CONVERGENCE RATE OF RESTARTED ACCELERATED GRADIENT*

CAN KIZILKALE†, SHIVKUMAR CHANDRASEKARAN‡, AND MING GU‡

Abstract. The accelerated gradient algorithm is known to have non-monotonic, periodic convergence behavior in the high momentum regime. If important function parameters like the condition number are known, the momentum can be adjusted to get linear convergence. Unfortunately these parameters are usually not accessible, so instead heuristics are used for deciding when to restart. One of the most intuitive and well known heuristics is to look at the inner product of the momentum and gradient vector and restart when this inner product is positive. In this paper we start by proving that the convergence rate of this adaptive restarting heuristic is linear for convex functions which may not be strongly convex. Next we introduce a new restarting criteria that we call "cone based restart", and prove linear convergence under the same conditions. Finally we extend the restart heuristic for non-smooth convex functions.

Key words. Accelerated gradient, restart, convex optimization, strong convexity.

AMS subject classifications. 68Q25, 68R10, 68U05

1. Introduction.

Nesterov's accelerated gradient algorithm [2] is well-known for achieving fast convergence despite not being more complex than the classical gradient descent algorithm. Although the algorithm was introduced more than three decades ago, it became very popular in the late 2000's due to its benefits in solving large problems in sparse signal recovery, machine learning, composite function optimization, etc., where higher order methods become infeasible.

The idea behind accelerated gradient scheme is the accumulation of momentum. At each step instead of just taking into account the gradient we also take into account the momentum vector which is essentially a weighted sum of all the previous steps. The momentum vector contains some second order information about the objective function which leads to accelerated convergence when used correctly.

A notable problem with the accelerated gradient algorithm (and momentum based methods in general) is that it exhibits non-monotonic convergence behavior. Especially when the function value seems to be decreasing the fastest, it begins to increase. This behavior seems to be periodic and lowers the convergence rate. An intuitive explanation of this behavior is that, as the momentum increases, the algorithm takes much larger steps towards the optimum point, leading to faster decrease in the function value, until the point where it overshoots. After that point the momentum vector makes the iterates move away from the optimum causing the function value to increase until the gradient of the objective function nullifies and corrects the direction.

One important observation is that when step sizes are chosen small enough the algorithm exhibits monotonic convergence until the first point of overshoot. The original algorithm lets the gradient slow down the algorithm once it overshoots. Yet we can obviously do better if we slow it down or stop it "artificially" when overshoot happens. Instead of slowing the algorithm using the gradient we restart it, which erases the history and starts the algorithm afresh using the current iterate as the

*Submitted to the editors DATE.

†Department of Computer Science, University of California Santa Barbara, Santa Barbara, CA (kizilkalecan@gmail.com)

‡ECE Department, University of California Santa Barbara, Santa Barbara, CA (shivchandrasekaran@ucsb.edu) Department of Mathematics, University of California Berkeley, Berkeley,CA (mgu@berkeley.edu)

initial point. If we know the condition number then one can exploit the periodic behavior of the non-monotonicity and employ periodic restarts at those points to achieve linear convergence [5]. When we don't have that information though it seems difficult to decide on the right periodicity and we currently cannot do better than ordinary accelerated gradient.

Some of the tests for detecting overshoot are the exact non-monotonicity test [1], and the gradient-mapping test [4], both of which seem to work well in practice.

In this paper we will focus on the gradient-mapping test based restart and prove that it exhibits linear convergence under strong convexity. To the best of our knowledge no such convergence result is known and prior analysis was restricted to quadratic functions [4].

2. The Algorithm.

We will assume the the objective function is strongly convex. There are several equivalent definitions of strong convexity. We will use the following one.

Definition 2.1. *A function $f : R^n \to R$ is strongly convex if*

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + (\mu/2)\|y - x\|_2^2, \tag{1}$$

for some constant $\mu \geq 0$.

We will also assume that the gradient of the objective function is Lipschitz.

Definition 2.2. *The gradient of f is Lipschitz if there exists a constant $L > 0$ such that*

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2. \tag{2}$$

In this paper we are interested in solving the general unconstrained convex optimization problem,

$$\min_{x \in R^n} f(x),$$

where $f : R^n \to R$ is a strongly convex, Lipschitz function.

The accelerated gradient algorithm is an instance of the general momentum based algorithms. These algorithms produce a sequence of iterates $x_k \in R^n$, for $k = 0, 1, 2, \ldots$.

Definition 2.3. *Generalized accelerated gradient update rule:*

$$y_k = x_k + \beta_k(x_k - x_{k-1}) \tag{3}$$

$$x_{k+1} = y_k - \alpha_k \nabla f(y_k), \tag{4}$$

where the term $\beta_k(x_k - x_{k-1})$ is the momentum *term at each step.*

It is well known that accelerated gradient has a guaranteed convergence rate of $O(k^{-2})$. However, for strongly convex functions, if the condition number μ and Lipschitz constant L are known, it can be improved to linear convergence, $O(c^{-k})$ [3]. Unfortunately both are unknown in many problems. Moreover, it is frequently impractical to estimate μ.

