Momentum accelerates evolutionary dynamics

Marc Harper ∗ and Joshua Safyan

Google, Inc., Mountain View, CA, United States of America

Author to whom any correspondence should be addressed.

E-mail: marcharper@gmail.com

Keywords: evolutionary game theory, momentum, Lyapunov convergence, accelerated gradient descent

Abstract

We combine momentum from machine learning with evolutionary dynamics, where momentum can be viewed as a simple mechanism of intergenerational memory similar to epigenetic mechanisms. Using information divergences as Lyapunov functions, we show that momentum accelerates the convergence of evolutionary dynamics including the continuous and discrete replicator equations and Euclidean gradient descent on populations. When evolutionarily stable states are present, these methods prove convergence for small learning rates or small momentum, and yield an analytic determination of the relative decrease in time to converge that agrees well with computations. The main results apply even when the evolutionary dynamic is not a gradient flow. We also show that momentum can alter the convergence properties of these dynamics, for example by breaking the cycling associated to the rock–paper–scissors landscape, leading to either convergence to the ordinarily non-absorbing equilibrium, or divergence, depending on the value and mechanism of momentum.

1. Introduction

Gradient descent is commonly used in machine learning (ML) and in many scientific fields, including to model biological systems. Evolutionary algorithms are frequently mentioned as an alternative to gradient descent, particularly when the function to be minimized is not differentiable. With a long history in ML [1], evolutionary algorithms have found broad application, including in reinforcement learning [2, 3], neural architecture search [4], AutoML [5], and meta-learning [6], among other areas. Some evolutionary algorithms and dynamics can be understood in terms of gradient descent. The replicator equation is a model of natural selection, recognizable in special cases as gradient descent on a non-Euclidean geometry of the probability simplex where the potential function is the population mean fitness, also known as the natural gradient [7]. In ML, the Adam optimizer is an approximation of the natural gradient with momentum [8].

Powerful Lyapunov methods exist to analyze the replicator equation and its long run behavior is relatively well-understood. We show that these methods inform the action of the ML concept of momentum, a method to carry forward prior values of the gradient into further iterations of the replicator dynamic or gradient descent.

Momentum as used in ML has plausible evolutionary interpretations. Mechanisms of memory are abundant in biological and cultural systems, capturing complex adaptive functions within the lifetimes of organisms, including epigenetics [9] and cultural transmission of information [10]. However, the bias of these extra-genetic forms of memory may only last a few generations, as opposed to information incorporated more permanently in the genome, for example into a highly conserved gene, which may encode a more fundamental physical adaptation (e.g. heat-shock proteins [11]). Hence we may simplistically model a short-term memory mechanism as having an exponentially-decaying impact on natural selection by carrying over some memory of the fitness function of earlier generations. This is similar to the description of some epigenetic mechanisms which can pass on acquired information for a small number of generations.

We show that the addition of a simple exponentially-decaying memory mechanism accelerates the convergence of trajectories of the replicator equation [12] and its Euclidean analog. This mechanism is called (Polyak) momentum in the ML literature [13, 14], where it is known to increase the rate of convergence of
gradient descent quadratically (in terms of condition number) [15]. We will also consider Nesterov momentum [16, 17], which has a look-ahead aspect.

Following a description of the replicator equation and important associated facts, we introduce momentum to the discrete replicator equation and give a Lyapunov function showing that the evolutionarily stable states, when they exist for a given fitness function, are unchanged for small magnitude momentum. We show analytically that the continuous replicator dynamic with momentum converges explicitly more quickly for typical values of momentum, slows for other regions, and reverses direction in some cases. Finally, we consider exceptional examples of nonzero learning rate and momentum that break typical dynamic behavior, such as the concentric cycles for the rock–paper–scissors landscapes.

Several authors have explored variations of the ideas presented here, including recent works exploring momentum and geometry [18–20], analysis of accelerated continuous replicator equations via Lyapunov methods [21], and earlier works regarding an aspect of memory to replicator dynamics [22], adding negative momentum to game dynamics [23], gradient descent on Riemannian manifolds [24, 25], evolutionary dynamics on Riemannian manifolds [26–29], and other interactions between game theory and ML [30–32]. Our contributions are as follows: (1) introducing momentum to the discrete replicator dynamic in a way compatible with recent work in ML, (2) demonstrating that momentum accelerates convergence for the replicator dynamic, and (3) Lyapunov stability theorems for evolutionary dynamics with momentum. We hope that this manuscript encourages further exploration of common ideas among researchers in ML and evolutionary dynamics.

