# On the Links Between Integration Methods and Optimization Algorithms

Damien Scieur, Vincent Roulet, Francis Bach, Alexandre d'Aspremont

ENS - INRIA

October 18, 2017

# Outline

A The gradient flow

# Outline

A The gradient flow

B Optimization algorithms and ODE: a small review

1. Gradient method
2. Nesterov's method and Accelerated mirror descent

# Outline

A The gradient flow

B Optimization algorithms and ODE: a small review

1. Gradient method
2. Nesterov's method and Accelerated mirror descent

C Linear multi-step methods

1. Generalization
2. How to model Nesterov as a linear multi-step method?

# The Gradient Flow

Let $\dot{x} = \frac{dx}{dt}$. The gradient flow is described by the ODE

$$\dot{x} = -\nabla f(x) \qquad ; \qquad x(0) = x_0$$

We can prove, if $f$ is strongly convex, that

$$\|x(t) - x^*\|^2 \quad \leq \quad e^{-\mu t}\|x_0 - x^*\|^2$$

If the step size is $h = \frac{1}{L}$, then for $t = kh$ we have

$$\|x(kh) - x^*\|^2 \quad \leq \quad e^{-k\frac{\mu}{L}}\|x_0 - x^*\|^2$$

We recover the rate of gradient method.

# Gradient method

- ODE Proposed:

$$\dot{x} = -\nabla f(x)$$

- Discretization on the interval $[t_k, t_{k+1}]$: **Forward Euler**.

$$\dot{x}(t) \approx \frac{x(t_k + h) - x(t_k)}{h} \qquad \nabla f(x(t)) \approx \nabla f(x(t_k))$$

Writing $x_k = x(t_k)$, and assuming $t_k = kh$, we have

$$x_{k+1} = x_k - h\nabla f(x_k)$$

# Proximal Gradient method

- ODE Proposed:

$$\dot{x} = -\nabla f(x)$$

- Discretization on the interval $[t_k, t_{k+1}]$: **Backward Euler**.

$$\dot{x}(t) \approx \frac{x(t_k + h) - x(t_k)}{h} \qquad \nabla f(x(t)) \approx \nabla f(x(t_{k+1}))$$

Writing $x_k = x(t_k)$, and assuming $t_k = kh$, we have

$$x_{k+1} = x_k - h\nabla f(x_{k+1}) = \mathrm{prox}_{hf}(x_k)$$

# Nesterov's method

Algorithm for convex functions (where $\beta_k = \frac{k-2}{k+1}$ ):

$$\begin{aligned} x_{k+1} &= y_k - \frac{1}{L} f(y_k) \\ y_{k+1} &= -\beta_k x_k + (1+\beta_k) x_{k+1} \end{aligned}$$

For strongly convex functions, $\beta = \frac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}}$ where $\kappa = \frac{\mu}{L}$ is the condition number.

Two different models:

- **(A)** Su, Boyd, Candes
- **(B)** Wibisono, Wilson, Jordan ; Wilson, Recht, Jordan.

# Model **(A)** (Su, Boyd, Candes)

The authors starts from the algorithm, then takes the limits when the step size $\frac{1}{L}$ goes to zero. The model becomes

$$\ddot{x} + \frac{r}{t}\dot{x} = -\nabla f(x),$$

where $r = 3$ is called the «magic constant».

They recover the initial algorithm using non-trivial **Forward Euler** approximation.

They proved that the ODE converge at the accelerated rate, but nothing about the convergence of the discretization.

# Model **(B)** (simplified) (Wibisono, Wilson, Recht, Jordan)

The authors introduce the following ODE,

$$\frac{d}{dt}(x + e^{-\alpha_t}\dot{x}) = -e^{\alpha_t+\beta_t}\nabla f(x),$$

where $\alpha_t$, $\beta_t$ should follow the *ideal scaling condition*

$$\dot{\beta}_t \leq e^{\alpha_t}.$$

The authors show that

- A (non-trivial) discretization of this ODE models Nesterov's gradient (**forward Euler**) and accelerated mirror descent (**backward Euler**).
- The ODE converges at the accelerated rate.
- There exists a "general" *discrete Lyapunov function* which proves the fast convergence of the discrete scheme (but without any links with the ODE).

