2023 9th International Conference on Control, Decision and Information Technologies
CoDIT 2023 | Rome, Italy / July 03-06, 2023 Technically co-sponsored by IEEE & IFAC

On Parameter Selection for First-Order Methods: A Matrix Analysis Approach

Eder Baron-Prada[1]*, Salman Alsubaihi[2], Khaled Alshehri[3], Fahad Albalawi[4]

2023 9th International Conference on Control, Decision and Information Technologies (CoDIT) | 979-8-3503-1140-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/CODIT58514.2023.10284153

***Abstract*— First-order convex optimization algorithms are popular due to their computational attractiveness and applicability to a wide range of domains such as machine learning and control. Despite the substantial progress being made over the last few decades, some open questions related to their convergence remain unaddressed. In addition, majority of the first-order methods assume strong convexity to analyze both the stability of the method and derive an explicit convergence rate. In this manuscript, we relax the strong convexity condition, and then, lay out two main contributions. First, we provide a methodology where one can analyze the speed of convergence of the algorithm using the contractive theory and linear algebra. Second, we find explicit values of the tuning parameters of the Double Momentum Algorithm (which unifies many of the popular algorithms), ensuring stability for gradient L- Lipchitz functions. In this work, while an explicit convergence rate is not provided, the foundational results serve as a stepping stone in that direction, as we provide an explicit non-asymptotic rate. Furthermore, our numerical experiments demonstrate superior performance of the proposed method. Beyond optimization, we also apply our method to two-sided markets in non-cooperative game theory.**

I. INTRODUCTION

First-order methods have evolved over the last six decades due to their mild computational burden, which make them appropriate tools for large-scale engineering applications [1], [2]. Significant contributions to such algorithms have been made from both a theoretical point of view and an application point of view (e.g., in machine learning [3] and control theory [4]). Improvements of first-order methods are still ongoing because of advancements to the so-called accelerated momentum algorithms with optimal convergence rates such as Nesterov method [5], and Heavy-ball method [6].

The gradient descent algorithm was the earliest developed first-order method proposed in the late 1950s for unconstrained convex optimization problems [7]. Such approach has found its way to produce iterates that converge to the optimal at a rate $O(1/k)$ where k is the number of iterations [8]. Furthermore, a linear convergence rate was also obtained under a gradient descent method when the objective function is strongly convex [9]. Years later, Nesterov proposed a first-order method famously known as Nesterov method that can produce iterates with a convergence rate of $O(1/k^2)$ [10]. Subsequent work in [11] attained a linear convergence rate when the objective function is strongly convex. Accelerated first-order can be seen as an algorithm that integrates the current iterates with gradient information evaluated at the current, and the past iteration [11]. In [12], the Heavy-ball achieved a better performance than both the gradient and Nesterov method with local linear convergence rate when the objective is twice continuously differentiable, strongly convex, and has Lipschitz continuous gradient.

Subsequently, different forms of the Heavy-ball method have been recently proposed to address both constrained [13] and distributed [14] optimization problems where their convergence rates outperformed conventional gradient-based methods. In addition, for nonconvex objective functions that have Lipschitz continuous gradient, sufficient conditions for the Heavy-ball trajectories to converge to the optimal solution were provided in [15]. However, original results on the global convergence rate of the Heavy-ball method for convex problems are not established yet. In [16], it was demonstrated that Heavy-ball method does not necessarily converge when the objective function is strongly convex. Therefore, one can conclude that there is no evidence whether the Heavy-ball can outperform both the Nesterov and the gradient descent methods [16].

In spite of the substantial developments in first-order convex optimization methods, some questions related to their rate of convergence remain open. Motivated by the above considerations, this manuscript introduces an extension to the current framework used to evaluate the convergence rate of iterative algorithms. First, we use the fixed point analysis made in [17] for an algorithm that we call, Double Momentum Algorithm. This algorithm encapsulates on it several well-known first-order optimization methods. Using the fixed point analysis, we reach to a time-varying matrix which characterizes the optimization method at each iteration. By iterating enough times, the algorithm will surely converge to the minimum of the objective function. However, the non-asymptotic rates of convergence obtained in each case are derived based on the spectral radius analysis.

Despite the fact that the spectral radius of a matrix (i.e., linear transformation) is a rigorous measure for proving the convergence, it does not draw any conclusions in the case when the spectral radius is equal to one (i.e., the optimization method may or may not converge to the minimum point). In addition, when the spectral radius of two different first-order optimization algorithms coincide, it is hard to tell which one of them will converge faster to the minimum. As a remedy, we first characterize a compact set shaped by the tuning parameters of the proposed Double Momentum Algorithm that can be avoided in order to ensure the convergence to the minimum. Subsequently, we propose an additional metric, which is the determinant of the transformation matrix to illustrate the convergence of the algorithm. Finally, we provide

*The research reported in this publication was done while the first author was a student at King Abdullah University of Science and Technology

[1]Eder Baron is with the Austrian Institute of Technology, 1210 Vienna,Austria, and also with the Automatic Control Laboratory, ETH Zurich, 8092 Zürich, Switzerland. ebaron@ethz.ch

[2]Salman Alsubaihi is with the National Center for Artificial Intelligence (NCAI), Saudi Data and Artificial Intelligence Authority (SDAIA), Riyadh, Kingdom of Saudi Arabia salsubaihi@sdaia.gov.sa

[3]Khaled Alshehri is with the Control and Instrumentation Engineering Department, King Fahd University of Petroleum and Minerals (KFUPM), Dhahran, Saudi Arabia. He is also with the Interdisciplinary Research Center for Smart Mobility and Logistics, KFUPM. kalshehri@kfupm.edu.sa

[4]Fahad Albalawi is with the Department of Electrical Engineering, College of Engineering, Taif University, P.O. Box 11099, Taif 21944, Saudi Arabia f.albiloi@tu.edu.sa

necessary and sufficient conditions that relate the spectral radius and the determinant of the transformation matrix to prove the stability of our proposed framework using tools from robust control theory and linear operators theory.

Remark 1: An important distinction between the proposed method and the triple momentum algorithm introduced in [18] is the relaxation of the strong convexity condition; hence: the proposed method is suitable for problems where the underlying function is only convex. The impact of our main findings is illustrated by applying our algorithm to two-sided markets in non-cooperative games, where numerical results demonstrate the superior performance of our approach compared to the first-order methods such as gradient descent, Nesterov, and Heavy-ball.

II. Mathematical Preliminaries

Consider the following unconstrained optimization problem

$$\min_{x\in\mathbb{R}^n} \quad f(x) \tag{1}$$

In this paper, our goal is to find a unifying approach for parameter selection in first-order methods for efficiently solving (1). Before we state our results, we discuss some mathematical preliminaries and our assumptions in this work.

Definition 1: Consider a differentiable function $f:\mathbb{R}^n\to\mathbb{R}$, then:

- f is convex if $f(\xi x+(1-\xi)y)\leq \xi f(x)+(1-\xi)f(y)$, where $\xi\in[0,1]$ and $x,y\in\mathbb{R}^n$.
- f has gradient L-Lipschitz if $||\nabla f(y)-\nabla f(x)||\leq L||y-x||$, for some $L>0$, and all $x,y\in\mathbb{R}^n$.

