arXiv:2311.10859v2 [quant-ph] 2 May 2025

A Quadratic Speedup in Finding Nash Equilibria of Quantum Zero-Sum Games

Francisca Vasconcelos[1], Emmanouil-Vasileios Vlatakis-Gkaragkounis[2], Panayotis Mertikopoulos[3], Georgios Piliouras[4], and Michael I. Jordan[5]

[1]UC Berkeley
[2]UC Berkeley, UW Madison
[3]Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, 38000 Grenoble, France
[4]Google DeepMind
[5]UC Berkeley, Inria Paris

Recent developments in domains such as non-local games, quantum interactive proofs, and quantum generative adversarial networks have renewed interest in quantum game theory and, specifically, quantum zero-sum games. Central to classical game theory is the efficient algorithmic computation of Nash equilibria, which represent optimal strategies for both players. In 2008, Jain and Watrous proposed the first classical algorithm for computing equilibria in quantum zero-sum games using the Matrix Multiplicative Weight Updates (MMWU) method to achieve a convergence rate of $\mathcal{O}(d/\epsilon^2)$ iterations to ϵ-Nash equilibria in the 4^d-dimensional spectraplex. In this work, we propose a hierarchy of quantum optimization algorithms that generalize MMWU via an extra-gradient mechanism. Notably, within this proposed hierarchy, we introduce the Optimistic Matrix Multiplicative Weights Update (OMMWU) algorithm and establish its average-iterate convergence complexity as $\mathcal{O}(d/\epsilon)$ iterations to ϵ-Nash equilibria. This quadratic speed-up relative to Jain and Watrous' original algorithm sets a new benchmark for computing ϵ-Nash equilibria in quantum zero-sum games.

Francisca Vasconcelos: francisca@berkeley.edu
Emmanouil-Vasileios Vlatakis-Gkaragkounis: vlatakis@wisc.edu

Contents

1 Introduction **3**
- 1.1 Motivation . . . 3
- 1.2 Prior Work . . . 4
- 1.3 Our Contributions and Methodology . . . 5
- 1.4 Paper Organization . . . 7

2 Preliminaries **10**
- 2.1 Quantum Zero-Sum Games . . . 10
- 2.2 Quantum Nash Equilibria . . . 11
- 2.3 Convexity, Smoothness, & Duality . . . 14
- 2.4 Bregman Divergence & Mirror Map . . . 16
- 2.5 Mirror & Proximal Steps . . . 18
  - 2.5.1 Mirror Step . . . 18
  - 2.5.2 Proximal Step . . . 20
  - 2.5.3 Regularizers . . . 21

3 Algorithms for Quantum Zero-Sum Games **22**
- 3.1 Matrix Dual Averaging (MDA) Method . . . 22
- 3.2 Matrix Mirror Prox (MMP) . . . 24
- 3.3 Optimistic Matrix Mirror-Prox (OMMP) Methods . . . 25

4 Convergence Analysis **26**
- 4.1 Monotonicity of $\mathcal{F}$ . . . 27
- 4.2 Error Decomposition . . . 27
- 4.3 Optimality Analysis of (A_t) . . . 28
- 4.4 Optimality Analysis of (C_t) . . . 28
- 4.5 An Upper Bound for (B_t) . . . 29
- 4.6 Average-Iterate Error Upper Bound . . . 30
- 4.7 Regularizer-Induced Dimensionality Dependence . . . 33

5 Experiments **34**

6 Discussion **36**

7 Acknowledgements **38**

A Complex Matrix Differentiation and Gradients **44**

B Mirror & Proximal Maps **45**

C Properties of Finite-Valued Zero-Sum Quantum Games **50**
- C.1 Properties of Spectraplexes . . . 50
- C.2 Properties of $\mathcal{F}$ . . . 51

D Convergence Analysis Proofs **56**

1 Introduction

1.1 Motivation

Ever since von Neumann's groundbreaking work in the 1920's [1], game theory has become a foundational pillar of modern mathematics, economics, and computer science. The central solution concept in non-cooperative game theory is that of a *Nash equilibrium*, i.e., a strategy of joint play in which no single player can benefit from a unilateral deviation [2]. Despite its mathematical significance, the computational perspective on general Nash equilibria is murkier, since even approximating such an equilibrium is a PPAD-complete problem for general classical games. On the other hand, Nash equilibria in two-player zero-sum games (i.e., where one player's gain is the other's loss) can be computed in polynomial time, and such games have found practical applications in a wide variety of domains, from resource allocation problems [3], to political strategy [4], and the training of machine learning models [5, 6, 7].

When the players' actions are the states of a quantum system, classical game theory no longer applies, and one must consider *quantum games*. These games constitute a natural framework for the study of the exchange and processing of quantum information, resulting in various useful applications to theoretical, computational, and cryptographic settings. The original interest in quantum games dates back to seminal work by Bell [8] and CSHS [9], explaining the Einstein-Podolsky-Rosen (EPR) Paradox. Since then, substantial work on quantum "non-local games" [10, 11, 12, 13, 14] has lead to novel methods for testing whether two or more spatially separated, non-communicating quantum devices can generate correlations not reproducible by any pair of classical devices—resulting in entanglement-based tests of quantum advantage. Much of this work culminated in establishing MIP*=RE [15], which, among many notable results, demonstrated that there exists an efficient reduction from the Halting Problem to deciding whether a two-player non-local game has entangled value at most 1 or $\frac{1}{2}$. Furthermore, it was recently proven that non-local games can be compiled into single-prover interactive games [16]. In general, quantum interactive proofs with competing provers can be modeled as competitive quantum refereed games [17, 18, 19] and multi-prover quantum interactive proofs [20, 21, 22, 23] can be modeled as cooperative quantum games. Finally, there has been substantial work on quantum coin-flipping [24, 25, 26, 27], a game model where two players directly exchange quantum information, studying how two parties with competing interests can carry out a fair coin flip across a quantum communication channel.

In this work, we consider the specific class of quantum *zero-sum* games, in which two players are in direct competition and do not share entanglement. One of the earliest works in quantum game theory [28] studied a "matching pennies"-type zero-sum game to prove that quantum strategies are at least as good as and, in some cases, can outperform classical strategies. Further work on quantum zero-sum games led to a proof that the quantum complexity class QRG(1), of problems having one-turned quantum refereed games, is contained in PSPACE [29]. Finally, in the context of quantum machine learning, there has been substantial recent interest in quantum generative adversarial networks (QGANs) [30, 31], for which the training of the competing generator and discriminator networks can be modeled as a quantum zero-sum game.

As in the classical setting, finding Nash equilibria of *general* quantum games is computationally prohibitive—PPAD-complete to be exact [32]. However, the situation is more favorable for quantum *zero-sum* games. As demonstrated by Jain and Watrous [29], Nash equilibria of d-qubit two-player quantum zero-sum games can be calculated to ϵ-accuracy in polynomial time and linear dependence on the number of qubits (logarithmic in the spec-

traplex dimensionality), i.e., $\mathcal{O}(d/\epsilon^2)$, even with payoff-based information on the players' side [33]. In view of this, equilibrium strategies in quantum zero-sum games can in principle be computed algorithmically, thus providing the required guarantees of implementability for proof schemes or cryptographic ciphers that rely on the computation of quantum Nash equilibria. Our goal is to pursue this objective further, bringing tools from online learning and computational learning theory to bear to improve the design of algorithms for the computation of approximate Nash equilibria in quantum zero-sum games.

1.2 Prior Work

Our work builds upon the work of Jain and Watrous [29] for non-interactive quantum zero-sum games. To provide some context, [29] leveraged feedback in the form of quantum channels (governed by superoperators) to introduce the Matrix Multiplicative Weights Update (MMWU) algorithm, a two-player variant of the matrix exponentiated gradient algorithm of Tsuda et al. [34], which is itself a special case of Mirror Descent [35, 36]. Jain, Piliouras and Sim [37] recently revisited the setting of MMWU in non-interactive quantum zero-sum games by studying its day-to-day behavior. They showed that a continuous-time analogue of MMWU, that they call *quantum replicator dynamics*, leads to cyclic behavior (formally Poincaré recurrence) in quantum zero-sum games with interior equilibrium as the quantum relative entropy between a fully mixed Nash equilibrium and the evolving state of the system is time-invariant. Beyond the setting of quantum zero-sum games, variants of the MMWU algorithm have also been used in many important applications such as proving QIP=PSPACE [38], solving SDPs [39], finding balanced separators [40], enhancing spectral sparsification [41], covariance optimization tasks [42, 43, 44], and matrix learning [45]. We refer the interested reader to the survey by Arora, Hazan, and Kale [46, Section 5], as well as the numerous follow-up works referring to it, for a more detailed treatment of the history and applications of MMWU.

For the quantum zero-sum games of interest in this work, despite recent work on no-regret learning dynamics for zero-sum and general quantum games [33, 47], to the best of our knowledge, the $\mathcal{O}(d/\epsilon^2)$ convergence rate of the MMWU algorithm is the state of the art for finding ϵ-approximate Nash equilibria. In this work, we ask:

Can we improve on the performance of MMWU in quantum zero-sum games?

More precisely, our paper's goal and main contribution is to provide an accelerated version of the MMWU algorithm which achieves ϵ-accuracy in linear, $O(d/\epsilon)$ time, instead of $O(d/\epsilon^2)$.

To achieve this quadratic speedup, our point of departure is the literature on classical *finite* games. The classical version of the MMWU algorithm—known, among other names, as the Multiplicative Weights Update (MWU) method [46]—similarly achieves an $\mathcal{O}_d(1/\epsilon^2)$ iteration complexity.[1] However, Nemirovski [48] and Auslender & Teboulle [49] showed that, by a suitable modification of the MWU method—now commonly known as the *mirror-prox* family of algorithms—it is possible to accelerate this rate to $\mathcal{O}_d(1/\epsilon)$, a rate which was recently shown by Ouyang and Xu [50] to be order-optimal in the setting of min-max convex-concave problems.

The main algorithmic insight of the mirror-prox template is the combination of an extra-gradient step in the spirit of Korpelevich [51] with an iterate-averaging mechanism à la Polyak & Juditsky [52]. Coupled with the smoothness of the underlying objective,

[1] Hereon, we will use the $\mathcal{O}_d(\cdot)$ notation to hide the dimensionality dependence of the iteration complexity. There will be explicit discussion of dimensionality dependence in the quantum setting in Section 4.7.

this combination yields a $\mathcal{O}(1/\epsilon)$ convergence rate, albeit with a possibly suboptimal dependence on the dimension of the problem. In the case of the simplex—which is the state space of classical, finite games—the dependence on the dimension can be improved dramatically by instantiating the mirror-prox with an entropic regularizer, which essentially boils down to a variant of MWU "with advice" [48]. The only notable drawback of the resulting algorithm is that it requires not one, but two oracle queries per iteration, which doubles the computation cost, and requires additional coordination / communication from the players' side. Building on an original idea by Popov [53], this "cost doubling" issue was mitigated by the so-called *optimistic mirror descent* (OMD) proposal of Raklhin and Sridharan [54], which optimistically reused past gradient information as a surrogate for the extra-gradient step of the mirror-prox algorithm. In this way, when applied to mixed extensions of classical, finite games, OMD ultimately achieves an order-optimal convergence rate in terms of both ϵ and d, all the while using a single oracle call per iteration. We summarize this hierarchy of optimization methods and the relevant trade-offs in Figure 1.

1.3 Our Contributions and Methodology

Our work focuses on developing an accelerated algorithm with similar, linear-time performance guarantees in the *quantum* setting. The main contributions of this work are thus twofold: *a*) we propose the *Optimistic Matrix Multiplicative Weights Update (OMMWU) algorithm* (Algorithm 1) for finding ϵ-Nash equilibria in quantum zero-sum games; and *b*) we show that OMMWU achieves the following iteration complexity guarantees.

Corollary 1 (OMMWU Iteration Complexity) *In a 4^d-dimensional spectraplex, OMMWU computes average-iterate ϵ-Nash equilibria in $\mathcal{O}(d/\epsilon)$ iterations.*

In order to achieve this result, we forge a link between quantum zero-sum games and established works in classical game theory and optimization. In addition to OMMWU—which surpasses the $\mathcal{O}(d/\epsilon^2)$ rate of MMWU with a single gradient call per iteration—we also propose a hierarchical family of algorithms and techniques that can be applied to various semidefinite programming problems (quantum or otherwise) and thus may be of independent interest.

Methodology. In terms of methodology, we move away from the *channel-based* view of quantum zero-sum games in [29] and instead adopt a *gradient-based* perspective, similar to prior work on learning in general quantum games [55, 56] (see also [47] connecting the same online-learning, regret-minimization perspective to new classes of quantum correlated equilibria). In this framework, Alice and Bob's feedback is characterized not by traditional quantum superoperators, but rather by mathematical gradient operators, which comprise the game's gradient feedback operator $\mathcal{F}$. We prove several important properties of $\mathcal{F}$, namely establishing Lipschitz continuity and monotonicity. In doing so, we conclude that solving a quantum zero-sum game to a desired accuracy essentially boils down to solving a smooth, semidefinite convex-concave problem with first-order oracle information.

Our approach is based on endowing the problem's feasible region—a product of spectraplexes—with an OMD template in the spirit of Rakhlin and Sridharan [54]. A difficulty that arises when trying to combine these elements is that the product structure of standard exponential weight algorithms does not carry over automatically to the non-commutative matrix variables that arise in the quantum setting. The key observation that allows us to retain the speed-up of a mirror prox method and the state-of-art dimensionality dependence is that the von Neumann entropy (which is the underlying regularizer of the MMWU algorithm)

Algorithm 1 Optimistic Matrix Multiplicative Weights Update (OMMWU)

Accuracy Parameter: ϵ
Regularization Function: $h : \{\mathcal{A}, \mathcal{B}\} \to \mathbb{R}$
Strong Convexity Parameter (of h): μ_h
Diameter (of $\mathcal{BD}_h$): $\mathcal{D}_h = \sup_{X,Y \in \mathcal{C}} \mathcal{BD}_h(X\|Y)$
Lipschitz Parameter (of $\mathcal{F}$): $\gamma_{\mathcal{F}}$
Proximal Map: $\mathrm{Prox}\Pi^{h,\eta}_{\mathcal{A},\mathcal{B}}(X, Y) := \arg\min_{C \in \mathcal{C}}\{\langle Y, C - X\rangle - \frac{1}{\eta}\mathcal{BD}_h(C||X)\}$.

$\eta \leftarrow \mu_h/(2\gamma_{\mathcal{F}})$ ▷ Step Size
$N \leftarrow \lceil \mathcal{D}_h/(\eta \cdot \epsilon)\rceil$ ▷ Number of Rounds
$(\alpha_0, \beta_0) \leftarrow \left(\frac{1}{2^n}\mathbb{1}_{\mathcal{A}}, \frac{1}{2^m}\mathbb{1}_{\mathcal{B}}\right)$ ▷ State Initialization
$(\hat{\alpha}_0, \hat{\beta}_0) \leftarrow (\alpha_0, \beta_0)$ ▷ Intermediate State Initialization

for $t \in [1, N-1]$ **do**

$\alpha_{t+1} \leftarrow \Lambda(\log \hat{\alpha}_t + \eta\mathcal{F}_\alpha(\beta_t))$ ▷ State Updates
$\beta_{t+1} \leftarrow \Lambda(\log \hat{\beta}_t + \eta\mathcal{F}_\beta(\alpha_t))$
$\hat{\alpha}_{t+1} \leftarrow \Lambda(\log \hat{\alpha}_t + \eta\mathcal{F}_\alpha(\beta_{t+1}))$ ▷ Momentum Updates
$\hat{\beta}_{t+1} \leftarrow \Lambda(\log \hat{\beta}_t + \eta\mathcal{F}_\beta(\alpha_{t+1}))$

end for

return $\left(\bar{\alpha} = \frac{1}{N}\sum_{t=0}^{N-1}\alpha_t, \bar{\beta} = \frac{1}{N}\sum_{t=0}^{N-1}\beta_t\right)$

can be encoded in a matrix exponentiation step, as per the original MMWU algorithm (see also [57] for generalizations of this connection for optimization over symmetric cones). This allows us to implement an optimistic update structure in the problem's dual space, where the primary arithmetic operation is ordinary addition—which is an abelian operation on the space of Hermitian matrices. Thus, drawing inspiration from a classical hierarchy of optimization methods, we propose a *design of quantum zero-sum game algorithms* (depicted in Figure 2) achieving an optimized convergence rate with a single gradient call per iteration.

Within this design, we demonstrate that the MMWU algorithm of [29] instantiates (via a von Neumann entropy regularizer) the *Matrix Dual Averaging (MDA)* method (Algorithm 4). Following a classical proof technique in the spirit of Ene and Nguyên [58], we also prove that another of the proposed algorithms in our hierarchy—the *Optimistic Matrix Mirror Prox (OMMP)* method (Algorithm 5)—yields a quadratic speedup relative to the convergence rate of MDA, while still requiring only one gradient call per iteration:

Theorem 1 (Main Result) *The OMMP method computes ϵ-Nash equilibria in $\mathcal{O}_d(1/\epsilon)$ steps.*

Thus, OMMP establishes a new benchmark for computing ϵ-Nash finite-valued quantum zero-sum games. It is by instantiating OMMP with a von Neumann entropy regularizer, that the OMMWU algorithm and $\mathcal{O}(d/\epsilon)$ iteration complexity is achieved.

Proof Insights. The core idea of an *optimistic* algorithm is to leverage the gradient from the previous iteration as a forecast for the subsequent iteration's gradient. This is predicated on the presumption that if the game's adversary employs a consistent (or stabilized) strategy, then the discrepancy between the gradients of successive iterations will be negligible. Therefore, the formulation of our algorithm is crafted by devising a parallel to such

methods, circumventing the limitations imposed by the non-commutativity of the matrix operator, which is a common hurdle in quantum problems. Upon establishing the algorithm in this manner, we demonstrate that the (quantum) superoperators, corresponding to the method's gradients, exhibit two significant characteristics: *monotonicity* and *Lipschitz continuity/smoothness*.

Monotonicity is a valuable property that allows us to: (i) transform any Nash equilibrium into a more manageable variational inequality problem, and (ii), akin to convexity in single-agent optimization, it guarantees a consistent progression towards the equilibrium.

Lipschitz continuity, in contrast, is the essential quality that accelerates the convergence rate from $O(1/\epsilon^2)$ to $O(1/\epsilon)$. To comprehend the significance of this attribute, let's view the situation through the lens of the minimization participant—named Bob, with his maximization counterpart being Alice. Bob finds himself involved in an online convex minimization game. In the absence of any predictive information about the forthcoming functions, which are influenced by Alice's choices, folklore results suggests that the optimal rate is $\Theta(1/\sqrt{T})$, or equivalently $\Theta(1/\epsilon^2)$ iterations. However, should Alice's strategy remain relatively static, Bob can anticipate that the upcoming minimization challenges will be similar, affording him a critical advantage for achieving a rate of $O(1/T)$, or $O(1/\epsilon)$.

This insight reveals that the algorithm's performance is deeply tied to the Lipschitz parameter of the gradient operator. Here, the concept of mirror proxies plays a vital role in the proof. Utilizing mirror steps, distinct regularizers enable either more assertive or more refined movements within the strategy space of the spectraplex for both parties. Contrary to the exponential dependency implied by a Frobenius norm, we show that the von Neumann entropy regularizer maintains a dimension-independent Lipschitz constant under a norm that is congruent with the specific needs of our problem.

Future Work. In addition to our novel methodology, we conclude with the following conjecture that aims to tighten our method's performance guarantees and is likely of interest to both the quantum and optimization communities:

Conjecture 1 *The rates of Theorem 1 are tight: specifically, there exists a quantum two-player, zero-sum game for which OMMP methods require $\Omega_d(1/\epsilon)$ iterations to compute ϵ-Nash equilibria.*

1.4 Paper Organization

The paper is divided as follows. Section 2 reviews preliminaries, such as the quantum zero-sum games setup and mathematical concepts necessary for our proofs. Section 3 presents the series of quantum optimization methods for computing approximate Nash equilibria of quantum zero-sum games, as depicted in Figure 2. It begins with the MMWU method of [29] and shows how the algorithm is an instantiation of the more general Matrix Dual Averaging (MDA) method. It then introduces the Matrix Mirror Prox (MMP) method, which achieves the optimal rate, but requires two gradients per iteration. The section concludes with our proposed OMMP algorithm, which requires only a single gradient call per iteration. Finally, Section 4 elaborates the convergence analysis of OMMP algorithm, proving that it achieves an $\mathcal{O}_d(1/\epsilon)$ rate. The paper concludes with a discussion of different implementations of the OMMP method, via different regularizers. We show that a von Neumann entropy regularizer, instantiating the OMMWU algorithm, achieves a better dimensionality dependence than the Frobenius regularizer. Thus, the OMMWU algorithm achieves the desired $\mathcal{O}(d/\epsilon)$ and a quadratic speedup relative to MMWU. For concision, we defer some details and proofs to the Appendix.

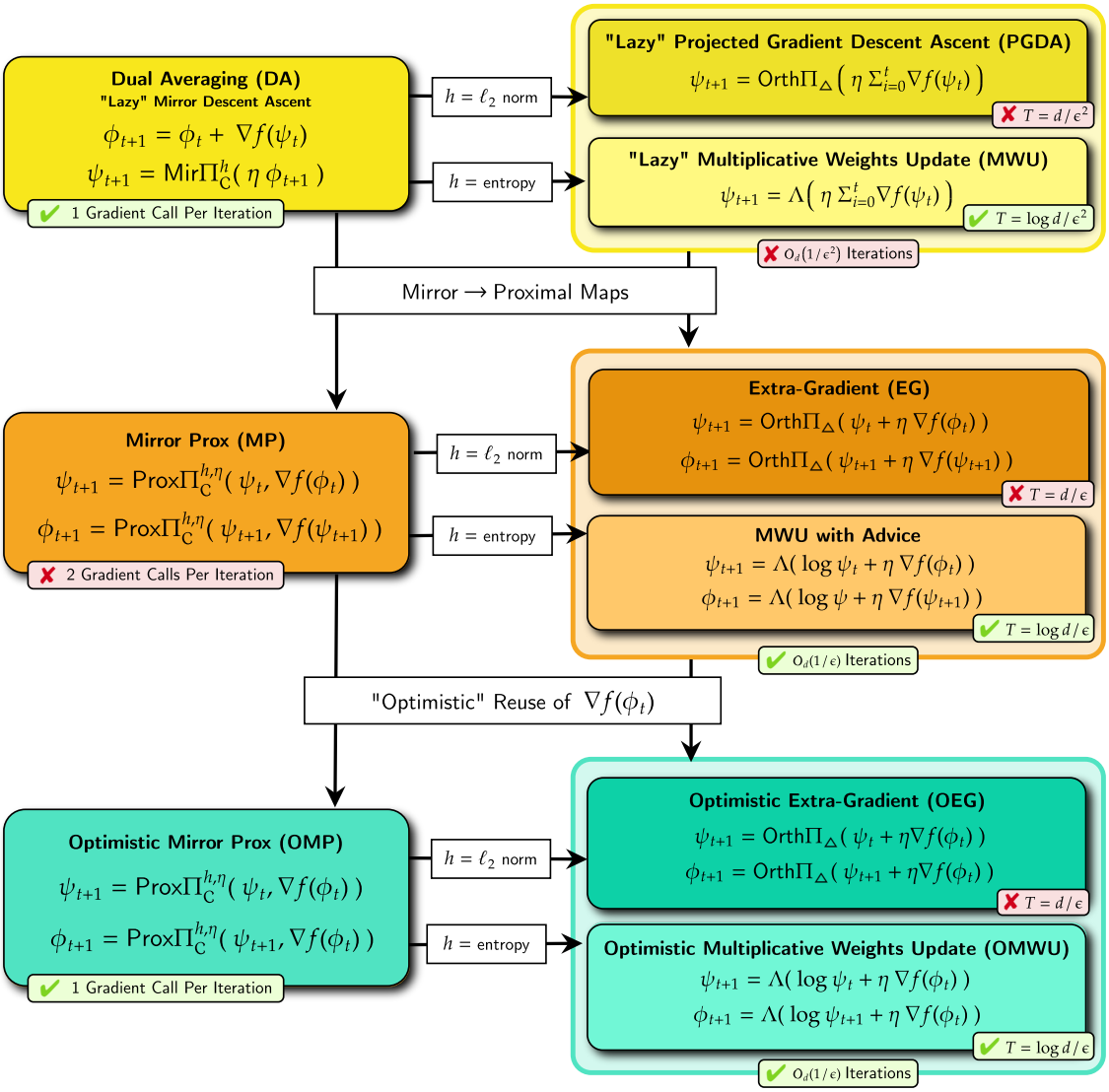


Figure 1: **Design of Classical Zero-Sum Game Algorithms.** This diagram provides the update rules for and relationships between different learning algorithms for classical zero-sum games. The left-hand side presents the Bregman generalized methods, parameterized by distance-generating/regularization function h. The right-hand side presents instantiations based on the ℓ_2 norm and entropy function, inducing an orthogonal projection $\mathrm{Orth}\Pi$ and logit map Λ, respectively. Note that moving from $h = \ell_2$ to $h =$entropy results in a logarithmic improvement in the total number of rounds T, by reducing the dependence on the simplex dimension d. Furthermore, moving from the mirror map $\mathrm{Mir}\Pi$ of the Dual Averaging method (a "lazy" variant of the classic Mirror Descent Ascent algorithm) to the proximal maps $\mathrm{Prox}\Pi$ of the Mirror Prox method results in a quadratic improvement in convergence, achieving the desired $O(1/\epsilon)$ rate. Furthermore, by reusing the "past gradient," the Single-Call Mirror Prox method retains the rate of Mirror Prox, but reduces the total number of gradient calls per iteration from two to one.

