Continuous-time Models for Stochastic Optimization Algorithms

Antonio Orvieto, Aurelien Lucchi

Accepted as a poster at NeurIPS | 2019
Unconstrained non-convex optimization

For some regular $f : \mathbb{R}^d \to \mathbb{R}$, find $x^* := \arg \min_{x \in \mathbb{R}^d} f(x)$.

Training loss of ResNet-110, no skip connections on CIFAR-10
(for more details, check [Li et al., 2018])

Usual Assumption: $f$ is $L$-smooth, i.e. $\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|$.

A recent trend is to model the dynamics of iterative gradient-based optimization algorithms with differential equations.
Tutorial: how is an ODE model constructed?

\[ x_{k+1} = x_k - h\nabla f(x_k) \]  \hspace{1cm} (GD)

Define curve \( y(t) \) as smooth interpolation: \( y(kh) = x_k \)

What is the law for \( y(t) \)?
1) by construction : \[ y(t + h) = y(t) - h\nabla f(y(t)) \];
2) thanks to smoothness: \[ y(t + h) = y(t) + \dot{y}(t) + \mathcal{O}(h^2) \].

In the limit \( h \to 0 \), \[ \dot{y}(t) = -\nabla f(y(t)) \]

*Solution \( y \in C^1(\mathbb{R}_+, \mathbb{R}^d) \) exists unique in since \( \nabla f \) is globally Lip.
## Some important ODE/SDE models

| Algorithm | Model | Perturbed model (stochastic grads) |
|-----------|-------|-----------------------------------|
| \( x_{k+1} = x_k - h \nabla f(x_k) \) | \( \dot{X} = -\nabla f(X) \) | \( dY = -\nabla f(Y) dt + \sigma dB \) |
| **(GD)** | | [Mertikopoulos and Staudigl, 2016] |
| \( x_{k+1} = x_k + \beta(x_k - x_{k-1}) - h \nabla f(x_k) \) | \( \dot{y} = -\alpha \dot{y} - \nabla f(y) \) | \( \begin{cases} 
\dot{v} = -\alpha v - \nabla f(y) \\
\dot{y} = v
\end{cases} \begin{cases} 
\dot{V} = -\alpha V dt - \nabla f(Y) dt + \sigma dB \\
\dot{Y} = Y dt
\end{cases} \) |
| **(HB)** | | [Polyak, 1964] [Orvieto et al., 2019] |
| \( \begin{cases} 
x_{k+1} = u_k - h \nabla f(u_k) \\
u_{k+1} = x_{k+1} + \frac{k}{k+3} (x_{k+1} - x_k)
\end{cases} \) | \( \dot{y} = -\frac{3}{t} \dot{y} - \nabla f(y) \) | \( \begin{cases} 
\dot{v} = -\frac{3}{t} v - \nabla f(y) \\
\dot{y} = v
\end{cases} \begin{cases} 
\dot{V} = -\frac{3}{t} V dt - \nabla f(Y) dt + \sigma dB \\
\dot{Y} = V dt
\end{cases} \) |
| **(NAG)** | | [Nesterov, 1983] [Su et al., 2016] [Krichene and Bartlett, 2017] |

and many more: primal-dual algorithms, adaptive methods, etc.
why should we care about SDE models?
do we really need to introduce these objects? what’s the gain?

- GD-ODE/SDE is the basis for many seminal contributions to the theory of SGD:
  1. asymptotic behavior [Kushner and Yin, 2003];
  2. connection to Bayesian inference [Mandt et al., 2017];
  3. generalization, width of minimas [Jastrzębski et al., 2017].

- NAG-ODE recently provided us with some novel insights of the acceleration phenomenon\(^1\):
  1. [Su et al., 2016] studied accel. with Bessel functions;
  2. [Wibisono et al., 2016] connected NAG to meta-learning and physics via the minimum action principle;
  3. [Krichene and Bartlett, 2017] studied the non-trivial interplay between noise and acceleration in NAG using stochastic analysis on the NAG-SDE;
  4. [Orvieto et al., 2019] showed NAG is equivalent to a linear gradient averaging system after the time-stretch \(\tau = t^2/8\).

\(^1\)For convex functions, there is a method (NAG) strictly faster than GD.
In this paper, inspired by this success.

- we build SDE models for SVRG and mini-batch SGD, which include the effect of **decaying learning rates and increasing batch-sizes**.

- We derive **convergence rates for our models**. We focus on non-convex functions relevant for machine learning.

- We derive equivalent **novel results for the algorithmic counterparts**, using the same Lyapunov functions. This proves the effectiveness of our SDE models.

- We provide a new interpretation for the distribution induced by SGD with decreasing stepsizes, which reveals an underlying **time warping** that can be used for designing Lyapunov functions.