If a function f is twice continuously differentiable, then, it is convex if and only if the Hessian satisfies $\nabla^2 f(\cdot)\succeq 0$. Despite of the value of the information that $\nabla^2 f(\cdot)$ provides, it is usually difficult to calculate it in a deterministic fashion. However, if the gradient of the function f is L-Lipschitz, it is possible to exploit the property $\nabla^2 f(\cdot)\preceq LI_n$,where I_n is an identity matrix of size $n\times n$, and L is the L-Lipschitz constant [19]. An attractive property of convex functions is that the first-order necessary condition of optimality is also sufficient, and any local minimum is also global [20]. Thus, an optimal solution x^* to (1) can be found by solving $\nabla f(x^*)=0$. The goal of first-order methods is to find algorithms converging to an x^* that satisfies this condition, and hence solving (1). Furthermore, x^* is unique if f is further strictly convex. Given the wide applicability of convex functions, and the strong mathematical guarantees that can be derived by their properties, throughout this paper, we have the following assumption.

Assumption 1: In solving (1), the function f is twice continuously differentiable, convex, and has a gradient L-Lipschitz.

III. Problem Statement

While there are several contributions advancing first-order methods, the convergence analysis done in the vast majority of those contributions is typically focused on the specific method at hand, which imposes a barrier to tune the parameters based on the user's objectives. This calls for a unified approach that takes into account the underlying trade-offs, and generalizes the well-known bounds and optimal parameter selections. Therefore, we consider an algorithm that includes two momentum formation, one considering the state of the variable to be optimized and one considering the gradient of the function, and we call it the *Double Momentum Algorithm*. Let x_k be a point in the feasible set of $f(\cdot)$ at time k, and denote the tuning parameters by $\beta,\ \alpha$ and η. Then, we have the update rule:

$$\begin{aligned} x_{k+1}\to&(1+\beta)x_k-\beta x_{k-1}-\\ &\alpha((1+\eta)\nabla f(x_k)-\eta\nabla f(x_{k-1})), \end{aligned} \tag{2}$$

where $\beta\in\mathbb{R},\ \eta\in\mathbb{R},\ \alpha\in\mathbb{R}_+$. The update rule (2) unifies a considerable amount of first-order methods with at most one step of memory. Some special and well-known cases are: gradient descent ($\beta=0,\ \eta=0$), Heavy-ball ($\eta=0$), optimistic gradient ($\beta=0,\ \eta=1$), or even, the early Nesterov method ($\beta=\eta$) [21]–[23]. As a remark, (2) is a linear mapping from $\mathbb{R}^n\to\mathbb{R}^n$. In other words, (2) can be seen as $x_{k+1}=A_k x_k$, where A_k is the characterization of the linear map. Given the generality of (2), in this paper, we answer the following key question:

Question 1 (Main Question): What are the conditions over the Double Momentum Algorithm, (2) *such that , when f is convex, twice differentiable and has a gradient L-Lipschitz, the iterative implementation of the aforementioned algorithm converges to the minimum?*

Addressing the above question generalizes existing bounds previously reported in [24]–[26]. Consequently, in this paper, we do this by addressing the following question:

Question 2: How can we find an analytical expression of the spectral radius and the determinant of (2) *as a function of $\alpha,\ \beta$ and η?*

In other words, we will give an asymptotic condition for convergence of the Double Momentum Algorithm and elaborate on the non-asymptotic convergence rate. Therefore, our problem can be rephrased as finding conditions where it is possible to minimize the determinant for (2), meanwhile, the spectral radius satisfies $\rho(A)\leq 1$. Consequently, addressing the above questions can be done by formulating and solving two problems. The first one, is to find the linear operator which encapsulates the dynamics of the Double Momentum Algorithm. Then, the second problem consists in bounding the maximum spectral radius that the algorithm can have, and then, calculate the determinant, which is the key to calculate the non-asymptotic convergence rate.

To the best of the authors' knowledge, the above approach and the claimed results have not been proposed in the literature. Hence, we analyze both the spectral radius and the determinant of the proposed linear iterative algorithm for functions that satisfy Assumption. 1. Then, we provide necessary and sufficient conditions that can guarantee convergence to the minimum.

IV. Main Results

The main results of this paper are organized as follows: First, we provide an illustration on how the determinant and spectral radius can be utilized as a necessary and sufficient conditions for the algorithm of (2) to converge to its minimum, and hence answer Question 1. Then, in Proposition 1, we provide an explicit expression of the spectral radius and the determinant of the Double Momentum Algorithm . Finally, equipped with Proposition 1, we answer Question 2 in Proposition 3.

A. Spectral Analysis

To analyze the magnitude of the iterative process of the algorithm, we use spectral radius analysis. Consider an $n\times n$

matrix A, and vector space $\mathcal{A} \in \mathbb{R}^n$, and let $A : \mathcal{A} \to \mathcal{A}$. For an appropriately defined A, we can represent how first-order methods iterate, and it is well-known that convergence to an optimal solution is guaranteed when $0 \leq \rho(A) < 1$, where $\rho(A) := \max_{1 \leq i \leq n} |\lambda_i|$ and λ_i is the i-th eigenvalue. The certification of convergence using the spectral radius as a measure is robust. When it is used as a convergence metric for an iterative algorithm, the spectral radius demonstrates how the iterative change in the vector x_{k+1} will contract the space until convergence to a single point if $\rho(A) < 1$. Nonetheless, these conditions can be relaxed by integrating sufficient and necessary conditions with the spectral characteristics of the matrix A.

As it was stated before, the spectral radius sufficient condition can be relaxed as a convergence certificate of the algorithm, making it necessary but not sufficient if we consider $\rho(A) \leq 1$ instead of $\rho(A) < 1$. Then, we need to formulate a sufficient condition based on the determinant of the matrix A. In addition, when two linear iterative algorithms have the same spectral radius, then one needs an additional metric to assess the convergence improvement of one over the other. For example, consider the following two matrices:

$$A_1 = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 1 \\ 0 & 0.7 \end{bmatrix}$$

Clearly, $\rho(A_1) = \rho(A_2) = 1$, but $det(A_1) \neq det(A_2)$ and as it will be shown later that the linear iterative process defined by A_2 will converge faster than the one governed by A_1 due to the fact that $det(A_1) = 0.7 < det(A_2)$. Roughly speaking, the determinant of A is a measure of how much the vector $z_{k+1} = [x_{k+1} \;\; x_k]^\top$ is expanded or contracted under the application of the transformation A to the vector $z_k = [x_k \;\; x_{k-1}]^\top$,i.e., we want to measure how much is the change of z_k when a transformation given by the matrix A is applied.

Example 1 (x with $n = 2$, inspired by [27]): Assume the matrix A has dimensions $2n \times 2n$, in consequence $z_k = [x_k \;\; x_{k-1}]^\top \in \mathbb{R}^4$. In this example, without loss of generality we can assume that x_k and x_{k-1} are orthonormal and define a paralelepid that we called X_k. The area of X_k is unitary, given that x_k and x_{k-1} are orthonormal as can be seen in Fig. 1. Then, A will change the shape and orientation of z_k, transform it into z_{k+1}. Where the new parallelepiped defined by the vectors $x_k + 1$ and x_k has an area defined by

$$\text{area}(X_{k+1}) = \text{area}(X_k)|\det(A)|.$$

Therefore, an iterative process where A is applied several times to the vector z_k will contract the area of the parallelepiped X_k if the determinant of A is less than one, and consequently, they will converge to the same point.
In other words, the determinant measures how much we are expanding or contracting a set of vectors that define a volume in the space. This concept is generalized in Lemma 1.