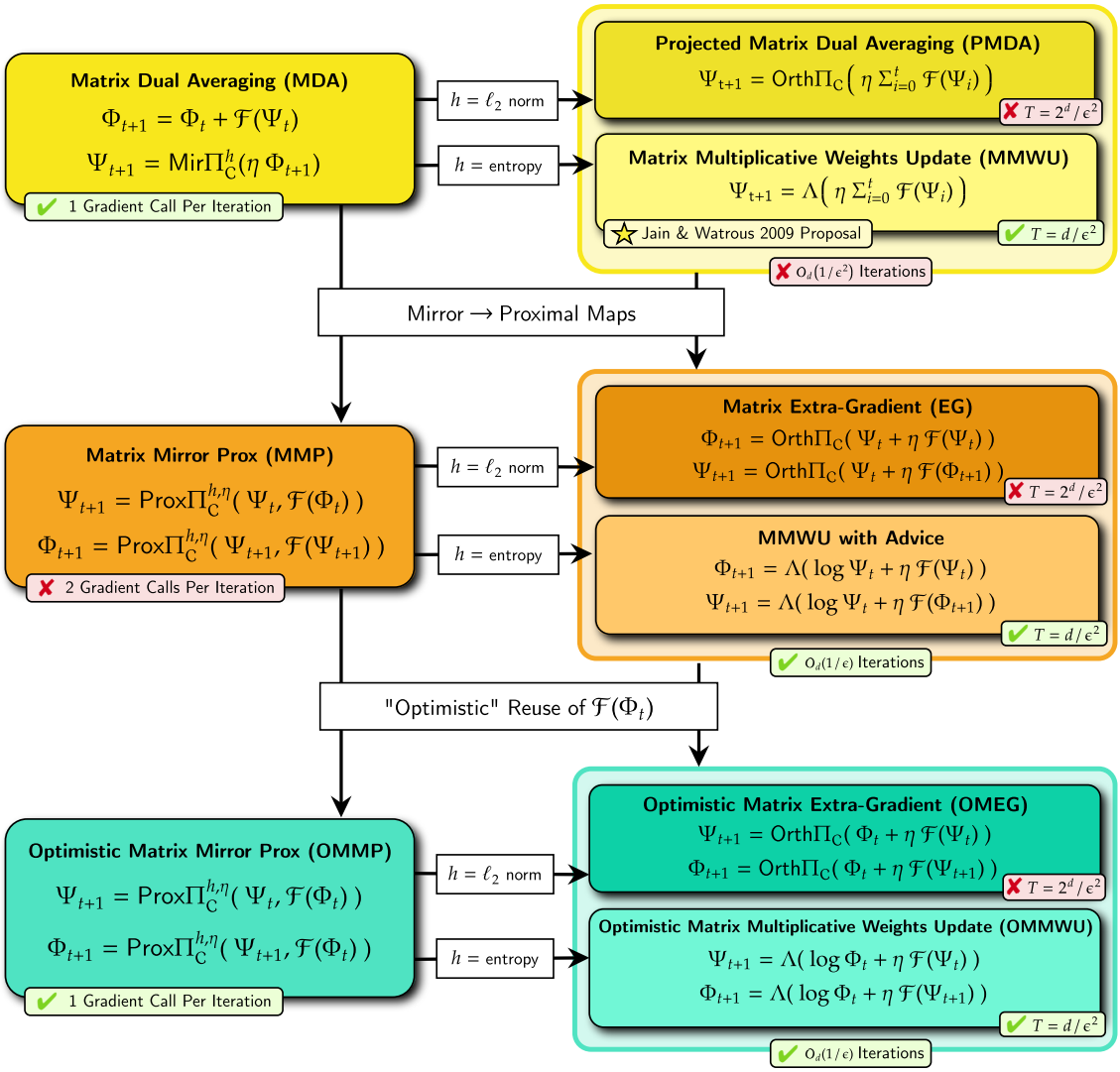


Figure 2: **Design of Quantum Zero-Sum Game Algorithms.** This diagram provides the update rules for and relationships between the learning algorithms for quantum zero-sum games as proposed in this work. The left-hand side presents the Bregman generalized methods, parameterized by distance-generating/regularization function h. The right-hand side presents instantiations based on the von Neumann entropy function and Frobenius (ℓ_2) norm, inducing an orthogonal projection $\mathrm{Orth}\Pi$ and logit map Λ, respectively. Note that moving from ℓ_2 to an entropy function as the regularizer results in a logarithmic improvement in the total number of rounds T, by reducing the dependence on the spectraplex dimension $D = 4^d$. Furthermore, moving from the mirror map $\mathrm{Mir}\Pi$ of the Matrix Dual Averaging method (Jain and Watrous' MMWU proposal [29]) to the proximal maps $\mathrm{Prox}\Pi$ of the Matrix Mirror Prox method results in a quadratic improvement in convergence, achieving the desired $O(1/\epsilon)$ rate. Furthermore, by reusing the "past gradient," the Single-Call Matrix Mirror Prox method retains the rate of Matrix Mirror Prox, but reduces the total number of gradient calls per iteration from two to one.

2 Preliminaries

2.1 Quantum Zero-Sum Games

In this work, we consider the restricted class of non-cooperative zero-sum quantum games. Within this setting, two competing players, Alice and Bob, each transmit a mixed quantum state to a referee (henceforth referred to as Roger), who performs a joint measurement on both states. Roger also possesses a utility function that attributes payoffs to the players based on the measurement outcomes. Since this is a zero-sum game, Bob's payoff will be the negative value of Alice's—e.g., if Alice wins \$5, Bob will lose \$5. We will now describe the game, as depicted in Figure 3, in more detail.

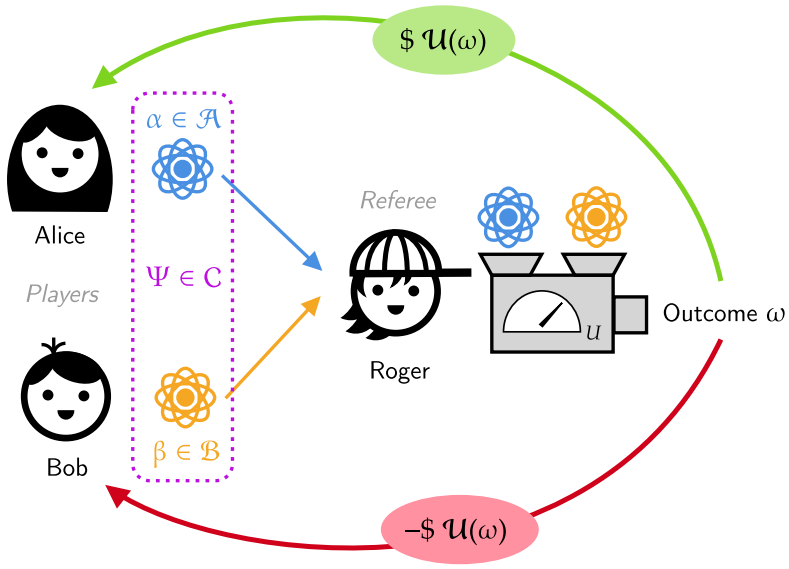


Figure 3: An illustration of a single round of the quantum zero-sum game.

Concretely, Alice and Bob, play the game by independently preparing multi-qubit quantum states α and β, respectively, to be sent to Roger. It is important to note that these are *mixed* states, corresponding to "mixed strategies," and that Alice and Bob do not share entanglement. Thus, Alice and Bob's set of possible moves is constrained to the set of density matrices lying in their respective *spectraplexes*. We will assume Alice prepares n-qubit states in her spectraplex,

$$\mathcal{A} = \{\alpha \in \mathcal{H}_+^n : \mathrm{Tr}(\alpha) = 1\}, \tag{1}$$

while Bob prepares m-qubit states in his spectraplex,

$$\mathcal{B} = \{\beta \in \mathcal{H}_+^m : \mathrm{Tr}(\beta) = 1\}, \tag{2}$$

where $\mathcal{H}_+^d$ denotes the set of $2^d \times 2^d$-dimensional, positive Hermitian matrices. In Appendix C.1, we show that all spectraplexes are compact and convex sets. Denote the joint set of their states as the direct sum

$$\Psi = (\alpha, \beta) \in \mathcal{C} = \mathcal{A} \oplus \mathcal{B}, \tag{3}$$

with tensor product denoted by $\Psi^{\otimes} = \alpha \otimes \beta$. As a general notational point, we will use lower-case variables (e.g., α or β) to refer to states in the individual spectraplexes of either

Alice ($\mathcal{A}$) or Bob ($\mathcal{B}$) and upper-case variables (e.g., Ψ or Φ) to denote joint states, which lie in the joint space ($\mathcal{C}$).

Alice and Bob send their states to Roger, who performs a joint measurement. This joint measurement has a finite number of possible measurement outcomes, Ω, defining an $(n+m)$-qubit *positive operator-valued measurement (POVM)*,

$$\{P_\omega\}_{\omega\in\Omega}, \text{ where } \sum_{\omega\in\Omega} P_\omega = \mathbb{1}. \tag{4}$$

Thus, Roger's probability of observing outcome $\omega \in \Omega$ is

$$p_\omega(\Psi) = \mathrm{Tr}\left(P_\omega^\dagger \Psi^{\otimes}\right). \tag{5}$$

Dependent on the measurement outcome, Roger gives Alice and Bob a reward/payoff. Let

$$\mathcal{U} : \Omega \to \mathbb{R} \tag{6}$$

denote Alice's *utility* function, which we will assume to be finite-valued and have image $[-1, 1]$ under normalization. We define Alice's *payoff observable* as

$$U = \sum_{\omega\in\Omega} \mathcal{U}(\omega) P_\omega, \tag{7}$$

such that Alice's *expected payoff* is

$$u_{\text{Alice}} := u(\Psi_t) = \mathrm{Tr}(U^\dagger \Psi^{\otimes}) = \sum_{\omega\in\Omega} \mathcal{U}(\omega)\mathrm{Tr}(P_\omega^\dagger \Psi^{\otimes}) = \sum_{\omega\in\Omega} \mathcal{U}(\omega) p_\omega(\Psi). \tag{8}$$

Since the game is zero-sum, this implies that Bob's utility is $-\mathcal{U}$, with payoff observable $-U$ and expected payoff $u_{\text{Bob}} := -u(\Psi)$. In playing the game, Alice and Bob aim to maximize their expected payoff, written as $\max_\alpha u(\alpha, \beta)$ and $\max_\beta u(\alpha, \beta)$, respectively. Since the game is zero-sum, in maximizing his expected payoff, Bob can be seen as trying to minimize Alice's payoff: $\max_\beta -u(\alpha, \beta) = \min_\beta u(\alpha, \beta)$.

Finally, since $u(\alpha, \beta)$ is a bilinear function and the spectraplexes are compact and convex sets, von Neumann's Min-Max Theorem [1] states

$$\max_\alpha \min_\beta u(\alpha, \beta) = \min_\beta \max_\alpha u(\alpha, \beta). \tag{9}$$

2.2 Quantum Nash Equilibria

A *Nash equilibrium* of the quantum game is a pair of states (α^*, β^*), such that each player has no incentive to change to a different state unilaterally:

$$u(\alpha^*, \beta^*) \geq u(\alpha, \beta^*), \ \ \forall \alpha \in \mathcal{A} \tag{10}$$

$$u(\alpha^*, \beta^*) \leq u(\alpha^*, \beta), \ \ \forall \beta \in \mathcal{B}. \tag{11}$$

Since $\mathcal{C}$ is convex and u is linear in α and β, the existence of Nash equilibria follows from Debreu's equilibrium existence theorem [59].

In playing the game, Alice and Bob receive feedback from Roger. In the original work on quantum zero-sum games, Jain and Watrous [29] define Alice's feedback as a quantum channel, given by the superoperator $\Xi : \mathcal{B} \to \mathcal{A}$, applied solely to Bob's state. The superoperator is parameterized by the payoff observable as

$$\Xi(\beta) = \mathrm{Tr}_{\mathcal{B}}[U(\mathbb{1}_{\mathcal{A}} \otimes \beta^\top)], \tag{12}$$

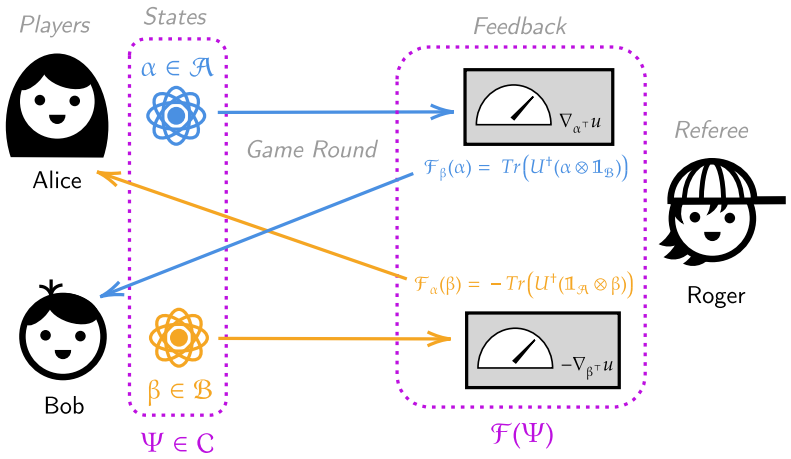


Figure 4: An illustration of the game feedback in a single round of the quantum zero-sum game (gradient-based view).

such that the expected utility can be expressed as

$$u(\alpha, \beta) = \mathrm{Tr}[U(\alpha \otimes \beta)] = \mathrm{Tr}\left[\alpha\, \mathrm{Tr}_\mathcal{B}[U(\mathbb{1}_\mathcal{A} \otimes \beta)]\right] = \mathrm{Tr}[\alpha\, \Xi(\beta^\top)]. \tag{13}$$

Similarly, Bob's feedback is defined by the quantum channel, corresponding to adjoint superoperator $\Xi^* : \mathcal{A} \to \mathcal{B}$, applied solely to Alice's state. This adjoint superoperator is defined as

$$\Xi^*(\alpha) = \mathrm{Tr}_\mathcal{A}[U(\alpha^\top \otimes \mathbb{1}_\mathcal{B})], \tag{14}$$

and is uniquely determined by the condition

$$u(\alpha, \beta) = \mathrm{Tr}[\alpha\, \Xi(\beta^\top)] = \mathrm{Tr}[\Xi^*(\alpha^\top)\, \beta]. \tag{15}$$

In this work, we will move away from thinking about Roger's feedback in terms of quantum channels towards a gradient-based view, as leveraged in recent work on learning general quantum games [55]. It is mathematically equivalent to think of the channel-based feedback in terms of gradients of the expected utility with respect to Alice and Bob's states. For a review of complex differentiation and gradients, refer to Appendix A. Namely, we define Alice's feedback, or *payoff gradient*, as

$$\mathcal{F}_\alpha(\beta) = \nabla_{\alpha^\top} u(\alpha, \beta) = \nabla_{\alpha^\top} \mathrm{Tr}\left[U^\dagger(\alpha \otimes \beta)\right] = \mathrm{Tr}_\mathcal{B}\left[U^\dagger(\mathbb{1}_n \otimes \beta)\right], \tag{16}$$

which is equivalent to $\Xi(\beta^\top)$ of Equation (12). Similarly Bob's feedback is defined as the payoff gradient

$$\mathcal{F}_\beta(\alpha) = -\nabla_{\beta^\top} u(\alpha, \beta) = -\nabla_{\beta^\top} \mathrm{Tr}\left[U^\dagger(\alpha \otimes \beta)\right] = -\mathrm{Tr}_\mathcal{A}\left[U^\dagger(\alpha \otimes \mathbb{1}_m)\right], \tag{17}$$

which is equivalent to $-\Xi^*(\alpha^\top)$ of Equation (14). Similarly to Equation (15), we have that

$$u(\alpha, \beta) = \mathrm{Tr}\left[\alpha\, \mathcal{F}_\alpha(\beta)\right] = -\mathrm{Tr}\left[\beta\, \mathcal{F}_\beta(\alpha)\right]. \tag{18}$$

Under the gradient-based view, $\mathrm{Tr}[\alpha\, \mathcal{F}_\alpha(\beta)] = \mathrm{Tr}[\alpha\, \nabla_{\alpha^\top} u(\alpha, \beta)]$ can be interpreted as the directional derivative of u, as a function of Alice's state, in the direction of Alice's state. Similarly, $-\mathrm{Tr}[\mathcal{F}_\beta(\alpha)\, \beta] = \mathrm{Tr}[\nabla_{\beta^\top} u(\alpha, \beta)\, \beta]$ can be interpreted as the directional derivative of u, as a function of Bob's state, in the direction of Bob's state. As will be demonstrated

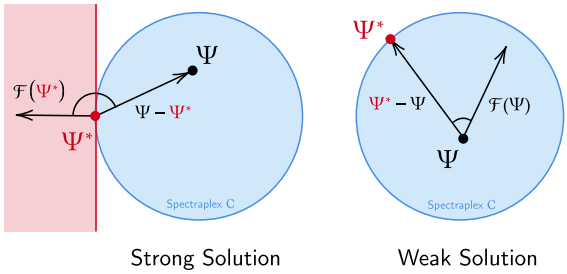


Figure 5: Intuitive illustrations of the strong and weak solutions Ψ^* of the variational inequalities given in Equation (20) and Equation (21), respectively. For strong solutions, the inner product between $\Psi - \Psi^*$ and $\mathcal{F}(\Psi^*)$ must be non-positive for all Ψ in the joint spectraplex $\mathcal{C}$. Meanwhile, for weak solutions, the inner product between $\Psi^* - \Psi$ and $\mathcal{F}(\Psi)$ must be non-negative for all $\Psi \in \mathcal{C}$.

throughout the work, moving away from a channel-based to this gradient-based view of the quantum zero-sum game feedback proves invaluable for mapping gradient-based insights from the classical games and optimization literature directly to the quantum setting.

Henceforth, we will denote the joint state as $\Psi = (\alpha, \beta)$, the joint feedback of both players, as follows:[2]

$$\mathcal{F}(\Psi) = \Big(\mathcal{F}_\alpha(\beta), \mathcal{F}_\beta(\alpha)\Big), \tag{19}$$

and the game's Nash equilibrium as $\Psi^* = (\alpha^*, \beta^*)$. Thus, as depicted in Figure 4, in each round t of the game, after playing the joint state Ψ_t, Alice will receive the individual payoff gradient $\mathcal{F}_{\alpha_t}(\beta_t)$ and Bob will receive $\mathcal{F}_{\beta_t}(\alpha_t)$, comprising joint feedback $\mathcal{F}(\Psi_t)$.

By using standard arguments [60], it can be shown that the solutions of the variational inequality

$$\operatorname{Tr}\left[(\Psi - \Psi^*)\,\mathcal{F}(\Psi^*)\right] \leq 0, \quad \forall\, \Psi \in \mathcal{C}, \tag{20}$$

are the game's Nash equilibria, Ψ^*.[3] Specifically, solutions satisfying Equation (20) are known as *strong solutions* to the variational inequality. Alternatively, *weak solutions* to the variational inequality are $\Psi^* \in \mathcal{C}$ satisfying

$$\operatorname{Tr}\left[(\Psi^* - \Psi)\,\mathcal{F}(\Psi)\right] \geq 0, \quad \forall\, \Psi \in \mathcal{C}. \tag{21}$$

We offer intuitive visualizations of both the weak and strong solutions of variational inequalities as vector inner products in Figure 5.

As we will discuss in Section 4 and prove in Appendix C.2, $\mathcal{F}$ is both monotone (Lemma 8) and Lipschitz-continuous (Lemma 9). Additionally, in Appendix C, we prove the following two important lemmas:

Lemma 1 *If $\mathcal{F}$ is monotone, then a strong solution is a weak solution.*

Lemma 2 *If $\mathcal{F}$ is Lipschitz continuous, then a weak solution is a strong solution.*

Therefore, by the monotonicity and Lipschitzness of $\mathcal{F}$, the strong and weak solutions (described, respectively, by Equation (20) and Equation (21)) are equally valid criteria for

[2] In a slight abuse of notation, let $\mathcal{F}(\Psi) = \mathcal{F}(\alpha, \beta)$.

[3] Note that in the direct sum notation, for matrices A_1, A_2, B_1, and B_2, where $\dim(A_1) = \dim(A_2) = n \times n$ and $\dim(B_1) = \dim(B_2) = m \times m$, we have that $\operatorname{Tr}[(A_1, B_1)(A_2, B_2)] = \operatorname{Tr}[A_1 A_2] + \operatorname{Tr}[B_1 B_2]$.

Nash equilibria of the finite-valued zero-sum quantum games under consideration. We will generally leverage the weak-solution criterion in our analysis.

Finally, in this work, we focus on the iteration complexity of our methods for computing an ϵ-*approximate Nash equilibrium*, $\tilde{\Psi}^* = (\tilde{\alpha}^*, \tilde{\beta}^*)$, which—as defined by [29]—must satisfy

$$\max_{\alpha} u(\alpha, \tilde{\beta}^*) - \epsilon||U||_\infty \leq u(\tilde{\alpha}^*, \tilde{\beta}^*) \leq \min_{\beta} u(\tilde{\alpha}^*, \beta) + \epsilon||U||_\infty,$$

where $||U||_\infty$ denotes the infinity norm of the payoff observable U. From this definition and the maximal utility of Nash equilibria it follows that

$$\epsilon||U||_\infty \geq u(\tilde{\alpha}^*, \tilde{\beta}^*) - \min_{\beta} u(\tilde{\alpha}^*, \beta) \geq 0 \tag{22}$$

$$\epsilon||U||_\infty \geq \max_{\alpha} u(\alpha, \tilde{\beta}^*) - u(\tilde{\alpha}^*, \tilde{\beta}^*) \geq 0. \tag{23}$$

Summing these equations gives the *duality gap*,

$$\mathcal{DG}(\alpha', \beta') = \max_{\alpha} u(\alpha, \beta') - \min_{\beta} u(\alpha', \beta), \tag{24}$$

which effectively serves as a measure of proximity between any proposed states, (α', β'), and an exact Nash equilibrium, (α^*, β^*). From Equation (22) it follows that for an ϵ-Nash equilibrium, the duality gap is bounded as

$$\epsilon' = 2\epsilon||U||_\infty \geq \mathcal{DG}(\alpha', \beta') \geq 0. \tag{25}$$

Therefore, to evaluate the error of a proposed solution $\tilde{\Psi}$, relative to any true Nash equilibrium, we leverage the notion of an *error/merit function*, which corresponds to the duality gap in the setting of min-max optimization:

$$\text{Error}(\tilde{\Psi}) = \sup_{\Psi \in \mathcal{C}} \text{Tr}\left[(\Psi - \tilde{\Psi})\, \mathcal{F}(\Psi)\right]. \tag{26}$$

It is easy to check that $\text{Error}(\tilde{\Psi}) \geq 0$ for all $\tilde{\Psi}$ and $\text{Error}(\tilde{\Psi}) = 0$ if and only if $\tilde{\Psi} = \Psi^*$.

2.3 Convexity, Smoothness, & Duality

Crucial to our analysis will be functional notions of convexity, smoothness, and duality [61, 62]. We begin by defining convexity and strong convexity of a differentiable function $f : \mathcal{C} \to \mathbb{R}$.

Definition 1 (Convex Function) *A differentiable function $f : \mathcal{C} \to \mathbb{R}$ is* convex *if*

$$f(X) \geq f(Y) + \langle \nabla f(Y), X - Y \rangle, \quad \forall X, Y \in \mathcal{C}. \tag{27}$$

Definition 2 (μ-Strongly Convex Function) *A convex differentiable function $f : \mathcal{C} \to \mathbb{R}$ is* μ-strongly convex *with respect to norm $\|\cdot\|$ if*

$$f(X) \geq f(Y) + \langle \nabla f(Y), X - Y \rangle + \frac{\mu}{2}\|X - Y\|^2, \quad \forall X, Y \in \mathcal{C}, \tag{28}$$

or equivalently, following from the definition of the Bregman divergence in Equation (40),

$$\mathcal{BD}_f(X\|Y) \geq \frac{\mu}{2}\|X - Y\|^2, \quad \forall X, Y \in \mathcal{C}. \tag{29}$$

Since strong convexity implies standard convexity (with $\mu = 0$), the optimal points (minima/maxima) of any strongly convex function must satisfy first-order optimality.

Fact 1 (First-Order Optimality Condition) *For an optimization problem,*

$$\min f(X) \quad \textit{subject to} \quad X \in \mathcal{X},$$

with convex differentiable f, a feasible point X is optimal if and only if

$$\langle \nabla f(X), Y - X \rangle \geq 0, \ \ \forall Y \in \mathcal{X}.$$

Another property of interest for the convex differentiable function f is smoothness, which implies Lipschitz continuity of ∇f.

Definition 3 (β-Smooth Function) *A convex differentiable function $f : \mathcal{C} \to \mathbb{R}$ is β-smooth with respect to norm $\|\cdot\|$ if*

$$f(X) \leq f(Y) + \langle \nabla f(Y), X - Y \rangle + \frac{\beta}{2}\|X - Y\|^2, \quad \forall X, Y \in \mathcal{C}, \tag{30}$$

or equivalently, following from the definition of the Bregman divergence in Equation (40),

$$\mathcal{BD}_f(X\|Y) \leq \frac{\beta}{2}\|X - Y\|^2, \quad \forall X, Y \in \mathcal{C}. \tag{31}$$

Before establishing the Lipschitz condition of ∇f, we introduce the *dual norm*.

Definition 4 (Dual Norm) *If the generic norm of a matrix X is denoted $\|X\|$, the dual norm is defined as*

$$\|X\|_* = \sup\{\langle Y, X \rangle : \|Y\| \leq 1\}. \tag{32}$$

As shown in [63, 64], the Frobenius norm $\|\cdot\|_F$ is self-dual, meaning $\|X\|_* = \|X\|_F$.

