CPSC 536C - Algorithms for Convex Optimization

Lecture : Accelerated Gradient Descent

11th March 2026

Lecturer: Trevor Tidy *Scribe: Hasti Karimi*

In previous lectures we have seen that gradient descent achieves a convergence rate of $O(\frac{1}{k})$ for smooth convex functions. In this lecture, we will see how to improve this convergence rate to $O(\frac{1}{k^2})$ using a technique called *accelerated gradient descent* [1].

1 Motivation for Acceleration

The standard gradient descent method can be thought of as a simple iterative process that takes steps proportional to the negative of the gradient at the current point ($-\nabla f(x_k)$). However, this method can be slow to converge, especially when the function has a large condition number (in narrow "valleys" it experiences high oscillations). Also, in flat regions of the function, the gradient can be very small, leading to tiny steps and slow progress towards the optimum.
The idea behind acceleration is to introduce a momentum term that helps the algorithm maintain a sense of direction and speed, allowing it to navigate through flat regions and narrow valleys more effectively. This momentum term can help the algorithm "push through" areas where the gradient is small.

2 Gradient Descent with Momentum

This approach is also known as the Heavy Ball method. The idea is like the name suggests: you descend the function as if you were rolling a heavy ball down the landscape of the function.
The ball has inertia, so it doesn't just respond to the current slope (gradient) but also retains some of its previous velocity. This allows it to build up speed in directions that consistently point towards the minimum, while dampening oscillations in directions that change frequently.

The update rule for gradient descent with momentum is as follows:

$$y_k = x_k + \beta(x_k - x_{k-1}) \qquad \text{(momentum step)}$$
$$x_{k+1} = y_k - \alpha\nabla f(x_k) \qquad \text{(gradient step)}$$

Where:

- y_k is the intermediate point at iteration k.
- β is the momentum coefficient (typically between 0 and 1).
- $x_k - x_{k-1}$ is the momentum carried over from the previous update.

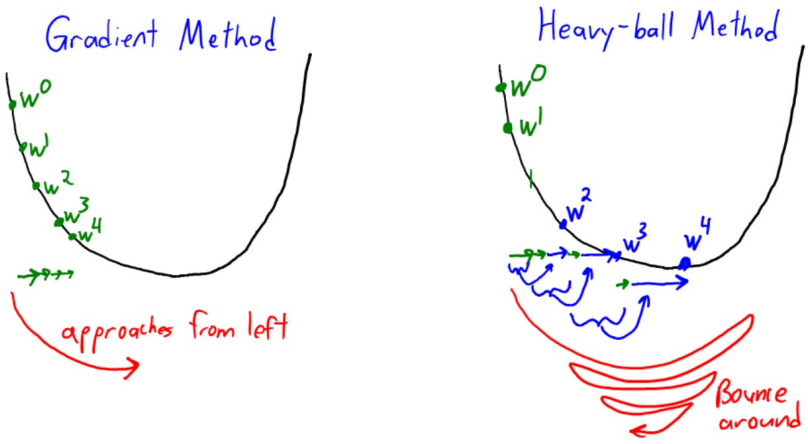

Figure 1: Heavy Ball method vs standard gradient descent.

- α is the learning rate.

However, theoretically, the Heavy Ball method does not guarantee an improved convergence rate over standard gradient descent. It can still converge at a rate of $O(\frac{1}{k})$ in the worst case. In fact, for strictly convex quadratic functions, it achieves an accelerated rate of $(1 - \frac{m}{L})^k$ where m is the strong convexity parameter and L is the smoothness parameter, but this does not generalize to all convex functions.