Lemma 1 (Adopted from [27]): Let A be an $n \times n$ matrix, and let $z_{k+1} : \mathbf{R}^n \to \mathbf{R}^n$ be the associated matrix transformation $z_{k+1} = Az_k$. If z_k defines any region in $\mathbf{R}^n$ called X_k, then

$$\text{vol}(X_{k+1}) = |\det(A)| \cdot \text{vol}(X_k)$$

Proof. The proof is structured in two main parts. Initially, we make the relationship between the absolute value of the determinant of a matrix and the volume of the parallelepiped formed by the vectors conforming to the matrix A. Therefore,

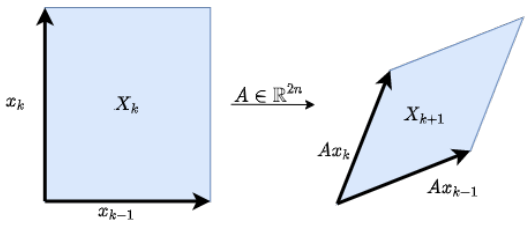

Fig. 1: The transformation process made at each iteration. We base our analysis in the possibility of calculating the determinant of A, which is the gain applied to the area of X_k, which results in the area of X_{k+1}.

we prove using linear transformation properties of A, that $\text{vol}(X_{k+1}) = |\det(A)| \cdot \text{vol}(X_k)$.

The volume of regions defined by vectors has four main properties met by the absolute value of the determinant of A. The first one is related to row replacement in A when a row replacement is made, it replaces the vector x_k with a vector parallel to x_{k-1}, and it is possible to notice that this movement does not affect the volume, given that the volume is calculated by multiplying the base of the parallelepiped and the height, as it is a parallel translation with respect to the base, it does not change the volume of X_k. The second main property is that scaling a row for a constant m, scales the volume by m. Consider Fig. 1(a), if x_k is scaled by m, the volume of the region will be $\text{Vol}(X) = m \cdot x_k \cdot x_{k-1} = m$. The third one relates to swapping the columns of the matrix. It is trivial that the determinant of a matrix will be maintained when the columns of a transformation matrix are swapped. It always has the same volume, independently of the order of the vectors in the plane. Finally, we have a unitary matrix transformation, which is the identity matrix. This matrix has a unitary volume since its vectors are defined to be unitary hypercubes. Therefore, it is possible to prove that $|det(A)| = vol(X_{k+1})$.

Now, as we already proved that the underlying volume of a linear transformation is measured exactly by the determinant of it, it readly follows that a linear transformation meets two properties. Namely, $A(x + y) = Ax + Ay$ and $A(mx) = m \cdot Ax$, where m is a scalar. Therefore, any volume defined by the vector $x \in \mathbb{R}^n$ will be scaled by $|det(A)|$. In such case, we can conclude that,

$$\text{vol}(X_{k+1}) = |\det(A)| \cdot \text{vol}(X_k).$$

■

B. Spectral radius and determinant of the Double Momentum Algorithm

In this subsection, we will derive the expressions representing the spectral radius and the determinant of the Double Momentum Algorithm. Furthermore, we will characterize the space where the tuning parameters make the algorithm stable. Proposition 1 makes use of the fixed point analysis to find the expressions for the determinant and the spectral radius of the Double Momentum Algorithm. In Proposition 2, we provide necessary conditions and prove that the spectral radius of the Double Momentum Algorithm is upper bounded by one when the tuning parameters fulfill specific requirements. Finally, Proposition 3 provides guarantees about the relations between

the tuning parameters and offers non-asymptotic bounds for multivariate convex functions.

***Assumption** 2:* The $\nabla^2 f(x)$ is a diagonal matrix.

Despite the fact that Assumption 2 seems to be a bit restrictive, the proposed Double Momentum Algorithm can be extended to convex functions that have a diagonally dominant Hessian (i.e., $|\nabla^2 f(x)_{i,i}| \geq \sum_{j \neq i} |\nabla^2 f(x)_{i,j}|$), which we relegate to future work. This work is intended to be a stepping stone towards the generalization of the triple momentum algorithm proposed in [18] to convex functions instead of only strongly convex functions. Furthermore, as we show later when we apply our method to two-sided markets in non-cooperative games, the Hessian of the objective function is not diagonal and the proposed method outperformed Nestrov, Gradient Descent and Heavy-ball methods.

Proposition 1 (Expressions for $\rho(A)$ & $det(A)$): Under Assumptions 1-2, the Double Momentum Algorithm has the following expression for the determinant

$$\det(A) = \prod_{\forall i} |\beta - \alpha\eta\nabla^2 f(z_{k-1}, i)|.$$

where $\nabla^2 f(z_{k-1}, i)$ makes reference to the diagonal terms (i, i) of the Hessian of $f(x)$. Besides, the following expression for the spectral radius is found

$$\rho(A) = \max_{\forall i} \left[\frac{1}{2} \left[1 + y^\top b \pm \left[y^\top b b^\top y - y^\top c + 1 \right]^{1/2} \right] \right] \tag{3}$$

where,

$$y := \begin{bmatrix} \eta\alpha & \alpha & \beta \end{bmatrix}^\top$$
$$b := \begin{bmatrix} -\nabla^2 f(z_k, i) & -\nabla^2 f(z_k, i) & 1 \end{bmatrix}^\top$$
$$c := \begin{bmatrix} -2\nabla^2 f(z_{k-1}, i) & 2\nabla^2 f(z_{k-1}, i) & 2 \end{bmatrix}^\top$$

Proof. Recall the update rule given by the Double Momentum Algorithm in (2)

$$\begin{aligned} x_{k+1} =& (1+\beta)x_k - \beta x_{k-1} \\ & - \alpha((1+\eta)\nabla f(x_k) - \eta\nabla f(x_{k-1})) \end{aligned} \tag{4}$$

Initially, consider arbitrary points $x, y \in \mathbb{R}^n$. Applying the transformation yields

$$\begin{aligned} x_{k+1} - y_{k+1} =& (1+\beta)x_k - \beta x_{k-1} - (1+\beta)y_k + \beta y_{k-1} \\ & - \alpha((1+\eta)\nabla f(x_k) - \eta\nabla f(x_{k-1})) \\ & + \alpha((1+\eta)\nabla f(y_k) - \eta\nabla f(y_{k-1})) \end{aligned} \tag{5}$$

Reorganizing the terms in (5)

$$\begin{aligned} x_{k+1} - y_{k+1} =& (1+\beta)(x_k - y_k) - \beta(x_{k-1} - y_{k-1}) \\ & - \alpha(1+\eta)(\nabla f(x_k) - \nabla f(y_k)) \\ & + \alpha\eta(\nabla f(x_{k-1}) - \nabla f(y_{k-1})) \end{aligned}$$