Proposition 1 (β-Lipschitz) *A convex differentiable function $f : \mathcal{C} \to \mathbb{R}$ is β-smooth with respect to norm $\|\cdot\|$ if its gradient ∇f is β-Lipschitz continuous,*

$$\|\nabla f(X) - \nabla f(Y)\|_* \leq \beta\|X - Y\| \ \ \forall X, Y \in \mathcal{C}. \tag{33}$$

For the equivalence of Proposition 1 and Definition 3, see [65]. Geometrically, β-smoothness can be interpreted as a convex quadratic *upper* bound,

$$g_X^+(Y) = f(Y) + \langle \nabla f(Y), X - Y \rangle + \frac{\beta}{2}\|X - Y\|^2, \tag{34}$$

of f such that $g_X^+(X) = f(X)$ and $g_X^+(Y) \geq f(Y)$, $\forall Y \in \mathcal{C}$. Meanwhile, μ-strong convexity can be interpreted as a convex quadratic *lower* bound,

$$g_X^-(Y) = f(Y) + \langle \nabla f(Y), X - Y \rangle + \frac{\mu}{2}\|X - Y\|^2, \tag{35}$$

of f such that $g_X^-(X) = f(X)$ and $g_X^-(Y) \leq f(Y)$, $\forall Y \in \mathcal{C}$. In this way, strong convexity is dual to smoothness.

Fenchel conjugacy relates two equivalent representations of a convex function. In the standard representation, a convex function f is represented by pairs $(X, f(X))$ consisting of coordinates X and corresponding function evaluations $f(X)$. Alternatively, as depicted in Figure 6, a convex function can be represented by its tangent at each point. These tangents are encoded as the pairs $(\theta, f^*(\theta))$, where $\theta = \nabla f(X)$ for some X is the gradient and $-f^*(\theta)$ is the intercept. The function f^*, which calculates the tangent's intercept $f^*(\theta)$ from a given gradient θ, is known as the Fenchel conjugate function, formally defined as follows.

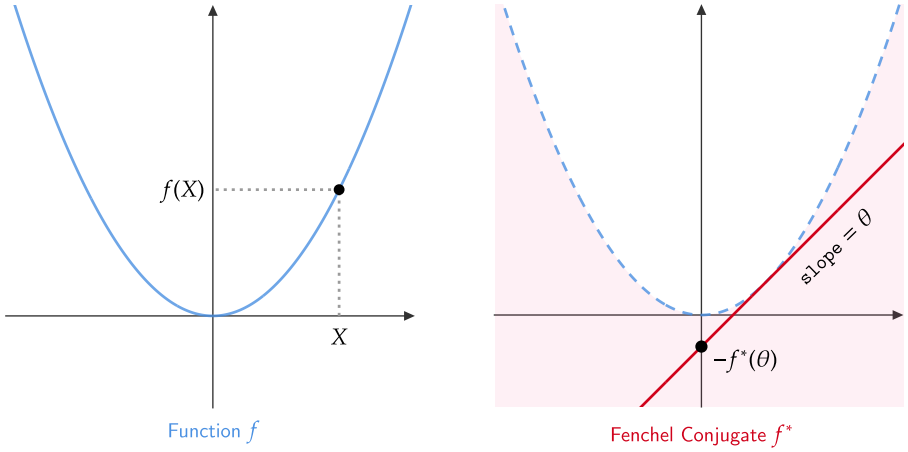


Figure 6: Illustration of a function f and its Fenchel conjugate f^*. The function f is represented by a collection of pairs $(X, f(X))$, while its Fenchel conjugate f^* is visualized as the set of tangents to f, characterized by pairs $(\theta, f^*(\theta))$. [Adapted from Figure 2.3 in [62].]

Definition 5 (Fenchel Conjugate) *The* Fenchel conjugate *of a function $f : \mathcal{C} \to \mathbb{R}$ is the function $f^* : \mathcal{C} \to \mathbb{R}$ such that*

$$f^*(\theta) = \max_{X \in \mathcal{C}} \{\langle X, \theta \rangle - f(X)\}. \tag{36}$$

Relating back to Definition 4 of the dual norm, if it were the case that for all X, $f(X) = 0$, then $f^*(\theta) = \max_{X \in \mathcal{C}} \langle X, \theta \rangle = ||\theta||_*$. Therefore, the Fenchel conjugate can be seen as maximizing the dual norm while also minimizing the penalty induced by function f. In this way, f can be seen as a "regularizer" function.

The definition of Fenchel conjugacy immediately implies the *Fenchel-Young Inequality*,

$$\forall X \in \mathcal{C}, \quad f^*(\theta) \geq \langle X, \theta \rangle - f(X). \tag{37}$$

Furthermore, taking the gradient of the Fenchel conjugate should result in the value of X which maximizes $\langle X, \theta \rangle - f(X)$, resulting in *Danskin's Theorem*, as follows. For more details, refer to Proposition 4.5.1 of [66].

Proposition 2 (Danskin's Theorem) *Let $f : \mathcal{C} \to \mathbb{R}$ be a strongly convex function. For all $X \in \mathcal{C}$, we have*

$$\nabla f^*(\theta) = \arg\min_{X \in \mathcal{C}} \{f(X) - \langle X, \theta \rangle\}.$$

Finally, as stated below and explained further in Corollary 3.5.11 of [67], there is a duality between the the convexity of f and the smoothness of f^*.

Lemma 3 (Strong-Smooth Duality) *A closed convex function $f : \mathcal{C} \to \mathbb{R}$ is α-strongly convex with respect to norm $\|\cdot\|$ if and only if f^* is $\frac{1}{\alpha}$-smooth with respect to the corresponding dual norm $\|\cdot\|_*$.*

2.4 Bregman Divergence & Mirror Map

Critical to understanding our proposed algorithm and its improved convergence over Jain and Watrous' MMWU proposal are the notions of "mirror maps" and Bregman divergence.

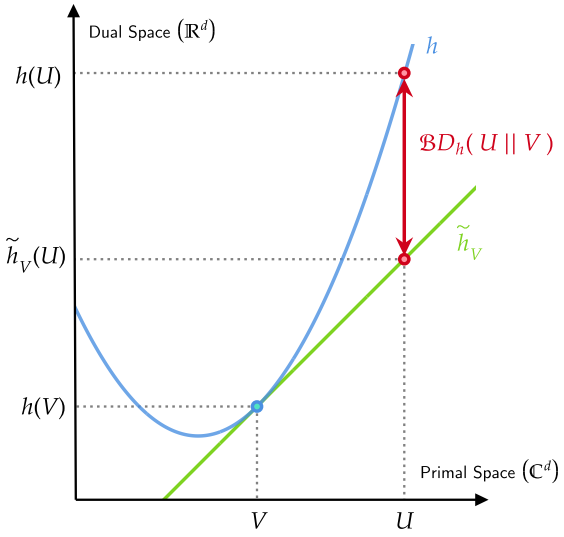


Figure 7: **Bregman Divergence.** The Bregman divergence $\mathcal{BD}_h(U||V)$ between two states in the primal space $U, V \in \mathcal{X}^d$, relative to regularizer h, is given by the difference in the dual space $(\mathbb{R}^d)$ between $h(U)$ and its linear approximation via the first-order Taylor expansion of h around V, $\tilde{h}_V(U)$.

Historically, the first proposals for constrained Gradient Descent Ascent leveraged Euclidean (ℓ_2 norm) projections in order to ensure that the algorithm output would lie within the feasible set. However, it was eventually realized that for certain problems, as we will discuss in the following subsection, regularization based on this ℓ_2 norm results in suboptimal constants in the convergence rates and, by leveraging different distance metrics, faster rates can be achieved. In order to develop algorithms which could encompass any such desired distance metric, the Bregman divergence was introduced.

The Bregman divergence is parameterized by a "distance-generating" or "regularizer" function. Formally, a *distance-generating function* (DGF) on the d-dimensional spectraplex $\mathcal{H}_+^d$ is defined as a convex, lower semi-continuous function $h : \mathbb{C}^d \to \mathbb{R} \cup \{\infty\}$, where the *effective* domain of h is $\mathcal{X}^d$ and $\mathbb{C}^d$ is the space of $2^d \times 2^d$ complex matrices. Denoting the space of $2^d \times 2^d$ real matrices as $\mathbb{R}^d$, let $\partial h : \mathcal{H}_+^d \to \mathbb{R}^d$ denote the *subdifferential* of h defined as[4]

$$\partial h(X) := \left\{ V \in \mathbb{R}^d : h(Y) \geq h(X) + \mathrm{Tr}[(Y - X)^\dagger V],\ \forall\, Y \in \mathcal{H}_+^d \right\}. \tag{38}$$

The *domain of subdifferentiability* of h is defined as

$$\mathrm{dom}\partial h := \{X \in \mathcal{X}^d : \partial h(X) \neq \emptyset\}. \tag{39}$$

Having defined the DGF h and its subdifferential ∂h, we are ready to define the Bregman divergence. Given two states $U, V \in \mathcal{H}_+^d$, the Bregman divergence induced by h is the non-negative function $\mathcal{BD}_h : \mathrm{dom}h \times \mathrm{dom}\partial h \to \mathbb{R}$ defined as

$$\mathcal{BD}_h(U||V) := h(U) - h(V) - \mathrm{Tr}[(U - V)^\dagger \partial h(V)], \tag{40}$$

where the subdifferential maps from states in the *feasible region* $\mathcal{H}_+^d$ of the *primal space* $\mathbb{C}^d$, to the *dual space* of subgradients $\mathbb{R}^d$. The function ∂h, which performs a primal-to-dual mapping, is often referred to as a *mirror map*.

[4]The subdifferential is the set-valued generalization of the derivative for convex functions which are not necessarily differentiable.

Intuitively, as illustrated in Figure 7, the Bregman divergence $\mathcal{BD}_h(U||V)$ between two states U and V in the primal space, relative to regularizer h, is given by the difference in the dual space between $h(U)$ and its linear approximation via the first-order Taylor expansion of h around V,

$$\widetilde{h}_V(U) = h(V) + \mathrm{Tr}[(U - V)^\dagger \partial h(V)]. \tag{41}$$

Since h is convex, this difference is always non-negative. Note that $\mathcal{BD}_h(U||V)$ is not necessarily equivalent to $\mathcal{BD}_h(V||U)$.

Finally, our analysis will use the well-known *Three-Point Identity*,

$$\mathrm{Tr}[(Z - X)^\dagger \nabla_{X^\top} \mathcal{BD}_h(X||Y)] = \mathcal{BD}_h(Z||Y) - \mathcal{BD}_h(X||Y) - \mathcal{BD}_h(Z||X), \tag{42}$$

which follows from the definition of the Bregman divergence [68].

2.5 Mirror & Proximal Steps

We will now introduce "mirror steps" and "proximal steps," which incorporate the mirror map generalization of distance in constrained optimization updates.[5] These steps implement the algorithmic state updates of the classical games algorithms discussed in the Prior Work (Section 1.2). Similar to a standard constrained optimization gradient step, the goal of these procedures is to produce an update that lies in the feasible set, closer to the optimum. However, by generalizing the Euclidean distance of traditional Gradient Descent Ascent steps, mirror and proximal steps enable incorporation of domain-specific knowledge about the geometry of the optimization task into the updates. This can lead to substantial improvements in convergence rate.

For intuition, let us first consider a classical single-agent case as an illustrative example. When optimizing a function f over a $(d-1)$-dimensional simplex, restricted by the ℓ_∞-norm, standard Projected Subgradient Descent (PSD) reaches a convergence rate of $\mathcal{O}(\sqrt{d/T})$. In contrast, classical Mirror Descent Ascent using a negative Shannon entropy regularizer significantly improves the rate to $\mathcal{O}(\sqrt{\log d/T})$.

Our objective is to adapt this methodology to the more challenging, exponentially larger $(4^d - 1)$-dimensional spectraplex $\mathcal{H}_+^d$, in a min-max framework. Specifically, we aim to incorporate domain-specific regularization techniques to achieve an $\mathcal{O}(d/T)$ convergence rate. This would offer an exponential dimensionality improvement from standard Projected Subgradient Ascent Descent, which has an $\mathcal{O}(2^d/\sqrt{T})$ convergence rate [70, 71], and a quadratic speed-up over the Jain-Watrous algorithm, which has an $\mathcal{O}(d/\sqrt{T})$ convergence rate [29].

Note that intuitions for the following definitions and proofs are provided in Appendix B.

2.5.1 Mirror Step

Given a function $f : \mathbb{C}^d \to \mathbb{R}$ that we aim to optimize, convex feasible set $\mathcal{H}_+^d \subset \mathbb{C}^d$, DGF h, subgradients ∂h, and initial primal state $X_t \in \mathcal{H}_+^d$, each call to a *mirror step* oracle generates an updated state X_{t+1} via Algorithm 2 (as depicted in Figure 8). In Step (3), the mirror map, ∂h^{-1}, can be implemented as follows.

[5] For further background, we refer the interested reader to [68] and Chapter 4 of [69].

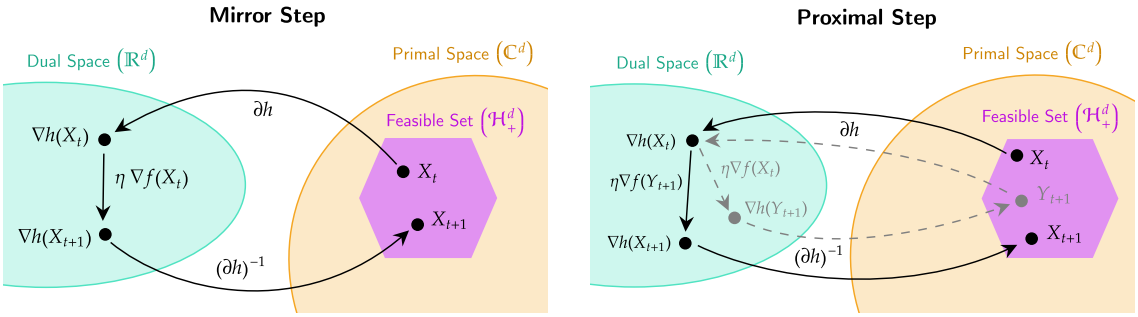


Figure 8: **Mirror & Proximal Step.**Visualizations of the mirror and proximal steps described in Algorithm 2 and Algorithm 3, respectively. [Adapted from Figures 4.1 and 4.2 in [69].]

Algorithm 2 Mirror Step

1. Initial primal state $X_t \in \mathcal{H}_+^d$ is mapped to a dual state $\nabla h(X_t) \in \partial h(X_t) \subset \mathbb{R}^d$.
2. Gradient $\nabla f : \mathbb{C}^d \to \mathbb{R}^d$ is calculated with respect to X_t.
3. In the dual space, the state is updated according to the gradient with step size η, resulting in a new dual state
$$\nabla h(X_{t+1}) = \nabla h(X_t) + \eta \nabla f(X_t) \quad \in \mathbb{R}^d.$$
4. Mapping back to the primal space, this corresponds to the updated feasible state
$$X_{t+1} \in \partial h^{-1}(\nabla h(X_{t+1}) + \eta \nabla f(X_t)) \quad \in \mathcal{H}_+^d.$$

Definition 6 (Mirror Map) *For a state D in the dual space $\mathcal{D}$, the* mirror map*, or* regularized best response*,*

$$\mathrm{Mir}\Pi_{\mathcal{P}}^{h}(D) := \arg\max_{P \in \mathcal{P}} \{\mathrm{Tr}[D^\dagger P] - h(P)\}, \tag{43}$$

implements the inverse mirror map, $\partial h^{-1} : \mathcal{D} \to \mathcal{P}$, mapping $D \in \mathcal{D}$ to the corresponding state P in the feasible region $\mathcal{P}$ of the primal space.

Therefore, the full mirror step procedure, mapping from X_t to X_{t+1}, is achieved via the mirror map of $Y_t = \nabla h(X_t) + \eta \nabla f(X_t)$ on the feasible set $\mathcal{H}_+^d$, as

$$X_{t+1} = \mathrm{Mir}\Pi_{\mathcal{H}_+^d}^{h}(\nabla h(X_t) + \eta \nabla f(X_t)). \tag{44}$$

Alternatively, the mirror step procedure can be expressed as a *mirror proximal map.*

Lemma 4 (Mirror Step) *For state X_t in the feasible set $\mathcal{H}_+^d$ of primal space $\mathbb{C}^d$,*

$$\mathrm{MirProx}\Pi_{\mathcal{X}^d}^{h,\eta,f}(X_t) := \arg\min_{X_{t+1} \in \mathcal{H}_+^d} \left\{ \mathrm{Tr}[\nabla f(X_t)^\dagger X_{t+1}] + \frac{1}{\eta} \mathcal{B}\mathcal{D}_h(X_{t+1} || X_t) \right\}, \tag{45}$$

implements the mirror step mapping to an updated feasbile set primal state $X_{t+1} \in \mathcal{H}_+^d$.

Algorithm 3 Proximal Step

1. From initial feasible primal state $X_t \in \mathcal{H}_+^d$, updated feasible primal state $Y_{t+1} \in \mathcal{H}_+^d$ is generated via the Mirror Step procedure of Algorithm 2.
2. Initial feasible primal state X_t is mapped to a dual state $\nabla h(X_t) \in \partial h(X_t) \subset \mathbb{R}^d$.
3. Gradient $\nabla f : \mathbb{C}^d \to \mathbb{R}^d$ is calculated with respect to Y_{t+1}.
4. In the dual space, the state is updated according to the gradient with step size η, resulting in a new dual state
$$\nabla h(X_{t+1}) = \nabla h(X_t) + \eta \nabla f(Y_{t+1}) \quad \in \mathbb{R}^d.$$
5. Mapping back to the primal space, this corresponds to the updated feasible state
$$X_{t+1} \in \partial h^{-1}(\nabla h(X_t) + \eta \nabla f(Y_{t+1})) \quad \in \mathcal{H}_+^d.$$

2.5.2 Proximal Step

We refer to the mirror prox projection as a *proximal map*. Its generalization comprises the *proximal step* oracle, described in Algorithm 3 and depicted in Figure 8.

Definition 7 (Proximal Map) *For a state P in the feasible set of the primal space $\mathcal{P}$ and a state D in the dual space, the* proximal map*,*

$$\begin{aligned}\text{Prox}\Pi_{\mathcal{P}}^{h}(P, D) :&= \arg\min_{P^* \in \mathcal{P}} \left\{ \text{Tr}[D^\dagger (P^* - P)] + \mathcal{BD}_h(P^* || P) \right\} \\ &= \arg\min_{P^* \in \mathcal{P}} \left\{ \text{Tr}[D^\dagger P^*] + \mathcal{BD}_h(P^* || P) \right\}, \end{aligned} \tag{46}$$

implements the proximal mapping to an updated feasible set primal state $P^* \in \mathcal{P}$.

Applying this definition to our problem of interest yields the following map.

Lemma 5 (Proximal Step) *For state $X_t \in \mathcal{H}_+^d$ in the feasible set of the primal space,*

$$\begin{aligned}\text{Prox}\Pi_{\mathcal{H}_+^d}^{h,\eta}(X_t, \nabla f(Y_{t+1})) :&= \arg\min_{X_{t+1} \in \mathcal{H}_+^d} \left\{ \text{Tr}[\nabla f(Y_{t+1})^\dagger (X_{t+1} - X_t)] + \frac{1}{\eta} \mathcal{BD}_h(X_{t+1} || X_t) \right\} \\ &= \arg\min_{X_{t+1} \in \mathcal{H}_+^d} \left\{ \text{Tr}[\nabla f(Y_{t+1})^\dagger X_{t+1}] + \frac{1}{\eta} \mathcal{BD}_h(X_{t+1} || X_t) \right\}, \end{aligned} \tag{47}$$

implements the proximal step mapping to an updated feasbile set primal state $X_{t+1} \in \mathcal{H}_+^d$.

Note that $\text{Prox}\Pi_{\mathcal{H}_+^d}^{h,\eta}(X_t, \nabla f(X_t)) = \text{MirProx}\Pi_{\mathcal{H}_+^d}^{h,\eta,f}(X_t)$, which, by Lemma 4, implies that the proximal step is a generalization of the mirror step and, thus, can be used to implement a mirror step. Thus, the proximal step procedure of Algorithm 3 can be achieved algorithmically via two proximal maps as:

$$Y_{t+1} = \text{Prox}\Pi_{\mathcal{H}_+^d}^{h,\eta}(X_t, \nabla f(X_t)) \tag{48}$$

$$X_{t+1} = \text{Prox}\Pi_{\mathcal{H}_+^d}^{h,\eta}(X_t, \nabla f(Y_{t+1})). \tag{49}$$

Note that the proximal step can be alternatively written as a single update:

$$X_{t+1} = \text{Prox}\Pi^{h,\eta}_{\mathcal{H}^d_+}\left(X_t, \nabla f\left(\text{Prox}\Pi^{h,\eta}_{\mathcal{H}^d_+}(X_t, \nabla f(X_t))\right)\right). \tag{50}$$

Standard gradient-based optimization methods are generally based on a single step in the direction of the gradient. In contrast, proximal step methods are based on a two-step approach:

1. From initial point X_t, apply a "look-ahead" step, resulting in a "prediction" Y_{t+1}.
2. Use prediction Y_{t+1} in a "correction" step that achieves the update X_{t+1}.

As we will discuss in Section 3.3, replacing mirror with proximal steps provides the desired acceleration from $\mathcal{O}_d(1/\epsilon^2)$ to $\mathcal{O}_d(1/\epsilon)$. Furthermore, the usage of the Bregman divergence enables a geometry-aware projection and, specifically, can be used to achieve a logarithmic improvement on dependence of the dimensionality of the spectraplex, as will be discussed in Section 4.7.

2.5.3 Regularizers

In order to guarantee the well-posedness of the mirror and proximal projections, we will restrict h to the class of functions which are strongly convex. In the classical games setting, as discussed in the Prior Work (Section 1.2), the most widely used regularizers are the Euclidean (ℓ_2) norm ($h(x) = \frac{1}{2}||x||_2^2$) and the negative entropy ($h(x) = \sum_i x_i \log(x_i)$). In this work, we will leverage the quantum analogs of these classical regularizers: the Frobenius norm ($h(X) = \frac{1}{2}||X||_F^2$) and the von Neumann entropy ($h(X) = \text{Tr}[X \log X]$). As highlighted in the following lemmas (proved in Appendix B), the mirror and proximal maps take well-known forms when these regularizers are used.

Lemma 6 (Bregman Divergence of the Frobenius Norm) *If h is the squared Frobenius norm,*

$$h(X) = \frac{1}{2}||X||_F^2 = \frac{1}{2}\text{Tr}[X^\dagger X], \tag{51}$$

then the Bregman divergence is the squared Frobenius distance,

$$\mathcal{BD}_h(X||Y) = \tfrac{1}{2}||X - Y||_F^2, \tag{52}$$

the corresponding mirror map (regularized best-response) is the orthogonal projection,

$$\text{Mir}\Pi^h_{\mathcal{C}}(Y) = \arg\min_{C\in\mathcal{C}}||Y - C||_F^2 = \mathit{Orth}\Pi_{\mathcal{C}}(Y), \tag{53}$$

and the corresponding proximal map is

$$\text{Prox}\Pi^h_{\mathcal{C}}(X, Y) = \arg\min_{C\in\mathcal{C}}||X + Y - C||_F^2 = \mathit{Orth}\Pi_{\mathcal{C}}(X + Y). \tag{54}$$

Lemma 7 (Bregman Divergence of the von Neumann Entropy) *If h is the von Neumann entropy,*

$$h(X) = \text{Tr}[X \log X], \tag{55}$$

then the Bregman divergence is the quantum relative entropy

$$\mathcal{BD}_h(X||Y) = \text{Tr}[X(\log X - \log Y)], \tag{56}$$

the corresponding mirror map (regularized best-response) is the logit map,

$$\mathrm{Mir}\Pi^h_{\mathcal{C}}(Y) = \frac{\exp(Y)}{\mathrm{Tr}[\exp(Y)]} = \Lambda(Y), \tag{57}$$

and the corresponding proximal map is

$$\mathrm{Prox}\Pi^h_{\mathcal{C}}(X, Y) = \frac{\exp(\log X + Y)}{\mathrm{Tr}[\exp(\log X + Y)]} = \Lambda(\log X + Y). \tag{58}$$

In Section 3, we will leverage these mirror and proximal steps (viewed as oracles) to craft efficient iterative algorithms for finding Nash equilibria of zero-sum quantum games.

3 Algorithms for Quantum Zero-Sum Games

As previously discussed, our goal is to efficiently find Nash equilibria, $\Psi^* = (\alpha^*, \beta^*)$, satisfying the variational inequality of Equation (20). This section describes existing algorithms, as well as our proposed algorithm for finding these Nash equilibria. Section 4 will demonstrate that our proposed algorithm achieves a quadratic speedup relative to existing algorithms.

3.1 Matrix Dual Averaging (MDA) Method

In 2008, Jain and Watrous introduced a parallel algorithm to find approximate Nash-equilibria of quantum zero-sum games [29]. Their algorithm was based on the Matrix Multiplicative Weights Update (MMWU) method [46, 34, 72, 37]. Here, we will instead present and assess a more general method, which we will refer to as Matrix Dual Averaging (MDA),[6] which, with different DGFs, parameterizes a family of algorithms that include Jain and Watrous' MMWU algorithm.

MDA algorithms are played over several rounds, $t \in [0, N]$, of the quantum zero-sum game. Suppose that, in round t, Alice and Bob play states $\Psi_t = (\alpha_t, \beta_t)$.[7] After performing his joint measurement, Roger returns feedback to Alice and Bob in the form of their respective payoff gradients, $\mathcal{F}_\alpha(\Psi_t)$ and $\mathcal{F}_\beta(\Psi_t)$, as defined in Equation (16) and Equation (17). Alice and Bob use this feedback to individually update their states so as to iteratively maximize their individual payoff. The general MDA family of algorithms (Algorithm 4) can be expressed concisely as the updates:

$$\Phi_{t+1} = \Phi_t + \mathcal{F}(\Psi_t) \tag{59}$$

$$\Psi_{t+1} = \mathrm{Mir}\Pi^h_{\mathcal{C}}(\eta\, \Phi_{t+1}), \tag{60}$$

where η is a chosen step size and h is a regularizing, convex DGF. As we will see, it is the choice of DGF h which instantiates different MDA algorithms, including MMWU. Furthermore, relating back to the depiction of the mirror update step in Figure 8, the intermediate states Φ are averaged in the dual space, but are mapped onto feasible states Ψ of the primal space by the mirror projection $\mathrm{Mir}\Pi^h_{\mathcal{C}}$. Overall, the algorithm only incorporates first-order derivatives in the update steps.