- We provide a dual interpretation of this last phenomenon as **landscape stretching**.
SDEs description

Below are the two SDEs corresponding to mini-batch SGD (MB-PGF) and SVRG (VR-PGF).

\[
\begin{align*}
\text{d}X(t) &= -\psi(t)\nabla f(X(t)) \, dt + \psi(t)\sqrt{\frac{h}{b(t)}} \sigma_{\text{MB}}(X(t)) \, d\B(t) \\
\text{d}X(t) &= -\psi(t)\nabla f(X(t)) \, dt + \psi(t)\sqrt{\frac{h}{b(t)}} \sigma_{\text{VR}}(X(t), X(t - \xi(t))) \, d\B(t)
\end{align*}
\]

where

- \( \xi : \mathbb{R}_+ \rightarrow [0, \Xi] \) is the \textit{staleness function} (linked to the pivot update frequency \( m \) in SVRG);
- \( \psi(\cdot) \in C^1(\mathbb{R}_+, [0, 1]) \) is the \textit{adjustment function} (encodes the relative decrease in the learning rate)
- \( b(\cdot) \in C^1(\mathbb{R}_+, \mathbb{R}_+) \) is the \textit{mini-batch size function};
- \( \{B(t)\}_{t \geq 0} \) is a \( \dim - \)dimensional Brownian Motion on some filtered probability space.
We derive matching convergence rates in continuous- and discrete-time, using the same Lyapunov functions. This proves the effectiveness of our SDE models.
Insight 1: time stretching

Using the SDE models, we can transform an algorithm to an equivalent one which is easier to study.

**Theorem.** Let \( \{X(t)\}_{t \geq 0} \) satisfy PGF and define \( \tau(\cdot) = \varphi^{-1}(\cdot) \), where \( \varphi(t) = \int_0^t \psi(s)ds \). For all \( t \geq 0 \), \( X(\tau(t)) = Y(t) \) in distribution, where \( \{Y(t)\}_{t \geq 0} \) has the stochastic differential

\[
dY(t) = -\nabla f(Y(t))dt + \sqrt{h \psi(\tau(t))/b(\tau(t))}\sigma(\tau(t)) dB(t).
\]

**Example.**

\( b(t) = 1, \sigma(s) = \sigma I_d \) and \( \psi(t) = 1/(t + 1); \) we have \( \varphi(t) = \log(t + 1) \) and \( \tau(t) = e^t - 1 \).

\[
dX(t) = -\frac{1}{t+1} \nabla f(X(t))dt - \frac{\sqrt{h} \sigma}{t+1} dB(t) \text{ is s.t. the sped-up solution } Y(t) = X(e^t - 1)
\]

satisfies

\[
dY(t) = -\nabla f(X(t))dt + \sqrt{h \sigma e^{-t}} dB(t).
\]

Verification of the Thm. on a 1d quadratic (100 samples): empirically \( X(t) \triangleq Y(\varphi(t)) \).
Insight 2: landspace stretching

For the sake of simplicity, let \( f(x) = \frac{1}{2} \|x\|^2 \). PGF with \( b(t) = 1, \sigma(s) = \sigma I_d, \psi(t) = \frac{1}{t+1} \) is

\[
dX(t) = -\frac{1}{t+1}X(t)dt + \frac{h\sigma}{t+1}dB(t).
\]

Using solution feedback (only possible with a continuous time formulation), we find that in expectation

\[
\mathbb{E}[dX] = CX^2 dt \rightarrow \frac{d\mathbb{E}[X]}{dt} = C\nabla (X^3/3).
\]

Hence, PGF on the quadratic \( \frac{1}{2} \|x\|^2 \) with learning rate decreasing as \( 1/t \) behaves in expectation like PGF with constant learning rate on a cubic.

i.e., we loose strong convexity hence we converge slower!
References
Anosov, D. V. (1967).  
Geodesic flows on closed riemannian manifolds of negative curvature.  
*Trudy Matematicheskogo Instituta Imeni VA Steklova*, 90:3–210.

Hairer, E., Lubich, C., and Wanner, G. (2006).  
*Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, volume 31.  
Springer Science & Business Media.

Jastrzębski, S., Kenton, Z., Arpit, D., Ballas, N., Fischer, A., Bengio, Y., and Storkey, A. (2017).  
Three Factors Influencing Minima in SGD.  
*ArXiv e-prints.*

Krichene, W. and Bartlett, P. L. (2017).  
Acceleration and Averaging in Stochastic Mirror Descent Dynamics.  
*ArXiv e-prints.*
Kushner, H. and Yin, G. (2003).

*Stochastic Approximation and Recursive Algorithms and Applications.*

Stochastic Modelling and Applied Probability. Springer New York.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. (2018).

Visualizing the loss landscape of neural nets.

In *Advances in Neural Information Processing Systems*, pages 6389–6399.

Mandt, S., Hoffman, M. D., and Blei, D. M. (2017).

Stochastic gradient descent as approximate bayesian inference.

*The Journal of Machine Learning Research*, 18(1):4873–4907.

Mertikopoulos, P. and Staudigl, M. (2016).

On the convergence of gradient-like flows with noisy gradient input.

*ArXiv e-prints.*

Nesterov, Y. E. (1983).
A method for solving the convex programming problem with convergence rate $o \left( \frac{1}{k^2} \right)$.

In *Dokl. Akad. Nauk SSSR*, volume 269, pages 543–547.

Orvieto, A., Kohler, J., and Lucchi, A. (2019).
The role of memory in stochastic optimization.
*arXiv preprint arXiv:1907.01678*.

Polyak, B. T. (1964).
Some methods of speeding up the convergence of iteration methods.
*USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17.

Su, W., Boyd, S., and Candès, E. J. (2016).
A differential equation for modeling nesterovs accelerated gradient method: theory and insights.
*Journal of Machine Learning Research*, 17(153):1–43.

Wibisono, A., Wilson, A. C., and Jordan, M. I. (2016).
A Variational Perspective on Accelerated Methods in Optimization.

ArXiv e-prints.