Next, we apply the mean value theorem on ∇f, with $z_k \in [x_k, y_k]$ and $z_{k-1} \in [x_{k-1}, y_{k-1}]$

$$\begin{aligned} x_{k+1} - y_{k+1} =& (1+\beta)(x_k - y_k) - \beta(x_{k-1} - y_{k-1}) - \\ & (\alpha + \alpha\eta)\nabla^2 f(z_k)(x_k - y_k) \\ & + \alpha\eta\nabla^2 f(z_{k-1})(x_{k-1} - y_{k-1}) \end{aligned}$$

Reorganizing terms yields

$$\begin{aligned} x_{k+1} - y_{k+1} =& \left((1+\beta)I_n - (\alpha + \alpha\eta)\nabla^2 f(z)\right)(x_k - y_k) \\ & - \left(\beta I_n - \alpha\eta\nabla^2 f(z_{k-1})\right)(x_{k-1} - y_{k-1}) \end{aligned} \tag{6}$$

Next, we can represent (6) using a state space discrete system representation as follows:

$$\begin{bmatrix} x_{k+1} - y_{k+1} \\ x_k - y_k \end{bmatrix} = A \begin{bmatrix} x_k - y_k \\ x_{k-1} - y_{k-1} \end{bmatrix}$$

where

$$A = \begin{bmatrix} (1+\beta)I_n - (\alpha + \alpha\eta)\nabla^2 f(z_k) & -\beta I_n + \alpha\eta\nabla^2 f(z_{k-1}) \\ I_n & 0_n \end{bmatrix}.$$

Without loss of generality we can assume that y_0 is any point in the plane, and let $y_0 = x^*$. With k recursive iterations, we have

$$\begin{bmatrix} x_{k+1} - x^* \\ x_k - x^* \end{bmatrix} = A^k \begin{bmatrix} x_1 - x^* \\ x_0 - x^* \end{bmatrix}$$

By the triangular inequality, we also obtain

$$\left|\left| \begin{bmatrix} x_{k+1} - x^* \\ x_k - x^* \end{bmatrix} \right|\right| \leq ||A^k|| \left|\left| \begin{bmatrix} x_1 - x^* \\ x_0 - x^* \end{bmatrix} \right|\right|. \tag{7}$$

When the Hessian $\nabla^2 f(\cdot)$ is diagonal and by the properties of f, we have $\nabla^2 f(\cdot) \preceq L I_n$. If we look closely we can notice that all the terms in A are diagonal matrices or scalars. Therefore, we can break the problem into n independent and identical problems of dimension 2×2, it suffices to focus on the eigenvalues and determinants of the following matrix

$$\begin{bmatrix} 1 + \beta - (\alpha + \alpha\eta)\nabla^2 f(z_k, i) & -\beta + \alpha\eta\nabla^2 f(z_{k-1}, i) \\ 1 & 0 \end{bmatrix} \tag{8}$$

It is clear that $0 \leq \nabla^2 f(z_k, i) \leq L$. The determinant of the matrix A, $\det(A)$, can be calculated easily given its structural characteristics of (8). By swapping the columns, which does not affect the determinant magnitude, but it does affect the sign [28], it readily follows that

$$\det(A) = \prod_{\forall i} |\beta - \alpha\eta\nabla^2 f(z_{k-1}, i)|.$$

Furthermore, from (8) is possible to derive the eigenvalues of A, using the eigenvalue problem $det(\lambda I - A) = 0$. Which leads to the following equality

$$[\lambda_1, \ldots, \lambda_i] = \left[\frac{1}{2} \left[1 + y^\top b \pm \left[y^\top b b^\top y - y^\top c + 1 \right]^{1/2} \right] \right]$$

where,

$$y := \begin{bmatrix} \eta\alpha & \alpha & \beta \end{bmatrix}^\top$$
$$b := \begin{bmatrix} -\nabla^2 f(z_k, i) & -\nabla^2 f(z_k, i) & 1 \end{bmatrix}^\top$$
$$c := \begin{bmatrix} -2\nabla^2 f(z_{k-1}, i) & 2\nabla^2 f(z_{k-1}, i) & 2 \end{bmatrix}^\top$$

Therefore, the spectral radius is $\max_{\forall i}\{\lambda_1, \ldots, \lambda_{2n}\}$, which is reflected in (3).

■

Therefore, as a consequence of Proposition 1, if we choose $\alpha = \frac{1}{L}$, following the common literature choices for the stepsize, it suffices to determine wisely the coefficients β and η such that the determinant of A is less than one, or in other words that A is a contractive linear operator. However, it is necessary to be aware that the spectral radius must be necessarily less than one, as is proved in Proposition 2

Proposition 2 (Necessary Condition): Under Assumptions 1-2, if $0 \leq \beta \leq 1, \quad 0 \leq \eta \leq 1$, and $\alpha = 1/L$, then the maximum spectral radius of (8), $\rho(A) = 1$.

Proof. The proof is structured in three steps. First, we evaluate the function between $\beta \in [0,1]$, $\eta \in [0,1]$ and $\nabla^2 f(x) * \alpha \in [0,1]$. Therefore, we obtain some boundary region where the spectral radius is greater than one. Second, we prove that (3) is a monotonic function in all variables, namely, β, η and $\nabla^2 f(x) * \alpha$ in the region where the spectral radius is greater than one. Finally, as we set that the region is compact, there is no point where we can guarantee that $\rho(A) \leq 1$. Therefore, it is possible to define a set with a spectral radius of less than one.

The derivatives of (2) with respect to each variable are

$$\frac{\partial \rho(A)}{\partial \delta} = \frac{-1-\eta \pm \frac{-\beta(\eta+1)+\eta^2\delta+2\eta\delta+\eta+\delta-1}{\sqrt{(\beta-(\eta+1)\delta+1)^2-4\beta+4\eta\delta}}}{2} \tag{9}$$

$$\frac{\partial \rho(A)}{\partial \eta} = \frac{\delta\left(-1 \pm \frac{-\beta+\delta\eta+\delta+1}{\sqrt{(\beta-\delta(\eta+1)+1)^2-4\beta+4\delta\eta}}\right)}{2} \tag{10}$$

$$\frac{\partial \rho(A)}{\partial \beta} = \frac{\left(1 \pm \frac{\eta\delta+\delta-\beta+1}{\sqrt{(-\eta\delta-\delta+\beta+1)^2+4\eta\delta-4\beta}}\right)}{2} \tag{11}$$

As (9),(10) and (11) do not have any change of sign for the set where the spectral radius is greater than one. In consequence, we can ensure that inside a 3-D polygon (that is shaped by β, η and α) there is no point where the spectral radius is less or equal than one.

■

Remark 2: Proposition 2 relies on the normalization of $\nabla^2 f(x)$ using the tuning parameter α. Since the tuning parameter $\alpha = \frac{1}{L}$ depends on the Lipschitz constant L, it is remarkable that the correct election of L is crucial for our approach. Given that, an overestimated L can result in a small δ, making the algorithm unstable.

The previous proof shows that the Double Momentum Algorithm has at most a unitary spectral radius inside the delimiting set, and therefore it meets the necessary condition for convergence. It is clear that depending on the choice of β and η, the determinant will decrease and therefore improve the convergence rate of the algorithm as it will be illustrated in the next proposition.