[6] The name is a reference to the analogous algorithm for classical games, i.e., Nesterov's Dual Averaging [70]. Dual Averaging is sometimes also known as "Lazy" Mirror Descent, since the key difference from standard Mirror Descent is that averaging happens in the dual space (rather than mapping from primal to dual space and back in each iteration).

[7] Note that, in the first round ($t = 0$), Alice and Bob play the maximally mixed state $\Psi_0 = (\frac{\mathbb{1}_n}{2^n}, \frac{\mathbb{1}_m}{2^m})$.

Algorithm 4 Matrix Dual Averaging

Accuracy Parameter: ϵ
Regularization Function: $h : \{\mathcal{A}, \mathcal{B}\} \to \mathbb{R}$
Diameter (of $\mathcal{BD}_h$): $\mathcal{D}_h = \sup_{X,Y \in \mathcal{C}} \mathcal{BD}_h(X\|Y)$
Regularized Best Response: $\mathrm{MirΠ}^h_{\mathcal{A},\mathcal{B}}(\cdot) = \arg\max_{x \in \{\mathcal{A},\mathcal{B}\}} \{\langle \cdot, x\rangle - h(x)\}$

$\eta \leftarrow \mu_h/(2\gamma_{\mathcal{F}})$ ▷ Step Size
$N \leftarrow \lceil \mathcal{D}_h/(\eta \cdot \epsilon^2) \rceil$ ▷ Number of Rounds
$(\alpha_0, \beta_0) \leftarrow \left(\frac{1}{2^n}\mathbb{1}_{\mathcal{A}}, \frac{1}{2^m}\mathbb{1}_{\mathcal{B}}\right)$ ▷ State Initialization

for $t \in [1, N-1]$ **do**

$\mathcal{CF}^{(t)}_{\alpha} \leftarrow \sum_{i=0}^{t-1} \mathcal{F}_{\alpha}(\Psi_i)$ ▷ Calculate Cumulative Feedback
$\mathcal{CF}^{(t)}_{\beta} \leftarrow \sum_{i=0}^{t-1} \mathcal{F}_{\beta}(\Psi_i)$

$\alpha_t \leftarrow \mathrm{MirΠ}^h_{\mathcal{A}}\left(\eta\, \mathcal{CF}^{(t)}_{\alpha}\right)$ ▷ State Updates
$\beta_t \leftarrow \mathrm{MirΠ}^h_{\mathcal{B}}\left(\eta\, \mathcal{CF}^{(t)}_{\beta}\right)$

end for

return $\left(\bar{\alpha} = \frac{1}{N}\sum_{t=0}^{N-1} \alpha_t, \bar{\beta} = \frac{1}{N}\sum_{t=0}^{N-1} \beta_t\right)$

We will now specify how each player chooses their next state via the previously described MDA method. Note that both Alice and Bob perform the same procedure, so we will only specify the procedure from Alice's perspective. Upon receiving feedback $\mathcal{F}_{\alpha}(\Psi_t)$ from Roger, Alice calculates her cumulative feedback,

$$\mathcal{CF}^{(t)}_{\alpha} = \sum_{i=0}^{t-1} \mathcal{F}_{\alpha}(\Psi_i), \tag{61}$$

which is simply the sum of the payoff gradients across all prior rounds. To calculate the next state she will play, α_{t+1}, Alice re-weights the cumulative feedback by the step size η and performs a mirror projection onto her spectraplex,

$$\alpha_{t+1} = \mathrm{MirΠ}^h_{\mathcal{A}}(\eta\, \mathcal{CF}^{(t)}_{\alpha}) = \arg\max_{\alpha \in \mathcal{A}} \{\langle \eta\, \mathcal{CF}^{(t)}_{\alpha}, \alpha\rangle - h(\alpha)\}. \tag{62}$$

We will now specify how different choices of h lead to different instantiations of the MDA algorithm. For example, following from Lemma 6, if the regularizer is the squared Frobenius norm,

$$h(\alpha) = \frac{1}{2}\|\alpha\|_F^2, \tag{63}$$

then the mirror projection maps Alice's next state onto the orthogonal projection

$$\alpha_{t+1} = \mathrm{OrthΠ}_{\mathcal{A}}(\eta\, \mathcal{CF}^{(t)}_{\alpha}). \tag{64}$$

Meanwhile, following from Lemma 7, if the regularizer is the negative von Neumann entropy,

$$h(\alpha) = \mathrm{Tr}[\alpha \log \alpha], \tag{65}$$

then the mirror projection maps Alice's next state onto the logit map

$$\alpha_{t+1} = \Lambda_{\mathcal{A}}\left(\eta\, \mathcal{CF}^{(t)}_{\alpha}\right) = \frac{\exp\left(\eta\, \mathcal{CF}^{(t)}_{\alpha}\right)}{\mathrm{Tr}\left(\exp\left(\eta\, \mathcal{CF}^{(t)}_{\alpha}\right)\right)} \tag{66}$$

which yields the MMWU algorithm of [29]. The following convergence rate for this procedure was established by [29].

Theorem 2 (MDA Rate) *MDA methods compute an ϵ-Nash equilibrium in $\mathcal{O}_d(1/\epsilon^2)$ steps.*

Thus, for quantum zero-sum games MMWU obtains a worse convergence rate than the best known algorithms for classical zero-sum games, which obtain an $\mathcal{O}_d(1/\epsilon)$ rate, as described in Section 1.2 and depicted in Figure 1. In the remainder of this work, we will show how the classically efficient Mirror Prox algorithm and its variants can be lifted to the quantum zero-sum game setting in order to obtain the desired $\mathcal{O}_d(1/\epsilon)$ rate.

3.2 Matrix Mirror Prox (MMP)

The classical Mirror Prox (MP) algorithm is described in Section 1.2, with updates given in Figure 1. In each step of the iterative MP optimization procedure, the gradient of the players' utilities is calculated followed by the execution of a mirror step (Algorithm 2), as illustrated in Figure 8. The mirror step can be implemented via a mirror map of the dual gradient, as in Equation (44), or the mirror map of the primal state, as in Equation (45). The addition of proximal steps—or extra-gradient steps—is crucial as they introduce a refinement to the basic gradient step. After the initial gradient is computed and the first mirror step is taken, the algorithm does not immediately proceed with this information. Instead, it calculates a second gradient at the point reached after the first mirror step. This second gradient offers a more accurate direction for the update because it accounts for the immediate effects of the initial step, thus incorporating more information about the curvature and constraints of the utility space. This two-pronged gradient approach is what distinguishes the MP algorithm from simpler gradient descent methods, offering a sophisticated tool for navigating the intricate landscapes of variational inequalities.

Drawing inspiration from classical MP, we propose an analog for the quantum zero-sum games of interest. This quantum analog, which we will refer to as the Matrix Mirror Prox (MMP) method, has updates:

$$\Phi_{t+1} = \mathrm{Prox}\Pi_{\mathcal{C}}^{h,\eta}(\Psi_t, \mathcal{F}(\Psi_t)) \tag{67}$$

$$\Psi_{t+1} = \mathrm{Prox}\Pi_{\mathcal{C}}^{h,\eta}(\Psi_t, \mathcal{F}(\Phi_{t+1})). \tag{68}$$

Notice that these update steps are of the same form as the proximal step updates given in Equation (48) and Equation (49). Alternatively, the updates can be expressed purely in terms of the joint states Ψ_t:

$$\Psi_{t+1} = \mathrm{Prox}\Pi_{\mathcal{C}}^{h,\eta}\Big(\Psi_t, \mathcal{F}\Big(\mathrm{Prox}\Pi_{\mathcal{C}}^{h,\eta}(\Psi_t, \mathcal{F}(\Psi_t))\Big)\Big). \tag{69}$$

As discussed in Section 2.5, the incorporation of proximal steps in the Matrix Mirror Prox algorithm offers two significant advantages over the conventional Projected Subgradient Descent Ascent method:

1. **Geometry-Awareness**: MMP replaces the Frobenius norm of PGDA with a generalized Bregman divergence. Similar to MDA, utilizing a mirror map with this Bregman divergence enables MMP to adapt to the inherent geometry of the problem space. This makes the algorithm highly versatile, capable of effectively handling a wide array of problems, including those with intricate constraints.

Algorithm 5 Optimistic Matrix Mirror Prox (OMMP)

Accuracy Parameter: ϵ
Regularization Function: $h : \{\mathcal{A}, \mathcal{B}\} \to \mathbb{R}$
Strong Convexity Parameter (of h): μ_h
Diameter (of $\mathcal{BD}_h$): $\mathcal{D}_h = \sup_{X,Y \in \mathcal{C}} \mathcal{BD}_h(X\|Y)$
Lipschitz Parameter (of $\mathcal{F}$): $\gamma_{\mathcal{F}}$
Proximal Map: $\text{Prox}\Pi^{h,\eta}_{\mathcal{A},\mathcal{B}}(X, Y) := \arg\min_{C \in \mathcal{C}} \{\langle Y, C - X\rangle - \frac{1}{\eta}\mathcal{BD}_h(C||X)\}$.

$\eta \leftarrow \mu_h/(2\gamma_{\mathcal{F}})$ ▷ Step Size
$N \leftarrow \lceil \mathcal{D}_h/(\eta \cdot \epsilon) \rceil$ ▷ Number of Rounds
$(\alpha_0, \beta_0) \leftarrow \left(\frac{1}{2^n}\mathbb{1}_{\mathcal{A}}, \frac{1}{2^m}\mathbb{1}_{\mathcal{B}}\right)$ ▷ State Initialization
$(\hat{\alpha}_0, \hat{\beta}_0) \leftarrow (\alpha_0, \beta_0)$ ▷ Optimistic-Momentum Initialization

for $t \in [1, N-1]$ **do**

$\alpha_{t+1} \leftarrow \text{Prox}\Pi^{h,\eta}_{\mathcal{A}}(\hat{\alpha}_t, \mathcal{F}_\alpha(\Psi_t))$ ▷ State Updates
$\beta_{t+1} \leftarrow \text{Prox}\Pi^{h,\eta}_{\mathcal{B}}(\hat{\beta}_t, \mathcal{F}_\beta(\Psi_t))$

$\hat{\alpha}_{t+1} \leftarrow \text{Prox}\Pi^{h,\eta}_{\mathcal{A}}(\hat{\alpha}_t, \mathcal{F}_\alpha(\Psi_{t+1}))$ ▷ Momentum Updates
$\hat{\beta}_{t+1} \leftarrow \text{Prox}\Pi^{h,\eta}_{\mathcal{B}}(\hat{\beta}_t, \mathcal{F}_\beta(\Psi_{t+1}))$

end for

return $\left(\bar{\alpha} = \frac{1}{N}\sum_{t=0}^{N-1} \alpha_t, \bar{\beta} = \frac{1}{N}\sum_{t=0}^{N-1} \beta_t\right)$

2. **Intermediate Proximal Step**: Proximal steps in MMP introduce an intermediate "prediction" state (i.e., the state Y_t in Section 2.5.2), which plays a crucial role. At a high level, this prediction state extrapolates the future trajectory of the descent/ascent, thereby guiding the subsequent state update. This leverages the intuition that the gradient at the next state offers a more informative update gradient than that at the current state.

Importantly, this combination of geometry-aware mirror maps and forward-looking intermediate proximal steps accelerates the convergence of MMP with von Neumann regularization to min-max points, thereby making MMP a potent tool for solving min-max problems efficiently.

As is the case for classical MP (discussed in Section 1.2), a key downside of MMP is that it requires two gradient calls per iteration. Specifically, in each round t, to calculate the updated joint state Ψ_{t+1}, Alice and Bob must query Roger with two distinct sets of states, Ψ_t and Φ_{t+1}, so as to obtain the respective feedback gradients $\mathcal{F}(\Psi_t)$ and $\mathcal{F}(\Phi_{t+1})$. Classically, there exist variants of MP with the same convergence rate that require only a single gradient call per iteration. The remainder of this work will focus on a proposed classically-inspired variant of MMP which requires only a single gradient call per iteration, and a proof that it obtains the desired $\mathcal{O}_d(1/\epsilon)$ convergence rate.

3.3 Optimistic Matrix Mirror-Prox (OMMP) Methods

We will now discuss the main contribution of this work, a novel approach to finding Nash equilibria of quantum zero-sum games, which we refer to as the Optimistic Matrix Mirror Prox (OMMP) method. This algorithm draws inspiration from seminal classical game

theory work on Optimistic Mirror Prox (OMP)[8] methods, as described in the Prior Work (Section 1.2). The OMMP algorithm, detailed in Algorithm 5, can be represented succinctly through the following update equations:

$$\Psi_{t+1} = \mathrm{Prox}\Pi_{\mathcal{C}}^{h,\eta}(\Phi_t, \mathcal{F}(\Psi_t)) \tag{70}$$

$$\Phi_{t+1} = \mathrm{Prox}\Pi_{\mathcal{C}}^{h,\eta}(\Phi_t, \mathcal{F}(\Psi_{t+1})) \tag{71}$$

Here, η is a predetermined step size, and $\mathrm{Prox}\Pi$ refers to the proximal map, as elaborated in Algorithm 3 and Definition 7.

Although the update equations for OMMP closely resemble those of the standard MMP algorithm, a critical difference exists. As alluded to in the previous section, the OMMP variant necessitates only a single gradient call per iteration. While it may appear that both Alice and Bob require two feedback gradients, $\mathcal{F}(\Psi_t)$ and $\mathcal{F}(\Psi_{t+1})$, for each round t, they will have already acquired $\mathcal{F}(\Psi_t)$ during round $t-1$. This allows them to reuse $\mathcal{F}(\Psi_t)$ and only make a query for a new gradient with one state, Ψ_{t+1}.

To gain some intuition for this algorithm, we will revisit the original proposal of "optimism" in this context, due to Popov [53]. In conventional Gradient Descent Ascent, both players update their strategies concurrently, moving in directions opposite to their individual gradients—akin to a tug-of-war. Optimistic Gradient Descent Ascent (OGDA) enhances this by incorporating a "look-ahead" feature. Players consider not only the current gradient but also anticipate the opponent's forthcoming strategic adjustments. This anticipatory or "optimistic" approach is formalized by adding an extra term to the update rule, essentially an estimate of the future gradient.

This anticipatory element is also how OMMP diverges from standard MMP. Rather than employing a new prediction, optimistic OMMP utilizes a past "inertia" point, Φ_t. Unlike MMP, which focuses solely on future states, OMMP integrates past gradient information to stabilize its learning trajectory.

4 Convergence Analysis

We will now prove one of the main results of the paper:

Theorem 1 (Main Result) *The OMMP method computes ϵ-Nash equilibria in $\mathcal{O}_d(1/\epsilon)$ steps.*

This establishes that our proposed OMMP method yields a quadratic speedup compared to previous MDA-based approaches. Concretely, we will leverage the proof structure of Ene and Nguyên [58] to demonstrate that, with an appropriately chosen step size η and for a given accuracy parameter ϵ, the *average iterate*

$$\overline{\Psi}_T = \frac{1}{T}\sum_{t=1}^{T} \Psi_t \tag{72}$$

converges to an ϵ-Nash equilibrium in $O_d(1/\epsilon)$ iterations.[9]

[8] There are a variety of classical algorithms of this general form, known as Popov's modified Arrow-Hurwicz algorithm, Past Extra-Gradient, Optimistic Extra-Gradient, or the Optimistic Gradient Variant.

[9] It's important to note the difference in the notation used to describe algorithmic convergence rates between the optimization and computer science communities. In the realm of optimization, *convergence rates* such as $\mathcal{O}(e^{-\rho T})$, $\mathcal{O}(1/T)$, and $\mathcal{O}(1/\sqrt{T})$ are commonly employed to indicate how swiftly an algorithm approximates the optimal solution as the iteration count T increases. In contrast, the computer science literature tends to recast these convergence rates as *number of iterations* required to achieve an ϵ-accurate solution, thereby giving a more direct account of computational efficiency. Expressed in terms of accuracy parameter ϵ, this leads to expressions like $\mathcal{O}(\log(1/\epsilon))$, $\mathcal{O}(1/\epsilon)$, and $\mathcal{O}(1/\epsilon^2)$.

In Section 4.7, we will treat the dimension-dependence hidden in the $\mathcal{O}_d(\cdot)$ notation, showing in particular that the von Neumann entropy regularizer reduces OMMP's dimension-dependence logarithmically relative to the Frobenius norm regularizer. Since OMMP with the von Neumann entropy regularizer instatiates the OMMWU algorithm, we thus prove that OMMWU obtains the desired $\mathcal{O}(d/\epsilon)$ iteration complexity and, thus, achieves a quadratic speedup to MMWU's $\mathcal{O}(d/\epsilon^2)$ iteration complexity.

4.1 Monotonicity of $\mathcal{F}$

Critical to our analysis is the monotonicity of the feedback operator $\mathcal{F}$, which is a key property of the finite-valued quantum zero sum games of interest.

Lemma 8 (Mononicity of $\mathcal{F}$) *$\mathcal{F}$ is monotone or, equivalently,*

$$\operatorname{Tr}\left[\left(\mathcal{F}(X)-\mathcal{F}(Y)\right)(X-Y)\right]\geq 0,\ \ \forall X,Y\in\mathcal{C}. \tag{73}$$

The proof of this result employs techniques involving the Pauli decomposition of the feedback operator, thereby deviating from the conventional proof for monotonicity of classical normal-form game operators. We defer the proof of $\mathcal{F}$'s monotonicity (in addition to many other properties of $\mathcal{F}$) to Appendix C.2.

4.2 Error Decomposition

Following from our discussion of Nash equilibria in Section 2.2 and the natural error function, given in Equation (26), the error of the average iterate is defined as

$$\operatorname{Error}(\overline{\Psi}_T)=\sup_{Z\in\mathcal{C}}\operatorname{Tr}\left[\mathcal{F}(Z)\left(\overline{\Psi}_T-Z\right)\right]=\frac{1}{T}\sup_{Z\in\mathcal{C}}\sum_{t=1}^{T}\operatorname{Tr}\left[\mathcal{F}(Z)\left(\Psi_t-Z\right)\right]. \tag{74}$$

From the monotonicity of $\mathcal{F}$, given in Equation (73), for Ψ_t and some state $Z\in\mathcal{C}$,

$$\operatorname{Tr}\left[\mathcal{F}(Z)\left(\Psi_t-Z\right)\right]\leq\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Psi_t-Z\right)\right], \tag{75}$$

which implies that the average-iterate error is upper bounded as

$$\operatorname{Error}(\overline{\Psi}_T)\leq\frac{1}{T}\sup_{Z\in\mathcal{C}}\sum_{t=1}^{T}\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Psi_t-Z\right)\right]. \tag{76}$$

In order to produce an explicit bound, we begin by decomposing the terms of the summation, as follows, resulting in a sum of three distinct terms: (A_t), (B_t), and (C_t).

$$\begin{aligned}
&\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Psi_t-Z\right)\right]=\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Psi_t-\Phi_t+\Phi_t-Z\right)\right]\\
&=\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Psi_t-\Phi_t\right)+\mathcal{F}(\Psi_t)(\Phi_t-Z)\right]=\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Psi_t-\Phi_t\right)\right]+\operatorname{Tr}\left[(\Psi_t)\left(\Phi_t-Z\right)\right]\\
&=\operatorname{Tr}\left[\left(\mathcal{F}(\Psi_t)-\mathcal{F}(\Psi_{t-1})+\mathcal{F}(\Psi_{t-1})\right)\left(\Psi_t-\Phi_t\right)\right]+\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Phi_t-Z\right)\right]\\
&=\operatorname{Tr}\left[\left(\mathcal{F}(\Psi_t)-\mathcal{F}(\Psi_{t-1})\right)\left(\Psi_t-\Phi_t\right)+\mathcal{F}(\Psi_{t-1})(\Psi_t-\Phi_t)\right]+\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Phi_t-Z\right)\right]\\
&=\underbrace{\operatorname{Tr}\left[\mathcal{F}(\Psi_t)\left(\Phi_t-Z\right)\right]}_{(A_t)}+\underbrace{\operatorname{Tr}\left[\left(\mathcal{F}(\Psi_t)-\mathcal{F}(\Psi_{t-1})\right)\left(\Psi_t-\Phi_t\right)\right]}_{(B_t)}+\underbrace{\operatorname{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(\Psi_t-\Phi_t\right)\right]}_{(C_t)}.
\end{aligned} \tag{77}$$

In the following analysis, we will show that terms (A_t) and (C_t) can be upper bounded via the optimality of the proximal updates in the OMMP algorithm. Furthermore, the term (B_t) can be upper bounded via the strong convexity, smoothness, and duality of the OMMP proximal updates.

4.3 Optimality Analysis of (A_t)

We begin by upper bounding the term (A_t). Recall from the definition of the OMMP algorithm in Section 3.3 that Φ_t is the outcome of the proximal projection of Equation (71), for a regularization function h. Therefore, Φ_t can be expressed as

$$\Phi_t = \arg\min_{Z \in \mathcal{C}} \phi_t(Z), \tag{78}$$

where $\phi_t(Z)$ is the proximal optimization term

$$\phi_t(Z) = \mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(Z - \Phi_{t-1}\right)\right] + \frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1}). \tag{79}$$

Note that ϕ_t is differentiable, with gradient

$$\nabla_{Z^\top}\phi_t(Z) = \mathcal{F}(\Psi_t) + \frac{1}{\eta}\nabla_{Z^\top}\mathcal{BD}_h(Z\|\Phi_{t-1}). \tag{80}$$

Using the definitions of Section 2.3, in the supplement (Appendix D) we further show that:

Proposition 3 *For a μ-strongly convex regularization function h with respect to the norm $\|\cdot\|$, ϕ_t is $\frac{\mu}{\eta}$-strongly convex with respect to the norm $\|\cdot\|$.*

Since ϕ_t is strongly convex and differentiable it satisfies the First-Order Optimality Condition (Fact 1) for a feasible momentum solution Φ_t,

$$\mathrm{Tr}\left[\nabla_{\Phi_t^\top}\phi_t(\Phi_t)\left(Z - \Phi_t\right)\right] \geq 0, \;\; \forall Z \in \mathcal{C}. \tag{81}$$

Plugging the gradient of Equation (80) into this expression gives

$$\begin{aligned}
&\mathrm{Tr}\left[\left(\mathcal{F}(\Psi_t) + \frac{1}{\eta}\nabla_{\Phi_t^\top}\mathcal{BD}_h(\Phi_t||\Phi_{t-1})\right)\left(\Phi_t - Z\right)\right] \leq 0 \\
&\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(\Phi_t - Z\right)\right] + \frac{1}{\eta}\mathrm{Tr}\left[\nabla_{\Phi_t^\top}\mathcal{BD}_h(\Phi_t||\Phi_{t-1})\left(\Phi_t - Z\right)\right] \leq 0 \\
&\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(\Phi_t - Z\right)\right] \leq \frac{1}{\eta}\mathrm{Tr}\left[-\nabla_{\Phi_t^\top}\mathcal{BD}_h(\Phi_t||\Phi_{t-1})\left(\Phi_t - Z\right)\right].
\end{aligned} \tag{82}$$

From our definition of (A_t) in Equation (77) and the three-point identity for the Bregman divergence given in Equation (42), it follows that, $\forall Z \in \mathcal{C}$,

$$(A_t) \leq \frac{1}{\eta}\Big(\mathcal{BD}_h(Z||\Phi_{t-1}) - \mathcal{BD}_h(\Phi_t||\Phi_{t-1}) - \mathcal{BD}_h(Z||\Phi_t)\Big). \tag{83}$$

4.4 Optimality Analysis of (C_t)

We will now upper bound (C_t), similarly to how we upper bounded (A_t). Recall from the definition of the OMMP algorithm in Section 3.3 that Ψ_t is the outcome of the proximal projection of Equation (70), for a regularization function h. Therefore, Ψ_t can be expressed as

$$\Psi_t = \arg\min_{Z \in \mathcal{C}} \psi_t(Z), \tag{84}$$

where $\psi_t(Z)$ is the proximal optimization term

$$\psi_t(Z) = \mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(Z - \Phi_{t-1}\right)\right] + \frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1}). \tag{85}$$

Note that ϕ_t is differentiable, with gradient

$$\nabla_{Z^\top}\psi_t(Z) = \mathcal{F}(\Psi_{t-1}) + \frac{1}{\eta}\nabla_{Z^\top}\mathcal{BD}_h(Z\|\Phi_{t-1}). \tag{86}$$

Similarly as in the previous section, in the supplement (Appendix D) we further show that:

Proposition 4 *For a μ-strongly convex regularization function h with respect to the norm $\|\cdot\|$, ψ_t is $\frac{\mu}{\eta}$-strongly convex with respect to the norm $\|\cdot\|$.*

Since ψ_t is strongly convex and differentiable, it satifies the First-Order Optimality Condition (Fact 1) for a feasible state solution Ψ_t,

$$\mathrm{Tr}\left[\nabla_{\Psi_t^\top}\psi_t(\Psi_t)\,(Z-\Psi_t)\right] \geq 0,\ \ \forall Z \in \mathcal{C}. \tag{87}$$

Plugging the gradient of Equation (86) into this expression gives, $\forall Z \in \mathcal{C}$,

$$\mathrm{Tr}\left[\left(\mathcal{F}(\Psi_{t-1}) + \frac{1}{\eta}\nabla_{\Psi_t^\top}\mathcal{BD}_h(\Psi_t||\Phi_{t-1})\right)(\Psi_t - Z)\right] \leq 0 \tag{88}$$

$$\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\,(\Psi_t - Z)\right] \leq \frac{1}{\eta}\mathrm{Tr}\left[\nabla_{\Psi_t^\top}\mathcal{BD}_h(\Psi_t||\Phi_{t-1})\,(Z-\Psi_t)\right]. \tag{89}$$

Since the expression holds for all $Z \in \mathcal{C}$, we can choose to set $Z = \Phi_t$, which implies

$$\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\,(\Psi_t - \Phi_t)\right] \leq \frac{1}{\eta}\mathrm{Tr}\left[\nabla_{\Psi_t^\top}\mathcal{BD}_h(\Psi_t||\Phi_{t-1})\,(\Phi_t-\Psi_t)\right]. \tag{90}$$

From our definition of (C_t) in Equation (77) and the Three-Point Tdentity for the Bregman divergence, given in Equation (42), it follows that

$$(C_t) \leq \frac{1}{\eta}\Big(\mathcal{BD}_h(\Phi_t||\Phi_{t-1}) - \mathcal{BD}_h(\Psi_t||\Phi_{t-1}) - \mathcal{BD}_h(\Phi_t||\Psi_t)\Big). \tag{91}$$

4.5 An Upper Bound for (B_t)

By the Cauchy-Schwarz inequality for Frobenius norm, we can upper bound (B_t) as

$$(B_t) = \mathrm{Tr}\left[(F(\Psi_t) - F(\Psi_{t-1}))\,(\Psi_t - \Phi_t)\right] \leq \|F(\Psi_t) - F(\Psi_{t-1})\|_F\|\Psi_t - \Phi_t\|_F. \tag{92}$$

or more generally, with respect to the norms $(\|\cdot\|, \|\cdot\|_*)$

$$(B_t) = \mathrm{Tr}\left[(F(\Psi_t) - F(\Psi_{t-1}))\,(\Psi_t - \Phi_t)\right] \leq \|F(\Psi_t) - F(\Psi_{t-1})\|_*\|\Psi_t - \Phi_t\|. \tag{93}$$

To further manipulate this upper bound for (B_t), we leverage the ideas from Fenchel duality and convexity introduced in Section 2.3.