To make the analysis shorter, we are going to investigate a simpler update rule given as follows.

Definition 2.4. *Simpler accelerated gradient update rule:*

$$x_{k+1} = x_k + \beta_k(x_k - x_{k-1}) - \alpha_k \nabla f(x_k).$$

The most notable difference in this version is that we are using $\nabla f(x_k)$ instead of $\nabla f(y_k)$. Under a sufficient smoothness condition it is straightforward to extend our analysis to the original case (Definition 2.3) if so desired. From now on we will refer to

$$x_{k+1} - x_k = \beta_k(x_k - x_{k-1}) - \alpha_k \nabla f(x_k),$$

as the **momentum** at step $k + 1$.

The gradient-mapping restart test was proposed in [4]. An ascent direction has a positive projection on the gradient.

Definition 2.5. *Gradient-mapping restart condition:*

$$\nabla f(x_k)^T (x_k - x_{k-1}) > 0.$$

The algorithm, which we will denote as MAGR, is shown in Algorithm 1.

Algorithm 1 Momentum accelerated gradient algorithm with gradient-mapping restart

```
Choose x_{-1} ∈ R^n
x_0 = x_{-1}
for k ≥ 0 do
  z_{k+1} = β_k(x_k − x_{k−1}) − α_k ∇f(x_k)
  if ∇f(x_k + z_{k+1})^T z_{k+1} > 0 then
    x_{k+1} = x_k − α_k ∇f(x_k)
  else
    x_{k+1} = x_k + z_{k+1}
  end if
end for
```

In Nesterov's original accelerated algorithm [2], the β_k's were chosen such that $\beta_{k+1} = \theta_k(1 - \theta_k)/(\theta_k^2 + \theta_{k+1})$, where θ_{k+1} solves $\theta_{k+1}^2 = (1 - \theta_k)/(\theta_k^2 + \theta_{k+1})$. In the next section we will do the convergence analysis for constant β_k rather than Nesterov's choice.

3. Convergence rate of MAGR.

Assuming that no restart was initiated,

$$(x_{k+1} - x_k)^T \nabla f(x_{k+1}) \le 0.$$

Then from equation (1) we have that,

$$f(x_k) \ge f(x_{k+1}) + \nabla f(x_{k+1})^T (x_k - x_{k+1}) + (\mu/2)\|x_{k+1} - x_k\|_2^2,$$

which implies that,

$$f(x_k) - f(x_{k+1}) \ge (\mu/2)\|x_{k+1} - x_k\|_2^2. \tag{5}$$

Therefore as long as there is no restart we do have monotonic decrease in the objective.

Strong convexity can be also used to bound the gradients at each step.

$$f(x^*) - f(x) - \nabla f(x)^T (x^* - x) \ge (\mu/2)\|x - x^*\|^2,$$

where x^* denotes the minimum of f, leading to,

$$
\begin{aligned}
f(x) - f(x^*) &\leq \nabla f(x)^T (x - x^*) - (\mu/2)\|x - x^*\|^2 \\
&\leq \|\nabla f(x)\| \|x - x^*\| - (\mu/2)\|x - x^*\|^2 \\
&= \frac{\|\nabla f(x)\|^2}{2\mu} - \left(\|x - x^*\| \sqrt{\frac{\mu}{2}} - \frac{\|\nabla f(x)\|}{\sqrt{2\mu}} \right)^2 \\
&\leq \frac{\|\nabla f(x)\|^2}{2\mu}. \qquad (6)
\end{aligned}
$$

Next observe that when there is no restart

$$(x_k - x_{k-1})^T \nabla f(x_k) \leq 0,$$

then

$$\|\beta_k(x_k - x_{k-1}) - \alpha_k \nabla f(x_k)\|_2^2 \geq \|\beta_k(x_k - x_{k-1})\|^2 + \alpha_k^2 \|\nabla f(x_k)\|^2, \qquad (7)$$

where the left hand side is the momentum at the *next* step $k+1$: $\|x_{k+1} - x_k\|^2$, if there is no restart in that step either.

Now let k_s denote the **first** iteration where we restart:

$$
\begin{aligned}
(x_{k_s} - x_{k_s-1})^T \nabla f(x_{k_s}) &\leq 0 \\
z_{k_s+1}^T \nabla f(x_{k_s} + z_{k_s+1}) &> 0.
\end{aligned}
$$

Assume that

$$c(f(x_0) - f(x^*)) = f(x_{k_s}) - f(x^*).$$

To show linear convergence it is sufficient to establish that c has an upper bound strictly smaller then 1.

In the rest of the the analysis, for the sake of simplicity, we will fix $\alpha_k = \alpha$ and $\beta_k = \beta$.