Proposition 3 (Sufficient Condition): Under Assumptions 1-2, if $0 \leq \beta < \eta \leq 1$, and $\alpha = 1/L$, then, the Double Momentum Algorithm has a determinant less than one, hence the algorithm of (2) is stable.

Proof. According to Proposition 1, the determinant expression for the Double Momentum Algorithm can be expressed as $\det(A) = \prod_{\forall i} |\beta - \alpha\eta\nabla^2 f(z_{k-1}, i)|$. Therefore, to evaluate the effect of A in the vector we consider the possible relationships between β and η inside the set defined by the Proposition 2. The five cases considered were

$$\frac{\beta}{\eta} > 1 \qquad \frac{\beta}{\eta} < 1 \qquad \frac{\beta}{\eta} = 1 \qquad \beta = 0 \qquad \eta = 0 \tag{12}$$

Therefore, using the cases in (12), we can derive bounds on the determinant and based on these bounds, one can determine if there are behaving as a contractive or expansive system. Consider N as the final number of iterations in which our method will be applied. Therefore, we have to calculate the effect of the iteration process from the A matrix of (8). This is defined as

$$\prod_{k=1}^{N} \prod_{\forall i} |\beta - \alpha\eta\nabla^2 f(z_{k-1}, i)| \tag{13}$$

As we defined, $\alpha = \frac{1}{L}$, and by Assumption 1 we know that $\nabla^2 f(z_{k-1}) \leq L$. Therefore, as it is usually difficult to have access to information regarding the behavior of $\nabla^2 f(z_{k-1})$, we can approximate it to the worst case scenario, which is $\nabla^2 f(z_{k-1}) = L$. Therefore, we can rewrite (13) as

$$\prod_{k=1}^{N} \prod_{\forall i} |\beta - \eta\delta_{k,i}| \tag{14}$$

Where $0 \leq \delta_k \leq 1$, therefore, we have to evaluate the convergence bound for expression (14) under the cases exposed in (12).

Starting from (14), we can derive the following bounds

$$\begin{aligned}\prod_{k=1}^{N} \prod_{\forall i} |\beta - \eta\delta_{k,i}| &= \beta^N \prod_{k=1}^{N} \prod_{\forall i} \left|1 - \frac{\eta}{\beta}\delta_{k,i}\right| \\ &= \prod_{\forall i} \beta^N \left[\left|1 - \frac{\eta}{\beta}\delta_i\right|\right]^N\end{aligned}$$

Making $n = 1$, from the expression $\beta^N \left[\left|1 - \frac{\eta}{\beta}\delta_i\right|\right]^N$, we can derive the convergence bounds for the three first cases from (12).

Therefore, if $\frac{\beta}{\eta} > 1$ or $\frac{\beta}{\eta} < 1$, the non-asymptotic bound related with these conditions is $\beta^N \left[1 - \frac{\eta\delta}{\beta}\right]^N < 1$. The following bound is particularized for $\frac{\beta}{\eta} = 1$ and in consequence the non-asymptotic bound can be characterized as $\beta^N [1-\delta]^N < 1$. This last particularization is the bound for the early Nesterov method [11]. Furthermore, for the cases when $\eta = 0$ and $\beta = 0$, we can derive directly from (14) the non-asymptotic bounds. From the case $\eta = 0$ we obtain a bound given by $\beta^N < 1$, and from the case where $\beta = 0$ we obtain the bound $[\eta\delta]^N < 1$. Utilizing both Lemma 1 and the fact that the aforementioned bounds are non-asymptotically decreasing, we can conclude that the Double Momentum Algorithm has a determinant less than one, hence the algorithm of (2) is stable.

■

Remark 3: Utilizing the preceding Propositions requires knowledge of L. Estimating the Lipschtz constant L for white box functions (defined as functions where their analytical form of the objective and derivatives are known explicitly) has been developed decades ago for both single-valued functions [29] and multivariate functions [30]. We note that our results can be used for black box functions (defined as functions where their analytical form is unknown and only the function value can be evaluated) that do implicitly meet our assumptions. To estimate L for single-valued black box functions, a number of stochastic approaches have been designed where the largest slope in a fixed size sample of slopes was proven to have an approximate Reverse Weibull distribution. Then, the Weibull distribution is fitted to the largest slopes and the location parameter used as an estimator of L [31]. A generalization to multivariate black box functions was further discussed in [32].

V. Application: Two-Sided Markets in Non-cooperative Games

It is worth mentioning that our results in this paper have potential applications in equilibrium computation in non-cooperative games. First, consider games in which each player has a single decision to make; examples include Cournot competition [33], pricing games [34], parameterized bidding [35], [36] and many others. Consider a non-cooperative M-person game, where player $i \in \{1, \dots, M\}$ minimizes a twice differentiable function $f_i(x_i, x_{-i})$ over x_i, where x_{-i} denotes the vector of decisions made by other players. The vector x^* constitutes a Nash equilibrium if $f_i(x_i^*, x_{-i}^*) \leq f_i(x_i, x_{-i}^*),\ \forall x_i,\ \forall i$ [37]. If each x_i is constrained over a compact and convex set $\mathcal{X}_i \subseteq \mathbb{R}$, and if f_i is strictly convex in x_i for each i, by Rosen's results [38], the game admits at least one Nash equilibrium, and it is unique and stable if the game is strictly convex (this is more restrictive than just saying f_i is strictly convex in each x_i). On the other hand, Li and Başar [39] also provided conditions for the existence and uniqueness of a stable Nash equilibrium when $\mathcal{X}_i = \mathbb{R}$ and f_i is strictly convex in x_i. With a slight abuse of notation, at iteration k, let $x_{-i,k}$ denote the latest decisions reported by other players from the perspective of player i, and consider sequential updates of the following form:

$$x_{i,k+1} = \underset{x_i \in \mathbb{R}}{\operatorname{argmin}} f_i(x_i, x_{-i,k}), \quad \forall i, k = 0, 1, \dots.$$

The above sequence is convergent when the conditions in [38] or [39] hold, and our results in this paper become useful here as they bring insights into how to compute the sequence for each i at iteration k fast. We also note that our results in this paper do not impose strict convexity, so it would be interesting to check to what extent strict convexity can be relaxed for convergence to a NE. Finally, we remark that x_i is n-dimensional for each i, the results in [38] and [39] remain valid. So, the general research direction of NE computation using the Double Momentum Algorithm seems promising and remains to have interesting open questions. To bring deeper insights into the impacts of our results, we consider the *Scalar-Parametrized Two-Sided Market Competition* purposed in [36], which has strong motivations from the electricity markets literature.

Consider a two-sided market with M consumers and N suppliers, so we have an $M+N$ non-cooperative game. The market maker receives demand bids and supply offers from market participants, and then clears the market. We note that here, due to the nature of the game, players are strategic in their bids and offers, as they take into consideration the effect of their bids/offers on the cleared market price and quantities. Due to their mathematical attractiveness and desirable efficiency guarantees, one might consider scalar-prametrized bids/offers [35], [36]. Here, we consider the game studied in [36]. For given market price p, a consumer sends $\theta_i^d > 0$ to the market maker, and his demand curve is represented by

$$D(\theta_i^d, p) := d_0 + \frac{\theta_i^d}{p},$$

where d_0 is a constant.[1] For a given p, $D(\theta_i^d, p)$ represents the quantity i is willing to buy. Similarly, for the supply-side,

[1]One can think of d_0 as the 'inelastic' portion of the demand.