For now, each discretization uses **forward** or **backward** Euler method (in a complicated way) on a non-trivial ODE.

Question: *What happen if we use a more sophisticated method on a simpler ODE?*

# Linear multi-step methods

Euler method (Forward - explicit):

$$x_{k+1} - x_k = -h\nabla f(x_k)$$

Euler method (backward - implicit):

$$x_{k+1} - x_k = -h\nabla f(x_{k+1})$$

# Linear multi-step methods

Euler method (Forward - explicit):

$$x_{k+1} - x_k = -h\nabla f(x_k)$$

Euler method (backward - implicit):

$$x_{k+1} - x_k = -h\nabla f(x_{k+1})$$

Generalization:

$$\sum_{i=0}^{s} \rho_i x_{k+i} = -h \sum_{i=0}^{s} \sigma_i \nabla f(x_{k+i})$$

# Linear multi-step methods

Euler method (Forward - explicit):

$$x_{k+1} - x_k = -h\nabla f(x_k)$$

Euler method (backward - implicit):

$$x_{k+1} - x_k = -h\nabla f(x_{k+1})$$

Generalization:

$$\sum_{i=0}^{s} \rho_i x_{k+i} = -h \sum_{i=0}^{s} \sigma_i \nabla f(x_{k+i})$$

Introducing the **shift operator** $E : Ex_k \to x_{k+1}\ ;\ E\nabla f(x_k) \to \nabla f(x_{k+1})$,

$$\left(\sum_{i=0}^{s} \rho_i E^i\right) x_k = -h \left(\sum_{i=0}^{s} \sigma_i E^i\right) \nabla f(x_k)$$

Let $\rho$ and $\sigma$ be two polynomials of degree $s$, with $\rho_s = 1$ (by convention). Then

$$\rho(E)x_k = -h\sigma(E)\nabla f(x_k)$$

A linear multi-step methods is uniquely defined by the pair $(\rho, \sigma)$.

# Linear multi-step methods

Many important characteristics:

- Consistency
- (Order of convergence)
- (Zero-stability)
- (A-stability)
- (G-stability)
- ...

## Consistency

Let $(\rho, \sigma)$ be a linear multi-step method which generates the sequence $x_k$ on the ODE $\dot{x} = -\nabla f(x)$, using step size $h$. Assume that the first iterates are exact, i.e.

$$(x_k, x_{k+1}, ..., x_{k+s-1}) = (x(t_k), x(t_{k+1}), ..., x(t_{k+s-1}))\,.$$

Then the method is ***consistent*** if and only if

$$\lim_{h\to 0} \frac{1}{h}\|x_k(h) - x(t_k)\| = 0$$

## Consistency

Let $(\rho, \sigma)$ be a linear multi-step method which generates the sequence $x_k$ on the ODE $\dot{x} = -\nabla f(x)$, using step size $h$. Assume that the first iterates are exact, i.e.

$$(x_k, x_{k+1}, ..., x_{k+s-1}) = (x(t_k), x(t_{k+1}), ..., x(t_{k+s-1}))\,.$$

Then the method is ***consistent*** if and only if

$$\lim_{h\to 0} \frac{1}{h}\|x_k(h) - x(t_k)\| = 0$$

This is equivalent to te condition (proof using Taylor expansion)

$$\rho(1) = 0 \qquad \text{and} \qquad \sigma(1) = \rho'(1)$$

***Intuition:***

- If we start at $x_0 = x^*$, the first condition ensures $x_k = x^*$ for all $k$.
- If the second condition is not satisfied, then the method presents artificial gain or damping.