2. Preliminaries

We briefly review the necessary background, recommending [15] for an overview of momentum and gradient descent and [33] for an overview of the replicator equation and information theory.

2.1. Gradient descent

First we describe gradient descent in Euclidean space. Let \( x \in \mathbb{R}^n \) be a real-valued vector, \( U: \mathbb{R}^n \rightarrow \mathbb{R} \) be a potential function (or simply a function to be optimized), \( f = \nabla U \) its gradient. Then discrete gradient descent takes the following form:

\[
x'_i = x_i + \alpha f_i(x)
\]

where \( \alpha \) is the learning rate, also commonly called the step size. In what follows it will be convenient to use the notation of time-scale dynamics [34]. Let

\[
x_{t, \alpha} = \frac{x'_t - x_t}{\alpha}
\]

be the ‘time-scale’ derivative, corresponding to either the ordinary derivative (in the limit that \( \alpha \to 0 \)) or a finite difference (\( \alpha > 0 \) and fixed) as needed. Gradient descent with learning rate \( \alpha \) is simply \( x_{t, \alpha} = f(x) \).

Since we will not consider dynamics with actively changing \( \alpha \) we simply write \( x_{t, \alpha} \), though we will consider how a family of dynamics changes as \( \alpha \to 0 \), as the difference equations converge to a continuous differential equation. Finally we note that a large step size can cause the dynamics to exit the simplex. We assume throughout that this does not occur\(^1\).

2.2. Gradient descent with momentum

Momentum adds a memory of the prior gradients to future iterations. We proceed in accordance with the ML literature [15]\(^2\). Gradient descent with (Polyak) momentum [37] \( \beta \in \mathbb{R} \) is given by:

\[
z'_i = \beta z_i + f_i(x)
\]

\[
x_{t, \alpha} = z'_i
\]

where \( f \) is the gradient as before. When \( \beta = 0 \) the momentum-free gradient descent is recovered.

---

\(^1\) See [35] for a method to maintain trajectories within the simplex.

\(^2\) Momentum can also be understood as a second order approximation, called the Heavy Ball Method [36]. See [17] for a second order ODE approach to Nesterov momentum.
2.3. Replicator dynamics and gradient descent

The replicator dynamic is an evolutionary dynamic describing the action of natural selection as well as the dynamics of iterated games [38]. Its theoretical properties are extensively studied in Evolutionary Game Theory (EGT) and the equation has applications in biology, economics, and other fields. The importance of geometry in the study of the replicator equation and related dynamics, including that special cases of the replicator equation are a form of gradient descent, has been studied in EGT [39] and Information Geometry [40].

In EGT one typically restricts to discrete probability distributions that represent populations of evolving organisms or players of a strategic game, hence it is necessary to reformulate the state space of gradient descent with respect to the Euclidean geometry on the simplex is a special case of the (orthogonal) projection dynamic, described below. Gradient descent with respect to the Fisher information metric (also known as the Shahshahani metric in EGT) is called the natural gradient in information geometry and ML [7]. In the case of a symmetric and linear fitness function, the replicator equation is the gradient of the mean fitness with respect to the same geometry. The more general form of the replicator equation is not always a gradient flow³, nevertheless it has a strong convergence theorem that is closely related to this geometric structure. Our results apply to this more general case where the fitness functions need not be linear or symmetric.

The continuous version of the replicator equation is obtained by letting \( \alpha \to 0 \). The denominator \( \bar{f}(x) \) on the right-hand side is often omitted as it can be eliminated with change in time scaling without altering the continuous trajectories. This gives the following standard form of the continuous dynamic:

\[
\dot{x}_i = \frac{dx_i}{dt} = x_i \left( f_i(x) - \bar{f}(x) \right) / f(x).
\]

Subtracting the mean fitness means that the rate of change of the \( i \)th population type is proportional to its excess fitness, which is how much more or less its fitness \( f_i \) is compared to the mean. Mathematically, subtracting the mean fitness keeps the derivative in the tangent space of the simplex.