LEMMA 3.1. *For fixed α and β and $k \leq k_s$,*

$$\|x_k - x_{k-1}\| \geq \alpha \sqrt{2\,\mu\,(f(x_{k_s}) - f(x^*)) \sum_{i=0}^{k-1} \beta^{2i}}.$$

Proof. When there is no restart we have

$$\gamma_k \equiv \|x_k - x_{k-1}\| = \|\beta_{k-1}(x_{k-1} - x_{k-2}) - \alpha_{k-1} \nabla f(x_{k-1})\|.$$

From (6) we know that at each step $k \leq k_s$,

$$\|\nabla f(x_k)\| \geq \sqrt{2\mu(f(x_{k_s}) - f(x^*))}.$$

Combining this with (7), we get

$$\gamma_k^2 \geq \beta^2 \gamma_{k-1}^2 + 2\alpha^2 \mu (f(x_{k_s}) - f(x^*)),$$

which yields the desired bound when combined with the fact that

$$\gamma_1 = \alpha \|\nabla f(x_0)\|. \qquad (8)$$

□

LEMMA 3.2. *Let k_s be the first restarting step. Then,*

$$f(x_0) - f(x^*) \geq (f(x_{k_s}) - f(x^*)) \left(1 + \mu^2\alpha^2 \sum_{k=0}^{k_s-1} \sum_{i=0}^{k} \beta^{2i}\right).$$

Proof. From (5), for $k < k_s$, and the fact that

$$f(x_k) - f(x^*) > f(x_{k_s}) - f(x^*),$$

we have,

$$f(x_k) - f(x_{k+1}) \geq \frac{\mu}{2} 2\mu\alpha^2 (f(x_{k_s}) - f(x^*)) \sum_{i=0}^{k} \beta^{2i}.$$

Therefore

$$f(x_0) - f(x_{k_s}) \geq \mu^2\alpha^2 (f(x_{k_s}) - f(x^*)) \sum_{k=0}^{k_s-1} \sum_{i=0}^{k} \beta^{2i},$$

which yields the desired bound. ∎

LEMMA 3.3. *If $0 < \beta < 1$*

$$k_s \leq \frac{1}{2\ln\beta} \ln\left(1 - \frac{1-\beta^2}{\mu^2\alpha^2}\right)_+ - 1.$$

Proof. From inequalities (6), (7), and (8), for fixed α and β we have:

$$\begin{aligned} \|x_{k+1} - x_k\|^2 &= \|\beta(x_k - x_{k-1}) - \alpha\nabla f(x_k)\|_2^2 \\ &\geq \|\beta(x_k - x_{k-1})\|^2 + 2\mu\alpha^2(f(x_k) - f(x^*)) \\ &\geq \beta^{2k}\|x_1 - x_0\|^2 \\ &\geq 2\mu\alpha^2\beta^{2k}\left(f(x_0) - f(x^*)\right). \end{aligned} \tag{9}$$

Strong convexity,

$$f(x_k) - f(x_{k+1}) \geq \nabla f(x_{k+1})^T (x_k - x_{k+1}) + \frac{\mu}{2}\|x_{k+1} - x_k\|^2,$$

and no restart, $\nabla f(x_{k+1})^T (x_k - x_{k+1}) \geq 0$, implies that:

$$f(x_k) - f(x_{k+1}) \geq \frac{\mu}{2}\|x_{k+1} - x_k\|^2.$$

Substituting (9) and summing over k we get,

$$f(x_0) - f(x^*) \geq f(x_0) - f(x_{k_s}) \geq \mu^2\alpha^2 \sum_{k=0}^{k_s} \beta^{2k}(f(x_0) - f(x^*))$$

Hence,

$$1 \geq \mu^2\alpha^2 \frac{1 - \beta^{2(k_s+1)}}{1 - \beta^2},$$

which yields, when $0 < \beta < 1$,

$$k_s \leq \frac{1}{2\ln\beta} \ln\left(1 - \frac{1-\beta^2}{\mu^2\alpha^2}\right)_+ - 1.$$ ∎

Note that this upper bound on k_s is probably not sharp but it will suffice for our purposes.

LEMMA 3.4. *If $\alpha < 1/L$ then $k_s \geq 2$. Also, for any $t \geq 2$, there exists an $\alpha > 0$, such that $k_s \geq t$.*

Proof. Since ∇f is Lipschitz continuous

$$\|\nabla f(x) - \nabla f\left(x - \alpha \nabla f(x)\right)\| \leq L\alpha \|\nabla f(x)\|.$$

If $0 < \alpha < L^{-1}$ then

$$\nabla f(x)^T \nabla f(x - \alpha \nabla f(x)) \geq \nabla f(x)^T \left(\nabla f(x) - L\alpha \nabla f(x)\right) \geq 0.$$

Therefore $k_s \geq 2$ since the initial momentum is zero.