3 Nesterov's Accelerated Gradient Descent

Rather than using the gradient at the current point x_k, Nesterov's method computes the gradient at the intermediate point y_k. This subtle change allows the algorithm to anticipate the future position of the iterates and adjust its momentum accordingly, leading to a provable convergence rate of $O(\frac{1}{k^2})$ for smooth convex functions which is optimal for first-order methods.
The update rule for Nesterov's Accelerated Gradient Descent is:

$$y_k = x_k + \beta(x_k - x_{k-1}) \qquad \text{(momentum step)}$$
$$x_{k+1} = y_k - \alpha \nabla f(y_k) \qquad \text{("lookahead" gradient step)}$$

3.1 Why does this work well?

The key insight is that, in Nesterov's method, we're literally calculating the gradient at a point in the direction where the momentum is taking us. This allows the algorithm to "look ahead" and adjust its trajectory before it actually reaches that point, which can help it avoid overshooting and oscillations that can occur in the Heavy Ball method.

- If we're soon approaching a minimum with momentum, we can start to slow down before we actually get there, which helps us converge faster. In contrast, the Heavy Ball method might overshoot the minimum and then have to correct itself, leading to slower convergence. Thus, in this case $\nabla f(y_k) > \nabla f(x_k)$ and therefore $-\alpha\nabla f(y_k) < -\alpha\nabla f(x_k)$.
- If we've already overshot the minimum, we can use the gradient at y_k to quickly correct our course and head back towards the minimum. In contrast, the Heavy Ball method might continue to oscillate around the minimum for a while before it settles down. Thus, in this case $\nabla f(y_k) < \nabla f(x_k)$ and therefore $-\alpha\nabla f(y_k) > -\alpha\nabla f(x_k)$.

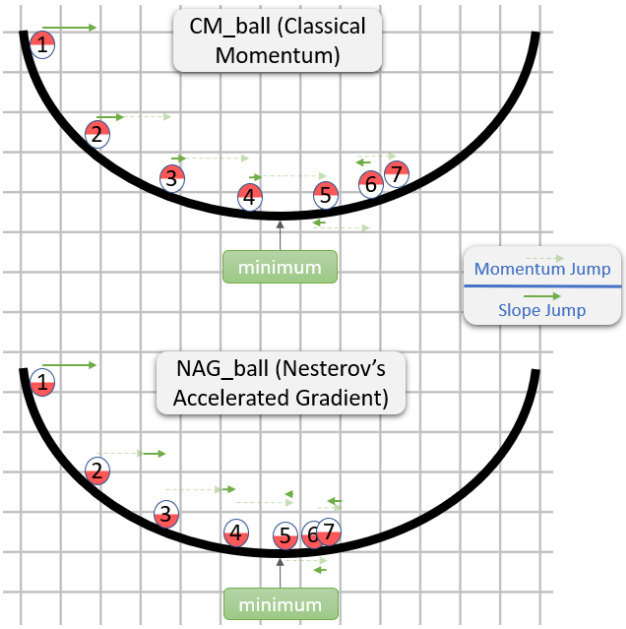

Figure 2: Nesterov's method vs Heavy Ball method.

3.2 AGD for Smooth Convex Functions and its Convergence Analysis

Theorem 3.1. *Suppose f is L-smooth and convex, with a minimizer x^* and minimum value $f^* = f(x^*)$. Then, Algorithm 1 achieves the following convergence rate:*

$$f(x_k) - f^* \leq \frac{2L\|x_0 - x^*\|^2}{(k+1)^2}$$

Proof. First, we will prove the following lemma which will be useful for our analysis:

Lemma 3.2. *For all $k \geq 0$, we have:*

$$f(x_{k+1}) \leq f(y_k) - \frac{1}{2L}\|\nabla f(y_k)\|^2$$

Algorithm 1 Nesterov's AGD, smooth convex

input: initial x_0, smoothness parameter L, number of iterations K
initialize: $x_{-1} = x_0, \alpha = \frac{1}{L}, \lambda_0 = 0, \beta_0 = 0$.
for $k = 0, 1, \ldots, K$ **do**
  $y_k = x_k + \beta_k(x_k - x_{k-1})$
  $x_{k+1} = y_k - \alpha \nabla f(y_k)$
  $\lambda_{k+1} = \frac{1+\sqrt{1+4\lambda_k^2}}{2}, \beta_{k+1} = \frac{\lambda_k - 1}{\lambda_{k+1}}$
end for
return x_K