Similarly, the analog of discrete Euclidean gradient descent on the simplex is known as the (orthogonal) projection dynamic, given by

\[
x_i^\Delta = f_i / f(x).
\]

where now \( \bar{f}(x) = \frac{1}{n} \sum f_k(x) \) is the (unweighted) average fitness. The continuous form is given by

\[
\dot{x}_i = f_i(x) - \frac{1}{n} \sum f_k(x).
\]

Again it is a gradient flow whenever the fitness function is itself a Euclidean gradient. When the fitness function is a linear symmetric matrix \( f(x) = Ax \), the replicator equation models the repeated play of games where \( A \) is the payoff matrix for the game. If \( A \) is symmetric, the replicator equation models \( n \)-alleles of a gene locus. In our computational examples we will use matrices of the form

\[
\begin{pmatrix}
0 & a & b \\
b & 0 & a \\
a & b & 0
\end{pmatrix}.
\]

³ When the fitness is given by a symmetric matrix \( A = A^T \) such that \( f_i(x) = Ax \) then the replicator dynamic is the gradient of the half-mean fitness \( U(x) = (1/2) x \cdot f(x) = \bar{f}(x) \) for the Fisher information geometry. [39, 40].
When \( a = -b = 1 \) this matrix is known as a rock–paper–scissors game and the (continuous) replicator dynamic cycles about the interior point of the simplex. Otherwise the trajectories converge to the center of the simplex or diverge to the boundary depending on the relative values and signs of \( a \) and \( b \). When \( a = b > 0 \) this matrix is a three dimensional version of the hawk-dove game.

In the case that the mean fitness \( f(x) \) is zero (e.g. for zero-sum games such as the rock–paper–scissors game when \( a = -b \)), it is common to either remove the denominator of the dynamics or to apply the softmax function [41] to the fitness function. Either allows the discrete dynamics to be well-defined. We choose to drop the denominator, so in the computational examples below we typically have that \( F_i(x) = x_i f_i(x) - f(x) \) for the replicator dynamic.

### 2.4. Lyapunov functions and evolutionarily stable states

As analytically solving non-linear differential or difference equations explicitly is often extremely difficult, a common method to demonstrate stability of a dynamical system and convergence to a rest point is to find a Lyapunov function [42, 43], often an energy-like or entropy-like quantity that is positive definite and decreasing along trajectories of the dynamic toward an equilibrium point [34]. The existence of a Lyapunov function is often sufficient to demonstrate local or asymptotic stability of the dynamic and bounds on convergence rate can often be determined. We describe how to obtain a Lyapunov function for the replicator equation, noting a generalization to a much larger class of evolutionary dynamics [28] for the interested reader.

Evolutionary dynamics are often studied in terms of evolutionarily stable states (ESS) [44], somewhat analogous to extrema of potential functions or stationary distributions. An ESS for a fitness function \( f \) is a state \( \hat{x} \) such that \( \hat{x} \cdot f(x) > x \cdot f(x) \) for all \( x \) in a neighborhood of \( \hat{x} \). It can also be defined in terms of robustness to invasion by mutant subpopulations, similar in concept to a Nash equilibrium, a mixture of strategies such that no player has an incentive to unilaterally deviate. In this sense it is a stable population state for the fitness function.

When a fitness function has an ESS, it is well-known in EGT that the KL-divergence is a (local) Lyapunov function of the dynamic. It can then be seen that interior trajectories of the replicator dynamic converge to the ESS, and for the standard replicator equation there can only be one such ESS interior to the simplex, meaning that the convergence is global [45]. We restate this result below and then generalize to the dynamics with momentum. An information-theoretic interpretation of theorem 1 is that the population is learning information about the environment and encoding that information in the population structure (the distribution over different types).

**Theorem 1.** Let \( \hat{x} \) be an ESS for a replicator dynamic. Then

\[
D(x) := D_{KL}(\hat{x}, x) = -\sum_i \hat{x}_i \log \hat{x}_i - \hat{x}_i \log \hat{x}_i
\]

is a local Lyapunov function for the discrete and continuous replicator dynamic.

Theorem 1 is often stated in alternative forms. The discrete time version with geometric considerations appears in [28] and is predated by a number of variations, going back at least to [46] and [47] in forms recognizable as information-theoretic (cross-entropy), and ultimately to [12]. Similarly, the Euclidean distance \( D(x) = \frac{1}{2} ||\hat{x} - x||^2 \) is a Lyapunov function for the projection dynamic [35, 48], also realizable as a Bregman divergence [19]. These functions can be derived directly from the underlying geometries, Fisher and Euclidean for the replicator and projection dynamics, respectively. Moreover, given an information divergence, an associated geometry and dynamic can be derived, and an analog of theorem 1 holds [49]. The proof of theorem 1 will be a special case of the proof of theorem 2.

