Adaptive Gradient Descent without Descent

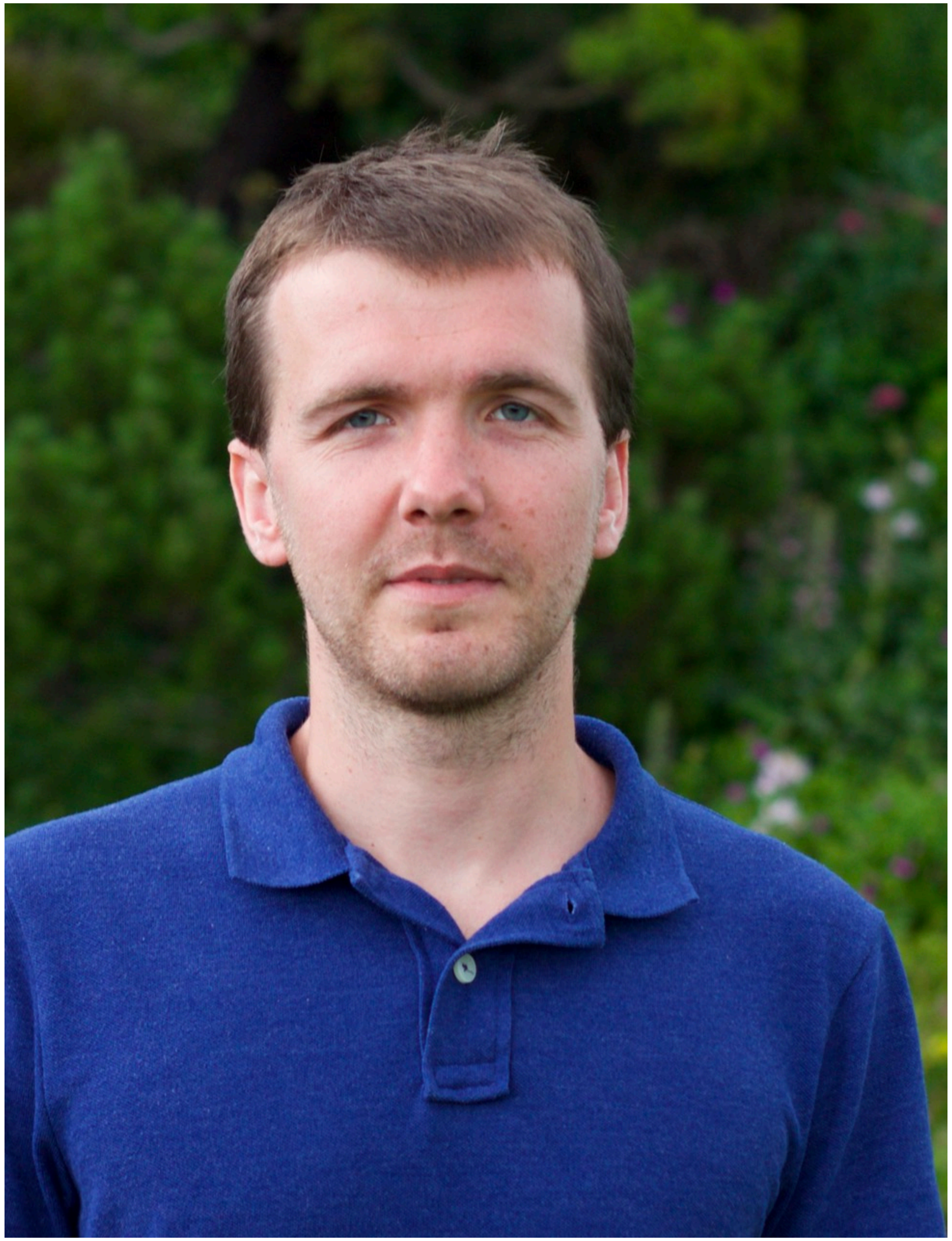
ICML 2020

Yura Malitsky, Konstantin Mishchenko

EPFL

جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology

Yura Malitsky



Gradient Descent

$$\min_{x \in \mathbb{R}^d} f(x)$$

$$x^{k+1} = x^k - \lambda_k \nabla f(x^k)$$

Gradient Descent

$$\min_{x \in \mathbb{R}^d} f(x)$$

$$x^{k+1} = x^k - \lambda_k \nabla f(x^k)$$

Gradient Descent

$$\min_{x \in \mathbb{R}^d} f(x)$$

$$x^{k+1} = x^k - \lambda_k \nabla f(x^k)$$

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$$

Gradient Descent

$$\min_{x \in \mathbb{R}^d} f(x)$$

$$x^{k+1} = x^k - \lambda_k \nabla f(x^k)$$

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|$$

$$\lambda_k < \frac{2}{L}$$

Some concerns

1. **How do we know *L*?**

Some concerns

1. **How do we know *L*?**
2. **What if *L* doesn't exist?**

$$\min_{\mathbf{U}\in\mathbb{R}^{n\times p},\mathbf{V}\in\mathbb{R}^{m\times p}} \frac{1}{2}\|\mathbf{A}-\mathbf{U}\mathbf{V}^\top\|_{\mathrm{F}}^2$$

Some concerns

1. **How do we know *L*?**
2. **What if *L* doesn't exist?**
3. **What if we can do better?**

Limitations of existing methods

1. **Bad guarantees (adaptive line search)**

$$\lambda_k \in \{2^p \lambda_{k-1} \mid p \in \mathbb{Z}\}$$

Limitations of existing methods

1. **Bad guarantees (adaptive line search)**
2. **Lack of adaptivity (line search with decreasing λ_k)**

