X}}(\sum_{i=0}^{t-1} \boldsymbol{\ell}^{(i)})$

Algorithm 6: Optimistic Fictitious Play (`OFP`)

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\mathrm{LMO}_{\mathcal{X}}$: LMO for $\mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \boldsymbol{m}^{(0)} = \mathbf{0}$
function NEXTSTRATEGY($\boldsymbol{m}^{(t)}$)
  return $\mathrm{LMO}_{\mathcal{X}}(\sum_{i=0}^{t-1} \boldsymbol{\ell}^{(i)} + \boldsymbol{m}^{(t)})$

G.4. O(BR)

Finally, in Algorithms 7 and 8, we present `BR` and `OBR`, as we implemented for our experiments. `OBR`, is an optimistic generalization of `BR`, based on the same idea used to generalize `FTPL` and `FTRL` to `OFTPL` and `OFTRL`. `BR` and `OBR` can be thought of as letting the regularization term go to 0 in `OMD` and `ROMD` respectively (letting the stepsize go to infinity).

Algorithm 7: Best Response (BR)

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\text{LMO}_{\mathcal{X}}$: LMO for $\mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \mathbf{0}$
function NEXTSTRATEGY()
  return $\text{LMO}_{\mathcal{X}}(\boldsymbol{\ell}^{(t-1)})$

Algorithm 8: Optimistic Best Response (OBR)

Data: $\mathcal{X} \subseteq \mathbb{R}^n$: convex and compact set
$\text{LMO}_{\mathcal{X}}$: LMO for $\mathcal{X}$
$\boldsymbol{\ell}^{(0)} = \boldsymbol{m}^{(0)} = \mathbf{0}$
function NEXTSTRATEGY($\boldsymbol{m}^{(t)}$)
  return $\text{LMO}_{\mathcal{X}}(\boldsymbol{\ell}^{(t-1)} + \boldsymbol{m}^{(t)} - \boldsymbol{m}^{(t-1)})$

G.5. Averaging and Restarting Pseudocode

When using different averaging schemes, the duality gap is computed with respect to the average iterate (as defined by that averaging scheme).

When using uniform averaging, the weight placed on the new iterate is $f(t) = \frac{1}{t+1}$.

When using linear averaging, the weight placed on the new iterate is $f(t) = \frac{2}{t+2}$.

When using quadratic averaging, the weight placed on the new iterate is $f(t) = \frac{6t+6}{(t+2)(2t+3)}$.

When using last iterate, the weight placed on the new iterate is $f(t) = 1$.

The "average" iterate is then set as follows: $\bar{\boldsymbol{x}}_i^{(t+1)} = \bar{\boldsymbol{x}}_i^{(t)} + f(t)(\boldsymbol{x}_i^{(t+1)} - \bar{\boldsymbol{x}}_i^{(t)})$.

Algorithm 9: Adaptive Restarting of Algorithm

Data: T: number of iterations to run algorithm
$\bar{\boldsymbol{x}}_1^{(0)} = \boldsymbol{x}_1^{(0)} \in \mathcal{X}_1$, $\bar{\boldsymbol{x}}_2^{(0)} = \boldsymbol{x}_2^{(0)} \in \mathcal{X}_2$
Alg_i for $i \in \{1, 2\}$: algorithm which generates iterates for each of the players
$r = 0$
$\xi = \max_{\boldsymbol{x}_2 \in \mathcal{X}_2} \langle \mathbf{A}\bar{\boldsymbol{x}}_1^{(0)}, \boldsymbol{x}_2 \rangle - \min_{\boldsymbol{x}_1 \in \mathcal{X}_1} \langle \mathbf{A}\boldsymbol{x}_1, \bar{\boldsymbol{x}}_2^{(0)} \rangle$
for $t = 0, \ldots, T-1$ **do**
  $\boldsymbol{x}_1^{(t+1)} = \text{Alg}_1()$
  $\boldsymbol{x}_2^{(t+1)} = \text{Alg}_2()$
  $\bar{\boldsymbol{x}}_1^{(t+1)} = \bar{\boldsymbol{x}}_1^{(t)} + f(r)(\boldsymbol{x}_1^{(t+1)} - \bar{\boldsymbol{x}}_1^{(t)})$
  $\bar{\boldsymbol{x}}_2^{(t+1)} = \bar{\boldsymbol{x}}_2^{(t)} + f(r)(\boldsymbol{x}_2^{(t+1)} - \bar{\boldsymbol{x}}_2^{(t)})$
  if $\max_{\boldsymbol{x}_2 \in \mathcal{X}_2} \langle \mathbf{A}\bar{\boldsymbol{x}}_1^{(t+1)}, \boldsymbol{x}_2 \rangle - \min_{\boldsymbol{x}_1 \in \mathcal{X}_1} \langle \mathbf{A}\boldsymbol{x}_1, \bar{\boldsymbol{x}}_2^{(t+1)} \rangle \leq \frac{\xi}{2}$ **then**
    $\xi = \max_{\boldsymbol{x}_2 \in \mathcal{X}_2} \langle \mathbf{A}\bar{\boldsymbol{x}}_1^{(t+1)}, \boldsymbol{x}_2 \rangle - \min_{\boldsymbol{x}_1 \in \mathcal{X}_1} \langle \mathbf{A}\boldsymbol{x}_1, \bar{\boldsymbol{x}}_2^{(t+1)} \rangle$
    $r = 0$