A similar, but more tedious, argument, shows that for all $t \geq 2$ there exists a small enough $\alpha > 0$ such that $k_s \geq t$. The basic idea is that for sufficiently small α the initial momentum can be kept as small as desired. Then the Lipschitz continuity is used as above to show that the restart condition will not be satisfied. ◻

Let k_j denote the number of iterations between the jth and $j-1$th restarts. Based on Lemmas 3.3 and 3.4, once $0 < \alpha < L^{-1}$ is fixed, we can choose $0 < \beta < 1$, such that there exist constants p and q which guarantee that

$$2 \leq p \leq k_j \leq q < \infty.$$

LEMMA 3.5. *Let r be the total number of iterations. Then*

$$f(x_r) - f(x^*) \leq (f(x_0) - f(x^*)) \left[\frac{1}{1 + \alpha^2\mu^2 \sum_{k=0}^{p-1} \sum_{i=0}^{k} \beta^{2i}}\right]^{\frac{r}{q}}.$$

Proof. Let $\hat{x}_j$ denote the point right at the beginning of the jth restart where $\hat{x}_0 = x_0$. From lemma (3.2) and $k_j \geq p$, right at the beginning of the jth restart we have,

$$f(\hat{x}_{j-1}) - f(x^*) \geq (f(\hat{x}_j) - f(x^*)) \left(1 + \mu^2\alpha^2 \sum_{k=0}^{p-1} \sum_{i=0}^{k} \beta^{2i}\right).$$

If there are a total of N restarts until iteration r this inequality leads to,

$$(f(x_r) - f(x^*)) \leq (f(\hat{x}_0) - f(x^*)) \left[\frac{1}{1 + \alpha^2\mu^2 \sum_{k=0}^{p-1} \sum_{i=0}^{k} \beta^{2i}}\right]^{N}.$$

From $k_j \leq q$ we have $N \geq \frac{r}{q}$ combining with $\hat{x}_0 = x_0$ the result follows. ◻

Now we have all the ingredients we need to state the main result of this paper.

THEOREM 3.6. *Convergence rate of MAGR is linear.*

Proof. The lower and upper bounds on p and q from Lemmas 3.4 and 3.3 combined with the result in Lemma 3.5 yields

$$(f(x_k) - f(x^*)) \leq (f(x_0) - f(x^*)) \left[\frac{1}{1 + \alpha\mu^2(\beta^2 + 1)}\right]^{\frac{k}{\frac{1}{2\ln\beta}\ln\left(1 - \frac{1-\beta^2}{\mu^2\alpha^2}\right)_+ - 1}}$$

Let

$$0 < \tau = \left[\frac{1}{1 + \alpha\mu^2(\beta^2 + 1)}\right]^{\frac{1}{\frac{1}{2\ln\beta}\ln\left(1 - \frac{1-\beta^2}{\mu^2\alpha^2}\right)_+ - 1}} < 1.$$

Then we see that MAGR converges like $O(\tau^k)$ which is linear as claimed. ◻

4. Non-Strongly Convex functions. In the previous section we have assumed strong convexity for our convergence proof, in this section we are going to see that it is not necessary and we can relax this requirement while still getting linear convergence.

Inequalities (5) and (6) are the two main ingredients in the analysis of the previous section:

- $f(x_k) - f(x_{k+1}) \geq \mu/2\|x_k - x_{k+1}\|^2$,
- $f(x) - f(x^*) \leq \|\nabla f(x)\|^2/2\mu$.

The generalized versions are:

$$f(x_k) - f(x_{k+1}) \geq c_1\|x_k - x_{k+1}\|^2, \tag{10}$$

and

$$f(x) - f(x^*) \leq c_2\|\nabla f(x)\|^2. \tag{11}$$

One can see that as long as there exists finite c_1 and c_2 such that the two conditions are satisfied, the rest of the analysis will hold and the algorithm will have linear convergence rate.

4.1. An Example. A simple example of a non-strongly convex function that satisfies the two conditions is $f(x) = x^T Ax/2$ where A is a symmetric positive semi-definite matrix with at least one zero eigenvalue. Since A is not full rank it is obvious that this objective function is not strongly convex.

However at every x that is not a minimum we have

$$\nabla f(x) = Ax = \sum_i c_i v_i \neq 0,$$

for some eigenvectors v_i where $\lambda_i > 0$. For $\hat{\lambda} = \min_{\lambda_i > 0} \lambda_i$ we have

$$f(x) - f(x^*) = \frac{x^T Ax}{2} \leq \frac{x^T A^2 x}{2\hat{\lambda}} = \frac{\|\nabla f(x)\|^2}{2\hat{\lambda}}.$$

So inequality (11) is satisfied.

Next note that

$$(x_k - x_{k+1})^T A(x_k - x_{k+1}) \geq \hat{\lambda}\|x_k - x_{k+1}\|^2.,$$

and

$$x_k^T Ax_k - x_{k+1}^T Ax_{k+1} = (x_k - x_{k+1})^T A(x_k - x_{k+1}) + 2(x_k - x_{k+1})^T Ax_{k+1}.$$

From the restart condition $\nabla f(x_{k+1})^T(x_k - x_{k+1}) \geq 0$, we conclude that,

$$x_k^T Ax_k - x_{k+1}^T Ax_{k+1} \geq (x_k - x_{k+1})^T A(x_k - x_{k+1}) \geq \hat{\lambda}\|x_k - x_{k+1}\|^2,$$

which yields inequality (10).

Therefore this example is not strongly convex yet it satisfies both inequalities (10) and (11), and hence has linear convergence rate. Although this example shows that strong-convexity is not necessary, the given example is still somewhat similar to its strongly-convex counterpart since in a subspace it is strongly convex. We will see that we can relax the requirement even more.