Proof. By the update rule of x_{k+1}, we have:

$$\begin{aligned} x_{k+1} &= y_k - \alpha \nabla f(y_k) \\ &= y_k - \frac{1}{L} \nabla f(y_k) && \text{(since } \alpha = \tfrac{1}{L}\text{)} \end{aligned}$$

By combining L-smoothness and the above inequality, we have:

$$\begin{aligned} f(x_{k+1}) &\leq f(y_k) + \langle \nabla f(y_k), x_{k+1} - y_k \rangle + \frac{L}{2} \|x_{k+1} - y_k\|^2 \\ &= f(y_k) + \langle \nabla f(y_k), -\frac{1}{L} \nabla f(y_k) \rangle + \frac{L}{2} \| - \frac{1}{L} \nabla f(y_k)\|^2 \\ &= f(y_k) - \frac{1}{L} \|\nabla f(y_k)\|^2 + \frac{1}{2L} \|\nabla f(y_k)\|^2 \\ &= f(y_k) - \frac{1}{2L} \|\nabla f(y_k)\|^2 \end{aligned}$$

□

Next, we will analyze the change in function value from x_k to x_{k+1}:

Lemma 3.3. *For all k, we have:*

$$f(x_{k+1}) - f(x_k) \leq -\frac{1}{2L} \|\nabla f(y_k)\|^2 + L \langle y_k - x_{k+1}, y_k - x_k \rangle$$

Proof.

$$\begin{aligned} & x_{k+1} = y_k - \alpha \nabla f(y_k) \\ \implies & \nabla f(y_k) = -L(x_{k+1} - y_k) \end{aligned} \tag{1}$$

Therefore, we get:

$$\begin{aligned} f(x_{k+1}) - f(x_k) &= f(x_{k+1}) - f(y_k) + f(y_k) - f(x_k) \\ &\leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + f(y_k) - f(x_k) && \text{By Lemma 3.2} \\ &\leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + \langle \nabla f(y_k), y_k - x_k \rangle && \text{By first order convexity of } f \\ &\leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + L\langle y_k - x_{k+1}, y_k - x_k \rangle && \text{Using equation (1)} \end{aligned}$$

□

We can now analyze the change in function value from x_k to x^* similarly.

Lemma 3.4. *For all k, we have:*

$$f(x_{k+1}) - f(x^*) \leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + L\langle y_k - x_{k+1}, y_k - x^* \rangle$$

Proof.

$$\begin{aligned} f(x_{k+1}) - f(x^*) &= f(x_{k+1}) - f(y_k) + f(y_k) - f(x^*) \\ &\leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + f(y_k) - f(x^*) && \text{By Lemma 3.2} \\ &\leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + \langle \nabla f(y_k), y_k - x^* \rangle && \text{By first order convexity of } f \\ &\leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + L\langle y_k - x_{k+1}, y_k - x^* \rangle && \text{Using equation (1)} \end{aligned}$$

□

Next, as we did in the analysis of gradient descent, we want to produce a telescoping argument to analyze the convergence rate of the algorithm. To do so, we will combine the above two lemmas together.

Definition 3.5. We will call Δ_k the gap at iteration k, which is defined as $\Delta_k = f(x_k) - f(x^*)$.

Rewriting the above two lemmas in terms of Δ_k, we have:

$$\Delta_{k+1} - \Delta_k \leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + L\langle y_k - x_{k+1}, y_k - x_k \rangle \tag{2}$$

$$\Delta_{k+1} \leq -\frac{1}{2L}\|\nabla f(y_k)\|^2 + L\langle y_k - x_{k+1}, y_k - x^* \rangle \tag{3}$$

Then, by adding up (eq.2)$\times\lambda_k(\lambda_k - 1)$ and (eq.3)$\times\lambda_k$ together, we have:

$$\lambda_k(\lambda_k - 1)(\Delta_{k+1} - \Delta_k) + \lambda_k\Delta_{k+1} \leq L\langle y_k - x_{k+1}, \lambda_k(\lambda_k - 1)(y_k - x_k) + \lambda_k(y_k - x^*) \rangle - \frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2$$