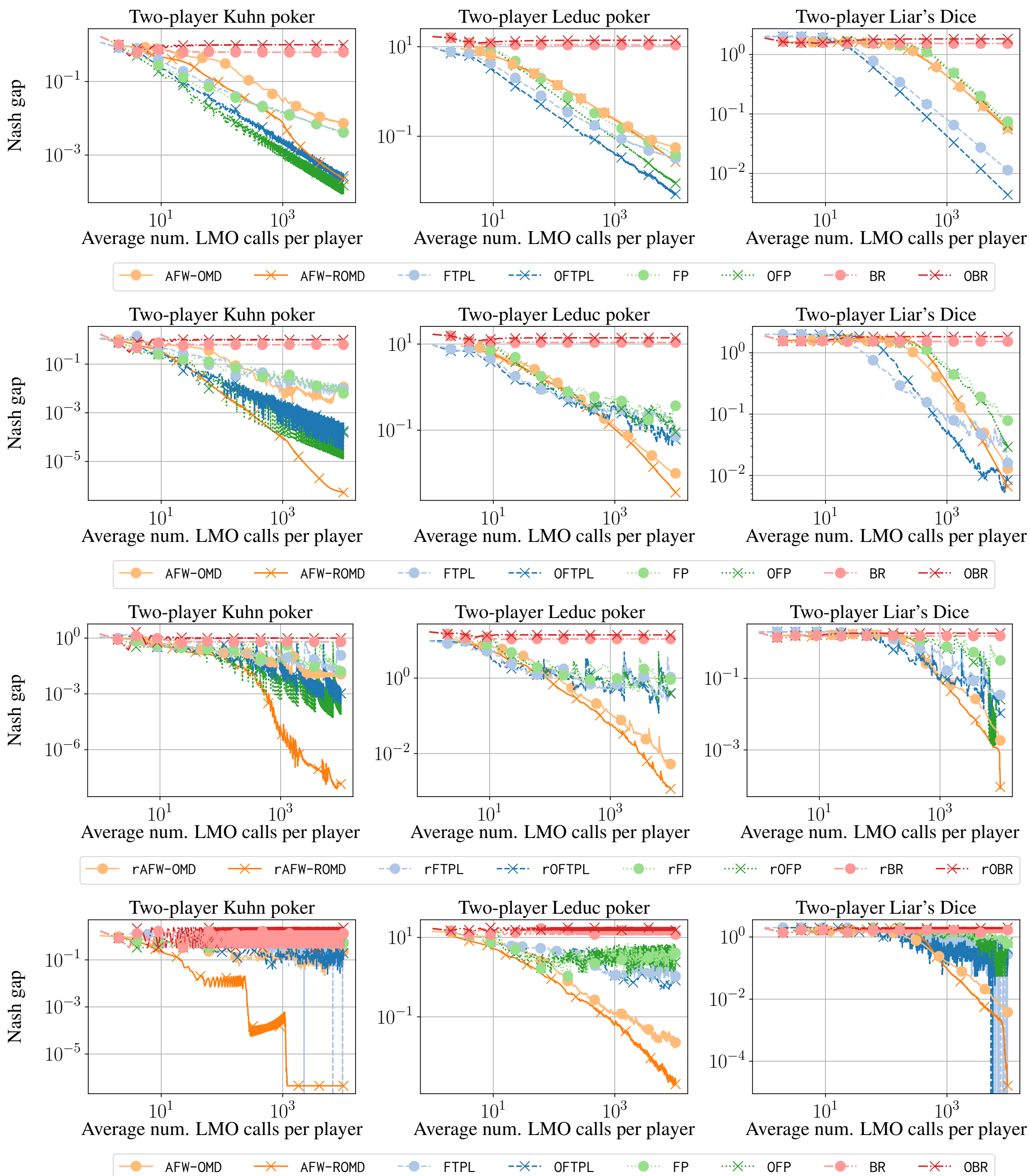

Figure 2: Convergence to NE as a function of average LMO calls per player for AFW-OMD, AFW-ROMD, FTPL, OFTPL, FP, OFP, BR, and OBR, for, from top to bottom, uniform, linear, quadratic, and last averaging, without using restarting.

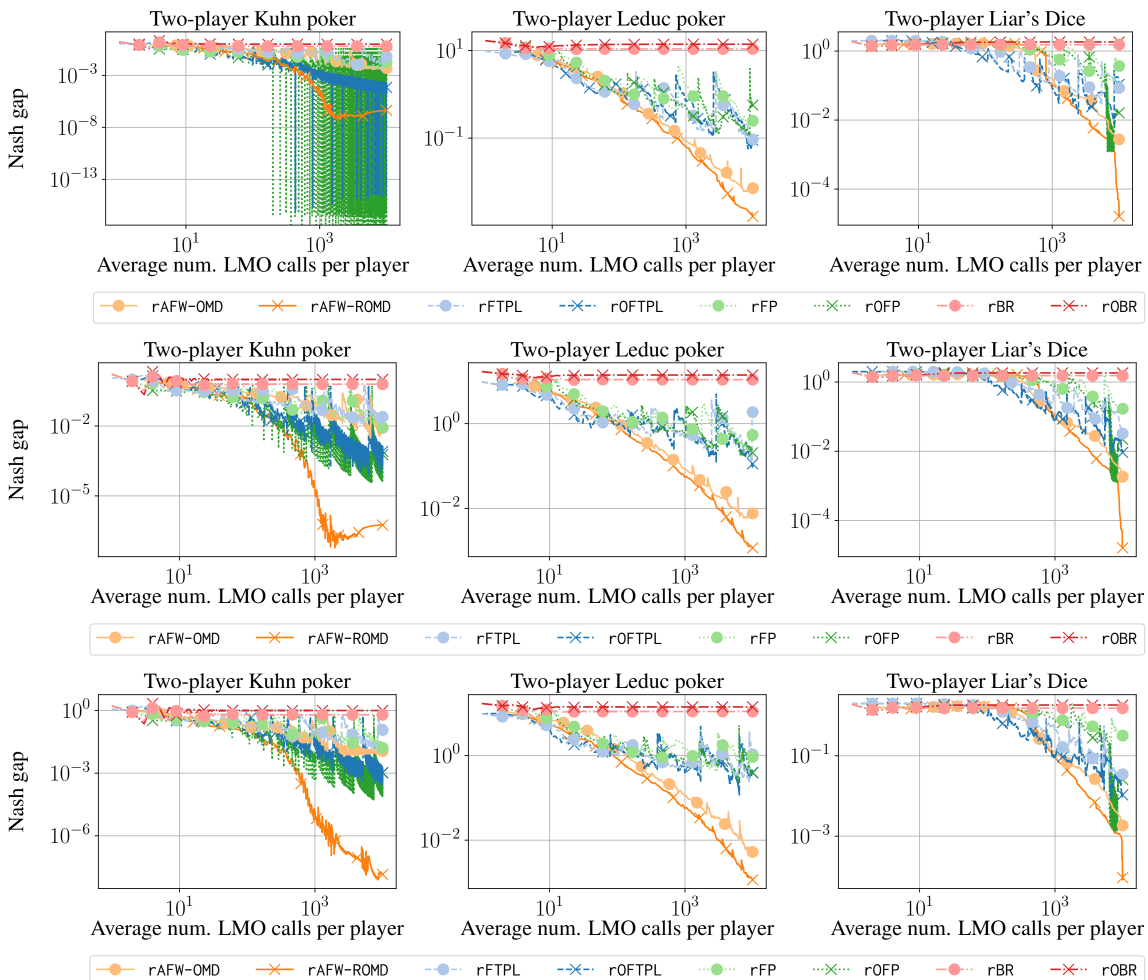

Figure 3: Convergence to NE as a function of average LMO calls per player for `AFW-OMD`, `AFW-ROMD`, `FTPL`, `OFTPL`, `FP`, `OFP`, `BR`, and `OBR`, for, from top to bottom, uniform, linear, and quadratic averaging, when using restarting.