4.2. Relaxed Criteria for Linear Convergence. For convex functions which are smooth on a compact set, there exists a constant M such that for all x we have $M \geq \|x - x^*\|$, which leads to the following lower-bound for $\|\nabla f(x)\|$:

$$\|\nabla f(x)\| M \geq \nabla f(x)^T (x - x^*) \geq f(x) - f(x^*) \implies \|\nabla f(x)\| \geq (f(x) - f(x^*))/M.$$

We will substitute strong-convexity with the following relaxed criteria. First,

$$\|\nabla f(x)\| \geq (f(x) - f(x^*))/M, \tag{12}$$

for some $M > 0$, and second,

$$f(x_k) - f(x_{k+1}) \geq m \|x_k - x_{k+1}\|^2, \tag{13}$$

for some $m > 0$ and $k \geq 2$.

Lemma 4.1. *Assuming fixed β, α and $k \leq k_s$, if the relaxed criteria (12) and (13) are satisfied then*

$$\|x_k - x_{k-1}\| \geq (f(x_{k_s}) - f(x^*)) \frac{\alpha}{M} \sqrt{\sum_{i=0}^{k-1} \beta^{2i}}.$$

Proof. The proof is similar to the one we have for Lemma 3.1. Taking

$$\gamma_k = \|x_k - x_{k-1}\|,$$

and replacing equation (6) with (12) we get.

$$\gamma_k^2 \geq \beta^2 \gamma_{k-1}^2 + \alpha^2 / M^2 (f(x_{k_s}) - f(x^*))^2.$$

The desired result follows. ∎

Lemma 4.2. *Let k_s be the first restarting step. Then,*

$$f(x_0) - f(x_{k_s}) \geq (f(x_{k_s}) - f(x^*))^2 \left(1 + m \frac{\alpha^2}{M^2} \sum_{k=0}^{k_s - 1} \sum_{i=0}^{k} \beta^{2i}\right).$$

Proof. By a similar argument to the proof of Lemma 3.2 we have,

$$f(x_k) - f(x^*) > f(x_{k_s}) - f(x^*),$$

and $f(x_k) - f(x_{k+1}) \geq m \|x_k - x_{k+1}\|^2$ combined with (4.1),

$$f(x_k) - f(x_{k+1}) \geq m \frac{\alpha^2}{M^2} (f(x_{k_s}) - f(x^*))^2 \sum_{i=0}^{k} \beta^{2i}.$$

summing this expression over k yields the desired bound. ∎

Lemma 4.3. *If $0 < \beta < 1$, and $B = \max_x (f(x) - f(x^*))$, then*

$$k_s \leq \frac{1}{2\beta} \ln \left(1 - \frac{1 - \beta^2}{\frac{m}{M^2} B}\right).$$

Proof. Following steps of the proof of Lemma 3.3:

$$\|x_k - x_{k+1}\|^2 \geq \beta^{2k}\|x_1 - x_0\|^2 \geq \beta^{2k}\|\nabla f(x_0)\|^2 \geq \frac{\beta^{2k}}{M^2}\|f(x_0) - f(x^*)\|^2.$$

Last inequality is a result of condition (12). Now using condition (13) we get

$$f(x_k) - f(x_{k+1}) \geq m\|x_k - x_{k+1}\|^2 \geq \beta^{2k}\frac{m}{M^2}(f(x_0) - f(x^*))^2.$$

Summing over k leads to

$$f(x_0) - f(x^*) \geq \frac{m}{M^2}(f(x_0) - f(x^*))^2 \sum_{k=0}^{k_s} \beta^{2k}.$$

We have assumed that the function is bounded, $B \geq f(x_0) - f(x^*)$. Replacing this in the inequality above we get

$$1 \geq B\frac{n}{M^2}\frac{1 - \beta^{2k_s+1}}{1 - \beta^2},$$

and conclude that

$$k_s \leq \frac{1}{2\ln\beta}\ln\left(1 - \frac{1 - \beta^2}{\frac{m}{M^2}B}\right).$$ ∎

Since Lemma 3.4 requires only Lipschitz continuity it carries on. Let k_j denote the number of iterations between the j-th and $(j-1)$-th restarts. Based on Lemmas 4.3 and 3.4, once again for fixed $\alpha \in (0, L^{-1})$, we can choose $\beta \in (0, 1)$, such that there exist constants p and q which guarantee that

$$2 \leq p \leq k_j \leq q < \infty.$$

LEMMA 4.4. *Let r be the total number of iterations. Then*

$$f(x_r) - f(x^*) \leq (f(x_0) - f(x^*))\left[\frac{1}{1 + \alpha^2\frac{m}{M^2}\sum_{k=0}^{p-1}\sum_{i=0}^{k}\beta^{2i}}\right]^{\frac{r}{q}}.$$

Proof. Similar to the proof of Lemma (3.5), by using Lemma (4.2) instead of Lemma (3.2), the desired result is achieved. ∎

THEOREM 4.5. *When conditions (12) and (13) are satisfied MAGR has linear convergence.*