$$\lambda_k \in \left\{ \lambda_{k-1}, \frac{1}{2}\lambda_{k-1}, \dots, \right\}$$

Limitations of existing methods

1. **Bad guarantees (adaptive line search)**
2. **Lack of adaptivity (line search with decreasing λ_k)**
3. **Provably divergent (Barzilai-Borwein)**

$$\lambda_k = \frac{\langle x^k - x^{k-1}, \nabla f(x^k) - \nabla f(x^{k-1}) \rangle}{\|\nabla f(x^k) - \nabla f(x^{k-1})\|^2}$$

Limitations of existing methods

1. **Bad guarantees (adaptive line search)**
2. **Lack of adaptivity (line search with decreasing λ_k)**
3. **Provably divergent (Barzilai-Borwein)**
4. **Use unknown information (Polyak stepsize)**

$$\lambda_k = \frac{f(x^k) - f(x^*)}{\|\nabla f(x^k)\|^2}$$

Limitations of existing methods

1. **Bad guarantees (adaptive line search)**
2. **Lack of adaptivity (line search with decreasing λ_k)**
3. **Provably divergent (Barzilai-Borwein)**
4. **Use unknown information (Polyak stepsize)**
5. **Rely on bounded gradients (Adagrad, Adam, etc.)**

Our method

$$L_k = \frac{\|\nabla f(x^k) - \nabla f(x^{k-1})\|}{\|x^k - x^{k-1}\|}$$

Our method

$$L_k = \frac{\|\nabla f(x^k) - \nabla f(x^{k-1})\|}{\|x^k - x^{k-1}\|}$$

$$\lambda_k = \min\left\{\sqrt{1+\theta_{k-1}}\lambda_{k-1}, \frac{1}{2L_k}\right\}$$

Our method

$$\theta_{k-1} = \frac{\lambda_{k-1}}{\lambda_{k-2}}, \ \theta_0 = +\infty$$

$$L_k = \frac{\|\nabla f(x^k) - \nabla f(x^{k-1})\|}{\|x^k - x^{k-1}\|}$$

$$\lambda_k = \min\left\{\sqrt{1+\theta_{k-1}}\lambda_{k-1}, \frac{1}{2L_k}\right\}$$