Following from Equation (84) and Equation (79), respectively, Ψ_t is the minimizer of ψ_t and Φ_t is the minimizer of ϕ_t. Re-expressing ϕ_t in terms of ψ_t,

$$\phi_t(U) = \psi_t(U) + \mathrm{Tr}\left[(\mathcal{F}(\Psi_t) - \mathcal{F}(\Psi_{t-1}))\,(U - \Phi_{t-1})\right] \tag{94}$$

implies that Φ_t can be expressed in terms of a minimization of ψ_t,

$$\Phi_t = \arg\min_{U\in\mathcal{C}}\left\{\psi_t(U) + \mathrm{Tr}\left[(\mathcal{F}(\Psi_t) - \mathcal{F}(\Psi_{t-1}))\,U\right]\right\}. \tag{95}$$

Since we assumed that h is μ-strongly convex, by Proposition 4 ψ_t must be $\frac{\mu}{\eta}$-strongly convex. Denoting the Fenchel conjugate of ψ_t as ψ_t^*, since ψ_t is strongly convex, Danskin's Theorem (Proposition 2) states that $\nabla\psi_t^*$ is well-defined, for all $V \in \mathcal{C}$, as

$$\nabla\psi_t^*(V) = \arg\min_{U\in\mathcal{C}} \{\psi_t(U) - \operatorname{Tr}[V\,U]\}. \tag{96}$$

In the case that $V = 0$, Equation (78) implies that

$$\nabla\psi_t^*(0) = \arg\min_{U\in\mathcal{C}} \psi_t(U) = \Psi_t. \tag{97}$$

Meanwhile, if $V = \mathcal{F}(\Psi_{t-1}) - \mathcal{F}(\Psi_t)$, Equation (95) implies that

$$\nabla\psi_t^*\left(\mathcal{F}(\Psi_{t-1}) - \mathcal{F}(\Psi_t)\right) = \arg\min_{U\in\mathcal{C}} \{\psi_t(U) + \operatorname{Tr}[(\mathcal{F}(\Psi_t) - \mathcal{F}(\Psi_{t-1}))\,U]\} = \Phi_t. \tag{98}$$

Therefore,

$$\|\Psi_t - \Phi_t\|_F = \|\nabla\psi_t^*(0) - \nabla\psi_t^*\left(\mathcal{F}(\Psi_{t-1}) - \mathcal{F}(\Psi_t)\right)\|_F. \tag{99}$$

Furthermore, since ψ_t is $\frac{\mu}{\eta}$-strongly convex with respect to the norm $\|\cdot\|$, Strong-Smooth Duality (Lemma 3) implies that ψ_t^* is $\frac{\eta}{\mu}$-smooth with respect to the norm $\|\cdot\|_*$. By Proposition 1, this implies that $\nabla\psi_t^*$ is $\frac{\eta}{\mu}$-Lipschitz continuous with respect to $\|\cdot\|_*$,

$$\|\nabla\psi_t^*(X) - \nabla\psi_t^*(Y)\| = \|\nabla\psi_t^*(X) - \nabla\psi_t^*(Y)\|_{**} \leq \frac{\eta}{\mu}\|X - Y\|_*. \tag{100}$$

Plugging this and Equation (99) into Equation (92) produces the final upper bound for (B_t),

$$(B_t) \leq \frac{\eta}{\mu}\|\mathcal{F}(\Psi_t) - \mathcal{F}(\Psi_{t-1})\|_*^2. \tag{101}$$

4.6 Average-Iterate Error Upper Bound

Combining the average-iterate error upper bound of Equation (76) and its decomposition, as expressed in Equation (77), we have that

$$\operatorname{Error}(\overline{\Psi}_T) \leq \frac{1}{T}\sup_{Z\in\mathcal{C}}\sum_{t=1}^{T}\operatorname{Tr}[\mathcal{F}(\Psi_t)\,(\Psi_t - Z)] = \frac{1}{T}\sup_{Z\in\mathcal{C}}\sum_{t=1}^{T}(A_t) + (B_t) + (C_t).$$

Plugging in the upper bounds for (A_t) from Equation (83), (B_t) from Equation (101), and (C_t) from Equation (91) gives

$$\begin{aligned}
\operatorname{Error}(\overline{\Psi}_T) &\leq \frac{1}{T}\sup_{Z\in\mathcal{C}}\sum_{t=1}^{T}\left(\frac{1}{\eta}\Big(\mathcal{BD}_h(Z\|\Phi_{t-1}) - (\mathcal{BD}_h(Z\|\Phi_t) + \mathcal{BD}_h(\Psi_t\|\Phi_{t-1}) + \mathcal{BD}_h(\Phi_t\|\Psi_t))\Big)\right. \\
&\qquad \left. + \frac{\eta}{\mu}\|\mathcal{F}(\Psi_t)^\top - \mathcal{F}(\Psi_{t-1})^\top\|_*^2\right) \\
&\leq \frac{1}{T}\sup_{Z\in\mathcal{C}}\underbrace{\frac{1}{\eta}\sum_{t=1}^{T}(\mathcal{BD}_h(Z\|\Phi_{t-1}) - \mathcal{BD}_h(Z\|\Phi_t))}_{(I)} - \frac{1}{\eta}\sum_{t=1}^{T}(\mathcal{BD}_h(\Psi_t\|\Phi_{t-1}) + \mathcal{BD}_h(\Phi_t\|\Psi_t)) \\
&\qquad + \frac{\eta}{\mu}\sum_{t=1}^{T}\|\mathcal{F}(\Psi_t)^\top - \mathcal{F}(\Psi_{t-1})^\top\|_*^2.
\end{aligned}$$

Via telescoping sums, the first term (I) can be reduced to

$$(I) = \frac{1}{\eta}\sum_{t=1}^{T}\left(\mathcal{BD}_h(Z\|\Phi_{t-1}) - \mathcal{BD}_h(Z\|\Phi_t)\right) = \frac{1}{\eta}\Big(\mathcal{BD}_h(Z\|\Phi_0) - \mathcal{BD}_h(Z\|\Phi_T)\Big). \tag{102}$$

Additionally, since h is μ-strongly convex with respect to $\|\cdot\|$,

$$\mathcal{BD}_h(X\|Y) \geq \frac{\mu}{2}\|X - Y\|^2. \tag{103}$$

Therefore, we can re-express the error as

$$\text{Error}(\overline{\Psi}_T) \leq \frac{1}{T}\sup_{Z\in\mathcal{C}}\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_0)\underbrace{-\frac{\mu}{2\eta}\|Z-\Phi_T\|^2 + -\frac{\mu}{2\eta}\sum_{t=1}^{T}\left(\|\Phi_{t-1}-\Psi_t\|^2 + \|\Phi_t - \Psi_t\|^2\right)}_{(II)}$$
$$+\underbrace{\frac{\eta}{\mu}\sum_{t=1}^{T}\|\mathcal{F}(\Psi_t)^\top - \mathcal{F}(\Psi_{t-1})^\top\|_*^2}_{(III)},$$

and proceed to upper bound terms (II) and (III). We begin by expanding (II) as

$$\begin{aligned}(II) &= -\frac{\mu}{2\eta}\|\Phi_T - Z\|^2 - \frac{\mu}{2\eta}\sum_{t=1}^{T}\left(\|\Phi_{t-1}-\Psi_t\|^2 + \|\Phi_t-\Psi_t\|^2\right)\\ &= \frac{\mu}{2\eta}\left(\|\Phi_0-\Psi_0\|^2 - \|\Phi_T-\Psi_T\|^2 - \|\Phi_T - Z\|^2 - \sum_{t=1}^{T}\left(\|\Phi_{t-1}-\Psi_t\|^2 + \|\Phi_{t-1}-\Psi_{t-1}\|^2\right)\right)\end{aligned}$$

Since we initialized $\Phi_0 = \Psi_0$,

$$\|\Phi_0 - \Psi_0\|^2 = 0. \tag{104}$$

Furthermore, since $\|\Phi_T - Z\|^2 \geq 0$ for all $Z \in \mathcal{C}$, including $Z = \Psi_T$,

$$(II) \leq -\frac{\mu}{2\eta}\sum_{t=1}^{T}\left(\|\Phi_{t-1}-\Psi_t\|^2 + \|\Phi_{t-1}-\Psi_{t-1}\|^2\right). \tag{105}$$

Defining

$$\Gamma := \sum_{t=1}^{T}\left(\|\Psi_t - \Phi_{t-1}\|^2 + \|\Psi_{t-1}-\Phi_{t-1}\|^2\right) \tag{106}$$

implies the following upper bound on (II)

$$(II) \leq -\frac{\mu}{2\eta}\Gamma. \tag{107}$$

In order to bound term (III), we will leverage an important property of finite-valued quantum zero-sum games. Namely, in Appendix D, we prove several properties of the $\mathcal{F}$ operator, including the following lemma regarding Lipschitz continuity.

Lemma 9 (Lipschitz Continuity of $\mathcal{F}$) *For finite-valued quantum zero-sum games, $\mathcal{F}$ is a Lipschitz-continuous operator, meaning there exists a Lipschitz constant $\gamma \in \mathbb{R}$ such that*

$$\|\mathcal{F}(Z) - \mathcal{F}(Z')\|_* \leq \gamma\|Z - Z'\|, \ \forall\, Z, Z' \in \mathcal{C}. \tag{108}$$

For the case of $(\|\cdot\|_F, \|\cdot\|_F)$ and $(\|\cdot\|_\infty, \|\cdot\|_1)$, $\gamma_{F,F} = \mathcal{O}(2^d)$ while $\gamma_{\infty,1} = \mathcal{O}(1)$.

Leveraging the Lipschitz continuity of $\mathcal{F}$, as given in Equation (108), and the fact that for an arbitrary norm $\|\cdot\|$

$$\|X-Y\|^2 \leq (\|X\|+\|Y\|)^2 \leq \|X\|^2+\|Y\|^2+2\|X\|\|Y\| \leq 2\|X\|^2+2\|Y\|^2, \tag{109}$$

(III) can be bounded as

$$\begin{aligned}
(III) &= \frac{\eta}{\mu}\sum_{t=1}^{T}\|\mathcal{F}(\Psi_t)-\mathcal{F}(\Psi_{t-1})\|_*^2 \leq \frac{\eta}{\mu}\sum_{t=1}^{T}\gamma^2\|\Psi_t-\Psi_{t-1}\|^2 \\
&= \frac{\gamma^2\eta}{\mu}\sum_{t=1}^{T}\|\Psi_t-\Phi_{t-1}+\Phi_{t-1}-\Psi_{t-1}\|^2 \\
&= \frac{\gamma^2\eta}{\mu}\sum_{t=1}^{T}\|(\Psi_t-\Phi_{t-1})-(\Psi_{t-1}-\Phi_{t-1})\|^2 \\
&\leq \frac{2\gamma^2\eta}{\mu}\Big(\sum_{t=1}^{T}\|\Psi_t-\Phi_{t-1}\|^2+\|\Psi_{t-1}-\Phi_{t-1}\|^2\Big) \\
&= \frac{2\gamma^2\eta}{\mu}\,\Gamma.
\end{aligned} \tag{110}$$

Therefore, combining the bound on (II) from Equation (107), with that of (III), from Equation (110), results in

$$(II)+(III) \leq \left(\frac{2\gamma^2\eta}{\mu}-\frac{\mu}{2\eta}\right)\Gamma. \tag{111}$$

Note that there exist step sizes η, such that $(II)+(III)\leq 0$. Namely,

$$\eta \leq \frac{\mu}{2\gamma} \quad\Longrightarrow\quad (II)+(III)\leq 0. \tag{112}$$

Recall that our objective was to upper bound $\mathrm{Error}(\overline{\Psi}_T)$, where

$$\mathrm{Error}(\overline{\Psi}_T) \leq \frac{1}{T}\sup_{Z\in\mathcal{C}}\frac{1}{\eta}\mathcal{BD}_h(\Phi_0\|Z)+(II)+(III), \tag{113}$$

If we choose the step size such that $\eta \leq \frac{\mu}{2\gamma}$, the upper bound simplifies to

$$\mathrm{Error}(\overline{\Psi}_T) \leq \frac{1}{T}\sup_{Z\in\mathcal{C}}\frac{1}{\eta}\mathcal{BD}_h(\Phi_0\|Z). \tag{114}$$

Since $\mathcal{BD}_h(\Phi_0\|Z)$ is bounded above by the Bregman diameter of $\mathcal{C}$, i.e.

$$\mathcal{BD}_h(\Phi_0\|Z) \leq \sup_{X,Y\in\mathcal{C}}\mathcal{BD}_h(X\|Y), \tag{115}$$

it has no time dependence, meaning

$$\mathrm{Error}(\overline{\Psi}_T) = \mathcal{O}_d\left(\frac{1}{T}\right),$$

where $\mathcal{O}_d$ is used to refer to the convergence dependence on the spectraplex dimension d, governed by the Bregman diameter of $\mathcal{C}$, as given in Equation (115). Therefore, setting $\mathrm{Error}(\overline{\Psi}_T)$ to a desired accuracy parameter, $\mathrm{Error}(\overline{\Psi}_T)=\epsilon$, we obtain that the number of steps necessary for average-iterate convergence to an ϵ-Nash equilibrium is

$$T = \mathcal{O}_d\left(\frac{1}{\epsilon}\right). \tag{116}$$

4.7 Regularizer-Induced Dimensionality Dependence

Following from this analysis, for a fixed dimension d, the family of algorithms encompassed by OMMP can compute an ϵ-Nash Equilibrium in $\mathcal{O}_d(1/\epsilon)$ steps. Note, however, that the $\mathcal{O}_d(\cdot)$ notation hides the dimension-dependence of the result, incurred by the $\frac{1}{\eta}\mathcal{BD}_h(\Phi_0\|Z)$ term. In order to achieve an explicit bound, we will demonstrate how different OMMP regularizers affect the Bregman divergence $\mathcal{BD}_h(\Phi_0\|Z)$ and step size η. In particular, we consider two regularizers—the Frobenius norm and the von Neumann entropy—which are quantum analogs of two of the most widely used classical regularizers (entropy and Euclidean norm). Note that OMMP instantiated with the Frobenius norm implements the *Optimistic Matrix Extra-Gradient (OMEG) algorithm* (updates given in Figure 2), while the von Neumann entropy regularizer implements the *Optimistic Matrix Multiplicative Weights Update (OMMWU) algorithm.*

We will first consider the Bregman divergence term, noting that in OMMP we initialize $\Phi_0 = \left(\frac{1}{2^n}\mathbb{1}_{\mathcal{A}}, \frac{1}{2^m}\mathbb{1}_{\mathcal{B}}\right)$. We offer an explicit proof for the Frobenius norm regularizer:

Proposition 5 *The Frobenius norm regularizer induces a maximum Bregman divergence that is constant:* $\max_{X\in\mathcal{C}} \mathcal{BD}_h(\Phi_0||X) = 2 = \mathcal{O}(1)$.

Proof. Using the derivation of the Bregman divergence for the Frobenius norm in Equation (52) and the fact that $\forall X \in \mathcal{C}$, $\mathrm{Tr}(X) = \mathrm{Tr}(\alpha) + \mathrm{Tr}(\beta) \leq 1 + 1 = 2$,

$$\begin{aligned}\max_{X\in\mathcal{C}} \mathcal{BD}_h(\Phi_0||X) = \max_{X\in\mathcal{C}} \tfrac{1}{2}||\Phi_0 - X||_F^2 &= \tfrac{1}{2}\max_{X\in\mathcal{C}}\left(||\Phi_0||_F^2 - 2\mathrm{Tr}(\Phi_0^\dagger X) + ||X||_F^2\right)\\ &\leq \tfrac{1}{2}\max_{X\in\mathcal{C}}\left(2 + ||X||_F^2\right) \leq \tfrac{1}{2}(2+2) = 2.\end{aligned}$$

□

In the case of the von Neumann entropy regularizer, as discussed in [73], for a $(4^d - 1)$-dimensional spectraplex, the corresponding bound on the Bregman divergence is equal to the maximum quantum entropy for a $2d$-qubit system, and hence linear in d—more precisely, equal to $\log\left(4^d - 1\right) = \mathcal{O}(d)$.

We now consider the step size. Following from Equation (112), the optimal step size is dependent on h's strong convexity parameter (μ_h) and $\mathcal{F}$'s Lipschitz constant ($\gamma_{\mathcal{F}}$), as

$$\eta = \Theta\left(\mu_h/\gamma_{\mathcal{F}}\right). \tag{117}$$

In scenarios utilizing the Frobenius regularizer, the modulus of strong convexity is 1, while γ_{Lipsch}, the Lipschitz constant with respect to $\|\cdot\|_F$ of the gradient for games with bounded utilities in $[-1, 1]$, is of the order of $\mathcal{O}(\sqrt{4^d}) = \mathcal{O}(2^d)$. In contrast, for the von Neumann entropy, the modulus of strong convexity is 1 with respect to the Shatten 1-norm (the matrix analog of the ℓ_1-norm) [74]. Moreover, for games with bounded utilities, the relative Lipschitz constant with respect to the Shatten 1-norm $\|\cdot\|_1$ is of the order of $\mathcal{O}(1)$.

In light of the preceding discussions, we conclude by stating the dimension-dependent iteration complexities of OMMWU and OMEG for achieving ϵ-Nash equilibria of quantum zero-sum games:

Corollary 1 (OMMWU Iteration Complexity) *In a 4^d-dimensional spectraplex, OMMWU computes average-iterate ϵ-Nash equilibria in $\mathcal{O}(d/\epsilon)$ iterations.*

Corollary 2 (OMEG Iteration Complexity) *In the 4^d-dimensional spectraplex, OMEG computes average-iterate ϵ-Nash equilibria in $\mathcal{O}(2^d/\epsilon)$ iterations.*

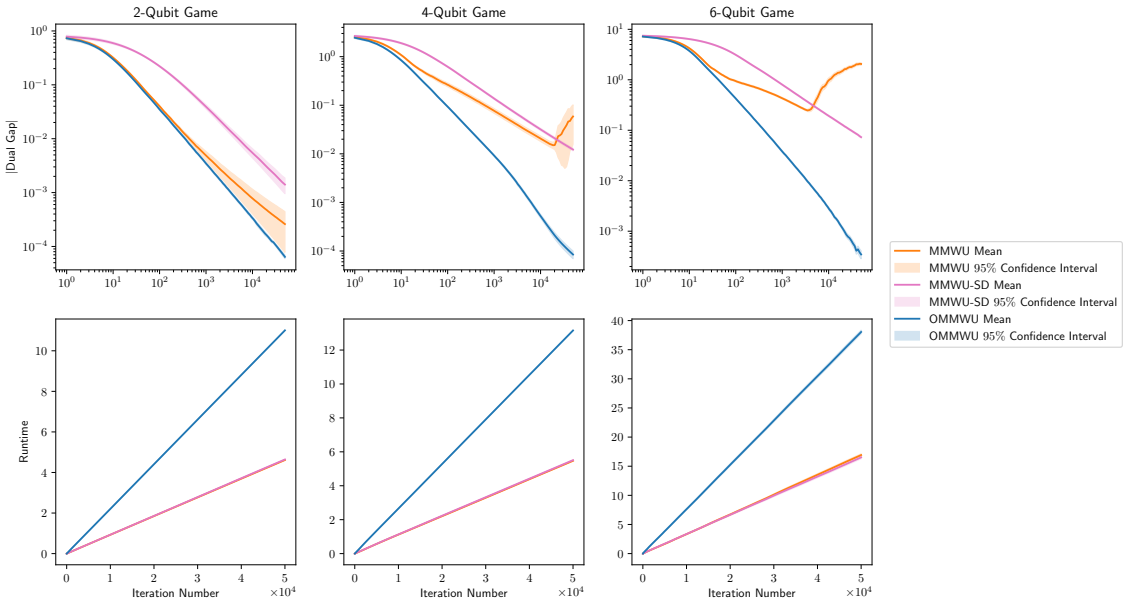


Figure 9: **Experimental Results I.** An assessment of the relative average-case performance of MMWU, MMWU-SD (MMWU with step-size decay), and OMMWU for quantum zero-sum games with a single step size. Experiments are run in the 2-, 4-, and 6-qubit settings, where Alice and Bob each play with 1-, 2-, and 3-qubits, respectively. In each setting, for 50 randomly generated games, the algorithms are run for 50,000 iterations. The figures report the absolute value of the dual gap and runtime as a function of iteration number. Although the OMMWU runtime is roughly double that of MMWU and MMWU-SD, it achieves a roughly quadratic improvement in convergence for average-case games involving 4 or more qubits. OMMWU is also more numerically stable than MMWU.

Not only does OMMWU match the dimension-dependece of MMWU, but it also logarithmically outperform the dimension dependence of OMEG. Thus, between the Frobenius norm (OMEG) and von Neumann entropy (OMMWU), the latter is the clear choice for efficient implementation of the OMMP method for quantum zero-sum games played in the spectraplex.

5 Experiments

In the last section, we proved that, for worst-case quantum zero-sum games, OMMWU obtains a quadratic speedup relative to MMWU. In practice, however, often more important than the worst-case performance is the average-case performance of an algorithm. Thus, in this section we explore the relative average-case performance of MMWU and OMMWU. In particular, we ran several numerical experiments involving randomly generated small-scale quantum zero-sum games to assess the relative average-case performance of these algorithms. The results of these experiments are reported in Figure 9 and Figure 10. The code and experimental data is provided at `https://github.com/FranciscaVasconcelos/qzsg`. For the reported runtimes, note that the experiments were run on a standard laptop (Apple M1 Pro).

Two distinct experiments were conducted, but, in both, the algorithms were run in 2-, 4-, and 6-qubit quantum zero-sum game settings. For each of these game sizes, Alice and Bob play with 1-, 2-, and 3-qubits, respectively. Furthermore, for each of these settings the quantum zero-sum game payoff observables are randomly generated, with full-rank POVMs.

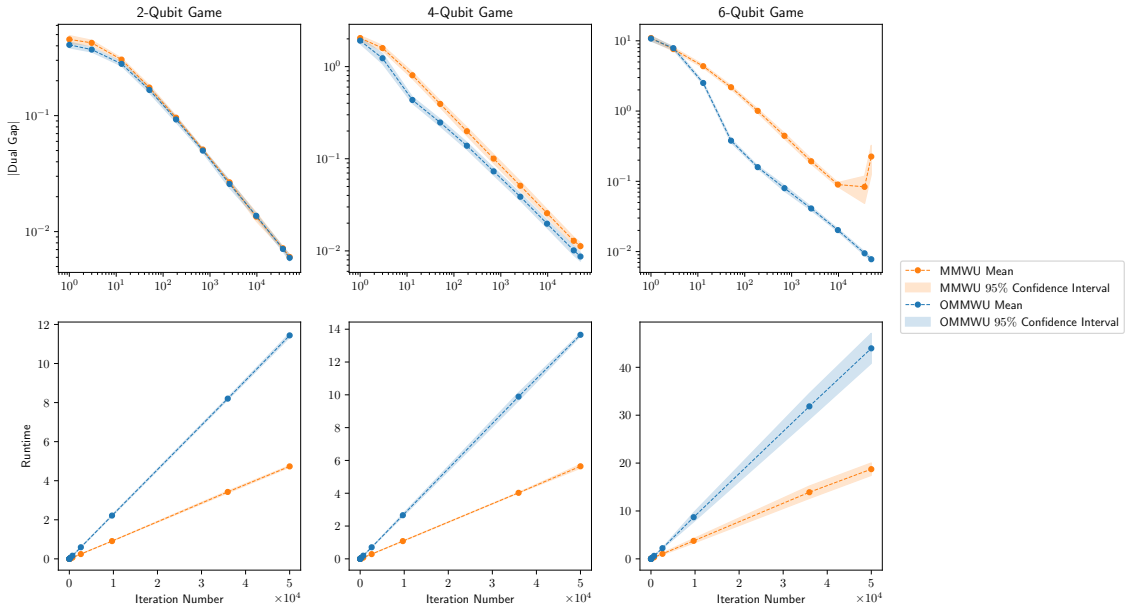


Figure 10: **Experimental Results II.** An assessment of the relative average-case performance of MMWU and OMMWU for quantum zero-sum games, with step size determined by iteration number. Experiments are run in the 2-, 4-, and 6-qubit settings, where Alice and Bob each play with 1-, 2-, and 3-qubits, respectively. In each setting, the algorithms are run for ten iteration numbers $t \in \{1, 3, ..., 49, 999\}$ for 10 randomly generated quantum zero-sum games. For each run, the step size of both MMWU and OMMWU are set to exactly $1/\sqrt{t}$. The figures report the absolute value of the dual gap and runtime for each iteration t (note that the dotted lines are interpolations). Although the runtime of OMMWU is roughly double that of MMWU and MMWU-SD, it achieves a notable improvement in convergence for average-case games of higher dimensions (the gains are roughly quadratic for 6-qubit games).