Appendix H. Additional Experimental Details

H.1. Comparisons of Algorithms across Averaging and Restarting Schemes

In this section, we present plots of all the algorithms that we evaluate for different choices of averaging and restarting. This is only relevant for two-player games since our performance measure in the other settings is the maximum (uniform) average of an individual player's regret (so iterate averaging is irrelevant).

We implement adaptive restarting by resetting the averaging process every time the duality gap halves. Adaptive restarting was recently shown effective in practice for EFGs [15]. Since adaptive restarting is applied as a heuristic in our experiments that has not previously applied to the algorithms we present as well as the ones we compare against, we label restarted variants of algorithms with a prepended "r" (e.g., the adaptive restarting heuristic applied to `OFTPL` is labeled as `rOFTPL`) to distinguish them from the original presentation of the algorithm.

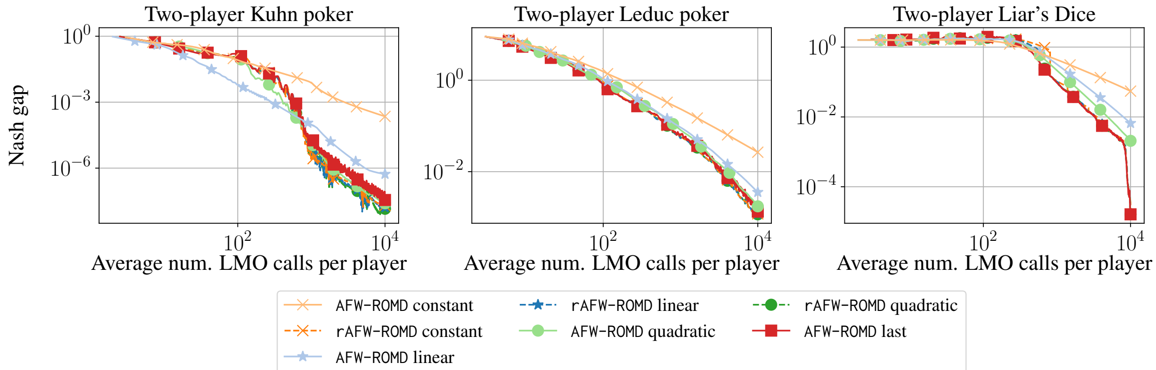

Figure 4: Convergence to NE as a function of average LMO calls per player for AFW-OMD for all combinations of averaging schemes and restarting.

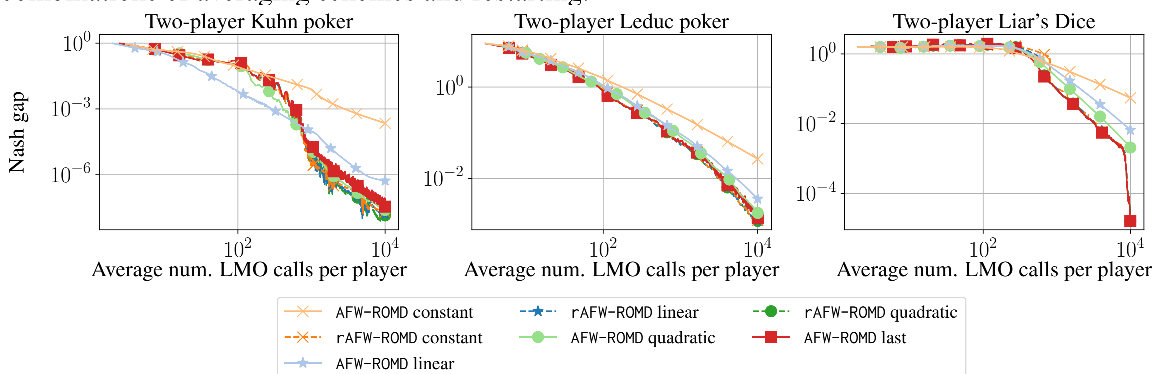

Figure 5: Convergence to NE as a function of average LMO calls per player for AFW-ROMD for all combinations of averaging schemes and restarting.

In Figure 2, we compare the algorithm on different choices of averaging for all algorithms when restarting is not used. In Figure 3, we compare the algorithm on different choices of averaging for all algorithms when restarting is used.

It can be seen that our algorithms are generally much more stable when averaging or restarting is applied relative to the other algorithms. This is to be expected because while our algorithm has last-iterate convergence guarantees, the other algorithms do not. Thus, even though (O)FTPL and (O)FP seem to perform well in the last-iterate case or with restarting on Kuhn and Liar's Dice, their behavior is extremely erratic. Observe that our algorithms generally outperform the other algorithms across different averaging and restarting schemes on Leduc and Liar's Dice. As mentioned before, (O)FTPL and (O)FP appear to be very unstable, and in Kuhn, while they are often able to find low duality gap solutions, they oscillate quite a bit.

H.2. Comparison of Averaging Schemes for AFW-OMD and AFW-ROMD

In Figures 4 and 5, we evaluate AFW-OMD and AFW-ROMD on two-player games on all combinations of averaging and restarting. As above, we label restarted variants of algorithms with a prepended "r" to distinguish them from the original presentation of the algorithm.