Our method

$$\theta_{k-1} = \frac{\lambda_{k-1}}{\lambda_{k-2}}, \ \theta_0 = +\infty$$

$$L_k = \frac{\|\nabla f(x^k) - \nabla f(x^{k-1})\|}{\|x^k - x^{k-1}\|}$$

$$\lambda_k = \min\left\{\sqrt{1+\theta_{k-1}}\lambda_{k-1}, \frac{1}{2L_k}\right\}$$

$$\frac{1}{2L} \le \lambda_k \le \frac{1}{2\mu}$$

Key ideas

1. **Two-step analysis**

Key ideas

1. **Two-step analysis**
2. **Only use convexity**

Key ideas

1. **Two-step analysis**
2. **Only use convexity**
3. **Let the proof give you a method**

Lyapunov function

$$0 \leq \Psi^{k+1} \leq \Psi^k$$

Lyapunov function

$$\Psi^{k+1} = \|x^{k+1} - x^*\|^2$$

$$+ 2\lambda_k(1 + \theta_k)(f(x^k) - f(x^*)) + \frac{1}{2}\|x^{k+1} - x^k\|^2$$

Lyapunov function

$$\Psi^{k+1} = \|x^{k+1} - x^*\|^2 + 2\lambda_k(1+\theta_k)(f(x^k) - f(x^*)) + \frac{1}{2}\|x^{k+1} - x^k\|^2$$

$$\Psi^{k+1} \le \Psi^k + 2\lambda_k^2\|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 - \frac{1}{2}\|x^k - x^{k-1}\|^2 + 2(\lambda_k^2/\lambda_{k-1} - \lambda_{k-1}(1+\theta_{k-1}))(f(x^{k-1}) - f(x^*))$$

Lyapunov function

$$\Psi^{k+1} = \|x^{k+1} - x^*\|^2 + 2\lambda_k(1+\theta_k)(f(x^k) - f(x^*)) + \frac{1}{2}\|x^{k+1} - x^k\|^2$$

$$\Psi^{k+1} \leq \Psi^k + 2\lambda_k^2\|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 - \frac{1}{2}\|x^k - x^{k-1}\|^2 + 2(\lambda_k^2/\lambda_{k-1} - \lambda_{k-1}(1+\theta_{k-1}))(f(x^{k-1}) - f(x^*))$$

Lyapunov function

$$\Psi^{k+1} \leq \Psi^k$$

$$+ 2\lambda_k^2 \|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 - \frac{1}{2}\|x^k - x^{k-1}\|^2$$

$$+ 2(\lambda_k \theta_k - \lambda_{k-1}(1 + \theta_{k-1}))(f(x^k) - f(x^*))$$

Lyapunov function

$$\Psi^{k+1} \le \Psi^k$$

$$+ 2\lambda_k^2 \|\nabla f(x^k) - \nabla f(x^{k-1})\|^2 - \frac{1}{2}\|x^k - x^{k-1}\|^2$$

$$+ 2(\lambda_k \theta_k - \lambda_{k-1}(1 + \theta_{k-1}))(f(x^k) - f(x^*))$$

1. $\lambda_k \langle \nabla f(x^k), x^* - x^k \rangle \le \lambda_k (f(x^*) - f(x^k))$

2. $\lambda_k \theta_k \langle x^{k-1} - x^k, \nabla f(x^k) \rangle \le \lambda_k \theta_k (f(x^{k-1}) - f(x^k))$

Convergence

$$f(\hat{x}^k) - f(x^*) = \mathcal{O}\left(\frac{1}{\sum_{t=1}^{k} \lambda_t}\right) = \mathcal{O}\left(\frac{1}{k}\right)$$

(convex *f*)

Only local smoothness is needed

Convergence

$$f(\hat{x}^k) - f(x^*) = \mathcal{O}\left(\frac{1}{\sum_{t=1}^{k} \lambda_t}\right) = \mathcal{O}\left(\frac{1}{k}\right)$$

(convex *f*)

Only local smoothness is needed

Converges linearly under local strong convexity

Experiments

https://github.com/ymalitsky/adaptive_GD

Experiments: log. reg.

$$\frac{1}{n}\sum_{i=1}^{n}\log(1+\exp(-b_i a_i^\top x))+\frac{\gamma}{2}\|x\|^2$$