In the first experiment (results reported in Figure 9), MMWU and OMMWU were run for 50,000 iterations for each of 50 randomly generated games. However, we found MMWU to be numerically unstable for larger iteration numbers, so we also implemented MMWU with a $1/\sqrt{t+1}$ step-size decay multiplicative factor, which we refer to as MMWU-SD.[10] For all three algorithms (MMWU, MMWU-SD, and OMMWU), Figure 9 reports the mean value and 95% confidence intervals across all 50 games for the absolute value of the dual gap and the runtime as a function of iteration number. As expected, the runtime of OMMWU is roughly twice that of MMWU and MMWU-SD. However, for the larger game sizes (in the 4- and 6-qubit game settings), OMMWU appears to obtain an approximate quadratic speedup, even in this average-case setting. Note that OMMWU is also much more stable numerically than MMWU. Therefore, even though each iteration of OMMWU has a longer runtime, its worst-case guarantees and apparent average-case convergence performance indicate that even for fairly small quantum zero-sum games, OMMWU can converge to high-precision Nash equilibria much faster than MMWU.

A potential concern with the first experiment is that Figure 9 reports the MMWU time average $\bar{t}$ for runs of t iterations with a *constant* step size, but the theoretical optimization guarantee of MMWU is a time average of $\bar{t}$ for runs of t iterations with an $O(1/t)$ step size. In particular, MMWU concerns *one* specific output with a predetermined step size, whereas OMMWU is an anytime guarantee, valid for any constant step size below a certain

[10] Step-size decay helps prevent error blow-ups as the number of iterations increases.

value. The standard workaround in the literature is to run MMWU with a variable step size, which we do with our implementation of MMWU-SD. To see why this is a valid workaround, note that the difference between a variable $1/\sqrt{t}$ policy and a constant $1/\sqrt{t}$ policy at the output of the algorithm is $O(\log t)$ and, thus, negligible. However, the $1/\sqrt{t}$ step size for MMWU provides an anytime guarantee which enables comparison with other anytime methods, such as OMMWU.

We also addressed the potential concern regarding step sizes by running a second experiment in which, for each of the game sizes (2-, 4-, and 6- qubits), MMWU and OMMWU are run for 10 randomly generated games for each of the ten iteration numbers

$$t \in \{1,\ 3,\ 13,\ 51,\ 189,\ 703,\ 2,610,\ 9,687,\ 35,949,\ 49,999\},$$

with a step size of exactly $1/\sqrt{t}$ (note that no hyperparameter tuning was performed). (results reported in Figure 10). Figure 10 reports the mean values of the duality gap of the final state reached after all t iterations and the corresponding runtime as dots (note that the dashed lines are solely interpolating between the ten iteration values). We also report 95% confidence intervals. The runtime of OMMWU is still double that of MMWU and there is not much difference from MMWU for the 2-qubit game. However, for 4-qubit games OMMWU already begins to outperform MMWU and for 6-qubit games there is nearly an order of magnitude improvement in accuracy relative to MMWU despite the constant increase in runtime. Also note that OMMWU is numerically stable, whereas MMWU becomes numerically unstable for larger iteration numbers. Overall, these experiments offer convincing evidence that OMMWU outperforms MMWU even for average-case games.

6 Discussion

In this work, we have aimed to broaden our understanding of solution methods for quantum zero-sum game algorithms. We demonstrated that the best existing method for finding ϵ-Nash equilibria for quantum zero-sum games—Jain and Watrous' Matrix Multiplicative Weight Updates (MMWU) algorithm [29]—fits within the framework as a particular instantiation of a general Matrix Dual Averaging (MDA) method. To improve upon MMWU, we leveraged a gradient-based view, which enabled us to draw from the large classical game theory and optimization literature for solving variational inequalities of monotone, Lipschitz operators. Making us of this gradient-based perspective, proximal steps, and optimistic updates we established convergence guarantees for a general Optimistic Matrix Mirror Prox (OMMP) method. When instantiated with the von Neumann entropy regularizer, OMMP implements the Optimistic Matrix Multiplicative Weights Update (OMMWU) algorithm, which we proved to have dimensionality dependence equivalent to that of MMWU. However, OMMWU establishes a new benchmark in convergence rate, achieving a quadratic speed-up relative MMWU, while still requiring only one gradient call per iteration.

Given these contributions several challenges remain and several interesting open problems arise:

1. **Alternative Methods**: In this work we primarily compared MMWU and methods related to OMMWU. It would be interesting to explore and compare other semi-definite programming methods for solving for approximate Nash-equilibria.

2. **Dimensionality of Quantum Games**: Our work focuses on finite quantum zero-sum games. Extending the applicability of our algorithms to games in infinite-dimensional Hilbert spaces remains an open challenge that could have theoretical applications to

operator analysis and practical implications to derive new optimal transport techniques.

3. **Robustness to Noise**: Quantum systems are inherently prone to noise. Important for practical implementations is how our method performs with different kind of feedback under noisy measurement conditions.

4. **Computational Scaling Challenges**: Our proposal requires knowledge of the entire system density matrix in each iteration, which is computationally prohibitive for large system sizes. As pointed out by [29], $\mathsf{QMA} \subseteq \mathsf{QRG}(1)$, which implies that the problem we are studying, as currently formulated, is in fact extremely hard in the worst case. Nevertheless, it would be valuable to explore more efficient classical representations of the system state or propose entirely quantum protocols/algorithms which can eliminate the need to classically learn/store the system state.

5. **Going Beyond 2-Player Zero-Sum**: Early forays into quantum potential games [75] and quantum zero-sum polymatrix games [47] are suggestive of possibilities for further research with novel classes of quantum games and solution concepts.

6. **Entangled Games**: In this work, we only consider quantum zero-sum games in which the players' states are unentangled. However, it would be interesting to explore the effects of entanglement between the players' states.

7. **No-Regret Dynamics**: Since OMMWU and MMP are no-regret frameworks, it would be interesting to explore generalizations of classical game-theoretic solution concepts, such as correlated equilibria and coarse correlated equilibria, for these algorithms. For example, such ideas have recently been explored in the quantum setting by [47, 76, 77]. In particular, [47] established the existence of entangled (i.e., non-separable) QCCEs which cannot be approached by any no-regret algorithm.

8. **Tightness of Convergence Rates**: Our first conjecture posits that the $O(1/\epsilon)$ convergence rate of 1-MMP is tight, but empirical or theoretical confirmation remains elusive. Establishing the tightness would clarify the limitations of our method.

Conjecture 1 *The rates of Theorem 1 are tight: specifically, there exists a quantum two-player, zero-sum game for which OMMP methods require $\Omega_d(1/\epsilon)$ iterations to compute ϵ-Nash equilibria.*

With respect to Conjecture 1, in classical game theory, there are two well-established algorithms which achieve a convergence rate of $\log(1/\epsilon)$, thereby surpassing the $O(1/\epsilon)$ of our OMMP analysis. However, it is unclear, for reasons elaborated below, whether the techniques of these approaches are applicable to quantum spectraplexes. Therefore, it remains an *open question* whether a $\log(1/\epsilon)$ convergence rate can be achieved or if there is some inherent limit in the quantum setting which prohibits such performance.

The first such classical approach (an older approach in the literature) leverages the *duality of linear programming (LP)* by transforming the computation of a Nash equilibrium into the dual of an LP. Consequently, the problem can be efficiently solved using the ellipsoid method or an interior-point method. However, both techniques exhibit strong dependence on the *dimensionality of the game*. Attempting to apply the same technique to a corresponding semidefinite program (SDP) would require verifying multiple properties to enable such a reduction, and the solution would likely be feasible only for very small quantum games.

A second (more recent) classical approach uses *gradient-based algorithms*, which are known to be *dimension-independent* (as long as the game has a succinct representation and access to a feedback oracle). This category includes our work. Specifically, it has been shown that *optimistic gradient descent-ascent* can achieve a $\log(1/\epsilon)$ convergence rate in constrained saddle-point optimization problems when the problem is restricted to polytopes such as the simplex. That is a surprising result since, in general, optimistic methods have a minimax lower bound of order $O(1/\epsilon^2)$ [78]. However, upon closely examining the proof, it becomes evident that the result hinges on the fact that a Nash equilibrium in a *classical zero-sum game* can be expressed as a finite sum of the normal vectors of the faces of the simplex polytope, and that pure Nash equilibria are finite in number. Unfortunately, these properties do not hold trivially in quantum games. Specifically, *pure Nash equilibria* in the quantum setting correspond to *density matrices of rank one*, which are not only infinite but also *uncountable*. Recent work [79] also establishes explicit examples of constrained sets where *optimistic algorithms* fail to achieve better than $1/\sqrt{\epsilon}$ convergence. However, the connection between these results and the *spectraplex* is not immediately clear.

7 Acknowledgements

Part of this work was done while the authors were visiting the Simons Institute for the Theory of Computing and the Institute for Mathematical Sciences at the National University of Singapore. We thank Rahul Jain for useful discussions. F.V. is supported by NSF grant DGE-2146752 and the Paul & Daisy Soros Fellowship for New Americans. E.V. is grateful for financial support by the Post-Doctoral FODSI-Simons Fellowship. P.M. is a member of the Archimedes Unit—Athena RC—Department of Mathematics, National & Kapodistrian University of Athens, and gratefully acknowledges financial support by the French National Research Agency (ANR) in the framework of the "Investissements d'avenir" program (ANR-15- IDEX-02), the LabEx PERSYVAL (ANR-11-LABX-0025-01), MIAI@Grenoble Alpes (ANR-19- P3IA-0003), and project MIS 5154714 of the National Recovery and Resilience Plan Greece 2.0 funded by the European Union under the NextGenerationEU Program. G.P. would like to acknowledge National Research Foundation, Singapore and DSO National Laboratories under its AI Singapore Program (AISG Award No: AISG2-RP-2020-016), NRF 2018 Fellowship NRF-NRFF2018- 07, NRF2019-NRF-ANR095 ALIAS grant, grant PIESGP-AI-2020-01, AME Programmatic Fund (Grant No.A20H6b0151) from the Agency for Science, Technology and Research (A*STAR) and Provost's Chair Professorship grant RGEPPV2101. M.J. is partially supported by the Vannevar Bush Faculty Fellowship program under grant number N00014-21-1-2941 and by the European Union (ERC-2022-SYG-OCEAN-101071601).

References

[1] John Von Neumann. "Zur Theorie der Gesellschaftsspiele". Mathematische Annalen **100**, 295–320 (1928).

[2] John Nash. "Non-Cooperative Games". Annals of Mathematics **54**, 286–295 (1951).

[3] Parvathy S. Pillai and Shrisha Rao. "Resource Allocation in Cloud Computing Using the Uncertainty Principle of Game Theory". IEEE Systems Journal **10**, 637–648 (2016).

[4] Steven J. Brams. "Game theory and politics". Free Press. (1975).

[5] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. "Mastering the game of Go with deep neural networks and tree search". Nature **529**, 484–489 (2016).

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial networks". Communications of the ACM **63**, 139–144 (2020).

[7] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. "Robust adversarial reinforcement learning". In Proceedings of the 34th International Conference on Machine Learning - Volume 70. Page 2817–2826. JMLR (2017).

[8] J. S. Bell. "On the Einstein Podolsky Rosen paradox". Physics Physique Fizika **1**, 195–200 (1964).

[9] John F. Clauser, Michael A. Horne, Abner Shimony, and Richard A. Holt. "Proposed Experiment to Test Local Hidden-Variable Theories". Phys. Rev. Lett. **23**, 880–884 (1969).

[10] N. David Mermin. "Simple unified form for the major no-hidden-variables theorems". Phys. Rev. Lett. **65**, 3373–3376 (1990).

[11] Lucien Hardy. "Nonlocality for two particles without inequalities for almost all entangled states". Phys. Rev. Lett. **71**, 1665–1668 (1993).

[12] P. K. Aravind. "Bell's Theorem Without Inequalities and Only Two Distant Observers". Journal of Genetic Counseling **15**, 397–405 (2002).

[13] Daniel M. Greenberger, Michael A. Horne, and Anton Zeilinger. "Going Beyond Bell's Theorem". Pages 69–72. Springer Netherlands. (1989).

[14] Ben W. Reichardt, Falk Unger, and Umesh Vazirani. "Classical command of quantum systems". Nature **496**, 456–460 (2013).

[15] Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen. "MIP* = RE". Communnications of the ACM **64**, 131–138 (2021).

[16] Yael Kalai, Alex Lombardi, Vinod Vaikuntanathan, and Lisa Yang. "Quantum Advantage from Any Non-local Game". In Barna Saha and Rocco A. Servedio, editors, Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023. Pages 1617–1628. ACM (2023).

[17] G. Gutoski. "Upper bounds for quantum interactive proofs with competing provers". In 20th Annual IEEE Conference on Computational Complexity (CCC'05). Pages 334–343. (2005).

[18] Gus Gutoski and John Watrous. "Quantum Interactive Proofs with Competing Provers". In STACS 2005. Pages 605–616. (2005).

[19] Gus Gutoski and John Watrous. "Toward a General Theory of Quantum Games". In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing. Page 565–574. STOC '07. ACM (2007).

[20] Hirotada Kobayashi and Keiji Matsumoto. "Quantum multi-prover interactive proof systems with limited prior entanglement". Journal of Computer and System Sciences **66**, 429–450 (2003).

[21] R. Cleve, P. Hoyer, B. Toner, and J. Watrous. "Consequences and limits of nonlocal strategies". In Proceedings of the 19th IEEE Annual Conference on Computational Complexity, 2004. Pages 236–249. (2004).

[22] Julia Kempe, Hirotada Kobayashi, Keiji Matsumoto, and Thomas Vidick. "Using Entanglement in Quantum Multi-Prover Interactive Proofs". Computational Complexity **18**, 273–307 (2009).

[23] Julia Kempe, Hirotada Kobayashi, Keiji Matsumoto, Ben Toner, and Thomas Vidick. "Entangled Games Are Hard to Approximate". SIAM Journal on Computing **40**, 848–877 (2011).

[24] Andris Ambainis. "A New Protocol and Lower Bounds for Quantum Coin Flipping". In Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing. Page 134–142. STOC '01. ACM (2001).

[25] R. W. Spekkens and Terry Rudolph. "Quantum Protocol for Cheat-Sensitive Weak Coin Flipping". Phys. Rev. Lett. **89**, 227901 (2002).

[26] Carlos Mochon. "Quantum weak coin flipping with arbitrarily small bias" (2007). arXiv:0711.4114.

[27] Carl A. Miller. "The Impossibility of Efficient Quantum Weak Coin Flipping". In Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing. Page 916–929. STOC '20. ACM (2020).

[28] David A. Meyer. "Quantum Strategies". Phys. Rev. Lett. **82**, 1052–1055 (1999).

[29] Rahul Jain and John Watrous. "Parallel Approximation of Non-interactive Zero-sum Quantum Games". In 2009 24th Annual IEEE Conference on Computational Complexity. Pages 243–253. (2009).

[30] Pierre-Luc Dallaire-Demers and Nathan Killoran. "Quantum generative adversarial networks". Phys. Rev. A **98**, 012324 (2018).

[31] Shouvanik Chakrabarti, Yiming Huang, Tongyang Li, Soheil Feizi, and Xiaodi Wu. "Quantum wasserstein GANs". In Proceedings of the 33rd International Conference on Neural Information Processing Systems. Pages 6781 – 6792. NIPS '19 (2019).

[32] John Bostanci and John Watrous. "Quantum game theory and the complexity of approximating quantum Nash equilibria". Quantum **6**, 882 (2022).

[33] Kyriakos Lotidis, Panayotis Mertikopoulos, Nicholas Bambos, and Jose Blanchet. "Payoff-based learning with matrix multiplicative weights in quantum games". In Proceedings of the 37th International Conference on Neural Information Processing Systems. NIPS '23 (2023).

[34] Koji Tsuda, Gunnar Rätsch, and Manfred K. Warmuth. "Matrix Exponentiated Gradient Updates for On-line Learning and Bregman Projection". Journal of Machine Learning Research **6**, 995–1018 (2005).

[35] Arkadi Semen Nemirovski and David Berkovich Yudin. "Problem complexity and method efficiency in optimization". Wiley. New York, NY (1983).

[36] Amir Beck and Marc Teboulle. "Mirror descent and nonlinear projected subgradient methods for convex optimization". Operations Research Letters **31**, 167–175 (2003).

[37] Rahul Jain, Georgios Piliouras, and Ryann Sim. "Matrix Multiplicative Weights Updates in Quantum Zero-Sum Games: Conservation Laws & Recurrence". In Proceedings of the 36th International Conference on Neural Information Processing Systems. NIPS '22 (2022).

[38] Rahul Jain, Zhengfeng Ji, Sarvagya Upadhyay, and John Watrous. "QIP = PSPACE". Journal of the ACM **58**, 30:1–30:27 (2011).

[39] Sanjeev Arora and Satyen Kale. "A Combinatorial, Primal-Dual Approach to Semidefinite Programs". In Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing. Page 227–236. STOC '07. ACM (2007).

[40] Li Chen, Rasmus Kyng, Maximilian Probst Gutenberg, and Sushant Sachdeva. "A Simple Framework for Finding Balanced Sparse Cuts via APSP". In Symposium on Simplicity in Algorithms (SOSA). Pages 42–55. SIAM (2023).

[41] Zeyuan Allen-Zhu, Zhenyu Liao, and Lorenzo Orecchia. "Spectral Sparsification and Regret Minimization Beyond Matrix Multiplicative Updates". In Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing. Page 237–245. STOC '15New York, NY, USA (2015). Association for Computing Machinery.

[42] Panayotis Mertikopoulos, E. Veronica Belmega, and Aris L. Moustakas. "Matrix exponential learning: Distributed optimization in MIMO systems". In 2012 IEEE International Symposium on Information Theory Proceedings. Pages 3028–3032. (2012).

[43] Panayotis Mertikopoulos and Aris L. Moustakas. "Learning in an Uncertain World: MIMO Covariance Matrix Optimization With Imperfect Feedback". IEEE Transactions on Signal Processing **64**, 5–18 (2016).

[44] Olivier Bilenne, Panayotis Mertikopoulos, and Elena Veronica Belmega. "Fast Optimization With Zeroth-Order Feedback in Distributed, Multi-User MIMO Systems". IEEE Transactions on Signal Processing **68**, 6085–6100 (2020).

[45] Sham M Kakade, Shai Shalev-Shwartz, and Ambuj Tewari. "Regularization techniques for learning with matrices". The Journal of Machine Learning Research **13**, 1865–1890 (2012).

[46] Sanjeev Arora, Elad Hazan, and Satyen Kale. "The Multiplicative Weights Update Method: a Meta-Algorithm and Applications". Theory of Computing **8**, 121–164 (2012).

[47] Wayne Lin, Georgios Piliouras, Ryann Sim, and Antonios Varvitsiotis. "No-Regret Learning and Equilibrium Computation in Quantum Games". Quantum **8**, 1569 (2024).

[48] Arkadi Nemirovski. "Prox-Method with Rate of Convergence O(1/t) for Variational Inequalities with Lipschitz Continuous Monotone Operators and Smooth Convex-Concave Saddle Point Problems". SIAM Journal on Optimization **15**, 229–251 (2004).

[49] Alfred Auslender and Marc Teboulle. "Interior projection-like methods for monotone variational inequalities". Mathematical Programming **104**, 39–68 (2005).

[50] Yuyuan Ouyang and Yangyang Xu. "Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems". Mathematical Programming **185**, 1–35 (2021).

[51] Galina M Korpelevich. "The extragradient method for finding saddle points and other problems". Matecon **12**, 747–756 (1976). url: https://cir.nii.ac.jp/crid/1571698600143951616.

[52] B. T. Polyak and A. B. Juditsky. "Acceleration of Stochastic Approximation by Averaging". SIAM Journal on Control and Optimization **30**, 838–855 (1992).

[53] Leonid Denisovich Popov. "A modification of the Arrow-Hurwicz method for search of saddle points". Mathematical notes of the Academy of Sciences of the USSR **28**, 845–848 (1980).

[54] Sasha Rakhlin and Karthik Sridharan. "Optimization, learning, and games with predictable sequences". Advances in Neural Information Processing Systems**26** (2013).

[55] Kyriakos Lotidis, Panayotis Mertikopoulos, and Nicholas Bambos. "Learning in quantum games" (2023). arXiv:2302.02333.

[56] Kyriakos Lotidis, Panayotis Mertikopoulos, and Nicholas Bambos. "The Stability of Matrix Multiplicative Weights Dynamics in Quantum Games". In 2023 62nd IEEE Conference on Decision and Control (CDC). Pages 1225–1232. (2023).

[57] Ilayda Canyakmaz, Wayne Lin, Georgios Piliouras, and Antonios Varvitsiotis. "Multiplicative updates for online convex optimization over symmetric cones" (2023). arXiv:2307.03136.

[58] Alina Ene and Huy Lê Nguyên. "Adaptive and Universal Algorithms for Variational Inequalities with Optimal Convergence". Proceedings of the AAAI Conference on Artificial Intelligence **36**, 6559–6567 (2022).

[59] Gerard Debreu. "A Social Equilibrium Existence Theorem". Proceedings of the National Academy of Sciences **38**, 886–893 (1952).

[60] Gesualdo Scutari, Daniel P. Palomar, Francisco Facchinei, and Jong-shi Pang. "Convex Optimization, Game Theory, and Variational Inequality Theory". IEEE Signal Processing Magazine **27**, 35–49 (2010).

[61] Stephen Boyd and Lieven Vandenberghe. "Convex Optimization". Cambridge University Press. (2004).

[62] Shai Shalev-Shwartz. "Online Learning and Online Convex Optimization". Foundations and Trends in Machine Learning **4**, 107–194 (2012).

[63] John Watrous. "The Theory of Quantum Information". Cambridge University Press. (2018).

[64] Carl D. Meyer and Ian Stewart. "Matrix Analysis and Applied Linear Algebra, Second Edition". Society for Industrial and Applied Mathematics. Philadelphia, PA (2023).

[65] Xingyu Zhou. "On the Fenchel Duality between Strong Convexity and Lipschitz Continuous Gradient" (2018). arXiv:1803.06573.

[66] D. Bertsekas, A. Nedic, and A. Ozdaglar. "Convex Analysis and Optimization". Athena Scientific optimization and computation series. Athena Scientific. (2003).

[67] Constantin Zalinescu. "Convex Analysis in General Vector Spaces". World scientific. (2002).

[68] Panayotis Mertikopoulos, Bruno Lecouat, Houssam Zenati, Chuan-Sheng Foo, Vijay Chandrasekhar, and Georgios Piliouras. "Optimistic mirror descent in saddle-point problems: Going the extra(-gradient) mile". In International Conference on Learning Representations. (2019). url: https://openreview.net/forum?id=Bkg8jjC9KQ.

[69] Sébastien Bubeck. "Convex optimization: Algorithms and complexity". Found. Trends Mach. Learn. **8**, 231–357 (2015).

[70] Yurii Nesterov. "Primal-dual subgradient methods for convex problems". Mathematical Programming **120**, 221–259 (2009).

[71] Andersen Ang. "Projected gradient algorithm". https://angms.science/doc/CVX/CVX_PGD.pdf (2023). Version: July 13, 2023. First draft: August 2, 2017.

[72] Satyen Kale. "Efficient algorithms using the multiplicative weights update method". PhD thesis. Princeton University. (2007).

[73] Anatoli Juditsky, Arkadi Nemirovski, and Claire Tauvel. "Solving Variational Inequalities with Stochastic Mirror-Prox Algorithm". Stochastic Systems **1**, 17–58 (2011).

[74] Yao-Liang Yu. "The strong convexity of von neumann's entropy". Lecture Notes for MSE 213: Optimization Theory (2020). Available online: https://web.stanford.edu/~sidford/courses/20fa_opt_theory/sidford_mse213_2020fa_chap_5_extensions.pdf.

[75] Wayne Lin, Georgios Piliouras, Ryann Sim, and Antonios Varvitsiotis. "Quantum Potential Games, Replicator Dynamics, and the Separability Problem" (2023). arXiv:2302.04789.

[76] Shengyu Zhang. "Quantum strategic game theory". In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference. Pages 39–59. (2012).

[77] Zhaohui Wei and Shengyu Zhang. "Full characterization of quantum correlated equilibria.". Quantum Information and Computation **13**, 846–860 (2013).

[78] Noah Golowich, Sarath Pattathil, and Constantinos Daskalakis. "Tight last-iterate convergence rates for no-regret learning in multi-player games". Advances in Neural Information Processing Systems **33**, 20766–20778 (2020).

[79] Chen-Yu Wei, Chung-Wei Lee, Mengxiao Zhang, and Haipeng Luo. "Linear last-iterate convergence in constrained saddle-point optimization" (2021). arXiv:2006.09517.

[80] D. Petz. "Bregman divergence as relative operator entropy". Acta Mathematica Hungarica **116**, 127–131 (2007).

[81] Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. "Fundamentals of convex analysis". Page 259. Grundlehren Text Editions. Springer Berlin, Heidelberg. (2001). 1 edition.

A Complex Matrix Differentiation and Gradients

In this appendix section, we review complex matrix differentiation and gradients - an invaluable tool for framing our quantum game feedback and analyzing our algorithm. Specifically, we will show why it is natural to define the gradient of a real-valued function $u : \mathbb{C}^d \to \mathbb{R}$ with respect to complex matrix $Z \in \mathbb{C}^d$ as

$$\nabla_{Z^\top} u(Z) \tag{118}$$

and how this relates to a directional derivative. Concretely, our goal is to compute the directional derivative of $u(Z(t))$ at Z in the direction of matrix dZ/dt.