# Nesterov's method viewed as a linear multi-step method

(Reminder) Nesterov's method:

$$\begin{aligned} x_{k+1} &= y_k - \frac{1}{L} f(y_k) \\ y_{k+1} &= -\beta_k x_k + (1+\beta_k) x_{k+1} \end{aligned}$$

# Nesterov's method viewed as a linear multi-step method

(Reminder) Nesterov's method:

$$
\begin{aligned}
x_{k+1} &= y_k - \frac{1}{L} f(y_k) \\
y_{k+1} &= -\beta_k x_k + (1+\beta_k) x_{k+1}
\end{aligned}
$$

If we expand $y_{k+1}$,

$$
y_{k+1} = -\beta_k \left( y_{k-1} - \frac{1}{L} \nabla f(y_{k-1}) \right) + (1+\beta_k) \left( y_k - \frac{1}{L} \nabla f(y_k) \right)
$$

# Nesterov's method viewed as a linear multi-step method

(Reminder) Nesterov's method:

$$\begin{aligned} x_{k+1} &= y_k - \frac{1}{L} f(y_k) \\ y_{k+1} &= -\beta_k x_k + (1+\beta_k) x_{k+1} \end{aligned}$$

If we expand $y_{k+1}$,

$$y_{k+1} = -\beta_k \left( y_{k-1} - \frac{1}{L} \nabla f(y_{k-1}) \right) + (1+\beta_k) \left( y_k - \frac{1}{L} \nabla f(y_k) \right)$$

If we separate $y_k$ and $\nabla f(y_k)$,

$$\beta_k y_{k-1} - (1+\beta_k) y_k + y_{k+1} = -\frac{1}{L} \left( -\beta_k \nabla f(y_{k-1}) + (1+\beta_k) \nabla f(y_k) \right)$$

# Nesterov's method viewed as a linear multi-step method

$$\beta_k y_{k-1} - (1+\beta_k) y_k + y_{k+1} = -\frac{1}{L}\left(-\beta_k \nabla f(y_{k-1}) + (1+\beta_k)\nabla f(y_k)\right)$$

We will check if the method is consistent $\big($i.e. $\rho(1) = 0$ and $\rho'(1) = \sigma(1)\big)$:

# Nesterov's method viewed as a linear multi-step method

$$\beta_k y_{k-1} - (1+\beta_k) y_k + y_{k+1} = -\frac{1}{L}\left(-\beta_k \nabla f(y_{k-1}) + (1+\beta_k)\nabla f(y_k)\right)$$

We will check if the method is consistent (i.e. $\rho(1) = 0$ and $\rho'(1) = \sigma(1)$):

- First condition: $\rho(1) = 0$

$$\rho(1) \quad = \quad \beta_k - (1+\beta_k) + 1 = 0 \qquad \textbf{OK}$$

# Nesterov's method viewed as a linear multi-step method

$$\beta_k y_{k-1} - (1+\beta_k) y_k + y_{k+1} = -\frac{1}{L}\left(-\beta_k \nabla f(y_{k-1}) + (1+\beta_k)\nabla f(y_k)\right)$$

We will check if the method is consistent (i.e. $\rho(1) = 0$ and $\rho'(1) = \sigma(1)$):

- First condition: $\rho(1) = 0$

$$\rho(1) \quad = \quad \beta_k - (1+\beta_k) + 1 = 0 \qquad \textbf{OK}$$

- Second condition: $\rho'(1) = \sigma(1) \Leftrightarrow h\rho'(1) = h\sigma(1)$

$$\begin{aligned} h\rho'(1) &= h(-(1+\beta_k)+2) = h(1-\beta_k) \\ h\sigma(1) &= \frac{1}{L}\left(-\beta_k + 1 + \beta_k\right) = \frac{1}{L} \end{aligned}$$

Since we need to have $\rho'(1) = \sigma(1)$, we conclude that $h = \frac{1}{L(1-\beta_k)}$.

# Nesterov's method viewed as a linear multi-step method

The parameters of Nesterov's method are thus (after identification)

- $\rho_k = [\beta_k \; ; \; -(1+\beta_k) \; ; \; 1]$
- $\sigma_k = (1-\beta_k) * [-\beta_k \; ; \; 1+\beta_k \; ; \; 0]$
- $h_k = \frac{1}{L(1-\beta_k)}$

Since $\beta_k \in ]0,1[$, the **step size is larger**.