### 3. Evolutionary dynamics with momentum

To introduce momentum to these dynamics we proceed in accordance with the ML literature [15]. The discrete replicator Equation with fitness function \( f \) and momentum \( \beta \) is given by:

\[
\begin{align*}
\dot{z}_i &= \beta z_i + F_i(x) \\
x_i^{n+1} &= z_i^n
\end{align*}
\]

(8)

\footnote{In the continuous case, the proof is an easy exercise using differentiation and the ESS definition. Since the KL-divergence is positive-definite, one need only show that the derivative is negative where \( x \) is defined by equation (5).}
where $F_i = -x_i \frac{f_i - f(x)}{f(x)}$ for the replicator Equation and we have suppressed the step size $\alpha$ in $x^\alpha_i$. Alternatively $F$ could be a gradient $\nabla U$. Similarly, we obtain the projection dynamic with momentum by instead substituting $F_i = \frac{Lf_i(x)}{f(x)}$ where the mean is again the unweighted average fitness. When $\beta = 0$ the usual momentum-free dynamics are obtained.

Note that $\beta$ is not a physical momentum in this setting. In evolutionary game theory, the parameter $\beta$ is known as the strength of selection or inverse temperature, by analogy with statistical thermodynamics. The term momentum is well-established in the ML literature, so we will refer to $\beta$ as the momentum throughout. See [50] for another usage of a concept of (angular) momentum with the replicator equation.

Another variation, known as Nesterov momentum, differs from Polyak momentum in that the function $F$ is evaluated at a look-ahead step weighted by the momentum. For both flavors of momentum the dynamic starts at some initial population state $x_0$ and the initial value can be chosen to be the zero vector.

$$z'_i = \beta z_i + F_i(x + \beta z_i)$$

$$x^\alpha_i = z'_i. \quad (9)$$

### 3.1. Lyapunov stability and momentum

Now we show that adding small amounts of momentum with a nonzero learning rate typically does not alter the ESS of these discrete dynamics. (We’ll also see later that the ESS of the continuous dynamics are not affected for typical values of momentum.) We state theorem 2 as a generalization of theorem 1 for small values of momentum $\beta$.

**Theorem 2.** For small momentum $\beta$, if $\bar{x}$ is an ESS for the fitness $f$, the KL divergence is a local Lyapunov function for the replicator dynamic with momentum and the Euclidean distance $D(x) = \frac{1}{2}||x - \bar{x}||^2$ is a local Lyapunov function for the projection dynamic with momentum. If the fitness function is continuous, this also holds for Nesterov momentum.

The proof of the theorem is straightforward and given in the appendix. We note that it holds for any learning rate $\alpha$, but the permissible values of $\beta$ may vary with both $\alpha$ and the fitness function. Below, we develop a similar result for the continuous dynamic (the limit that $\alpha \to 0$) which works for any $\beta \neq 1$. In general there cannot be a variant of theorem 2 for Polyak momentum, arbitrary learning rate $\alpha$, and arbitrary momentum $\beta$: the hypothesis that at least one of $\alpha$ and $\beta$ is small is necessary (see examples in figures 1 and 2).

### 4. Effect of momentum on time to converge

While it is good to know that the addition of some memory to the replicator Equation does not alter the stable states, a more interesting effect is the acceleration of convergence. This is why momentum is of interest in ML. For evolutionary processes, this acceleration suggests one reason why epigenetic mechanisms may have evolved and persisted.

We can again use Lyapunov methods to see that the time to converge decreases with momentum. Empirically we find that the convergence takes fewer steps by a factor of approximately $(1 - \beta)$ of the momentum-free case, which we justify analytically. First we note that this factor makes sense intuitively given the iterative nature of momentum in equation (9) since

$$\frac{1}{1 - \beta} = 1 + \beta + \beta^2 + \cdots.$$

Now we hold momentum constant and allow the learning rate to converge to zero, yielding a continuous replicator dynamic with momentum associated to the discrete replicator dynamic. From equation (9), as the discrete dynamic converges, we set $z'_i = z_i$ to find that $z_i = \frac{F_i}{1 - \beta}$. Letting $\alpha \to 0$ we obtain, after substituting in $z_i$ to the second equation

$$\frac{dx_i}{dt} = \frac{1}{1 - \beta} x_i \left( f_i(x) - \bar{f}(x) \right) \quad (10)$$

where a factor of $\bar{f}(x)$ has been removed for brevity, corresponding to a scaling of time $^5$. Setting $\beta = 0$ recovers the standard definition of the replicator equation as in the discrete case. The leading factor of