The lower and upper bounds on p and q from Lemmas 3.4 and 4.3 combined with the result in Lemma 4.4 yields

$$(f(x_k) - f(x^*)) \leq (f(x_0) - f(x^*))\left[\frac{1}{1 + \alpha\frac{m}{M^2}(\beta^2 + 1)}\right]^{\frac{k}{\frac{1}{2\ln\beta}\ln\left(1 - \frac{1-\beta^2}{\frac{m}{M^2}B}\right)_+^{-1}}}$$

Let

$$0 < \tau = \left[\frac{1}{1 + \alpha\frac{m}{M^2}(\beta^2 + 1)}\right]^{\frac{k}{\frac{1}{2\ln\beta}\ln\left(1 - \frac{1-\beta^2}{\frac{m}{M^2}B}\right)_+^{-1}}} < 1.$$

Then we see that MAGR converges like $O(\tau^k)$ which is linear as claimed.

5. Cone based restart.

We now introduce a new gradient based restart criteria which we will call "cone based restart". As we will see in the experiments the corresponding Algorithm 2 has very similar convergence behaviour and speed to MAGR, but it has some nice properties that makes it easier to guarantee linear convergence. Moreover the coefficient c in Algorithm 2 makes it possible to tune the algorithm.

Algorithm 2 Momentum accelerated gradient algorithm with cone based restart

Choose $x_{-1} \in R^n$
Choose $c > 1/\sqrt{2}$
$x_0 = x_{-1}$
$g_r = \nabla f(x_0)$
for $k \geq 0$ **do**
  $z_{k+1} = \beta_k(x_k - x_{k-1}) - \alpha_k \nabla f(x_k)$
  if $\nabla f(x_k + z_{k+1})^T g_r < c\|\nabla f(x_k + z_{k+1})\|\|g_r\|$ **then**
    $x_{k+1} = x_k - \alpha_k \nabla f(x_k)$
    $g_r = \nabla f(x_{k+1})$
  else
    $x_{k+1} = x_k + z_{k+1}$
  end if
end for

The restart condition

$$\nabla f(x_k + z_{k+1})^T g_r < c\,\|\nabla f(x_k + z_{k+1})\|\|g_r\|,$$

guarantees that all of the gradients until the next restart lie in the cone centered around the initial gradient g_r. Observe that when the assumptions in Lemma 4.5 are satisfied the conclusions hold true for cone based restart too. For strongly convex objective functions it is easy to see that they indeed are satisfied since the selection $c > \sin(\pi/4)$ guarantees that for all k we have $(x_k - x_{k+1})^T \nabla f(x_{k+1}) > 0$ when there is no restart. For the same selection of c we can further observe that there is a $\mu > 0$ such that

$$\nabla f(x_i)^T \nabla f(x_0) \geq \mu \|\nabla f(x_i)\|\|\nabla f(x_0)\|,$$

for $i < k_s$.

For the non-strongly convex example we have in (16), assuming compactness, let $\|x - y\| \leq M$. Then from convexity, when there is no restart at step k, we have

$$f(x_k) - f(x_{k+1}) \geq (x_k - x_{k+1})^T \nabla f(x_{k+1}) > \alpha\mu \sum_{i=0}^{k} \left(\sum_{l=0}^{i-1} \beta^l \right) \|\nabla f(x_i)\|\|\nabla f(x_{k+1})\|.$$

Since the domain is assumed to be compact we have

$$\|\nabla f(x_i)\| \geq (f(x_i) - f(x^*))/M \geq (f(x_{k+1}) - f(x^*))/M.$$

Therefore

$$\begin{aligned} f(x_k) - f(x_{k+1}) &\geq \alpha\mu \sum_{i=0}^{k} \left(\frac{1-\beta^i}{1-\beta} \right) (f(x_{k+1}) - f(x^*))^2/M \\ &\geq \frac{\alpha\mu (f(x_{k_s}) - f(x^*))^2}{(1-\beta)M} \sum_{i=0}^{k} (1-\beta^i). \end{aligned}$$

Summing up both sides until the restart we get,

$$f(x_0) - f(x_{k_s}) \geq \frac{\alpha\mu(f(x_{k_s}) - f(x^*))^2}{(1-\beta)M} \sum_{k=0}^{k_s} \sum_{i=0}^{k} (1-\beta^i). \tag{14}$$

This inequality is very similar to the one in Lemma 4.2, and the the proofs follow similar steps afterwards. A nice problem to try the algorithm out is given in (16) and the corresponding experiments show linear convergence of Cone Based Restart (and MAGR) and how close the convergence behavior is.

6. An algorithm for non-smooth functions.

In this section we consider the case when the objective function is non-smooth. The usual approach is to replace the objective function with a smooth but approximate one. We on the other hand, will give an extension of MAGR which can be used for non-smooth convex functions. We will call it NSMAGR (Algorithm 3).