Setting λ such that $\lambda_{k+1}(\lambda_{k+1}-1)=\lambda_k^2$ allows us to create a telescoping sum in the LHS, which will be useful for our analysis.

$$\begin{aligned}\lambda_k^2\Delta_{k+1}-\lambda_{k-1}^2\Delta_k &= \lambda_k(\lambda_k-1)(\Delta_{k+1}-\Delta_k)+\lambda_k\Delta_{k+1}\\ &\le L\langle y_k-x_{k+1},\lambda_k(\lambda_k-1)(y_k-x_k)+\lambda_k(y_k-x^*)\rangle-\frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2\end{aligned}$$

To make the RHS of the above inequality telescoping as well, we can use the power of β to help us with the analysis. We make the following claim:

Claim 3.6. *The following equality holds:*

$$L\langle y_k-x_{k+1},\lambda_k(\lambda_k-1)(y_k-x_k)+\lambda_k(y_k-x^*)\rangle-\frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2=\frac{L}{2}(\|z_k-x^*\|^2-\|z_{k+1}-x^*\|^2)$$

Where $z_k=\lambda_k y_k-(\lambda_k-1)x_{k-1}$.

Proof. We will first factor out λ_k from the inner product:

$$\begin{aligned}RHS =& L\langle y_k-x_{k+1},\lambda_k(\lambda_k-1)(y_k-x_k)+\lambda_k(y_k-x^*)\rangle-\frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2\\ =& L\lambda_k\langle y_k-x_{k+1},(\lambda_k-1)(y_k-x_k)+(y_k-x^*)\rangle-\frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2\\ =& L\lambda_k\langle y_k-x_{k+1},\lambda_k y_k-(\lambda_k-1)x_k-x^*\rangle-\frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2 \qquad (*)\end{aligned}$$

Now, we want to simplify the second vector in the inner product, that is $\lambda_k y_k-(\lambda_k-1)x_k-x^*$. Note that:

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

First, we will write z_k in terms of x_k and x_{k-1}:

$$\begin{aligned}z_k=\lambda_k y_k-(\lambda_k-1)x_{k-1} &= \lambda_k(x_k+\beta_k(x_k-x_{k-1}))-(\lambda_k-1)x_{k-1}\\ &= \lambda_k\beta_k(x_k-x_{k-1})+\lambda_k x_k-(\lambda_k-1)x_{k-1}\\ &= \lambda_{k-1}x_k-(\lambda_{k-1}-1)x_{k-1}\end{aligned}$$

Also, by setting $\beta_k=\frac{\lambda_{k-1}-1}{\lambda_k}$, we have:

$$\begin{aligned}\lambda_k y_k-(\lambda_k-1)x_k-x^* &= \lambda_k(x_k+\beta_k(x_k-x_{k-1}))-(\lambda_k-1)x_k-x^*\\ &= \lambda_k\beta_k(x_k-x_{k-1})+x_k-x^*\\ &= (\lambda_{k-1}-1)(x_k-x_{k-1})+x_k-x^*\\ &= \lambda_{k-1}x_k-(\lambda_{k-1}-1)x_{k-1}-x^*\\ &= z_k-x^*\end{aligned}$$

Where the last equality follows from the definition of z_k in terms of x_k and x_{k-1}. Now by substituting the above equality into eq.*, we have:

$$\begin{aligned} RHS =& L\lambda_k\langle y_k - x_{k+1}, z_k - x^*\rangle - \frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2 \\ =& L\lambda_k\langle \frac{1}{L}\nabla f(y_k), z_k - x^*\rangle - \frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2 && \text{gradient step} \\ =& \lambda_k\langle \nabla f(y_k), z_k - x^*\rangle - \frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2 \end{aligned}$$