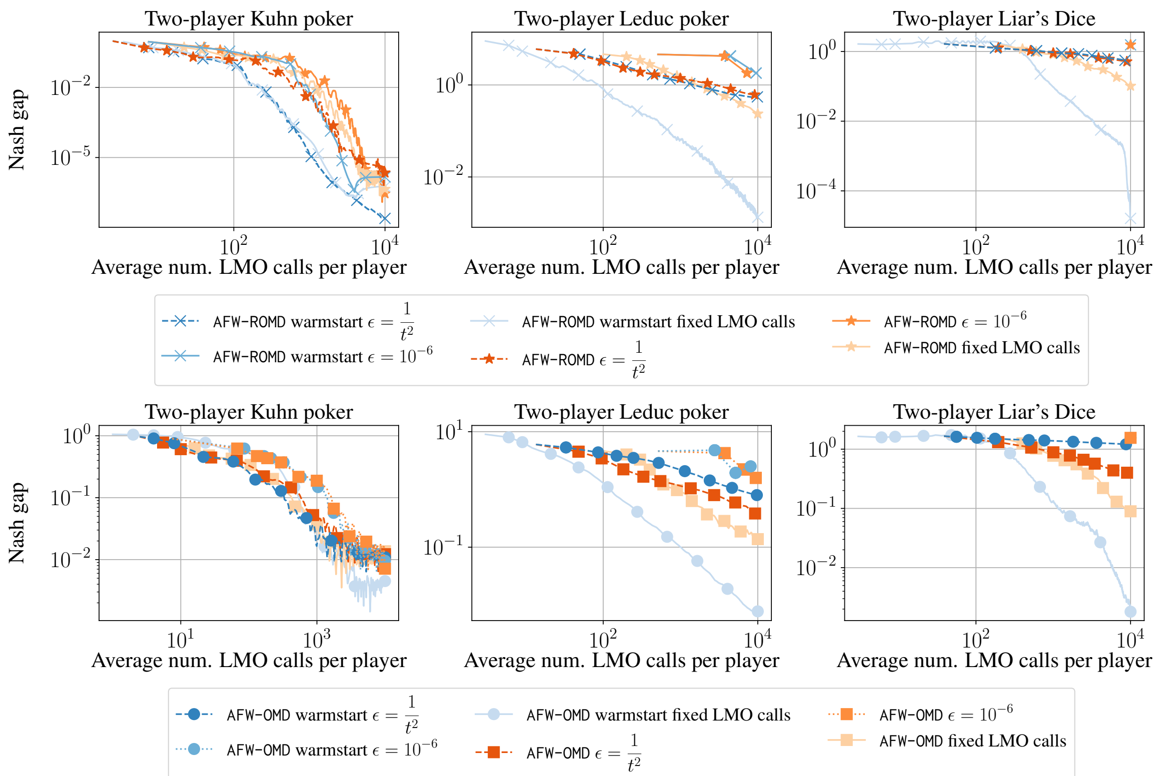

Figure 6: Convergence to NE as a function of average LMO calls per player for `AFW-ROMD` (top row) and `AFW-OMD` (bottom row) for different choices of warmstarting and termination criteria.

We observe here that for `AFW-OMD`, the restarted schemes (using uniform averaging, linear averaging, or quadratic averaging) are all very similar and generally perform best.

For `AFW-ROMD`, we observe that the restarted schemes and the last iterate perform quite similarly and generally better than the non-restarted schemes.

H.3. Comparison of Warmstart and APO Termination Criteria

We also test the performance of our algorithm when using different choices of termination criteria for the approximate prox call, and whether to warmstart. It can be seen in Figure 6, that in two-player Leduc poker and two-player Liar's Dice, using warm starting and a fixed number of LMO calls per iteration leads to the best performance. In the multiplayer setting, in Figure 7, it can be observed again that using warmstarting and a fixed number of LMO calls leads to the best performance of our algorithm in three-player Liar's Dice and three-player Goofspiel (3 ranks). For both two-player and three-player Kuhn, the choice of using warm starting and a fixed number of LMO calls is competitive.

H.4. Parameter Choices

In Tables 2 to 11, for each figure presented on experimental results, we provide the specific parameter choices used for each algorithm. The columns, in order, correspond to the game, the algorithm, the averaging scheme, whether restarting was used or not, the max number of LMO calls allowed during each iteration of the algorithm, the stepsize or noise (the latter in the case of `FTPL` and `OFTPL`, and

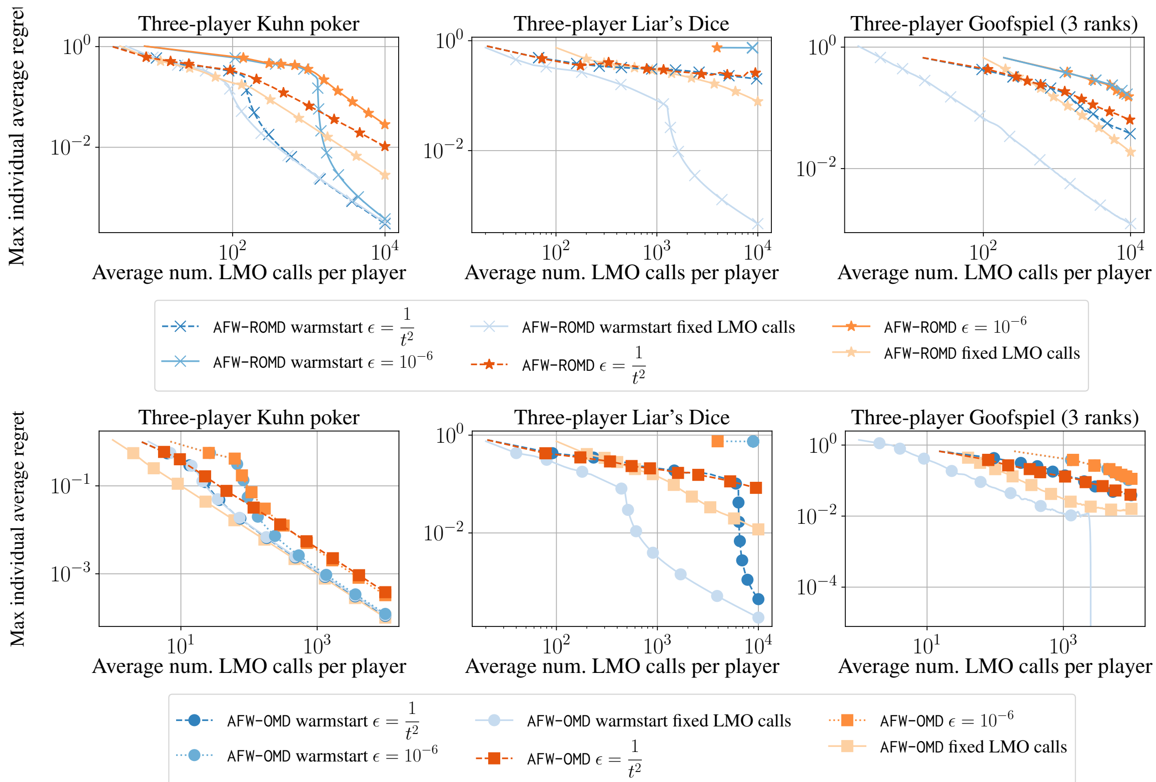

Figure 7: Convergence to CCE as a function of average LMO calls per player for `AFW-ROMD` (top row) and `AFW-OMD` (bottom row) for different choices of warm starting and termination criteria.

the former in all other cases), whether warmstarting was used (for our algorithms), and the APO termination criterion (for our algorithms).