---

$^5$ Equation (10) can also be obtained by scaling the Fisher information metric $\frac{1}{\bar{f}_i} \delta_{ij} \to \frac{1 - \beta}{\bar{f}_i} \delta_{ij}$. A similar form where the resultant coefficient on the dynamic is simply $\beta$ appears in [49], where $\beta$ is known as the intensity of selection or inverse temperature.
Figure 1. Examples of altered convergence time for Polyak (top 2) and Nesterov (bottom 2) momentum. In all cases we use a fitness with $a = 2$ and $b = 1$ and $\alpha = 1/200$. As $\beta$ increases, the dynamics typically converge faster, and the trajectories are not identical since $\alpha > 0$. However, for Polyak momentum (top), as the value of $\beta$ becomes closer to 1, the Lyapunov quantity eventually fails to be monotonic along the entirety of the trajectory (it is at best local). Contrast with the Nesterov momentum trajectories (bottom) for the same parameters, which in this case are all monotonically decreasing.

Figure 2. For large values of momentum the dynamic may fail to converge as in the momentum free case if $\alpha$ is not sufficiently small. For all trajectories here $\alpha = 0.01$, $a = 2$, and $b = -1$. Lowering $\alpha$ to 0.001 restores convergence of the red $\beta = 0.9$ curve.

$1/(1 - \beta)$ can similarly be eliminated by change of time scale in the continuous case without altering the trajectories of the continuous dynamic, however we retain it to argue explicitly that the convergence time decreases as $\beta$ increases within $(0, 1)$, increasing relative to the base case $\beta = 0$ for $\beta \in (0, 1)$. Similarly, the convergence slows down for $\beta \in (-\infty, 0)$. Note that traditionally in EGT, scaling the continuous replicator Equation this way would not be considered particularly interesting since the trajectories (and stable points) do not change, however the increased rate of convergence is of paramount importance in ML (and perhaps to actual evolving populations).
Figure 3. Graphical depiction of theorem 3 in terms of the properties of the dynamic coefficient $\frac{1}{1-\beta}$. As $\beta$ varies the convergence and trajectory velocity changes in accordance with the coefficient $1/(1-\beta)$. The trajectory velocity is increasing with $\beta$ on $(-\infty, 1)$ and $(1, \infty)$, the orientation is reversed on $(1, \infty)$, and the velocity is faster than the momentum free case ($\beta = 0$) for $(0, 1)$ and $(1, 2)$.

Let $V_\beta = D(\hat{x}, x_\beta)$ be the KL-divergence with $\hat{x}$ an ESS and $x_\beta$ denoting that the trajectories evolve in time according to the replicator dynamic with momentum $\beta$ as in equation (10). An easy calculation shows that

$$\frac{dV_\beta}{dt} = \frac{1}{1-\beta} \frac{dV_0}{dt}$$

where we have used equation (10) and the chain rule, i.e. we effectively scale the derivative of the Lyapunov quantity by the leading factor. From this simple fact follows theorem 3, which shows that the dynamic convergence and trajectory velocity is altered accordingly to $1/(1-\beta)$.

**Theorem 3.** Let $V_\beta$ be defined as above. Then we have that:

1. For $-\infty < \beta < 1$, the ESS of the dynamic with (Polyak) momentum are the same as for the momentum free case; equivalently the KL-divergence is still a Lyapunov function for $\beta < 1$.
2. For $1 < \beta < \infty$, the directionality of the trajectory is reversed (so any ESS for $\beta < 1$ is no longer an ESS)
3. The speed of the convergence is increasing on the intervals $(-\infty, 1)$, and decreasing on $(1, \infty)$ (with direction reversed in the latter case)
4. In particular, the speed of convergence is faster than the momentum free dynamic for $0 < \beta < 1$ and the ESS are unchanged.

For the continuous dynamic, in the case that the dynamic converges to an ESS, equation 11 also shows that it takes $\approx (1-\beta)$ as much time for the dynamic to be within $\varepsilon$ of the ESS when compared to the momentum-free dynamic, as measured by the KL-divergence, starting from the same initial point. Thus the trajectories converge more quickly as $\beta$ ranges from 0 to 1 and the convergence slows for $\beta < 0$ (see figure 3).