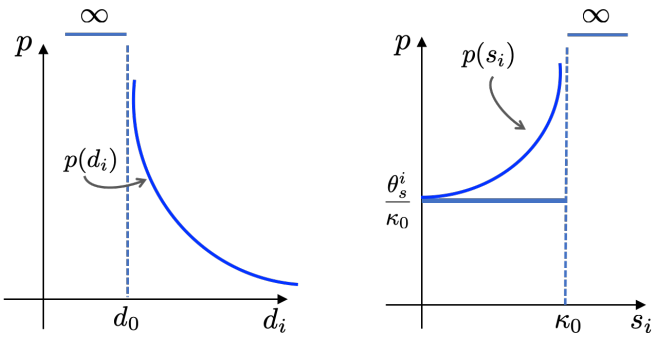

Fig. 2: Illustrations of the demand bid and supply offer [36]

a supplier sends $\theta_i^s > 0$, and his supply curve is given by

$$S(\theta_i^s, p) := \kappa_0 - \frac{\theta_i^s}{p},$$

where κ_0 is the capacity. Figure 2 illustrates the demand and supply curves. Because of the game, the price p is in fact a function of all θ's, and because of the supply-demand balance and competition in electricity markets, it is given by

$$p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s) := \frac{\mathbf{1}^T\boldsymbol{\theta}^d + \mathbf{1}^T\boldsymbol{\theta}^s}{N\kappa_0 - Md_0},$$

where $\mathbf{1}$ is a vector of ones of appropriate dimension, $\boldsymbol{\theta}^d$ is the vector of all parameters sent from consumers, and $\boldsymbol{\theta}^s$ is similarly defined for suppliers. With the above $p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s)$, each player maximizes her payoff. For consumer i, her payoff is given by

$$f_i^d := U_i\left(D\left(\theta_i^d, p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s)\right)\right) - p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s) D\left(\theta_i^d, p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s)\right),$$

where the first term corresponds to the utility of consumption, and the second term corresponds to the cost. For supplier i, the payoff is given by

$$f_i^s := p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s) S\left(\theta_i^s, p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s)\right) - C_i\left(S\left(\theta_i^s, p(\boldsymbol{\theta}^d, \boldsymbol{\theta}^s)\right)\right),$$

where the first term corresponds to the revenue, and the second term corresponds to the cost. Under mild assumptions, the above game admits a unique Nash equilibrium [36]. Here, we demonstrate that for the stylized example discussed in [36], using our Double Momentum Algorithm, we can converge to the equilibrium fast. For that, we have the following result.

Proposition 4: Let the utility functions be logarithmic and the costs be quadratic of the following form

$$U_i(d_i) := a_i \log(d_i), \qquad C_i(s_i) := \frac{b_i}{2} s_i^2,$$

where $a_i, b_i > 0$ for each player i. Then, the Lipschitz constants are given by

$$L_i^d = \frac{\bar{c}_1(\bar{c}_1 + 2)}{\bar{c}_2^2}, \qquad L_i^s = \frac{2\bar{c}_1(2\kappa_0 + \bar{c}_1)}{\bar{c}_3^2},$$

where

$$\bar{c}_1 = N\kappa_0 - Md_0 > 0,$$
$$\bar{c}_2 = \mathbf{1}^T\theta_{-i}^d + \mathbf{1}^T\boldsymbol{\theta}^s > 0,$$
$$\bar{c}_3 = \mathbf{1}^T\theta_{-i}^s + \mathbf{1}^T\boldsymbol{\theta}^d > 0.$$

Proof. We first rewrite the payoff equations for each side of the game. At first, we rewrite the payoff equation from the demand side as follows:

$$f_i^d = a_i \log(d_0 + \frac{\bar{c}_1\theta_i^d}{\theta_i^d + \bar{c}_2}) - \bar{r}\theta_i^d - \bar{q} \tag{15}$$

where

$$\bar{r} = \frac{d_0}{\bar{c}_1} + 1, \qquad \bar{q} = \frac{d_0\bar{c}_2}{\bar{c}_1}.$$

We want to show that f_i^d is gradient Lipschitz, i.e. show that $|f_i^{d^{''}}(\theta_i^d)| \leq L_i^d$. For the sake of simplicity, we let $a_i = 1$ for all $i = 1, \ldots, M$. At the end, we will scale the final Lipschitz bound by a_i in case $a_i \neq 1$ for some $i = 1, \ldots, M$.

The first and second derivatives of f_i^d are as follows:

$$f_i^{d^{'}}(\theta_i^d) = \frac{\bar{c}_1\bar{c}_2}{(\bar{c}_1+1)\theta_i^{d^2} + \bar{c}_2(\bar{c}_1+2)\theta_i^d + \bar{c}_2^2} - \bar{r}$$

$$f_i^{d^{''}}(\theta_i^d) = -\frac{2\bar{c}_1\bar{c}_2(\bar{c}_1+1)\theta_i^d + \bar{c}_1\bar{c}_2^2(\bar{c}_1+2)}{\Big((\bar{c}_1+1)\theta_i^{d^2} + \bar{c}_2(\bar{c}_1+2)\theta_i^d + \bar{c}_2^2\Big)^2}$$

Since the parameters $\bar{c}_1$ and $\bar{c}_2$ are strictly positive, and θ_i^d is non-negative, we have:

$$|f_i^{d^{''}}(\theta_i^d)| = \frac{2\bar{c}_1\bar{c}_2(\bar{c}_1+1)x + \bar{c}_1\bar{c}_2^2(\bar{c}_1+2)}{\Big((\bar{c}_1+1)\theta_i^{d^2} + \bar{c}_2(\bar{c}_1+2)\theta_i^d + \bar{c}_2^2\Big)^2}$$

$$\leq \frac{2\bar{c}_1\bar{c}_2(\bar{c}_1+1)\theta_i^d + \bar{c}_1\bar{c}_2^2(\bar{c}_1+2)}{\Big(\bar{c}_2(\bar{c}_1+2)\theta_i^d + \bar{c}_2^2\Big)^2}$$

due to the fact that $(\bar{c}_1+1)\theta_i^{d^2} \geq 0$

$$= \frac{2\bar{c}_1\bar{c}_2(\bar{c}_1+1)\theta_i^d + \bar{c}_1\bar{c}_2^2(\bar{c}_1+2)}{\bar{c}_2^2(\bar{c}_1+2)^2\theta_i^{d^2} + 2\bar{c}_2^3(\bar{c}_1+2)\theta_i^d + \bar{c}_2^4}$$

$$\leq \frac{2\bar{c}_2(\bar{c}_1+1)\theta_i^d + \bar{c}_1\bar{c}_2^2(\bar{c}_1+2)}{2\bar{c}_2^3(\bar{c}_1+2)\theta_i^d + \bar{c}_2^4}$$