In order to derive this derivative, we will start by introducing a few important properties of Z and u. To begin, note that Z is a Hermitian matrix of the form

$$Z = X + iY, \tag{119}$$

where X is a real symmetric matrix and Y is a real skew-symmetric matrix. Thus, can use the chain rule to derive du/dt:

$$\frac{du}{dt} = \sum_{i,j} \frac{\partial u}{\partial X_{ij}} \frac{dX_{ij}}{dt} + \frac{\partial u}{\partial Y_{ij}} \frac{dY_{ij}}{dt}. \tag{120}$$

From the definition of the Wirtinger derivative, we also have that

$$\frac{\partial u}{\partial Z_{ij}} = \frac{\partial u}{\partial X_{ij}} - i \frac{\partial u}{\partial Y_{ij}} \tag{121}$$

Finally, we note the following property of symmetric and skew-symmetric matrices:

Proposition 6 *For symmetric matrix A and skew-symmetric matrix B,*

$$\mathrm{Tr}(AB) = \mathrm{Tr}(A^\top B) = \sum_{i,j} A_{ij} B_{ij} = 0. \tag{122}$$

Proof. $\mathrm{Tr}(AB) = \mathrm{Tr}(A^\top B) = \mathrm{Tr}(AB^\top) = -\mathrm{Tr}(AB) \implies \mathrm{Tr}(AB) = 0$ □ Since X is symmetric, $\partial u/\partial X$ and dX/dt are symmetric. Similarly, since Y is skew-symmetric, $\partial u/\partial Y$ and dY/dt are symmetric. Therefore, by Proposition 6,

$$\sum_{i,j} \frac{\partial u}{\partial X_{ij}} \frac{dY_{ij}}{dt} = \mathrm{Tr}\left(\frac{\partial u}{\partial X} \frac{dY}{dt} \right) = 0 \tag{123}$$

$$\sum_{i,j} \frac{\partial u}{\partial Y_{ij}} \frac{dX_{ij}}{dt} = \mathrm{Tr}\left(\frac{\partial u}{\partial Y} \frac{dX}{dt} \right) = 0 \tag{124}$$

Now, we can combine the previous properties to obtain du/dt expressed in terms of the complex matrix Z. Starting from Equation (120), then leveraging Equation (123) and Equation (124), we get that

$$\frac{du}{dt} = \sum_{i,j} \frac{\partial u}{\partial X_{ij}} \frac{dX_{ij}}{dt} + \frac{\partial u}{\partial Y_{ij}} \frac{dY_{ij}}{dt} \tag{125}$$

$$= \sum_{i,j} \left(\frac{\partial u}{\partial X_{ij}} \frac{dX_{ij}}{dt} + \frac{\partial u}{\partial Y_{ij}} \frac{dY_{ij}}{dt} \right) + i\,\mathrm{Tr}\left(\frac{\partial u}{\partial X} \frac{dY}{dt} \right) - i\,\mathrm{Tr}\left(\frac{\partial u}{\partial Y} \frac{dX}{dt} \right) \tag{126}$$

$$= \sum_{i,j} \frac{\partial u}{\partial X_{ij}} \frac{dX_{ij}}{dt} + i \frac{\partial u}{\partial X_{ij}} \frac{dY_{ij}}{dt} - i \frac{\partial u}{\partial Y_{ij}} \frac{dX_{ij}}{dt} + \frac{\partial u}{\partial Y_{ij}} \frac{dY_{ij}}{dt} \tag{127}$$

$$= \sum_{i,j} \left(\frac{\partial u}{\partial X_{ij}} - i \frac{\partial u}{\partial Y_{ij}} \right) \left(\frac{dX_{ij}}{dt} + i \frac{dY_{ij}}{dt} \right). \tag{128}$$

By Equation (119) and Equation (121), this achieves the desired directional derivative, which is expressed in terms of the gradient $\nabla_{Z^\top} u$:

$$\frac{du}{dt} = \sum_{i,j} \left(\frac{\partial u}{\partial X_{ij}} - i \frac{\partial u}{\partial Y_{ij}} \right) \left(\frac{dX_{ij}}{dt} + i \frac{dY_{ij}}{dt} \right) = \sum_{i,j} \frac{\partial u}{\partial Z_{ij}} \frac{dZ_{ij}}{dt} \tag{129}$$

$$= \mathrm{Tr}\left(\left(\frac{\partial u}{\partial Z} \right)^\top \frac{dZ}{dt} \right) = \mathrm{Tr}\left(\frac{\partial u}{\partial Z^\top} \frac{dZ}{dt} \right) = \mathrm{Tr}\left(\nabla_{Z^\top} u \cdot \frac{dZ}{dt} \right). \tag{130}$$

B Mirror & Proximal Maps

In this appendix we offer further intuition for definitions and provide intuitive proofs of key lemmas regarding the mirror and proximal maps/steps introduced in Section 2.5.

Definition 6 (Mirror Map) *For a state D in the dual space $\mathcal{D}$, the* mirror map*, or* regularized best response*,*

$$\mathrm{Mir}\Pi^h_{\mathcal{P}}(D) := \arg\max_{P \in \mathcal{P}} \{\mathrm{Tr}[D^\dagger P] - h(P)\}, \tag{43}$$

implements the inverse mirror map, $\partial h^{-1} : \mathcal{D} \to \mathcal{P}$, mapping $D \in \mathcal{D}$ to the corresponding state P in the feasible region $\mathcal{P}$ of the primal space.

Proof. Since $\mathcal{H}^d$ is the set of all complex matrices, this is an unconstrained minimization problem. We can thus solve for the maximum by setting the gradient of the minimization term to zero, as

$$0 = \nabla_{P^\top} \Big(\mathrm{Tr}[D^\dagger P] - h(P) \Big) = D - \partial h(P), \tag{131}$$

which implies the solution

$$\partial h(P) = D \quad \Longrightarrow \quad P = \partial^{-1} h(D). \tag{132}$$

□

Lemma 4 (Mirror Step) *For state X_t in the feasible set $\mathcal{H}^d_+$ of primal space $\mathbb{C}^d$,*

$$\mathrm{MirProx}\Pi^{h,\eta,f}_{\mathcal{X}^d}(X_t) := \arg\min_{X_{t+1} \in \mathcal{H}^d_+} \left\{ \mathrm{Tr}[\nabla f(X_t)^\dagger X_{t+1}] + \frac{1}{\eta} \mathcal{BD}_h(X_{t+1} || X_t) \right\}, \tag{45}$$

implements the mirror step mapping to an updated feasbile set primal state $X_{t+1} \in \mathcal{H}^d_+$.

Proof. The constrained minimization problem has Lagrangian

$$\mathcal{L} = \mathrm{Tr}[\nabla f(X_t)^\dagger X_{t+1}] - \frac{1}{\eta} \Big(h(X_{t+1}) - h(X_t) - \mathrm{Tr}[\nabla h(X_t)^\dagger (X_{t+1} - X_t)] \Big) - \lambda(\mathrm{Tr}[X_{t+1} - 1]), \tag{133}$$

with gradients

$$\begin{cases} \nabla_{X^\top_{t+1}} \mathcal{L} & = \nabla f(X_t) - \frac{1}{\eta} \left(\nabla h(X_{t+1}) - \nabla h(X_t) \right) - \lambda \mathbb{1} \\ \nabla_\lambda \mathcal{L} & = \mathrm{Tr}[X_{t+1}] - 1. \end{cases}. \tag{134}$$

Setting both of these gradients to zero gives us the equations

$$\begin{cases} \partial h(X_{t+1}) & = \partial h(X_t) + \eta \nabla f(X_t) - \eta\lambda \mathbb{1} \\ \mathrm{Tr}[X_{t+1}] & = 1, \end{cases}, \tag{135}$$

which implies the solution

$$X_{t+1} = \partial^{-1} h(\partial h(X_t) + \eta \nabla f(X_t) - \eta\lambda \mathbb{1}), \tag{136}$$

with λ chosen to ensure that X_t lies in the feasible set $\mathcal{X}^d$; i.e., it satisfies the constraint

$$\mathrm{Tr}[X_{t+1}] = \mathrm{Tr}[\partial^{-1} h(\partial h(X_t) + \eta \nabla f(X_t) - \eta\lambda \mathbb{1})] = 1. \tag{137}$$

□

Lemma 5 (Proximal Step) *For state $X_t \in \mathcal{H}_+^d$ in the feasible set of the primal space,*

$$\begin{aligned} \mathrm{Prox}\Pi_{\mathcal{H}_+^d}^{h,\eta}(X_t, \nabla f(Y_{t+1})) :&= \underset{X_{t+1} \in \mathcal{H}_+^d}{\arg\min} \left\{ \mathrm{Tr}[\nabla f(Y_{t+1})^\dagger (X_{t+1} - X_t)] + \frac{1}{\eta} \mathcal{BD}_h(X_{t+1} || X_t) \right\} \\ &= \underset{X_{t+1} \in \mathcal{H}_+^d}{\arg\min} \left\{ \mathrm{Tr}[\nabla f(Y_{t+1})^\dagger X_{t+1}] + \frac{1}{\eta} \mathcal{BD}_h(X_{t+1} || X_t) \right\}, \end{aligned} \tag{47}$$

implements the proximal step mapping to an updated feasbile set primal state $X_{t+1} \in \mathcal{H}_+^d$.

Proof. The constrained minimization problem has Lagrangian

$$\mathcal{L} = \mathrm{Tr}[\nabla f(Y_{t+1})^\dagger (X_{t+1} - X_t)] - \frac{1}{\eta} \mathcal{BD}_h(X_{t+1} || X) - \lambda(\mathrm{Tr}[X_{t+1}] - 1), \tag{138}$$

where

$$\mathcal{BD}_h(X_{t+1} || X) = h(X_{t+1}) - h(X_t) - \mathrm{Tr}[\partial h(X_t)^\dagger (X_{t+1} - X_t)]. \tag{139}$$

The Lagrangian has gradients

$$\begin{cases} \nabla_{X_{t+1}^\top} \mathcal{L} & = \nabla f(Y_{t+1}) - \frac{1}{\eta} \left(\partial h(X_{t+1}) - \partial h(X_t) \right) - \lambda \mathbb{1} \\ \nabla_\lambda \mathcal{L} & = \mathrm{Tr}[X_{t+1}] - 1. \end{cases} \tag{140}$$

Setting both of these gradients to zero gives us the equations

$$\begin{cases} \partial h(X_{t+1}) & = \partial h(X_t) + \eta \nabla f(Y_{t+1}) - \eta\lambda \mathbb{1} \\ \mathrm{Tr}[X_{t+1}] & = 1, \end{cases} \tag{141}$$

which implies the solution

$$X_{t+1} = \partial^{-1} h(\partial h(X_t) + \eta \nabla f(Y_{t+1}) - \eta\lambda \mathbb{1}), \tag{142}$$

with λ chosen to ensure that X_t lies in the feasible set $\mathcal{X}^d$, i.e., it satisfies the constraint

$$\mathrm{Tr}[X_t] = \mathrm{Tr}[\partial^{-1} h(\partial h(X_t) + \eta \nabla f(Y_{t+1}) - \eta\lambda \mathbb{1})] = 1. \tag{143}$$

□

Lemma 6 (Bregman Divergence of the Frobenius Norm) *If h is the squared Frobenius norm,*

$$h(X) = \frac{1}{2}||X||_F^2 = \frac{1}{2}\mathrm{Tr}[X^\dagger X], \tag{51}$$

then the Bregman divergence is the squared Frobenius distance,

$$\mathcal{BD}_h(X||Y) = \tfrac{1}{2}||X - Y||_F^2, \tag{52}$$

the corresponding mirror map (regularized best-response) is the orthogonal projection,

$$\mathrm{Mir}\Pi_{\mathcal{C}}^h(Y) = \arg\min_{C\in\mathcal{C}}||Y - C||_F^2 = Orth\Pi_{\mathcal{C}}(Y), \tag{53}$$

and the corresponding proximal map is

$$\mathrm{Prox}\Pi_{\mathcal{C}}^h(X, Y) = \arg\min_{C\in\mathcal{C}}||X + Y - C||_F^2 = Orth\Pi_{\mathcal{C}}(X + Y). \tag{54}$$

Proof. We begin by noting that

$$\nabla h(Y) = \frac{1}{2}\nabla||Y||_F^2 = Y, \tag{144}$$

which implies that the mirror map ∇h is the identity in this case. Therefore,

$$\nabla^{-1}h(Y) = \nabla h(Y) = Y. \tag{145}$$

Furthermore, since

$$\begin{aligned}
||X - Y||_F^2 &= ||X||_F^2 + ||Y||_F^2 - 2\,\mathrm{Tr}[X^\dagger Y] && (146)\\
&= ||X||_F^2 + (2||Y||_F^2 - ||Y||_F^2) - 2\,\mathrm{Tr}[X^\dagger Y] && (147)\\
&= ||X||_F^2 - ||Y||_F^2 + 2\,\mathrm{Tr}[Y^\dagger Y] - 2\,\mathrm{Tr}[X^\dagger Y] && (148)\\
&= ||X||_F^2 - ||Y||_F^2 - 2\,\mathrm{Tr}[(X - Y)^\dagger Y], && (149)
\end{aligned}$$

the Bregman divergence, as desired, is

$$\begin{aligned}
\mathcal{BD}_h(X||Y) &= h(X) - h(Y) - \mathrm{Tr}[(X - Y)^\dagger \nabla h(Y)] && (150)\\
&= \frac{1}{2}||X||_F^2 - \frac{1}{2}||Y||_F^2 - \mathrm{Tr}[(X - Y)^\dagger \nabla h(Y)] && (151)\\
&= \frac{1}{2}||X||_F^2 - \frac{1}{2}||Y||_F^2 - \mathrm{Tr}[(X - Y)^\dagger Y] && (152)\\
&= \frac{1}{2}\left(||X||_F^2 - ||Y||_F^2 - 2\mathrm{Tr}[(X - Y)^\dagger Y]\right) && (153)\\
&= \frac{1}{2}||X - Y||_F^2. && (154)
\end{aligned}$$

The corresponding regularized best-response is

$$\mathrm{Mir}\Pi_{\mathcal{C}}^h(Y) = \arg\min_{C\in\mathcal{C}}\left\{\mathrm{Tr}[Y^\dagger C] - \frac{1}{2}||C||_F^2\right\},$$

which has Lagrangian

$$\mathcal{L} = \mathrm{Tr}[Y^\dagger C] - \frac{1}{2}||C||_F^2 - \lambda(\mathrm{Tr}[C] - 1), \tag{155}$$

with gradients

$$\nabla\mathcal{L} = \begin{cases} \nabla_{C^\top}\mathcal{L} & = Y - C - \lambda\mathbb{1} \\ \nabla_\lambda\mathcal{L} & = -\mathrm{Tr}[C] + 1. \end{cases} \tag{156}$$

Note, however, that these are also gradients of the Lagrangian

$$\mathcal{L}' = \frac{1}{2}||Y - C||_F^2 - \lambda(\mathrm{Tr}[C] - 1), \tag{157}$$

corresponding to the minimization of the orthogonal projection

$$\mathrm{Orth}\Pi_{\mathcal{C}}(Y) = \arg\min_{C\in\mathcal{C}}||Y - C||_F^2.$$

Therefore, as desired,

$$\mathrm{Mir}\Pi_{\mathcal{C}}^h(Y) = \arg\min_{C\in\mathcal{C}}||Y - C||_F^2 = \mathrm{Orth}\Pi_{\mathcal{C}}(Y).$$

Finally, the corresponding proximal projection is given by

$$\mathrm{Prox}\Pi_{\mathcal{C}}^h(X, Y) = \arg\min_{C\in\mathcal{C}}\left\{\mathrm{Tr}[Y^\dagger(X - C)] + \frac{1}{2}||C - X||_F^2\right\}, \tag{158}$$

which has Lagrangian

$$\mathcal{L} = \mathrm{Tr}[Y^\dagger(X - C)] + \frac{1}{2}||C - X||_F^2 - \lambda(\mathrm{Tr}[C] - 1), \tag{159}$$

with gradients

$$\nabla\mathcal{L} = \begin{cases} \nabla_{C^\top}\mathcal{L} & = X + Y - C - \lambda\mathbb{1} \\ \nabla_\lambda\mathcal{L} & = -\mathrm{Tr}[C] + 1. \end{cases} \tag{160}$$

Note, however, that these are also gradients of the Lagrangian

$$\mathcal{L}' = \frac{1}{2}||X + Y - C||_F^2 - \lambda(\mathrm{Tr}[C] - 1), \tag{161}$$

corresponding to the minimization of the orthogonal projection

$$\mathrm{Orth}\Pi_{\mathcal{C}}(X + Y) = \arg\min_{C\in\mathcal{C}}||X + Y - C||_F^2.$$

Therefore, as desired,

$$\mathrm{Prox}\Pi_{\mathcal{C}}^h(X, Y) = \arg\min_{C\in\mathcal{C}}||X + Y - C||_F^2 = \mathrm{Orth}\Pi_{\mathcal{C}}(X + Y).$$

□

Lemma 7 (Bregman Divergence of the von Neumann Entropy) *If h is the von Neumann entropy,*

$$h(X) = \mathrm{Tr}[X\log X], \tag{55}$$

then the Bregman divergence is the quantum relative entropy

$$\mathcal{BD}_h(X||Y) = \mathrm{Tr}[X(\log X - \log Y)], \tag{56}$$

the corresponding mirror map (regularized best-response) is the logit map,

$$\mathrm{Mir}\Pi_{\mathcal{C}}^h(Y) = \frac{\exp(Y)}{\mathrm{Tr}[\exp(Y)]} = \Lambda(Y), \tag{57}$$

and the corresponding proximal map is

$$\mathrm{Prox}\Pi_{\mathcal{C}}^h(X, Y) = \frac{\exp(\log X + Y)}{\mathrm{Tr}[\exp(\log X + Y)]} = \Lambda(\log X + Y). \tag{58}$$

Proof. We will begin with an informal proof of the Bregman divergence of the von Neumann entropy of density matrices (for the original, rigorous proof refer to [80]). In doing so, we will leverage the following facts about density matrices X and Y:

1. $\nabla_{X^\top} \mathrm{Tr}[X \log X] = \mathbb{1} + \log X$
2. $\mathrm{Tr}[X] = \mathrm{Tr}[Y] = 1$
3. Density matrices are Hermitian, which implies $X^\dagger = X$ and $Y^\dagger = Y$.

Thus, the Bregman divergence is the relative quantum entropy as follows,

$$\begin{aligned}
\mathcal{BD}_h(X||Y) &= h(X) - h(Y) - \mathrm{Tr}[(X-Y)^\dagger \nabla h(Y)] \\
&= \mathrm{Tr}[X \log X] - \mathrm{Tr}[Y \log Y] - \mathrm{Tr}[(X-Y)^\dagger \nabla \mathrm{Tr}[Y \log Y]] \\
&= \mathrm{Tr}[X \log X] - \mathrm{Tr}[Y \log Y] - \mathrm{Tr}[(X-Y)^\dagger (\mathbb{1} + \log Y)] \\
&= \mathrm{Tr}[X \log X] - \mathrm{Tr}[Y \log Y] - \mathrm{Tr}[X] + \mathrm{Tr}[Y] - \mathrm{Tr}[X \log Y] + \mathrm{Tr}[Y \log Y] \\
&= \mathrm{Tr}[X(\log X - \log Y)].
\end{aligned}$$

The regularized best-response,

$$\mathrm{Mir}\Pi^h_{\mathcal{C}}(Y) = \arg\min_{C \in \mathcal{C}} \{\mathrm{Tr}[Y^\dagger C] - \mathrm{Tr}[C \log C]\}$$

has Lagrangian

$$\mathcal{L}(Y) = \mathrm{Tr}[Y^\dagger C] - \mathrm{Tr}[C \log C] - \lambda \cdot (\mathrm{Tr}[C] - 1),$$

with gradients

$$\nabla \mathcal{L} = \begin{cases} \nabla_{C^\top} \mathcal{L}(Y) & = Y - (\mathbb{1} + \log C) - \lambda \mathbb{1} = Y - \log C - \alpha \mathbb{1} \\ \nabla_\lambda \mathcal{L}(Y) & = -\mathrm{Tr}[C] + 1, \end{cases}$$

where $\alpha = \lambda + 1$. Setting the gradients to zero results in the system of equations

$$\begin{cases} \log C & = Y - \alpha \mathbb{1} \\ \mathrm{Tr}[C] & = 1 \end{cases} \implies \begin{cases} C & = \exp(Y - \alpha \mathbb{1}) = \exp(Y) \cdot e^{-\alpha} \mathbb{1} \\ \mathrm{Tr}[C] & = 1. \end{cases}$$

We can solve for α by plugging the first equation into the second and using the properties of matrix exponentials as follows.

$$\mathrm{Tr}[\exp(Y) \cdot e^{-\alpha} \mathbb{1}] = e^{-\alpha} \mathrm{Tr}[\exp(Y)] = 1 \quad \implies \quad e^{-\alpha} = \frac{1}{\mathrm{Tr}[\exp(Y)]}.$$

Substituting this back into our equation for C gives

$$C = \frac{\exp(Y)}{\mathrm{Tr}[\exp(Y)]}.$$

Therefore, as desired, the regularized best response is the logit map

$$\mathrm{Mir}\Pi^h_{\mathcal{C}}(Y) = \frac{\exp(Y)}{\mathrm{Tr}[\exp(Y)]} = \Lambda(Y).$$

Finally, the corresponding proximal projection is given by

$$\mathrm{Prox}\Pi^h_{\mathcal{C}}(X, Y) = \arg\min_{C \in \mathcal{C}} \left\{ \mathrm{Tr}[Y^\dagger (X - C)] + \mathrm{Tr}[C(\log C - \log X)] \right\}, \tag{162}$$

which has Lagrangian

$$\mathcal{L}(Y) = \mathrm{Tr}[Y^\dagger(X - C)] + \mathrm{Tr}[C(\log C - \log X)] - \lambda \cdot (\mathrm{Tr}[C] - 1),$$

with gradients

$$\nabla\mathcal{L} = \begin{cases} \nabla_{C^\top}\mathcal{L}(Y) & = -Y + \mathbb{1} + \log C - \log X - \lambda\mathbb{1} = \log C - Y - \log X + \alpha\mathbb{1} \\ \nabla_\lambda\mathcal{L}(Y) & = -\mathrm{Tr}[C] + 1, \end{cases}$$

where $\alpha = \lambda + 1$. Setting the gradients to zero results in the system of equations

$$\begin{cases} \log C & = Y + \log X - \alpha\mathbb{1} \\ \mathrm{Tr}[C] & = 1 \end{cases} \implies \begin{cases} C & = \exp(Y + \log X) \cdot e^{-\alpha}\mathbb{1} \\ \mathrm{Tr}[C] & = 1 \end{cases}$$

We can solve for α by plugging the first equation into the second and using the properties of matrix exponentials as follows.

$$\mathrm{Tr}[\exp(Y + \log X) \cdot e^{-\alpha}\mathbb{1}] = e^{-\alpha}\mathrm{Tr}[\exp(Y + \log X)] = 1 \quad \implies \quad e^{-\alpha} = \frac{1}{\mathrm{Tr}[\exp(Y + \log X)]}.$$

Substituting this back into our equation for C gives

$$C = \frac{\exp(Y + \log X)}{\mathrm{Tr}[\exp(Y + \log X)]}.$$

Therefore, as desired, the regularized best response is the logit map

$$\mathrm{Prox}\Pi^h_{\mathcal{C}}(X, Y) = \frac{\exp(Y + \log X)}{\mathrm{Tr}[\exp(Y + \log X)]} = \Lambda(Y + \log X).$$

□

C Properties of Finite-Valued Zero-Sum Quantum Games

In this appendix we will prove several important properties of finite-valued zero-sum quantum games.

C.1 Properties of Spectraplexes

Lemma 10 *The spectraplex $\mathcal{X} = \{X \in \mathcal{H}^k_+ : \mathrm{Tr}(X) = 1\}$ is a convex set.*

Proof. By the definition of convex set, $\mathcal{X}$ is a convex set iff, for any $X_1, X_2 \in \mathcal{X}$ and $t \in (0, 1]$,

$$\tilde{X} = (1 - t) \cdot X_1 + t \cdot X_2 \tag{163}$$

such that $\tilde{X} \in \mathcal{H}^k_+$ and $\mathrm{Tr}(\tilde{X}) = 1$.