Game	**Algorithm**	**Averaging**	**Restarts**	m	η	**Warmstart**	**APO term. crit.**
Two-player Kuhn poker	`AFW-OMD`	quadratic	no	1	0.08	yes	fixed LMO calls
	`AFW-ROMD`	quadratic	no	5	1.28	yes	fixed LMO calls
	`FTPL`	last	no	3	20.48	N/A	N/A
	`OFTPL`	last	no	3	20.48	N/A	N/A
	`FP`	constant	no	1	1.28	N/A	N/A
	`OFP`	linear	no	1	1.28	N/A	N/A
	`BR`	quadratic	no	1	1.28	N/A	N/A
	`OBR`	quadratic	no	1	1.28	N/A	N/A
Two-player Leduc poker	`AFW-OMD`	quadratic	no	3	1.28	yes	fixed LMO calls
	`AFW-ROMD`	last	no	2	1.28	yes	fixed LMO calls
	`FTPL`	constant	no	1	0.32	N/A	N/A
	`OFTPL`	constant	no	1	0.01	N/A	N/A
	`FP`	constant	no	1	1.28	N/A	N/A
	`OFP`	constant	no	1	1.28	N/A	N/A
	`BR`	quadratic	no	1	1.28	N/A	N/A

	OBR	linear	no	1	1.28	N/A	N/A
Two-player Liar's Dice	AFW-OMD	last	no	3	10.24	yes	fixed LMO calls
	AFW-ROMD	last	no	3	10.24	yes	fixed LMO calls
	FTPL	last	no	1	0.32	N/A	N/A
	OFTPL	last	no	1	0.08	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	linear	no	1	1.28	N/A	N/A
	BR	last	no	1	1.28	N/A	N/A
	OBR	last	no	1	1.28	N/A	N/A

Table 2: The parameters for all algorithms shown in Figure 1 for two-player games. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	**Algorithm**	**Averaging**	**Restarts**	m	η	**Warmstart**	**APO term. crit.**
Three-player Kuhn poker	AFW-OMD	constant	no	4	40.96	yes	fixed LMO calls
	AFW-ROMD	constant	no	5	1.28	yes	fixed LMO calls
	FTPL	constant	no	2	0.02	N/A	N/A
	OFTPL	constant	no	2	0.01	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	constant	no	1	1.28	N/A	N/A
	BR	constant	no	1	1.28	N/A	N/A
	OBR	constant	no	1	1.28	N/A	N/A
Three-player Liar's Dice	AFW-OMD	constant	no	20	40.96	yes	fixed LMO calls
	AFW-ROMD	constant	no	4	20.48	yes	fixed LMO calls
	FTPL	constant	no	1	0.01	N/A	N/A
	OFTPL	constant	no	1	0.01	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	constant	no	1	1.28	N/A	N/A
	BR	constant	no	1	1.28	N/A	N/A
	OBR	constant	no	1	1.28	N/A	N/A
Three-player Goofspiel (3 ranks)	AFW-OMD	constant	no	1	81.92	yes	fixed LMO calls
	AFW-ROMD	constant	no	2	40.96	yes	fixed LMO calls
	FTPL	constant	no	1	0.04	N/A	N/A
	OFTPL	constant	no	2	0.04	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	constant	no	1	1.28	N/A	N/A
	BR	constant	no	1	1.28	N/A	N/A
	OBR	constant	no	1	1.28	N/A	N/A

Table 3: The parameters for all algorithms shown in Figure 1 for multiplayer games. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	Algorithm	Averaging	Restarts	m	η	Warmstart	APO term. crit.
Two-player Kuhn poker	AFW-OMD	constant	no	3	0.32	yes	fixed LMO calls
	AFW-ROMD	constant	no	4	1.28	yes	fixed LMO calls
	FTPL	constant	no	1	0.04	N/A	N/A
	OFTPL	constant	no	1	0.08	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	constant	no	1	1.28	N/A	N/A
	BR	constant	no	1	1.28	N/A	N/A
	OBR	constant	no	1	1.28	N/A	N/A
Two-player Leduc poker	AFW-OMD	constant	no	3	2.56	yes	fixed LMO calls
	AFW-ROMD	constant	no	3	1.28	yes	fixed LMO calls
	FTPL	constant	no	1	0.32	N/A	N/A
	OFTPL	constant	no	1	0.01	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	constant	no	1	1.28	N/A	N/A
	BR	constant	no	1	1.28	N/A	N/A
	OBR	constant	no	1	1.28	N/A	N/A
Two-player Liar's Dice	AFW-OMD	constant	no	3	10.24	yes	fixed LMO calls
	AFW-ROMD	constant	no	2	5.12	yes	fixed LMO calls
	FTPL	constant	no	1	0.02	N/A	N/A
	OFTPL	constant	no	1	0.01	N/A	N/A
	FP	constant	no	1	1.28	N/A	N/A
	OFP	constant	no	1	1.28	N/A	N/A
	BR	constant	no	1	1.28	N/A	N/A
	OBR	constant	no	1	1.28	N/A	N/A
Two-player Kuhn poker	AFW-OMD	linear	no	1	0.32	yes	fixed LMO calls
	AFW-ROMD	linear	no	2	2.56	yes	fixed LMO calls
	FTPL	linear	no	1	0.64	N/A	N/A
	OFTPL	linear	no	1	0.04	N/A	N/A
	FP	linear	no	1	1.28	N/A	N/A
	OFP	linear	no	1	1.28	N/A	N/A
	BR	linear	no	1	1.28	N/A	N/A
	OBR	linear	no	1	1.28	N/A	N/A
Two-player Leduc poker	AFW-OMD	linear	no	3	1.28	yes	fixed LMO calls
	AFW-ROMD	linear	no	5	1.28	yes	fixed LMO calls
	FTPL	linear	no	1	0.32	N/A	N/A