Returning to the discrete dynamic, for continuous fitness functions and smaller $\alpha$, we also roughly have that time-scale derivatives of the KL-divergence scale by $1/(1-\beta)$, though we cannot as easily compare directly along trajectories and the associated trajectories will not trace out the same curves (as seen in the examples above), so there is not a direct analog of equation (11). Nevertheless we may reasonably predict that it takes approximately $(1-\beta)$ as many steps as the momentum free case to be within $\varepsilon$ of the ESS compared to the dynamic without momentum ($\beta = 0$), demonstrated in computational examples. This approximation improves as the learning rate $\alpha \to 0$. Computationally we also find that the dynamic with Nesterov momentum exhibits a similar behavior (figure 4). While the argument of theorem 3 does not directly apply to Nesterov momentum, for small $\beta$ and continuous fitness function, a continuity argument suggests that the same approximation holds.

For completeness, we note that theorem 3 also holds for the projection dynamic in an analogous manner and should similarly apply to other Riemannian geometries as described in [28].
Figure 4. Left: convergence speed up for Polyak momentum: convergence time for small learning rates are well approximated by 
\((1 - \beta)\) times the momentum free convergence time \((\beta = 0)\) of iterations for small learning rates. Right: the dynamic with Nesterov momentum is also fairly well approximated by a constant factor times the momentum free convergence time, but is clearly not scaled by the same factor. The fitness function is defined by \(a = 1 = b\).

Figure 5. For the rock–paper–scissors fitness \((a = 1, b = -1)\), momentum \(\beta = 0.65\), and learning rate \(\alpha = 1/200\), the replicator equation cycles indefinitely with constant KL-divergence based on the initial point. Adding momentum with a non-zero learning rate can cause the cycling to break into either convergence or divergence. In this case Nesterov momentum causes the dynamic to converge while Polyak momentum causes the dynamic to slowly diverge to the boundary.

Theorem 4. Theorem 3 also holds for the projection dynamic with the Lyapunov function \(\frac{1}{2}\|x - \hat{x}\|^2\).

5. Momentum can break cycling into convergence or divergence

For the rock–paper–scissors fitness with \(a = -b \neq 0\), the replicator equation is not a gradient. Since the mean fitness is zero (the game is zero-sum as the payoff matrix is skew-symmetric), the gradient flow is degenerate (the dynamic is motionless). However the replicator equation with this fitness function is not degenerate and the phase portrait consists of concentric cycles of constant KL-divergence from the interior center of the simplex. The cycles are non-absorbing and the KL-divergence is an integral of motion. In the continuous case \((\alpha \to 0)\), momentum alters the time to cycle around the central point, and possibly also reverses the directionality of the cycles, in accordance with the inequalities in theorem 3 (see figure 5).

In contrast, for non-zero learning rate \(\alpha\), we find computationally that the momentum can cause the trajectories to converge inward or diverge outward. For Polyak momentum, the memory of the prior iterations causes the divergence, preventing the dynamic from turning sufficiently. For Nesterov momentum, it is the look-ahead aspect of the momentum that induces the convergence by causing the dynamic to turn more quickly.
6. Discussion

We’ve shown that momentum can accelerate evolutionary dynamics in the probability simplex just as it does for gradient descent in the ML literature. Lyapunov methods, commonly used to analyze dynamical systems but not yet as commonly applied in ML, allow us to show analytically and explicitly that momentum decreases the time to converge for values of momentum typically used in ML, and otherwise cause divergence or slowdown of trajectories for momentum outside of the interval $[0, 1)$. Crucially, we have shown that learning rate and momentum interact so that preservation of the convergence properties of the dynamic are guaranteed only for small $\beta$ or $\alpha$ despite the frequently realized speed up in convergence for larger values of momentum. These assumptions are needed for convergence results for arbitrary fitness landscapes and we note that some specific examples given above (in figure 3) allow for larger values of $\beta$. However, there must be limitations on the ability to speed up convergence using momentum. If we were free to choose any value of $\beta$, we could choose it arbitrarily close to one and achieve arbitrary increases in the rate of convergence.

Interpreting the results, we have shown that the convergence of evolutionary dynamics can be accelerated by a mechanism of memory that can be viewed as a simple model of intergenerational information exchange such as epigentics. This may also apply to immunity or cultural exchanges of information and explain the origin and persistence of extra-genetic information exchange in lineages and populations.

Data availability statement

No new data were created or analysed in this study.

Acknowledgments

The authors thank Jean Ortega, Karol Langner, and the anonymous reviewers for comments on manuscript drafts.

Conflict of interest

The authors declare no funding sources or competing interests.

Code

The code to generate the trajectories and plots in this manuscript is available as a Python library pyed at https://github.com/marcharper/pyed. Ternary plots were generated with the python-ternary library [51].