Algorithm 3 Non-mmooth momentum accelerated gradient algorithm with gradient-mapping restart

```
Choose x_{-1} ∈ R^n
Choose μ ∈ (0,1)
x_0 = x_{-1}
for k ≥ 0 do
    Choose g_k ∈ ∂f(x_k)
    z_{k+1} = β_k(x_k − x_{k−1}) − α_k g_k
    Choose ĝ ∈ ∂f(x_k + z_{k+1})
    if ĝ^T z_{k+1} > 0 then
        if ĝ^T g_k < 0 then
            β_{k+1} = μβ_k
            x_{k+1} = x_k + z_{k+1}
        else
            x_{k+1} = x_k − α_k g_k
            β_{k+1} = β_k
        end if
    else
        x_{k+1} = x_k + z_{k+1}
    end if
end for
```

In the algorithm NSMAGR, g_k and $\hat{g}$ are sub-gradients of the objective function at x_k and x_k+z_{k+1} respectively. As with any gradient based non-smooth algorithm we are using sub-gradients instead of gradients. The main difference between NSMAGR and MAGR is that we impose a second condition for restart: the algorithm will not restart if there is an abrupt change in the gradient. To check if there is an abrupt change we compute the inner product of the sub-gradients at the current and the projected next step. If it is negative we don't restart; instead we decrease the step-size . In the algorithm, we have taken $\beta_{k+1} = \mu\beta_k$. However one can select different reduction schemes as long as it does not hinder the convergence rate.

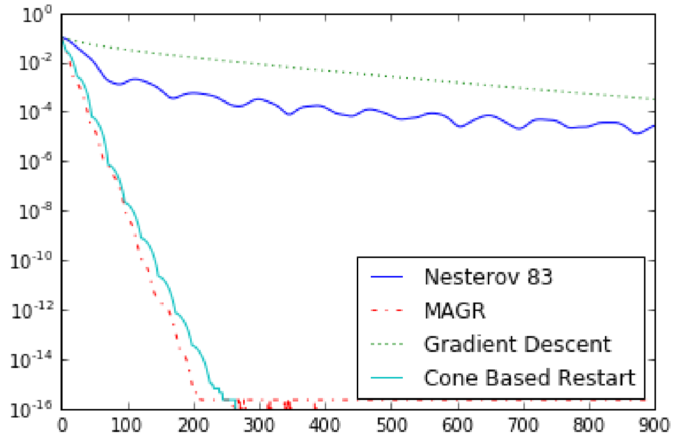

FIG. 1. *Optimizing the smooth version for $\rho = 1$. The vertical axis depicts $\frac{(f(x_n)-f^*)}{f^*}$, and the horizontal axis depicts the iteration number n.*

6.1. A non-smooth example. We tried the non-smooth version of MAGR on minimizing the following function:

$$f(x) = \max_{i=1,\dots,m} (a_i^T x - b_i). \tag{15}$$

One possible smooth approximation of the function is:

$$f(x) \approx \rho \log \left(\sum_{i=1}^{m} \exp((a_i^T x - b_i)/\rho) \right). \tag{16}$$

Although the smooth approximation converges to the original function as $\rho \to 0$, there will be numerical issues as ρ becomes small.

In our numerical experiments we took $\mu = 0.99$ and $\alpha_k = (r+1)/(r-1)$ where r is the number of steps taken after the latest restart.

From Figures 1 and 2, one can observe that accelerated gradient with restart is drastically better than both vanilla accelerated gradient and gradient descent. Classic accelerated gradient is eventually beaten by gradient descent, yet MAGR stays the fastest.

For the non-smooth case the results are very encouraging. Though the algorithms are no longer monotonic, the decrease rate of NSMAGR is still linear (see Figure 3). Yet both accelerated gradient and the vanilla gradient descent get very slow, and cannot get close to the optimum. If we look at the minimum value achieved up to each step (see Figure 4), we can see this better.

We also do a comparison on how fast and accurate NSMAGR is compared to MAGR on the smoothed version of the problem. In Figure 4 we can see that although MAGR on the smoothed version of the problem is fast in the beginning, it converges to a point which is not close to the optimum, while NSMAGR gets very close to the optimum. Also one has to note that using the smoothed function adds a constant overhead at each step, so in fact NSMAGR is also faster in terms of flop count per iteration.

7. Conclusions. Recent analysis of accelerated gradient methods have been based on ODEs [5, 6]. The rough idea is to analyze the continuous case, where

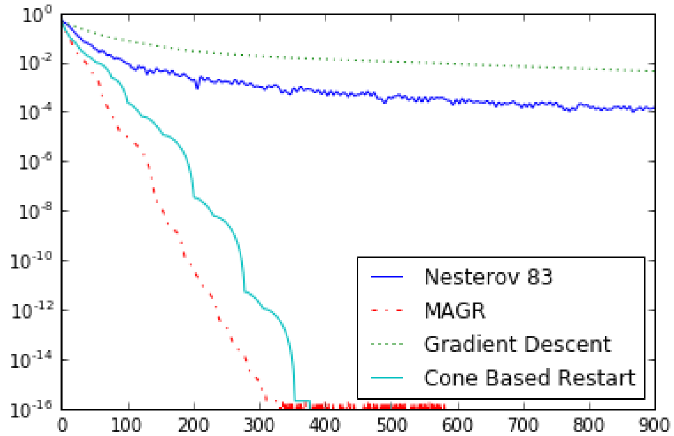

FIG. 2. *Optimizing the smooth version for $\rho = 0.1$. The vertical axis depicts $\frac{(f(x_n)-f^*)}{f^*}$, and the horizontal axis depicts the iteration number n.*