	OFTPL	linear	no	1	0.08	N/A	N/A
	FP	linear	no	1	1.28	N/A	N/A
	OFP	linear	no	1	1.28	N/A	N/A
	BR	linear	no	1	1.28	N/A	N/A
	OBR	linear	no	1	1.28	N/A	N/A
Two-player Liar's Dice	AFW-OMD	linear	no	3	10.24	yes	fixed LMO calls
	AFW-ROMD	linear	no	2	10.24	yes	fixed LMO calls
	FTPL	linear	no	1	0.02	N/A	N/A
	OFTPL	linear	no	2	0.08	N/A	N/A
	FP	linear	no	1	1.28	N/A	N/A
	OFP	linear	no	1	1.28	N/A	N/A
	BR	linear	no	1	1.28	N/A	N/A
	OBR	linear	no	1	1.28	N/A	N/A
Two-player Kuhn poker	AFW-OMD	quadratic	no	1	0.08	yes	fixed LMO calls
	AFW-ROMD	quadratic	no	5	1.28	yes	fixed LMO calls
	FTPL	quadratic	no	1	0.32	N/A	N/A
	OFTPL	quadratic	no	1	0.04	N/A	N/A
	FP	quadratic	no	1	1.28	N/A	N/A
	OFP	quadratic	no	1	1.28	N/A	N/A
	BR	quadratic	no	1	1.28	N/A	N/A
	OBR	quadratic	no	1	1.28	N/A	N/A
Two-player Leduc poker	AFW-OMD	quadratic	no	3	1.28	yes	fixed LMO calls
	AFW-ROMD	quadratic	no	3	1.28	yes	fixed LMO calls
	FTPL	quadratic	no	1	0.32	N/A	N/A
	OFTPL	quadratic	no	1	0.16	N/A	N/A
	FP	quadratic	no	1	1.28	N/A	N/A
	OFP	quadratic	no	1	1.28	N/A	N/A
	BR	quadratic	no	1	1.28	N/A	N/A
	OBR	quadratic	no	1	1.28	N/A	N/A
Two-player Liar's Dice	AFW-OMD	quadratic	no	2	10.24	yes	fixed LMO calls
	AFW-ROMD	quadratic	no	2	10.24	yes	fixed LMO calls
	FTPL	quadratic	no	1	0.32	N/A	N/A
	OFTPL	quadratic	no	1	0.08	N/A	N/A
	FP	quadratic	no	1	1.28	N/A	N/A
	OFP	quadratic	no	1	1.28	N/A	N/A
	BR	quadratic	no	1	1.28	N/A	N/A
	OBR	quadratic	no	1	1.28	N/A	N/A
Two-player Kuhn poker	AFW-OMD	last	no	1	0.32	yes	fixed LMO calls
	AFW-ROMD	last	no	2	2.56	yes	fixed LMO calls
	FTPL	last	no	3	20.48	N/A	N/A
	OFTPL	last	no	20	0.32	N/A	N/A
	FP	last	no	1	1.28	N/A	N/A
	OFP	last	no	1	1.28	N/A	N/A

	BR	last	no	1	1.28	N/A	N/A
	OBR	last	no	1	1.28	N/A	N/A
Two-player Leduc poker	AFW-OMD	last	no	1	1.28	yes	fixed LMO calls
	AFW-ROMD	last	no	2	1.28	yes	fixed LMO calls
	FTPL	last	no	10	2.56	N/A	N/A
	OFTPL	last	no	200	0.16	N/A	N/A
	FP	last	no	1	1.28	N/A	N/A
	OFP	last	no	1	1.28	N/A	N/A
	BR	last	no	1	1.28	N/A	N/A
	OBR	last	no	1	1.28	N/A	N/A
Two-player Liar's Dice	AFW-OMD	last	no	4	5.12	yes	fixed LMO calls
	AFW-ROMD	last	no	3	10.24	yes	fixed LMO calls
	FTPL	last	no	1	0.32	N/A	N/A
	OFTPL	last	no	2	0.02	N/A	N/A
	FP	last	no	1	1.28	N/A	N/A
	OFP	last	no	1	1.28	N/A	N/A
	BR	last	no	1	1.28	N/A	N/A
	OBR	last	no	1	1.28	N/A	N/A

Table 4: The parameters for all algorithms shown in Figure 2. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	**Algorithm**	**Averaging**	**Restarts**	m	η	**Warmstart**	**APO term. crit.**
Two-player Kuhn poker	rAFW-OMD	constant	yes	4	0.32	yes	fixed LMO calls
	rAFW-ROMD	constant	yes	5	1.28	yes	fixed LMO calls
	rFTPL	constant	yes	1	5.12	N/A	N/A
	rOFTPL	constant	yes	1	0.01	N/A	N/A
	rFP	constant	yes	1	1.28	N/A	N/A
	rOFP	constant	yes	1	1.28	N/A	N/A
	rBR	constant	yes	1	1.28	N/A	N/A
	rOBR	constant	yes	1	1.28	N/A	N/A
Two-player Leduc poker	rAFW-OMD	constant	yes	3	1.28	yes	fixed LMO calls
	rAFW-ROMD	constant	yes	4	1.28	yes	fixed LMO calls
	rFTPL	constant	yes	1	0.64	N/A	N/A
	rOFTPL	constant	yes	2	0.16	N/A	N/A
	rFP	constant	yes	1	1.28	N/A	N/A
	rOFP	constant	yes	1	1.28	N/A	N/A
	rBR	constant	yes	1	1.28	N/A	N/A
	rOBR	constant	yes	1	1.28	N/A	N/A
Two-player Liar's Dice	rAFW-OMD	constant	yes	2	10.24	yes	fixed LMO calls

	`rAFW-ROMD`	constant	yes	3	10.24	yes	fixed LMO calls
	`rFTPL`	constant	yes	1	0.08	N/A	N/A
	`rOFTPL`	constant	yes	2	0.02	N/A	N/A
	`rFP`	constant	yes	1	1.28	N/A	N/A
	`rOFP`	constant	yes	1	1.28	N/A	N/A
	`rBR`	constant	yes	1	1.28	N/A	N/A
	`rOBR`	constant	yes	1	1.28	N/A	N/A
Two-player Kuhn poker	`rAFW-OMD`	linear	yes	4	0.32	yes	fixed LMO calls
	`rAFW-ROMD`	linear	yes	2	2.56	yes	fixed LMO calls
	`rFTPL`	linear	yes	1	0.08	N/A	N/A
	`rOFTPL`	linear	yes	2	0.04	N/A	N/A
	`rFP`	linear	yes	1	1.28	N/A	N/A
	`rOFP`	linear	yes	1	1.28	N/A	N/A
	`rBR`	linear	yes	1	1.28	N/A	N/A
	`rOBR`	linear	yes	1	1.28	N/A	N/A
Two-player Leduc poker	`rAFW-OMD`	linear	yes	3	1.28	yes	fixed LMO calls
	`rAFW-ROMD`	linear	yes	3	1.28	yes	fixed LMO calls
	`rFTPL`	linear	yes	1	0.32	N/A	N/A
	`rOFTPL`	linear	yes	1	0.01	N/A	N/A
	`rFP`	linear	yes	1	1.28	N/A	N/A
	`rOFP`	linear	yes	1	1.28	N/A	N/A
	`rBR`	linear	yes	1	1.28	N/A	N/A
	`rOBR`	linear	yes	1	1.28	N/A	N/A
Two-player Liar's Dice	`rAFW-OMD`	linear	yes	2	10.24	yes	fixed LMO calls
	`rAFW-ROMD`	linear	yes	3	10.24	yes	fixed LMO calls
	`rFTPL`	linear	yes	2	0.16	N/A	N/A
	`rOFTPL`	linear	yes	1	0.16	N/A	N/A
	`rFP`	linear	yes	1	1.28	N/A	N/A
	`rOFP`	linear	yes	1	1.28	N/A	N/A
	`rBR`	linear	yes	1	1.28	N/A	N/A
	`rOBR`	linear	yes	1	1.28	N/A	N/A
Two-player Kuhn poker	`rAFW-OMD`	quadratic	yes	3	0.64	yes	fixed LMO calls
	`rAFW-ROMD`	quadratic	yes	5	1.28	yes	fixed LMO calls
	`rFTPL`	quadratic	yes	1	0.08	N/A	N/A
	`rOFTPL`	quadratic	yes	3	0.04	N/A	N/A
	`rFP`	quadratic	yes	1	1.28	N/A	N/A
	`rOFP`	quadratic	yes	1	1.28	N/A	N/A
	`rBR`	quadratic	yes	1	1.28	N/A	N/A
	`rOBR`	quadratic	yes	1	1.28	N/A	N/A
Two-player Leduc poker	`rAFW-OMD`	quadratic	yes	2	1.28	yes	fixed LMO calls
	`rAFW-ROMD`	quadratic	yes	3	1.28	yes	fixed LMO calls
	`rFTPL`	quadratic	yes	1	0.32	N/A	N/A
	`rOFTPL`	quadratic	yes	1	0.32	N/A	N/A