Appendix. Proof of equation (11)

The proof in the momentum free case is well-known and we extend it. Taking the derivative of the KL-divergence, using the definition of the continuous replicator equation, gives the following:

$$\frac{d}{dt}D(\hat{x}||x) = -\sum_i \frac{\hat{x}_i}{x_i} \frac{dx_i}{dt} = -\sum_i \hat{x}_i (f_i - \bar{f}) = \hat{x} \cdot f - \hat{x} \cdot \bar{f}.$$ 

This quantity is less than zero if $\hat{x}$ is an ESS (which proves theorem 1 in the continuous case). If we instead use the replicator equation with momentum (equation (10)), a factor of $\frac{1}{1-\beta}$ is present on the right hand side of the equation above, proving equation (11), as the constant factor does not affect the sign or the convergence to zero (when the factor is positive).

A.1. Proof of theorem 2

Proof. Since the KL-divergence is positive and zero only at $\hat{x}$, essentially one just needs to show that the quantity

$$D^{\Delta}(x) = \frac{D(\hat{x}||x') - D(\hat{x}||x)}{\alpha}$$

is less than zero when $\hat{x}$ is an ESS (for fixed $\alpha$) to establish it as a discrete Lyapunov function.

A straightforward algebraic calculation and Jensen’s inequality shows that this quantity is bounded by

$$-\log \left( \sum_i \frac{\hat{x}_i}{x_i} \frac{x'_i}{x_i} \right) = -\log \left( 1 + \alpha \beta \sum_i \frac{\hat{x}_i}{x_i} \frac{z_i}{x_i} + \alpha \left( \frac{\hat{x} \cdot f}{x \cdot f} - 1 \right) \right)$$
which is less than 0 for sufficiently small $\beta$ and the inequality defining an ESS.

When $\alpha \to 0$ it is easy to directly use ordinary differentiation to prove the continuous version of theorem 2. Proof for the projection dynamic using the Euclidean distance is analogous and omitted. In the case of Nesterov momentum, continuity of the fitness function and small $\beta$ reduces to the Polyak momentum case.