We begin by proving that $\tilde{X}$ is positive semi-definite, $\tilde{X} \in \mathcal{H}^k_+$, which is true iff $\forall\, |v\rangle \in \mathbb{C}^k$,

$$\langle v|\, \tilde{X}\, |v\rangle \geq 0. \tag{164}$$

Given that $X_1, X_2 \in \mathcal{H}_+^k$,

$$\forall\, |v\rangle \in \mathbb{C}^k : \quad \langle v|\, X_1\, |v\rangle \geq 0 \quad \text{and} \quad \langle v|\, X_2\, |v\rangle \geq 0. \tag{165}$$

Therefore, since $t \geq 0$ and $(1-t) \geq 0$,

$$\langle v|\, \tilde{X}\, |v\rangle = \langle v|\, (1-t) \cdot X_1 + t \cdot X_2\, |v\rangle = (1-t)\, \langle v|\, X_1\, |v\rangle + t\, \langle v|\, X_2\, |v\rangle \geq 0. \tag{166}$$

We now prove that $\mathrm{Tr}(\tilde{X}) = 1$. Since $\mathrm{Tr}[X_1] = \mathrm{Tr}[X_2] = 1$,

$$\mathrm{Tr}(\tilde{X}) = \mathrm{Tr}\Big[(1-t) \cdot X_1 + t \cdot X_2\Big] = (1-t) \cdot \mathrm{Tr}[X_1] + t \cdot \mathrm{Tr}[X_2] = 1 - t + t = 1 \tag{167}$$

□

Lemma 11 *The spectraplex $\mathcal{X} = \{X \in \mathcal{H}_+^k : \mathrm{Tr}(X) = 1\}$ is a compact set.*

Proof. See page 162 of [81]. □

C.2 Properties of $\mathcal{F}$

Lemma 8 (Mononicity of $\mathcal{F}$) *$\mathcal{F}$ is monotone or, equivalently,*

$$\mathrm{Tr}\left[(\mathcal{F}(X) - \mathcal{F}(Y))\,(X - Y)\right] \geq 0, \;\; \forall X, Y \in \mathcal{C}. \tag{73}$$

Proof. For this proof, we will leverage the Pauli decompositions of the involved operators. Note that every matrix M can be decomposed in terms of the Pauli matrices $\mathcal{P} = \{I, X, Y, Z\}$, as

$$M = \sum_{P \in \mathcal{P}^{\otimes n}} \widehat{M}(P) P,$$

where the tensored Pauli matrices P constitute an orthonormal basis, with respect to trace inner product, and possess corresponding Pauli coefficients $\widehat{M}(P)$. Note that for all Paulis, we have $P = P^\dagger$. The Pauli coefficients of an n-qubit system can also be explicitly defined as

$$\widehat{M}(P) = \frac{1}{2^n} \mathrm{Tr}(P^\dagger M).$$

The payoff observable U can be decomposed as

$$U = \sum_{R \in \mathcal{P}^{\otimes (nm)}} \widehat{U}(R) R = \sum_{P \in \mathcal{P}^{\otimes n}} \sum_{Q \in \mathcal{P}^{\otimes m}} \widehat{U}(P, Q) P \otimes Q,$$

where $\widehat{U}(P,Q) = \widehat{U}(P \otimes Q)$. Plugging this into the payoff gradients defined in Equation (16) and Equation (17), we get that

$$\begin{aligned}
\mathcal{F}_\alpha(\alpha,\beta) &= \nabla_{\alpha^\top} \mathrm{Tr}\left(U^\dagger(\alpha\otimes\beta)\right) \\
&= \nabla_{\alpha^\top} \mathrm{Tr}\left(\left(\sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q) P\otimes Q\right)^\dagger (\alpha\otimes\beta)\right) \\
&= \nabla_{\alpha^\top} \mathrm{Tr}\left(\sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* P^\dagger\alpha\otimes Q^\dagger\beta\right) \\
&= \sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot \nabla_{\alpha^\top} \mathrm{Tr}\left(P^\dagger\alpha\otimes Q^\dagger\beta\right) \\
&= \sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot \nabla_{\alpha^\top} \mathrm{Tr}\left(P^\dagger\alpha\right)\mathrm{Tr}\left(Q^\dagger\beta\right) \\
&= \sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot 2^m \widehat{\beta}(Q)\, P^\dagger
\end{aligned}$$

and, similarly,

$$\begin{aligned}
\mathcal{F}_\beta(\alpha,\beta) &= -\nabla_{\beta^\top} \mathrm{Tr}\left(U^\dagger(\alpha\otimes\beta)\right) \\
&= -\sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot \nabla_{\beta^\top} \mathrm{Tr}\left(P^\dagger\alpha\right)\mathrm{Tr}\left(Q^\dagger\beta\right) \\
&= -\sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot 2^n \widehat{\alpha}(P)\, Q^\dagger.
\end{aligned}$$

Thus, plugging these results into Equation (19), the operator $\mathcal{F}$ can be decomposed as

$$\mathcal{F}(\alpha,\beta) = \begin{pmatrix} 2^m & \sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot \widehat{\beta}(Q)\, P^\dagger \\ -2^n & \sum_{P\in\mathcal{P}^{\otimes n}}\sum_{Q\in\mathcal{P}^{\otimes m}} \widehat{U}(P,Q)^* \cdot \widehat{\alpha}(P)\, Q^\dagger \end{pmatrix}. \tag{168}$$

Now, let us consider the two distinct games states, $X = (x_1, x_2)$ and $Y = (y_1, y_2)$, where $x_1, y_1 \in \mathcal{A}$ and $x_2, y_2 \in \mathcal{B}$, with difference

$$X - Y = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} x_1 - y_1 \\ x_2 - y_2 \end{pmatrix}.$$

Following from Equation (168), the difference of the payoff gradients is

$$F(X) - F(Y) = F(x_1,x_2) - F(y_1,y_2) = \begin{pmatrix} 2^m \sum_{P,Q} \widehat{U}(P,Q)^* \cdot (\widehat{x_2}(Q) - \widehat{y_2}(Q))\ P^\dagger \\ 2^n \sum_{P,Q} \widehat{U}(P,Q)^* \cdot (\widehat{y_1}(P) - \widehat{x_1}(P))\ Q^\dagger \end{pmatrix}$$

Plugging these into the left-hand side of Equation (73), we get that

$$
\begin{aligned}
&\mathrm{Tr}\left[(\mathcal{F}(X)-\mathcal{F}(Y))(X-Y)\right] = \\
&\quad = \mathrm{Tr}\left[\begin{pmatrix} 2^m \sum_{P,Q} \widehat{U}(P,Q)^* \cdot (\widehat{x_2}(Q) - \widehat{y_2}(Q))\ P^\dagger \\ 2^n \sum_{P,Q} \widehat{U}(P,Q)^* \cdot (\widehat{y_1}(P) - \widehat{x_1}(P))\ Q^\dagger \end{pmatrix} \begin{pmatrix} x_1 - y_1 \\ x_2 - y_2 \end{pmatrix}\right] \\
&\quad = \mathrm{Tr}\left[\sum_{P,Q} \widehat{U}(P,Q)^* \left(2^m\,(\widehat{x_2}(Q) - \widehat{y_2}(Q))\ P^\dagger(x_1-y_1) + 2^n\,(\widehat{y_1}(P) - \widehat{x_1}(P))\ Q^\dagger(x_2-y_2)\right)\right] \\
&\quad = \sum_{P,Q} \widehat{U}(P,Q)^* \left(2^m\,(\widehat{x_2}(Q) - \widehat{y_2}(Q))\ \mathrm{Tr}\left[P^\dagger(x_1-y_1)\right] + 2^n\,(\widehat{y_1}(P) - \widehat{x_1}(P))\ \mathrm{Tr}\left[Q^\dagger(x_2-y_2)\right]\right) \\
&\quad = \sum_{P,Q} \widehat{U}(P,Q)^* \Big(2^m\,(\widehat{x_2}(Q) - \widehat{y_2}(Q))\,2^n\,(\widehat{x_1}(P) - \widehat{y_1}(P)) + 2^n\,(\widehat{y_1}(P) - \widehat{x_1}(P))\,2^m\,(\widehat{x_2}(Q) - \widehat{y_2}(Q))\Big) \\
&\quad = 2^{n+m} \sum_{P,Q} \widehat{U}(P,Q)^* \Big((\widehat{x_2}(Q) - \widehat{y_2}(Q))\,(\widehat{x_1}(P) - \widehat{y_1}(P)) - (\widehat{x_1}(P) - \widehat{y_1}(P))\,(\widehat{x_2}(Q) - \widehat{y_2}(Q))\Big) \\
&\quad = 0.
\end{aligned}
$$

□

Lemma 9 (Lipschitz Continuity of $\mathcal{F}$) *For finite-valued quantum zero-sum games, $\mathcal{F}$ is a Lipschitz-continuous operator, meaning there exists a Lipschitz constant $\gamma \in \mathbb{R}$ such that*

$$\|\mathcal{F}(Z) - \mathcal{F}(Z')\|_* \leq \gamma \|Z - Z'\|, \ \ \forall\, Z, Z' \in \mathcal{C}. \tag{108}$$

For the case of $(\|\cdot\|_F, \|\cdot\|_F)$ and $(\|\cdot\|_\infty, \|\cdot\|_1)$, $\gamma_{F,F} = \mathcal{O}(2^d)$ while $\gamma_{\infty,1} = \mathcal{O}(1)$.

Proof. In Lemma 12, we proved that $\mathcal{F}$ is linear, which implies that for any $Z, Z' \in \mathcal{C}$,

$$\mathcal{F}(Z) - \mathcal{F}(Z') = \mathcal{F}(Z - Z').$$

In order to prove that $\mathcal{F}$ is Lipschitz continuous, we simply need to prove that there exists a $\gamma \in \mathbb{R}$ such that, for all $Z \neq Z'$,[11]

$$\frac{||\mathcal{F}(Z - Z')||_F}{||Z - Z'||_F} \leq \gamma.$$

Letting $Y = Z - Z'$, this is equivalent to proving

$$\sup_{Y \neq 0} \frac{||\mathcal{F}(Y)||_F}{||Y||_F} \leq \gamma. \tag{169}$$

Vectorizing Y, as a vector of size $|\mathcal{C}|$, the Frobenius norm can be expressed as

$$||Y||_F = \sqrt{\sum_i |Y_i|^2}.$$

Furthermore, since $\mathcal{F}$ is a linear operator, it maps Y to some new joint state $Y' = \mathcal{F}(Y) \in \mathcal{C}$, with entries

$$Y'_j = \sum_i c_{ij} Y_i.$$

[11] In the case that $Z = Z'$, trivially $Z - Z' = 0$ and $\mathcal{F}(Z - Z') = 0$, meaning any value for γ suffices.

for some linear coefficients $c_{ij} \in \mathbb{C}$. Note that since the setting is restricted to finite-valued quantum zero-sum games, with a finite-valued utility function $\mathcal{U}$, the coefficients c_{ij} are guaranteed to be finite-valued. Therefore, the right-hand side of Equation (169) can be re-expressed as

$$\sup_{Y\neq 0} \frac{||\mathcal{F}(Y)||_F}{||Y||_F} = \sup_{Y\neq 0} \frac{||Y'||_F}{||Y||_F} = \sup_{Y\neq 0} \sqrt{\frac{\sum_j |\sum_i c_{ij} Y_i|^2}{\sum_i |Y_i|^2}}.$$

Letting $c^* = \max(\{c_{i,j}\})$,

$$\sup_{Y\neq 0} \sqrt{\frac{\sum_j |\sum_i c_{ij} Y_i|^2}{\sum_i |Y_i|^2}} \leq \sup_{Y\neq 0} \sqrt{\frac{\sum_j |\sum_i c^* Y_i|^2}{\sum_i |Y_i|^2}} = \sup_{Y\neq 0} \sqrt{\frac{c^* \sum_j |\sum_i Y_i|^2}{\sum_i |Y_i|^2}}$$

and leveraging the triangle inequality,

$$\sup_{Y\neq 0} \sqrt{\frac{c^* \sum_j |\sum_i Y_i|^2}{\sum_i |Y_i|^2}} \leq \sup_{Y\neq 0} \sqrt{\frac{c^* \sum_j \sum_i |Y_i|^2}{\sum_i |Y_i|^2}} = \sup_{Y\neq 0} \sqrt{\frac{c^* |\mathcal{C}| \cdot \sum_i |Y_i|^2}{\sum_i |Y_i|^2}} = \sqrt{c^* |\mathcal{C}|}.$$

Thus, for any $\gamma \geq \sqrt{c^*|\mathcal{C}|}$, Equation (169) is satisfied. Recall that $c^* = \Theta(1)$ for games in $[-1, 1]$. Therefore, the linear operator $\mathcal{F}$ is Lipschitz continuous where $\gamma_{F,F} = \mathcal{O}(\sqrt{4^d})$. Applying similar argumentation for the case of real vector and norm pairs (ℓ_∞, ℓ_1), we get $\gamma_{\infty,1} = \mathcal{O}(1)$. Indeed, it is easy to see

$$\sup_{Y\neq 0} \frac{||\mathcal{F}(Y)||_\infty}{||Y||_1} = \sup_{Y\neq 0} \frac{||Y'||_\infty}{||Y||_1} = \sup_{Y\neq 0} \sqrt{\frac{\max_j |\sum_i c_{ij} Y_i|}{\sum_i |Y_i|}} \leq c^* = \mathcal{O}(1).$$

□

Lemma 1 *If $\mathcal{F}$ is monotone, then a strong solution is a weak solution.*

Proof. Denote a strong solution, satisfying Equation (20), as Ψ^S. Since $\mathcal{F}$ is monotone, from the definition of monotonicty, $\forall\, \Psi \in \mathcal{C}$,

$$\mathrm{Tr}\left[(\mathcal{F}(\Psi) - \mathcal{F}(\Psi^S))\,(\Psi - \Psi^S)\right] \geq 0, \tag{170}$$

which implies that,

$$\mathrm{Tr}\left[\mathcal{F}(\Psi)\,(\Psi - \Psi^S)\right] - \mathrm{Tr}\left[\mathcal{F}(\Psi^S)\,(\Psi - \Psi^S)\right] \geq 0. \tag{171}$$

Since Ψ^S is a strong solution, by Equation (20),

$$\mathrm{Tr}\left[\mathcal{F}(\Psi)\,(\Psi - \Psi^S)\right] \geq \mathrm{Tr}\left[\mathcal{F}(\Psi^S)\,(\Psi - \Psi^S)\right] \geq 0. \tag{172}$$

Therefore, $\forall\, \Psi \in \mathcal{C}$,

$$\mathrm{Tr}\left[\mathcal{F}(\Psi)\,(\Psi^S - \Psi)\right] \leq 0, \tag{173}$$

meaning Ψ^S satisfies Equation (21) and is, thus, also a weak solution. □

Lemma 2 *If $\mathcal{F}$ is Lipschitz continuous, then a weak solution is a strong solution.*

Proof. Denote a weak solution, satisfying Equation (21), as Ψ^W. In Lemma 10 we proved that spectraplexes are convex, which implies that $\mathcal{A}$, $\mathcal{B}$, and, thus, $\mathcal{C}$ are convex sets. Therefore, by the definition of convex set, for any $\Psi \in \mathcal{C}$ and $t \in (0, 1]$,

$$\Psi^W + t \cdot (\Psi - \Psi^W) \in \mathcal{C}. \tag{174}$$

Plugging this state into Equation (21) gives

$$\mathrm{Tr}\left[\mathcal{F}(\Psi^W + t \cdot (\Psi - \Psi^W))\left(\Psi^W + t \cdot (\Psi - \Psi^W) - \Psi^W\right)\right] \geq 0 \tag{175}$$

$$t \cdot \mathrm{Tr}\left[\mathcal{F}(\Psi^W + t \cdot (\Psi - \Psi^W))\left(\Psi - \Psi^W\right)\right] \geq 0 \tag{176}$$

$$\mathrm{Tr}\left[\mathcal{F}(\Psi^W + t \cdot (\Psi - \Psi^W))\left(\Psi - \Psi^W\right)\right] \geq 0. \tag{177}$$

Since $\mathcal{F}$ is Lipschitz continuous, we can take the following limit,

$$\lim_{t \to 0} \mathrm{Tr}\left[\mathcal{F}(\Psi^W + t \cdot (\Psi - \Psi^W))\left(\Psi - \Psi^W\right)\right] \geq 0. \tag{178}$$

The resulting expression,

$$\mathrm{Tr}\left[\mathcal{F}(\Psi^W)\left(\Psi - \Psi^W\right)\right] \geq 0, \ \ \forall \Psi \in \mathcal{C}, \tag{179}$$

implies that Ψ^W satisfies Equation (20) and is also a strong solution. □

Lemma 12 (Linearity of $\mathcal{F}$) *$\mathcal{F} : \mathcal{A} \times \mathcal{B} \mapsto \mathcal{C}$ is a linear operator. Therefore, $\mathcal{F}$ must satisfy the following properties:*

1. *For any $\lambda \in \mathbb{C}$, $\mathcal{F}(\lambda\alpha, \lambda\beta) = \lambda\mathcal{F}(\alpha, \beta)$.*
2. *For $\alpha, \alpha' \in \mathcal{A}$ and $\beta, \beta' \in \mathcal{B}$, $\mathcal{F}(\alpha + \alpha', \beta + \beta') = \mathcal{F}(\alpha, \beta) + \mathcal{F}(\alpha', \beta')$.*

Proof. From Equation (16) and Equation (17), it follows that

$$\begin{aligned}
\mathcal{F}_\alpha(\alpha, \beta) &= \mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes \beta)\right] = \mathcal{F}_\alpha(\beta) \in \mathcal{A} \\
\mathcal{F}_\beta(\alpha, \beta) &= -\mathrm{Tr}_{\mathcal{A}}\left[U^\dagger(\alpha \otimes \mathbb{1}_m)\right] = \mathcal{F}_\beta(\alpha) \in \mathcal{B} \\
\mathcal{F}(\alpha, \beta) &= (\mathcal{F}_\alpha(\alpha, \beta), \mathcal{F}_\beta(\alpha, \beta)) \in \mathcal{C}
\end{aligned}$$

such that $\mathcal{F} : \mathcal{A} \times \mathcal{B} \mapsto \mathcal{C}$.

We will now prove that $\mathcal{F}$ satisfies the *multiplicative property*. If $\lambda \in \mathbb{C}$, then

$$\begin{aligned}
\mathcal{F}_{\lambda\alpha}(\lambda\alpha, \lambda\beta) &= \mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes \lambda\beta)\right] = \lambda\mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes \beta)\right] = \lambda\mathcal{F}_\alpha(\alpha, \beta) \\
\mathcal{F}_{\lambda\beta}(\lambda\alpha, \lambda\beta) &= -\mathrm{Tr}_{\mathcal{A}}\left[U^\dagger(\lambda\alpha \otimes \mathbb{1}_m)\right] = -\lambda\mathrm{Tr}_{\mathcal{A}}\left[U^\dagger(\alpha \otimes \mathbb{1}_m)\right] = \lambda\mathcal{F}_\beta(\alpha, \beta),
\end{aligned}$$

which implies, as desired,

$$\mathcal{F}(\lambda\alpha, \lambda\beta) = \left(\mathcal{F}_{\lambda\alpha}(\lambda\alpha, \lambda\beta), \mathcal{F}_{\lambda\beta}(\lambda\alpha, \lambda\beta)\right) = \left(\lambda\mathcal{F}_\alpha(\alpha, \beta), \lambda\mathcal{F}_\beta(\alpha, \beta)\right) = \lambda\mathcal{F}(\alpha, \beta).$$

We will now prove that $\mathcal{F}$ satisfies the *additive property*. If $\alpha, \alpha' \in \mathcal{A}$ and $\beta, \beta' \in \mathcal{B}$, then

$$\begin{aligned}
\mathcal{F}_{\alpha+\alpha'}(\alpha+\alpha', \beta+\beta') &= \mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes (\beta+\beta'))\right] \\
&= \mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes \beta) + U^\dagger(\mathbb{1}_n \otimes \beta')\right] \\
&= \mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes \beta)\right] + \mathrm{Tr}_{\mathcal{B}}\left[U^\dagger(\mathbb{1}_n \otimes \beta')\right] \\
&= \mathcal{F}_\alpha(\alpha, \beta) + \mathcal{F}_{\alpha'}(\alpha', \beta') \\
\mathcal{F}_{\beta+\beta'}(\alpha+\alpha', \beta+\beta') &= -\mathrm{Tr}_{\mathcal{A}}\left[U^\dagger((\alpha+\alpha') \otimes \mathbb{1}_m)\right] \\
&= -\mathrm{Tr}_{\mathcal{A}}\left[U^\dagger(\alpha \otimes \mathbb{1}_m) + U^\dagger(\alpha' \otimes \mathbb{1}_m)\right] \\
&= -\mathrm{Tr}_{\mathcal{A}}\left[U^\dagger(\alpha \otimes \mathbb{1}_m)\right] - \mathrm{Tr}_{\mathcal{A}}\left[U^\dagger(\alpha' \otimes \mathbb{1}_m)\right] \\
&= \mathcal{F}_\beta(\alpha, \beta) + \mathcal{F}_{\beta'}(\alpha', \beta'),
\end{aligned}$$

which implies, as desired,

$$\begin{aligned}
\mathcal{F}(\alpha+\alpha', \beta+\beta') &= \Big(\mathcal{F}_\alpha(\alpha, \beta) + \mathcal{F}_{\alpha'}(\alpha', \beta'), \mathcal{F}_\beta(\alpha, \beta) + \mathcal{F}_{\beta'}(\alpha', \beta')\Big) \\
&= \Big(\mathcal{F}_\alpha(\alpha, \beta), \mathcal{F}_\beta(\alpha, \beta)\Big) + \Big(\mathcal{F}_{\alpha'}(\alpha', \beta'), \mathcal{F}_{\beta'}(\alpha', \beta')\Big) \\
&= \mathcal{F}(\alpha, \beta) + \mathcal{F}(\alpha', \beta').
\end{aligned}$$

□

D Convergence Analysis Proofs

Proposition 7 *If $g(X)$ is a μ-strongly convex function and $h(X)$ is an α-strongly convex function, then $f(X) = g(X) + h(X)$ is a $\mu + \alpha$-strongly convex function.*

Proof. By the defintion of strong convexity (Definition 2), $\forall X, Y \in \mathcal{C}$

$$g(X) \geq g(Y) + \langle \nabla g(Y), X - Y \rangle + \frac{\mu}{2}\|X - Y\|_F^2, \tag{180}$$

$$h(X) \geq h(Y) + \langle \nabla h(Y), X - Y \rangle + \frac{\alpha}{2}\|X - Y\|_F^2. \tag{181}$$

Summing together these two expressions gives

$$g(X) + h(X) \geq g(Y) + h(Y) + \langle \nabla g(Y) + \nabla h(Y), X - Y \rangle + \frac{\mu+\alpha}{2}\|X - Y\|_F^2 \tag{182}$$

$$\implies f(X) \geq f(Y) + \langle \nabla f(Y), X - Y \rangle + \frac{\mu+\alpha}{2}\|X - Y\|_F^2, \tag{183}$$

which implies f is $(\mu + \alpha)$-strongly convex. □

Proposition 3 *For a μ-strongly convex regularization function h with respect to the norm $\|\cdot\|$, ϕ_t is $\frac{\mu}{\eta}$-strongly convex with respect to the norm $\|\cdot\|$.*

Proof. To prove that $\phi_t(Z)$ is $\frac{\mu}{\eta}$-strongly convex, we will leverage Proposition 7. Namely, by proving that $\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(Z-\Phi_{t-1}\right)\right]$ is convex (i.e., 0-strongly convex) and that $\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1})$ is $\frac{\mu}{\eta}$-strongly convex, then

$$\phi_t(Z)=\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(Z-\Phi_{t-1}\right)\right]+\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1}), \tag{184}$$

must be $0+\frac{\mu}{\eta}=\frac{\mu}{\eta}$-strongly convex.

We begin by proving that $\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1})$ is $\frac{\mu}{\eta}$-strongly convex. From the definition of strong convexity (Definition 2), if h is μ-strongly convex, then the Bregman divergence $\mathcal{BD}_h$ must also be μ-strongly convex. This implies that

$$\frac{1}{\eta}\mathcal{BD}_h(X\|Y)\geq\frac{\mu}{2\eta}\|X-Y\|_F^2,\quad\forall X,Y\in\mathcal{C}, \tag{185}$$

meaning the function $\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1})$ must be $\frac{\mu}{\eta}$-strongly convex.

We now prove that $f(Z)=\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(Z-\Phi_{t-1}\right)\right]$ is convex. For all $X,Y\in\mathcal{C}$,

$$f(X)-f(Y)=\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(X-\Phi_{t-1}\right)\right]-\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(Y-\Phi_{t-1}\right)\right]=\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(X-Y\right)\right]. \tag{186}$$

Since $\forall Z\in\mathcal{C}$, $\nabla f(Z)=\mathcal{F}(\Psi_t)$, this implies that

$$f(X)-f(Y)=\mathrm{Tr}\left[\mathcal{F}(\Psi_t)\left(X-Y\right)\right]=\mathrm{Tr}\left[\nabla f(Y)\left(X-Y\right)\right], \tag{187}$$

meaning, by Definition 1, f is convex. □

Proposition 4 *For a μ-strongly convex regularization function h with respect to the norm $\|\cdot\|$, ψ_t is $\frac{\mu}{\eta}$-strongly convex with respect to the norm $\|\cdot\|$.*

Proof. To prove that $\psi_t(Z)$ is $\frac{\mu}{\eta}$-strongly convex, we will leverage Proposition 7. Namely, by proving that $\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(Z-\Phi_{t-1}\right)\right]$ is convex (i.e. 0-strongly convex) and that $\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1})$ is $\frac{\mu}{\eta}$-strongly convex, then

$$\psi_t(Z)=\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(Z-\Phi_{t-1}\right)\right]+\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1}), \tag{188}$$

must be $0+\frac{\mu}{\eta}=\frac{\mu}{\eta}$-strongly convex.

We begin by proving that $\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1})$ is $\frac{\mu}{\eta}$-strongly convex. From the definition of strong convexity (Definition 2), if h is μ-strongly convex, then the Bregman divergence $\mathcal{BD}_h$ must also be μ-strongly convex. This implies that

$$\frac{1}{\eta}\mathcal{BD}_h(X\|Y)\geq\frac{\mu}{2\eta}\|X-Y\|_F^2,\quad\forall X,Y\in\mathcal{C}, \tag{189}$$

meaning the function $\frac{1}{\eta}\mathcal{BD}_h(Z\|\Phi_{t-1})$ must be $\frac{\mu}{\eta}$-strongly convex.

We now prove that $f(Z)=\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(Z-\Phi_{t-1}\right)\right]$ is convex. For all $X,Y\in\mathcal{C}$,

$$f(X)-f(Y)=\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(X-\Phi_{t-1}\right)\right]-\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(Y-\Phi_{t-1}\right)\right]=\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(X-Y\right)\right]. \tag{190}$$

Since $\forall Z\in\mathcal{C}$, $\nabla f(Z)=\mathcal{F}(\Psi_t)$, this implies that

$$f(X)-f(Y)=\mathrm{Tr}\left[\mathcal{F}(\Psi_{t-1})\left(X-Y\right)\right]=\mathrm{Tr}\left[\nabla f(Y)\left(X-Y\right)\right], \tag{191}$$

meaning, by Definition 1, f is convex. □