	`rFP`	quadratic	yes	1	1.28	N/A	N/A
	`rOFP`	quadratic	yes	1	1.28	N/A	N/A
	`rBR`	quadratic	yes	1	1.28	N/A	N/A
	`rOBR`	quadratic	yes	1	1.28	N/A	N/A
Two-player Liar's Dice	`rAFW-OMD`	quadratic	yes	2	10.24	yes	fixed LMO calls
	`rAFW-ROMD`	quadratic	yes	2	10.24	yes	fixed LMO calls
	`rFTPL`	quadratic	yes	1	0.32	N/A	N/A
	`rOFTPL`	quadratic	yes	1	0.16	N/A	N/A
	`rFP`	quadratic	yes	1	1.28	N/A	N/A
	`rOFP`	quadratic	yes	1	1.28	N/A	N/A
	`rBR`	quadratic	yes	1	1.28	N/A	N/A
	`rOBR`	quadratic	yes	1	1.28	N/A	N/A

Table 5: The parameters for all algorithms shown in Figure 3. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	**Algorithm**	**Averaging**	**Restarts**	m	η	**Warmstart**	**APO term. crit.**
Two-player Kuhn poker	`AFW-OMD`	constant	no	1	0.32	yes	fixed LMO calls
	`rAFW-OMD`	constant	yes	1	0.32	yes	fixed LMO calls
	`AFW-OMD`	linear	no	1	0.08	yes	fixed LMO calls
	`rAFW-OMD`	linear	yes	1	0.32	yes	fixed LMO calls
	`AFW-OMD`	quadratic	no	5	0.32	yes	fixed LMO calls
	`rAFW-OMD`	quadratic	yes	1	0.32	yes	fixed LMO calls
	`AFW-OMD`	last	no	1	0.32	yes	fixed LMO calls
Two-player Leduc poker	`AFW-OMD`	constant	no	3	1.28	yes	fixed LMO calls
	`rAFW-OMD`	constant	yes	3	0.64	yes	fixed LMO calls
	`AFW-OMD`	linear	no	3	1.28	yes	fixed LMO calls
	`rAFW-OMD`	linear	yes	3	0.64	yes	fixed LMO calls
	`AFW-OMD`	quadratic	no	3	1.28	yes	fixed LMO calls
	`rAFW-OMD`	quadratic	yes	3	0.64	yes	fixed LMO calls
	`AFW-OMD`	last	no	4	0.64	yes	fixed LMO calls
Two-player Liar's Dice	`AFW-OMD`	constant	no	3	10.24	yes	fixed LMO calls
	`rAFW-OMD`	constant	yes	3	10.24	yes	fixed LMO calls
	`AFW-OMD`	linear	no	2	10.24	yes	fixed LMO calls
	`rAFW-OMD`	linear	yes	2	10.24	yes	fixed LMO calls
	`AFW-OMD`	quadratic	no	2	10.24	yes	fixed LMO calls
	`rAFW-OMD`	quadratic	yes	2	10.24	yes	fixed LMO calls
	`AFW-OMD`	last	no	3	10.24	yes	fixed LMO calls

Table 6: The parameters for all algorithms shown in Figure 4. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	Algorithm	Averaging	Restarts	m	η	Warmstart	APO term. crit.
Two-player Kuhn poker	`AFW-ROMD`	constant	no	4	1.28	yes	fixed LMO calls
	`rAFW-ROMD`	constant	yes	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
	`AFW-ROMD`	linear	no	2	2.56	yes	fixed LMO calls
	`rAFW-ROMD`	linear	yes	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
	`AFW-ROMD`	quadratic	no	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
	`rAFW-ROMD`	quadratic	yes	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
	`AFW-ROMD`	last	no	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
Two-player Leduc poker	`AFW-ROMD`	constant	no	3	1.28	yes	fixed LMO calls
	`rAFW-ROMD`	constant	yes	3	1.28	yes	fixed LMO calls
	`AFW-ROMD`	linear	no	3	1.28	yes	fixed LMO calls
	`rAFW-ROMD`	linear	yes	3	1.28	yes	fixed LMO calls
	`AFW-ROMD`	quadratic	no	5	1.28	yes	fixed LMO calls
	`rAFW-ROMD`	quadratic	yes	3	1.28	yes	fixed LMO calls
	`AFW-ROMD`	last	no	3	1.28	yes	fixed LMO calls
Two-player Liar's Dice	`AFW-ROMD`	constant	no	2	5.12	yes	fixed LMO calls
	`rAFW-ROMD`	constant	yes	3	10.24	yes	fixed LMO calls
	`AFW-ROMD`	linear	no	2	10.24	yes	fixed LMO calls
	`rAFW-ROMD`	linear	yes	3	10.24	yes	fixed LMO calls
	`AFW-ROMD`	quadratic	no	2	10.24	yes	fixed LMO calls
	`rAFW-ROMD`	quadratic	yes	3	10.24	yes	fixed LMO calls
	`AFW-ROMD`	last	no	3	10.24	yes	fixed LMO calls