due to the fact that $\bar{c}_2^2(\bar{c}_1+2)^2\theta_i^{d^2} \geq 0$. Now, we apply long division to reach to the following equation

$$= \frac{\bar{c}_1(\bar{c}_1+1)}{\bar{c}_2^2(\bar{c}_1+2)} + \frac{\bar{c}_1(\bar{c}_1^2+3\bar{c}_1+3)}{2\bar{c}_2(\bar{c}_1+2)^2\theta_i^d + \bar{c}_2^2(\bar{c}_1+2)}$$

$$:= \overline{|f_i^{d^{''}}(\theta_i^d)|}$$

Note that the terms $\bar{c}_1(\bar{c}_1^2+3\bar{c}_1+3)$, $2\bar{c}_2(\bar{c}_1+2)^2$ and $\bar{c}_2^2(\bar{c}_1+2)$ are strictly positive. Therefore, for finite and strictly positive parameters $\bar{c}_1$ and $\bar{c}_2$, we have:

$$|f_i^{d^{''}}(\theta_i^d)| \leq \overline{|f_i^{d^{''}}(\theta_i^d)|} \leq \overline{|f_i^{d^{''}}(0)|} < \infty \quad \forall \theta_i^d \geq 0.$$

Furthermore, note that this bound is actually tight, since $|f_i^{d^{''}}(0)| = \overline{|f_i^{d^{''}}(0)|}$. Finally, we have:

$$|f_i^{d^{''}}(\theta_i^d)| \leq \overline{|f_i^{d^{''}}(0)|} = \frac{\bar{c}_1(\bar{c}_1+2)}{\bar{c}_2^2} = L_i^d$$

On the other hand, the payoff equation for each player from the supply side can be rewritten as follows:

$$f_i^s = -\frac{b_i}{2}[\kappa_0 - \frac{\bar{c}_1\theta_i^s}{\theta_i^s + \bar{c}_3}]^2 + \bar{c}_4\theta_i^s + \bar{c}_5 \tag{16}$$

where

$$\bar{c}_4 = \frac{\kappa_0}{\bar{c}_1} - 1 > 0, \qquad \bar{c}_5 = \frac{\kappa_0\bar{c}_3}{\bar{c}_1} > 0.$$

We want to show that f_i^s is gradient Lipschitz, i.e. show that $|f_i^{s^{''}}(\theta_i^s)| \leq L_i^s$. For the sake of simplicity, we let $b_i/2 = 1$ for all $i = 1, \ldots, N$. At the end, we will scale the final Lipschitz bound by $b_i/2$ in case $b_i/2 \neq 1$ for some $i = 1, \ldots, N$. The first and second derivatives of f_i^s are as follows:

$$f_i^{s^{'}}(\theta_i^s) = -2\bar{c}_1\bar{c}_3\,\frac{(\bar{c}_1-\kappa_0)\theta_i^s - \kappa_0\bar{c}_3}{(\theta_i^s+\bar{c}_3)^3}$$

$$f_i^{s^{''}}(\theta_i^s) = -2\bar{c}_1\bar{c}_3\,\frac{2(\kappa_0-\bar{c}_1)\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)}{(\theta_i^s+\bar{c}_3)^4}$$

Since the parameters $\bar{c}_1, \bar{c}_3$ and κ_0 are strictly positive, and θ_i^s is non-negative, we have:

$$|f_i^{s^{''}}(\theta_i^s)| = \left| -2\bar{c}_1\bar{c}_3\,\frac{2(\kappa_0-\bar{c}_1)\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)}{(\theta_i^s+\bar{c}_3)^4} \right|$$

$$= 2\bar{c}_1\bar{c}_3\,\frac{|2(\kappa_0-\bar{c}_1)\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)|}{(\theta_i^s+\bar{c}_3)^4}$$

By using triangle inequality

$$\leq 2\bar{c}_1\bar{c}_3\,\frac{2|\kappa_0-\bar{c}_1|\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)}{(\theta_i^s+\bar{c}_3)^4}$$

$$\leq 2\bar{c}_1\bar{c}_3\,\frac{2(\kappa_0+\bar{c}_1)\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)}{(\theta_i^s+\bar{c}_3)^4}$$

$$= 2\bar{c}_1\bar{c}_3\,\frac{2(\kappa_0+\bar{c}_1)\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)}{\theta_i^{s4} + 4\bar{c}_3\theta_i^{s3} + 6\bar{c}_3^2\theta_i^{s2} + 4\bar{c}_3^3\theta_i^s + \bar{c}_3^4}$$

Due to the fact that $\theta_i^{s4} + 4\bar{c}_3\theta_i^{s3} + 6\bar{c}_3^2\theta_i^{s2} \geq 0$

$$\leq 2\bar{c}_1\bar{c}_3\,\frac{2(\kappa_0+\bar{c}_1)\theta_i^s + \bar{c}_3(2\kappa_0+\bar{c}_1)}{4\bar{c}_3^3\theta_i^s + \bar{c}_3^4}$$

By applying the long division to the previous expression, we reach to the following two terms

$$= \frac{\bar{c}_1(\kappa_0+\bar{c}_1)}{\bar{c}_3^2} + \frac{\bar{c}_1(3\kappa_0+\bar{c}_1)}{4\bar{c}_3\theta_i^s + \bar{c}_3^2}$$

$$:= \overline{|f_i^{s^{''}}(\theta_i^s)|}$$

Note that the terms $\bar{c}_1(3\kappa_0+\bar{c}_1)$, $4\bar{c}_3$ and $\bar{c}_3^2$ are strictly positive. Therefore, for finite and strictly positive parameters $\bar{c}_1, \bar{c}_3$ and κ_0, we have:

$$|f_i^{s^{''}}(\theta_i^s)| \leq \overline{|f_i^{s^{''}}(\theta_i^s)|} \leq \overline{|f_i^{s^{''}}(0)|} < \infty \quad \forall \theta_i^s \geq 0.$$

Furthermore, note that this bound is actually tight, since $|f_i^{s^{''}}(0)| = \overline{|f_i^{s^{''}}(0)|}$. Finally, we have:

$$|f_i^{s^{''}}(\theta_i^s)| \leq \overline{|f_i^{s^{''}}(0)|} = \frac{2\bar{c}_1(2\kappa_0+\bar{c}_1)}{\bar{c}_3^2} = L_i^s$$

■

Now, equipped with the above analytical expressions of the Lipschitz constants, we are now ready to apply our algorithm. Similar to [36], we have the following values:

$$M = 5,\ a = [1, 1, 1.5, 2, 2],$$
$$N = 6,\ b = [0.1, 0.2, 0.3, 0.4, 0.5, 0.5],$$
$$d_0 = 1,\ \kappa_0 = 9.25.$$

Method	Iterations	Computation Time (Sec)
Nestrov	33,613	66
Heavy-ball	47,903	99
Gradient Descent	59,363	122
Double Momentum	23,488	49

TABLE I: Iteration and computation time for convergence to Nash equilibrium. Here, for Nestrov, we used $\eta = \beta = 0.2$, while for the Double Momentum algorithm, we used $\eta = 0.2, \beta = 0.8$ for all players. For Heavy-ball, $\beta = 0.2$ for all players, while $\eta = 0$ by definition.