# Nesterov's method viewed as a linear multi-step method

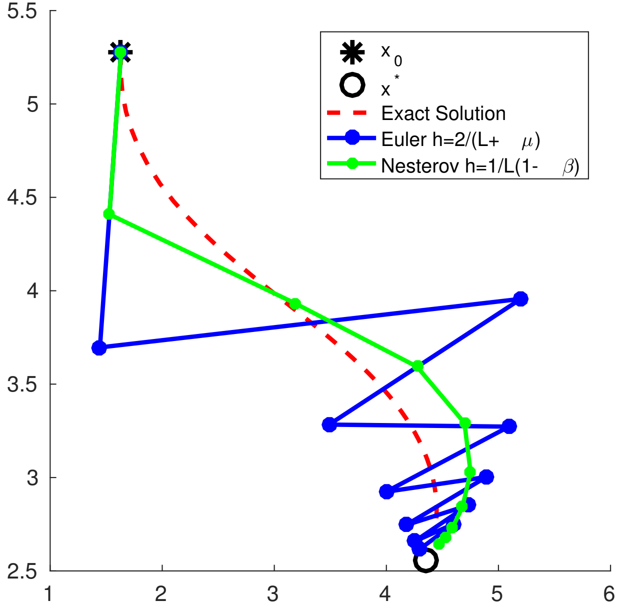
The parameters of Nesterov's method are thus (after identification)

- $\rho_k = [\beta_k \; ; \; -(1+\beta_k) \; ; \; 1]$
- $\sigma_k = (1-\beta_k) * [-\beta_k \; ; \; 1+\beta_k \; ; \; 0]$
- $h_k = \frac{1}{L(1-\beta_k)}$

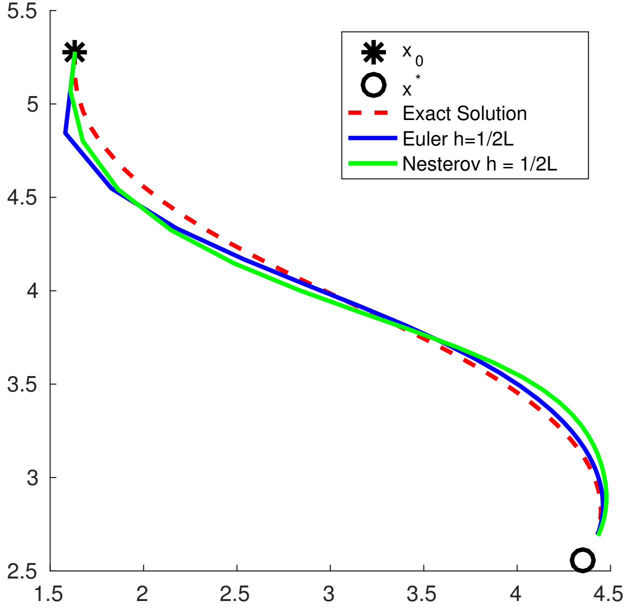
Since $\beta_k \in ]0,1[$, the **step size is larger**.

Intuitive estimation of the rate of convergence:

- (Convex case) If $\beta_k = \frac{k-2}{k+1}$, $h = \frac{1}{L}\frac{k+1}{3}$. It means that we go $\approx k$ times faster than usual gradient method. The rate is $\approx \frac{1}{k^2}$.
- (Strongly convex case) If $\beta_k = \frac{1-\sqrt{\kappa}}{1+\sqrt{\kappa}}$, $h = \frac{1}{L}\frac{1+\sqrt{\kappa}}{2}$. We go $\approx \sqrt{\kappa}$ times faster. The rate is $\approx (1-\sqrt{\kappa})^k$



5.5
5
4.5
4
3.5
3
2.5
1
2
3
4
5
6
$x_0$
$x^*$
Exact Solution
Euler h=2/(L+ $\mu$)
Nesterov h=1/L(1- $\beta$)



5.5
5
4.5
4
3.5
3
2.5
1.5
2
2.5
3
3.5
4
4.5
x 0
x *
Exact Solution
Euler h=1/2L
Nesterov h = 1/2L

# Conclusion

Main contributions:

- Using simple arguments, we have strong links between many algorithms in optimization and integration methods.
- The approach is straightforward, and without any magic tricks we are able to understand why Nesterov's method is faster.
- Using consistency, zero-stability and some optimality argument, we are able to derive a family of two-steps methods (which contains Nesterov's gradient and Polyak's heavy ball).

Future work:

- Proof of convergence.
- The convex case is not so clear: the method change over time.
- Extension to non-Euclidean case?