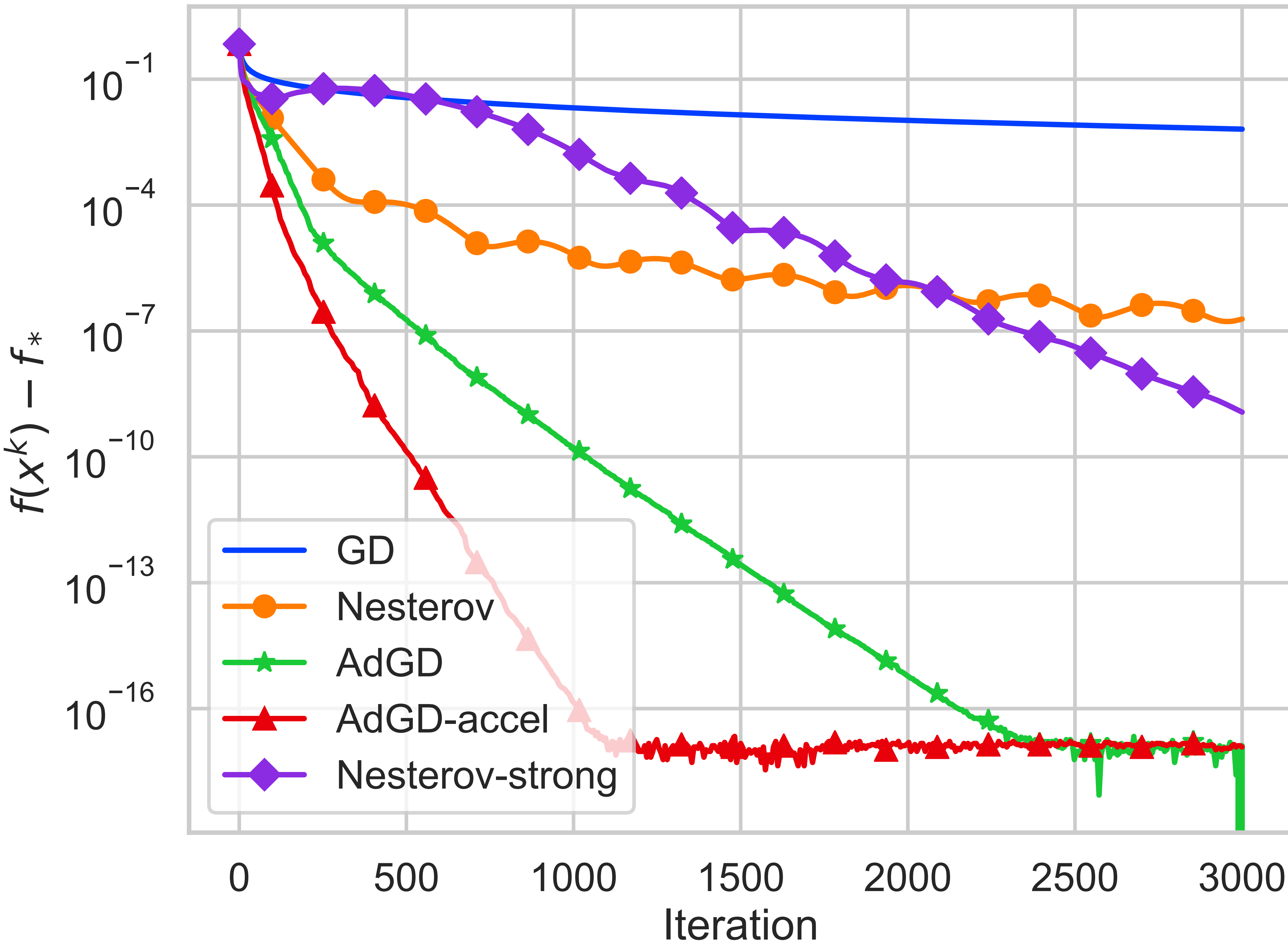
Experiments: log. reg.