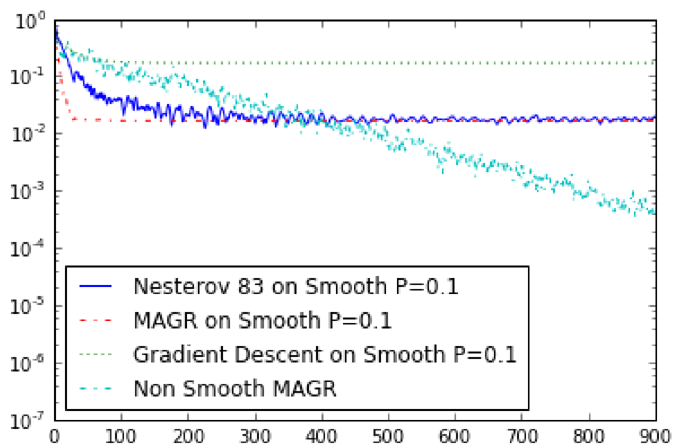

FIG. 3. *Non-smooth problem minimization. The vertical axis depicts $\frac{(f(x_n)-f^*)}{f^*}$, and the horizontal axis depicts iteration number n.*

step size is arbitrarily small, and then expand the analysis by quantizing the continuous path. Here however we used the classical approach in proving the convergence rate. With the restart condition the algorithm becomes monotonic. The momentum vector in the worst case grows like $O(\sqrt{k})$, and even in this case we have shown linear convergence rate. For additional experimental results on how effective this restart rule is the reader can also refer to [1, 4].

This paper has shown that the gradient-mapping based restart scheme will improve the convergence rate of momentum based algorithms to linear. Although this was suspected to be the case in practice we have now proved it to be true under the assumptions (12) and (13), which are quite a bit less restrictive than the assumption of strong convexity. The proposed cone based restarting condition has very similar convergence behavior, yet it is much easier to prove linear convergence. Since it also has the flexibility of the tuning parameter c we believe it may serve well where MAGR

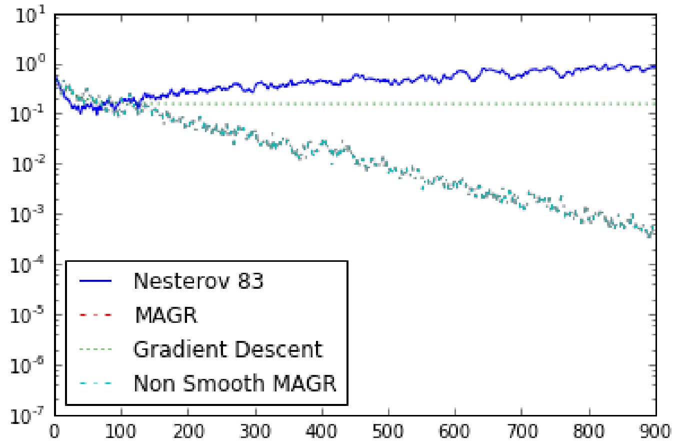

FIG. 4. *Minimums achieved at each step for the non-smooth problem. The vertical axis depicts* $\frac{(f(x_n)-f^*)}{f^*}$*, and the horizontal axis depicts iteration number* n*.*

can not.

It is easy to give examples of non-smooth functions where restart actually worsens the convergence rate (becomes comparable to that of standard gradient descent). However reusing earlier gradients seems to be capable of resolving this issue, and we have given a non-smooth version of MAGR that shows promising results.

Looking forward, it might be possible to find a more general analysis that covers an even larger class of restarting schemes. We would also like to study the convergence rate of our proposed non-smooth MAGR algorithm.

REFERENCES

[1] P. Giselsson and S. Boyd, *Monotonicity and restart in fast gradient methods*, in Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on, IEEE, 2014, pp. 5058–5063.

[2] Y. Nesterov, *A method of solving a convex programming problem with convergence rate o (1/k2)*, in Soviet Mathematics Doklady, vol. 27, 1983, pp. 372–376.

[3] Y. Nesterov, *Smooth minimization of non-smooth functions*, Mathematical programming, 103 (2005), pp. 127–152.

[4] B. Odonoghue and E. Candes, *Adaptive restart for accelerated gradient schemes*, Foundations of computational mathematics, 15 (2015), pp. 715–732.

[5] W. Su, S. Boyd, and E. J. Candes, *A differential equation for modeling nesterovs accelerated gradient method: theory and insights*, Journal of Machine Learning Research, 17 (2016), pp. 1–43.

[6] A. C. Wilson, B. Recht, and M. I. Jordan, *A lyapunov analysis of momentum methods in optimization*, arXiv preprint arXiv:1611.02635, (2016).