**ORCID ID**

Marc Harper 🏫 https://orcid.org/0000-0003-0918-219X

**References**

[1] Goldberg D E and Henry Holland J H 1988 Genetic algorithms and machine learning Mach. Learn. 3 95–99
[2] Moriatry D E, Schultz A C and Grefenstette J J 1999 Evolutionary algorithms for reinforcement learning J. Artif. Intell. Res. 11 241–76
[3] Salimans T, Ho J, Chen Xi, Sidor S and Sutskever I 2017 Evolution strategies as a scalable alternative to reinforcement learning (arXiv:1703.03864)
[4] Elsken T, Hendrik Metzen J and Hutter F 2019 Neural architecture search: a survey J. Mach. Learn. Res. 20 1–21
[5] Neal R, Liang C, So D B, and Le Q V 2020 AutoML-zero: Evolving machine learning algorithms from scratch (arXiv:2003.03384)
[6] Fernando C, Sygnowski J, Osindero S, Wang J, Schaul T, Teplyashin D, Sprechmann P, Pritzel A and Rusu A 2018 Meta-learning by the baldwin effect Proc. Genetic and Evolutionary Computation Conf. Companion pp 1313–20
[7] Amari S-I 1998 Natural gradient works efficiently in learning Neural Comput. 10 251–76
[8] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization (arXiv:1412.6980)
[9] Murtagroyd C, Patchev A V, Wu Y, Miche V, Bockmühl Y, Fischer D, Holboer E, Wotjat C T, Almeida O F X and Spengler D 2009 Dynamic dna methylation programs persistent adverse effects of early-life stress Nat. Neurosci. 12 1559
[10] Pagel M 2009 Human language as a culturally transmitted replicator Nat. Rev. Genet. 10 405
[11] Schlesinger M I 1990 Heat shock proteins J. Biol. Chem. 265 12111–4
[12] Taylor P D and Jonker L B 1978 Evolutionary stable strategies and game dynamics Math. Biosc. 40 145–56
[13] Qian N 1999 On the momentum term in gradient descent learning algorithms Neural Netw. 12 145–51
[14] Sato Y, Flammarion N and Fazel M 2019 Escaping from saddle points on riemannian manifolds (arXiv:1807.04740)
[15] Gidel G, Askari Hemmat R, Pezeshki M, Huang G, Lepriol R, Lacoste-Julien S and Mitliagkas I 2018 Negative momentum for improved game dynamics (arXiv:1703.03864)
[16] Goh G 2017 Why momentum really works Distill 2 e6
[17] Nesterov Y 1983 A method for solving a convex programming problem with convergence rate $o(1/k^2)$ Sov. Math. Dokl 269 543
[18] Su W, Boyd S and Candès E J 2016 A differential equation for modeling nesterov’s accelerated gradient method: theory and insights J. Mach. Learn. Res. 17 5312–54
[19] Zhang H and Sra S 2018 Towards riemannian accelerated gradient methods (arXiv:1806.02812)
[20] Ahn K and Sra S 2020 From nesterov’s estimate sequence to riemannian acceleration (arXiv:2001.08876)
[21] Defazio A 2019 On the curved geometry of accelerated optimization Advances in Neural Information Processing Systems pp 1764–73
[22] Krichene W, Bayen A and Bartlett P L 2016 Adaptive averaging in accelerated descent dynamics Advances in Neural Information Processing Systems pp 29
[23] Sato Y, Akiyama E and Crutchfield J P 2003 Stability and diversity in collective adaptation Physica D 210 21–57
[24] Gidel G, Askar Hemmat R, Pezeshki M, Huang G, Lepriol R, Lacoste-Julien S and Miltagaskas I 2018 Negative momentum for improved game dynamics (arXiv:1807.04740)
[25] Tripuraneni N, Flammarion N, Bach F and Jordan M I 2018 Averaging stochastic gradient descent on riemannian manifolds (arXiv:1802.09128)
[26] Sun Y, Flammarion N and Fazel M 2019 Escaping from saddle points on riemannian manifolds Advances in Neural Information Processing Systems pp 7276–86
[27] Wang Y and Li W 2020 Information newton’s notion: second-order optimization method in probability space (arXiv:2001.04341)
[28] Harper M and Fryer D 2015 Lyapunov functions for time-scale dynamics on riemannian geometries of the simplex Dyn. Games Appl. 5 318–33
[29] Schuurmans D and Zinkevich M A 2016 Deep learning games Advances in Neural Information Processing Systems pp 1678–86
[30] Balduzzi D, Racaniere S, Martens J, Foerster J, Tuyls K and Graepel T 2018 The mechanics of n-player differentiable games (arXiv:1802.05642)
[31] Bartoszewicz Z and Piotrowska E 2011 Lyapunov functions in stability of nonlinear systems on time scales Dyn. Games Appl. 1 15–19
[32] Bartosiewicz Z and Piotrowska E 2011 Lyapunov functions in stability of nonlinear systems on time scales Dyn. Games Appl. 1 15–19
[33] Lohmar R and Sandholm W H 2008 The projection dynamic and the geometry of population games J. Econ. Theory 315–64
[34] Oshika T and Sato Y 2009 The projection dynamic and the geometry of population games J. Differ. Equ. Appl. 15 285–90
[35] 2010 Global convergence of the heavy-ball method for convex optimization (arXiv:1805.10228)
[36] 2015 Global convergence of the heavy-ball method for convex optimization (arXiv:1805.10228)
[37] 2016 Deep learning games Advances in Neural Information Processing Systems pp 1678–86
[38] 2017 On the properties of the softmax function with application in game theory and reinforcement learning (arXiv:1704.08005)
[39] 2016 A lyapunov analysis of momentum methods in optimization (arXiv:1611.02635)
[43] Wilson A 2018 *Lyapunov Arguments in Optimization* (University of California)
[44] Maynard Smith J 1982 *Evolution and the Theory of Games* (Cambridge University Press)
[45] Hofbauer J and Sigmund K 1998 *Evolutionary Games and Population Dynamics* (Cambridge University Press)
[46] Bomze I M 1991 Cross entropy minimization in uninvadable states of complex populations *J. Math. Biol.* **30** 73–87
[47] Akin E and Losert V 1984 Evolutionary dynamics of zero-sum games *J. Math. Biol.* **20** 231–58
[48] Nagurney A and Zhang D 1997 Projected dynamical systems in the formulation, stability analysis and computation of fixed-demand traffic network equilibria *Transp. Sci.* **31** 147–58
[49] Harper M 2011 Escort evolutionary game theory *Physica D* **240** 1411–5
[50] Wang Y, Chen X and Wang Z 2017 Testability of evolutionary game dynamics based on experimental economics data *Physica A* **486** 455–64
[51] Harper M et al 2020 python-ternary: ternary plots in python *Zenodo* (https://arxiv.org/abs/10.5281/zenodo.594435)