Data

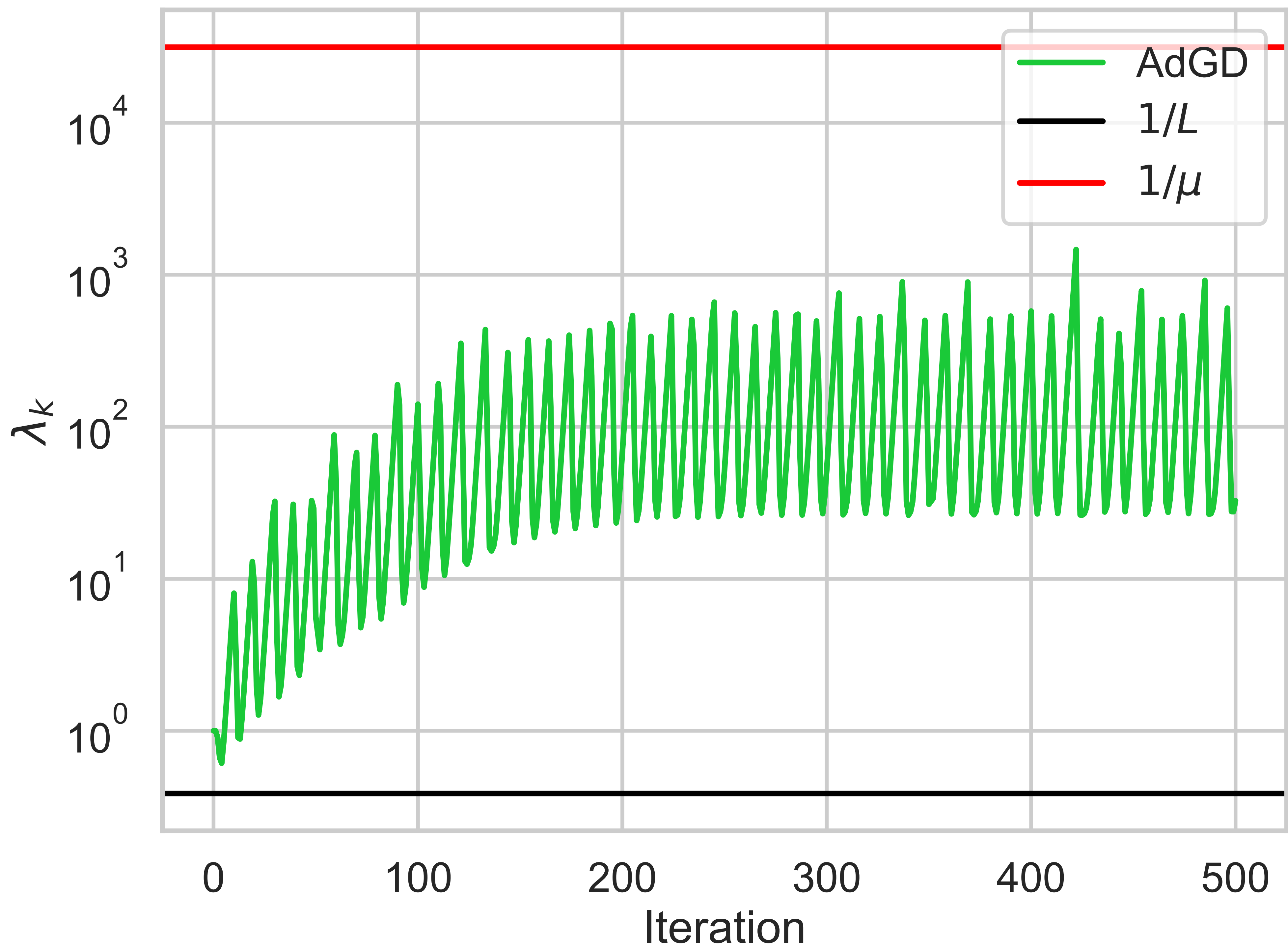
$$\frac{1}{n}\sum_{i=1}^{n}\log(1+\exp(-b_i a_i^\top x)) + \frac{\gamma}{2}\|x\|^2$$

Regularization

Experiments: log. reg.