Table 7: The parameters for all algorithms shown in Figure 5. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	Algorithm	Averaging	Restarts	m	η	Warmstart	APO term. crit.
Two-player Kuhn poker	`AFW-ROMD`	quadratic	no	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
	`AFW-ROMD`	quadratic	no	N/A	1.28	yes	$\epsilon = 10^{-6}$
	`AFW-ROMD`	quadratic	no	5	1.28	yes	fixed LMO calls
	`AFW-ROMD`	last	no	N/A	1.28	no	$\epsilon = \frac{1}{t^2}$
	`AFW-ROMD`	last	no	N/A	1.28	no	$\epsilon = 10^{-6}$
	`AFW-ROMD`	last	no	20	1.28	no	fixed LMO calls

Two-player Leduc poker	AFW-ROMD	last	no	N/A	0.64	yes	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	last	no	N/A	2.56	yes	$\epsilon = 10^{-6}$
	AFW-ROMD	last	no	3	1.28	yes	fixed LMO calls
	AFW-ROMD	last	no	N/A	0.64	no	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	last	no	N/A	2.56	no	$\epsilon = 10^{-6}$
	AFW-ROMD	quadratic	no	100	0.64	no	fixed LMO calls
Two-player Liar's Dice	AFW-ROMD	last	no	N/A	5.12	yes	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	quadratic	no	N/A	5.12	yes	$\epsilon = 10^{-6}$
	AFW-ROMD	last	no	3	10.24	yes	fixed LMO calls
	AFW-ROMD	last	no	N/A	5.12	no	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	quadratic	no	N/A	0.04	no	$\epsilon = 10^{-6}$
	AFW-ROMD	last	no	200	5.12	no	fixed LMO calls

Table 8: The parameters for AFW-ROMD in Figure 6. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	**Algorithm**	**Averaging**	**Restarts**	m	η	**Warmstart**	**APO term. crit.**
Two-player Kuhn poker	AFW-OMD	linear	no	N/A	0.64	yes	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	linear	no	N/A	0.64	yes	$\epsilon = 10^{-6}$
	AFW-OMD	linear	no	1	0.08	yes	fixed LMO calls
	AFW-OMD	linear	no	N/A	0.64	no	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	linear	no	N/A	0.64	no	$\epsilon = 10^{-6}$
	AFW-OMD	linear	no	10	0.64	no	fixed LMO calls
Two-player Leduc poker	AFW-OMD	quadratic	no	N/A	2.56	yes	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	last	no	N/A	10.24	yes	$\epsilon = 10^{-6}$
	AFW-OMD	quadratic	no	3	1.28	yes	fixed LMO calls
	AFW-OMD	quadratic	no	N/A	2.56	no	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	last	no	N/A	5.12	no	$\epsilon = 10^{-6}$
	AFW-OMD	quadratic	no	100	2.56	no	fixed LMO calls
Two-player Liar's Dice	AFW-OMD	linear	no	N/A	20.48	yes	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	last	no	N/A	0.64	yes	$\epsilon = 10^{-6}$
	AFW-OMD	last	no	3	10.24	yes	fixed LMO calls
	AFW-OMD	linear	no	N/A	20.48	no	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	linear	no	N/A	0.02	no	$\epsilon = 10^{-6}$
	AFW-OMD	quadratic	no	200	10.24	no	fixed LMO calls

Table 9: The parameters for AFW-OMD in Figure 6. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	Algorithm	Averaging	Restarts	m	η	Warmstart	APO term. crit.
Three-player Kuhn poker	AFW-ROMD	constant	no	N/A	1.28	yes	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	constant	no	N/A	1.28	yes	$\epsilon = 10^{-6}$
	AFW-ROMD	constant	no	5	1.28	yes	fixed LMO calls
	AFW-ROMD	constant	no	N/A	1.28	no	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	constant	no	N/A	1.28	no	$\epsilon = 10^{-6}$
	AFW-ROMD	constant	no	4	1.28	no	fixed LMO calls
Three-player Liar's Dice	AFW-ROMD	constant	no	N/A	5.12	yes	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	constant	no	N/A	0.02	yes	$\epsilon = 10^{-6}$
	AFW-ROMD	constant	no	20	10.24	yes	fixed LMO calls
	AFW-ROMD	constant	no	N/A	10.24	no	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	constant	no	N/A	20.48	no	$\epsilon = 10^{-6}$
	AFW-ROMD	constant	no	100	5.12	no	fixed LMO calls
Three-player Goofspiel (3 ranks)	AFW-ROMD	constant	no	N/A	20.48	yes	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	constant	no	N/A	40.96	yes	$\epsilon = 10^{-6}$
	AFW-ROMD	constant	no	2	40.96	yes	fixed LMO calls
	AFW-ROMD	constant	no	N/A	20.48	no	$\epsilon = \frac{1}{t^2}$
	AFW-ROMD	constant	no	N/A	40.96	no	$\epsilon = 10^{-6}$
	AFW-ROMD	constant	no	100	20.48	no	fixed LMO calls

Table 10: The parameters for AFW-ROMD in Figure 7. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.

Game	Algorithm	Averaging	Restarts	m	η	Warmstart	APO term. crit.
Three-player Kuhn poker	AFW-OMD	constant	no	N/A	40.96	yes	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	constant	no	N/A	40.96	yes	$\epsilon = 10^{-6}$
	AFW-OMD	constant	no	4	40.96	yes	fixed LMO calls
	AFW-OMD	constant	no	N/A	81.92	no	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	constant	no	N/A	40.96	no	$\epsilon = 10^{-6}$
	AFW-OMD	constant	no	1	40.96	no	fixed LMO calls
Three-player Liar's Dice	AFW-OMD	constant	no	N/A	40.96	yes	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	constant	no	N/A	0.04	yes	$\epsilon = 10^{-6}$
	AFW-OMD	constant	no	20	40.96	yes	fixed LMO calls
	AFW-OMD	constant	no	N/A	40.96	no	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	constant	no	N/A	0.02	no	$\epsilon = 10^{-6}$
	AFW-OMD	constant	no	100	40.96	no	fixed LMO calls
Three-player Goofspiel (3 ranks)	AFW-OMD	constant	no	N/A	40.96	yes	$\epsilon = \frac{1}{t^2}$
	AFW-OMD	constant	no	N/A	81.92	yes	$\epsilon = 10^{-6}$
	AFW-OMD	constant	no	1	81.92	yes	fixed LMO calls

AFW-OMD	constant	no	N/A	81.92	no	$\epsilon = \frac{1}{t^2}$
AFW-OMD	constant	no	N/A	81.92	no	$\epsilon = 10^{-6}$
AFW-OMD	constant	no	20	40.96	no	fixed LMO calls

Table 11: The parameters for AFW-OMD in Figure 7. The final two columns are irrelevant for algorithms that are not ours and thus are marked as N/A. Additionally, if the number of LMO calls is not fixed for our algorithms, then N/A is used in the maximum number of LMO calls per iteration (m) column.