Fact 3.7. *The following equality holds:*

$$2\langle a, b\rangle - \|a\|^2 = \|b\|^2 - \|a - b\|^2$$

Exercise 3.8. *By setting $g = \nabla f(y_k)$, $a = \sqrt{\frac{\lambda_k}{L}}g$ and $b = \sqrt{\frac{\lambda_k}{L}}(z_k - x^*)$, prove the following:*

$$\lambda_k\langle \nabla f(y_k), z_k - x^*\rangle - \frac{\lambda_k^2}{2L}\|\nabla f(y_k)\|^2 = \frac{L}{2}(\|z_k - x^*\|^2 - \|z_{k+1} - x^*\|^2)$$

Which proves the claim.

□

Therefore, we have:

$$\lambda_k^2\Delta_{k+1} - \lambda_{k-1}^2\Delta_k \leq \frac{L}{2}(\|z_k - x^*\|^2 - \|z_{k+1} - x^*\|^2)$$

Now, by summing up the above inequality for $k = 0, 1, \ldots, k$, we have:

$$\begin{aligned} & \lambda_k^2\Delta_{k+1} + \lambda_0^2\Delta_1 \leq \frac{L}{2}(\|z_0 - x^*\|^2 - \|z_{k+1} - x^*\|^2) \\ \implies & \lambda_k^2\Delta_{k+1} + 0 \leq \frac{L}{2}\|z_0 - x^*\|^2 && (*) \end{aligned}$$

Fact 3.9. *By the choice of $\lambda_0 = 0$, we have $z_0 = x_0$.*

Proof.

$$\begin{aligned} z_0 &= \lambda_0 y_0 - (\lambda_0 - 1)x_{-1} \\ &= 0 \cdot y_0 - (-1)x_0 \\ &= x_0 \end{aligned}$$

□

Lemma 3.10. *By the choice of $\lambda_1 = 1$, we have $\lambda_k \geq \frac{k+1}{2}$ for all $k \geq 1$.*

Proof. We will prove the lemma by induction. For $k = 1$, we have $\lambda_1 = 1 \geq \frac{1+1}{2}$. Now, suppose the lemma holds for all $k' < k$. Then, we have:

$$\begin{aligned}\lambda_k &= \frac{1 + \sqrt{1 + 4\lambda_{k-1}^2}}{2} \\ &\geq \frac{1 + \sqrt{1 + k^2}}{2} \\ &\geq \frac{1 + k}{2}\end{aligned}$$

□

Now, by substituting the above lemma into equation.*, we have:

$$\begin{aligned}\Delta_{k+1} &\leq \frac{L\|z_0 - x^*\|^2}{2\lambda_k^2} \\ &\leq \frac{2L}{(k+1)^2}\|z_0 - x^*\|^2 \\ &= \frac{2L}{(k+1)^2}\|x_0 - x^*\|^2\end{aligned}$$

Which gives us the desired convergence rate of $O(\frac{1}{k^2})$ for Nesterov's Accelerated Gradient Descent.

□

Theorem 3.11. *Equivalantely, we can also show that Nesterov's AGD has $O(\frac{1}{\sqrt{\epsilon}})$ iteration complexity to achieve an ϵ-optimal solution, i.e., to find x_k such that $f(x_k) - f^* \leq \epsilon$, we need at most $O(\frac{1}{\sqrt{\epsilon}})$ iterations.*

Proof. Using the result above, it suffices to find k such that $\frac{2L\|x_0 - x^*\|^2}{(k+1)^2} \leq \epsilon$.

$$\begin{aligned}&\frac{2L\|x_0 - x^*\|^2}{(k+1)^2} \leq \epsilon \\ &\implies (k+1)^2 \geq \frac{2L\|x_0 - x^*\|^2}{\epsilon} \\ \implies k &= O(\frac{1}{\sqrt{\epsilon}})\end{aligned}$$

□

References

[1] A method for solving the convex programming problem with convergence rate o (1/k2). In *Dokl akad nauk Sssr*, volume 269, page 543, 1983.