With the above values, we use the popular choices of $\alpha_i^d = \frac{1}{L_i^d}$ and $\alpha_i^s = \frac{1}{L_i^s}$, and then, we utilize our results to choose β's and η's, and compare the performance with gradient descent, Heavy-ball, and Nestrov. Clearly, as shown in Table I, the Double Momentum algorithm is superior. Finally, we note that we have experimented with various parameter values and combinations and initial conditions, and similar conclusions were observed.

VI. Conclusion

For twice differentiable and gradient L-Lipschitz convex functions, we provided a parameter selection criteria for the Double Momentum Algorithm ensuring convergence to the optimal value in non-asymptotic rate. Then, we introduced for the first time the determinant of a matrix as a new metric for proving the convergence of the algorithm where necessary and sufficient conditions that connect the spectral radius of a linear transformation with the determinant were given. Finally, we demonstrated the proposed framework through non-cooperative games application. Providing an explicit convergence rate and relaxing Assumption 2 to convex functions with diagonally dominant Hessian are interesting directions for future work.

References

[1] O. Devolder, F. Glineur, and Y. Nesterov, "First-order methods of smooth convex optimization with inexact oracle," *Mathematical Programming*, vol. 146, pp. 37–75, 2014.

[2] Y. Drori and M. Teboulle, "Performance of first-order methods for smooth convex minimization: a novel approach," *Mathematical Programming*, vol. 145, pp. 451–482, 2014.

[3] R. Xin, S. Kar, and U. A. Khan, "Decentralized stochastic optimization and machine learning: A unified variance-reduction framework for robust performance and fast convergence," *IEEE Signal Processing Magazine*, vol. 37, pp. 102–113, 2020.

[4] X. Wu and J. Lu, "Fenchel dual gradient methods for distributed convex optimization over time-varying networks," *IEEE Transactions on Automatic Control*, vol. 64, pp. 4629–4636, 2019.

[5] Y. Nesterov, "Universal gradient methods for convex optimization problems," *Mathematical Programming*, vol. 152, pp. 381–404, 2015.

[6] E. Ghadimi, H. R. Feyzmahdavian, and M. Johansson, "Global convergence of the heavy-ball method for convex optimization," in *2015 European control conference (ECC)*, 2015, pp. 310–315.

[7] S. Boyd, S. P. Boyd, and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[8] T. Sun, P. Yin, D. Li, C. Huang, L. Guan, and H. Jiang, "Non-ergodic convergence analysis of heavy-ball algorithms," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 5033–5040.

[9] H. Karimi, J. Nutini, and M. Schmidt, "Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2016, pp. 795–811.

[10] Y. Nesterov, "A method of solving a convex programming problem with convergence rate $O(1/k^2)$," in *Sov. Math. Dokl*, vol. 27, 1983, pp. 543–547.

[11] ——, *Introductory lectures on convex optimization: A basic course*. Springer Science & Business Media, 2003.

[12] B. T. Polyak, "Some methods of speeding up the convergence of iteration methods," *Ussr computational mathematics and mathematical physics*, vol. 4, pp. 1–17, 1964.

[13] E. Ghadimi, I. Shames, and M. Johansson, "Multi-step gradient methods for networked optimization," *IEEE Transactions on Signal Processing*, vol. 61, pp. 5417–5429, 2013.

[14] H. Wang and P. C. Miller, "Scaled heavy-ball acceleration of the richardson-lucy algorithm for 3d microscopy image restoration," *IEEE Transactions on Image Processing*, vol. 23, pp. 848–854, 2013.

[15] S. Zavriev and F. Kostyuk, "Heavy-ball method in nonconvex optimization problems," *Computational Mathematics and Modeling*, vol. 4, pp. 336–341, 1993.

[16] L. Lessard, B. Recht, and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, pp. 57–95, 2016.

[17] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE transactions on Neural Networks*, vol. 10, pp. 626–634, 1999.

[18] B. Van Scoy, R. A. Freeman, and K. M. Lynch, "The fastest known globally convergent first-order method for minimizing strongly convex functions," *IEEE Control Systems Letters*, vol. 2, pp. 49–54, 2017.

[19] F. A. Graybill, *Matrices with applications in statistics*. Duxbury, 1983.

[20] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge: Cambridge University Press, 2010.

[21] Y. Nesterov, *Lectures on convex optimization second edition*. Cham, Switzerland: Springer Nature Switzerland AG, 2018.

[22] A. Jung, "A fixed-point of view on gradient methods for big data," *Frontiers in Applied Mathematics and Statistics*, vol. 3, p. 18, 2017.

[23] S. Sun, Z. Cao, H. Zhu, and J. Zhao, "A Survey of Optimization Methods from a Machine Learning Perspective," *IEEE Transactions on Cybernetics*, vol. 50, pp. 3668–3681, 2020.

[24] E. K. Ryu and S. Boyd, "Primer on monotone operator methods," *Appl. Comput. Math*, vol. 15, pp. 3–43, 2016.

[25] A. Mokhtari, A. Ozdaglar, and S. Pattathil, "A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach," in *International Conference on Artificial Intelligence and Statistics*, 2020, pp. 1497–1507.

[26] K. Mishchenko, D. Kovalev, E. Shulgin, P. Richtárik, and Y. Malitsky, "Revisiting stochastic extragradient," in *International Conference on Artificial Intelligence and Statistics*, 2020, pp. 4573–4582.

[27] D. Margalit and J. Rabinoff, "Interactive Linear Algebra," pp. 1–9, 2019.

[28] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[29] B. O. Shubert, "A sequential method seeking the global maximum of a function," *SIAM Journal on Numerical Analysis*, vol. 9, pp. 379–388, 1972.

[30] R. H. Mladineo, "An algorithm for finding the global maximum of a multimodal, multivariate function," *Mathematical Programming*, vol. 34, pp. 188–200, 1986.

[31] G. Wood and B. Zhang, "Estimation of the lipschitz constant of a function," *Journal of Global Optimization*, vol. 8, pp. 91–103, 1996.

[32] B. Zhang, "Topics in lipschitz global optimisation," Ph.D. dissertation, University of Canterbury. Mathematics and Statistics, 1995.

[33] G. Perakis and W. Sun, "Efficiency analysis of cournot competition in service industries with congestion," *Management Science*, vol. 60, pp. 2684–2700, 2014.

[34] X. Vives, "Edgeworth and modern oligopoly theory," *European Economic Review*, vol. 37, pp. 463–476, 1993.

[35] R. Johari and J. N. Tsitsiklis, "Parameterized supply function bidding: Equilibrium and efficiency," *Operations Research*, vol. 59, pp. 1079–1089, 2011.

[36] M. Ndrio, K. Alshehri, and S. Bose, "A scalar-parameterized mechanism for two-sided markets," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 16 952–16 957, 2020, 21st IFAC World Congress.

[37] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM Series in Classics in Applied Mathematics, 1998.

[38] J. B. Rosen, "Existence and uniqueness of equilibrium points for concave n-person games," *Econometrica*, vol. 33, pp. 520–534, 1965.

[39] S. Li and T. Başar, "Distributed algorithms for the computation of noncooperative equilibria," *Automatica*, vol. 23, pp. 523–533, 1987.