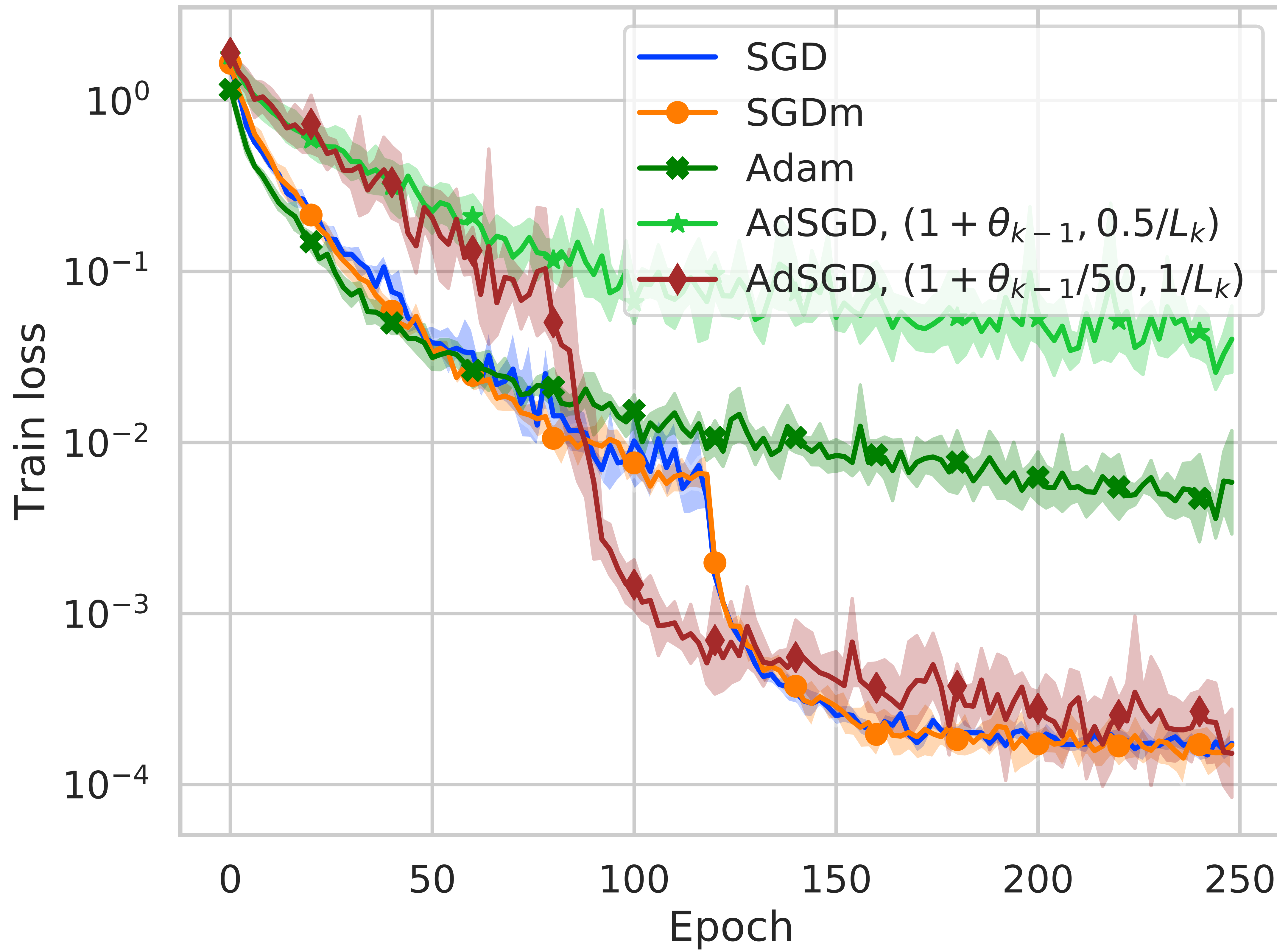
Experiments: log. reg.

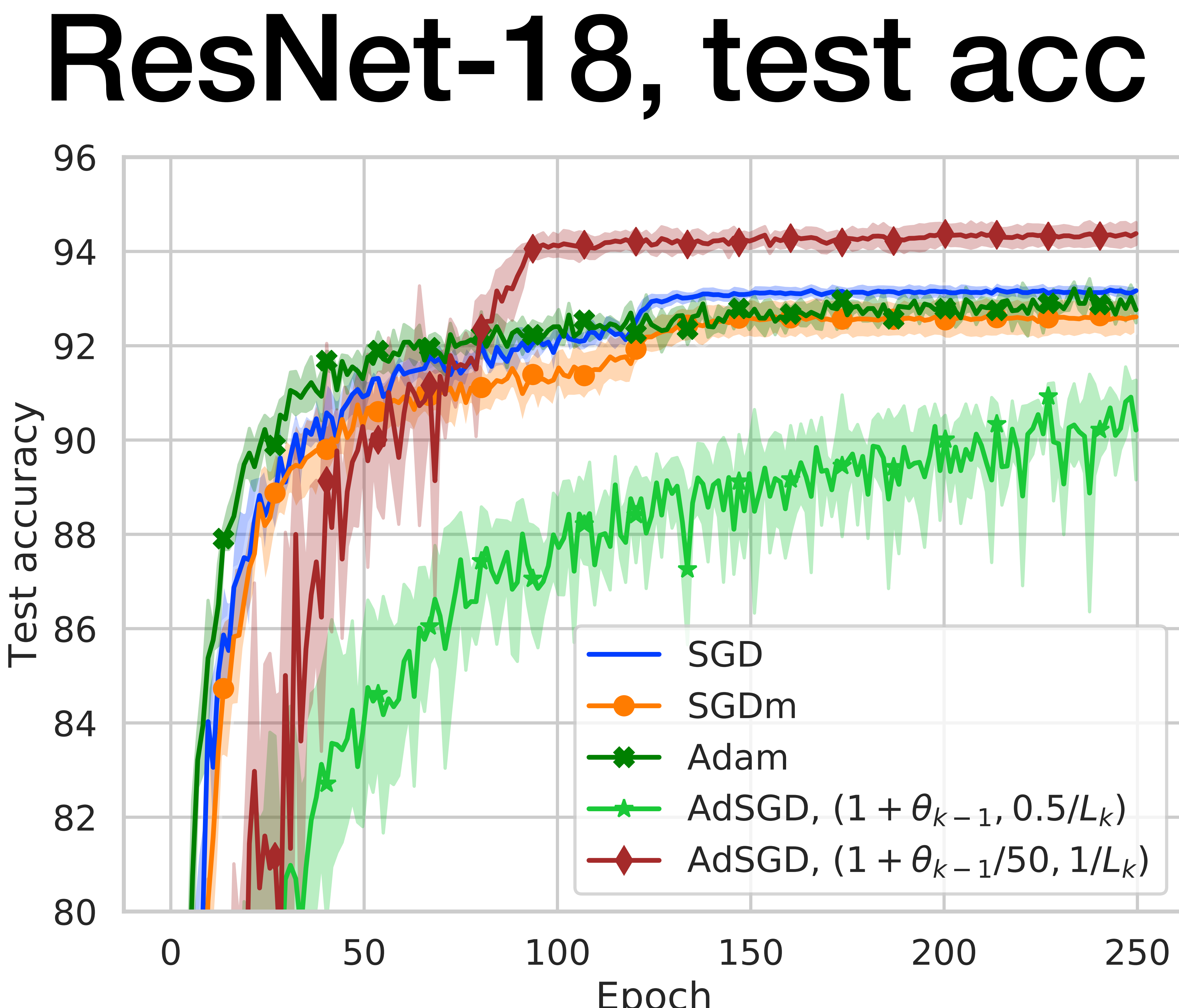


Neural nets, Cifar-10

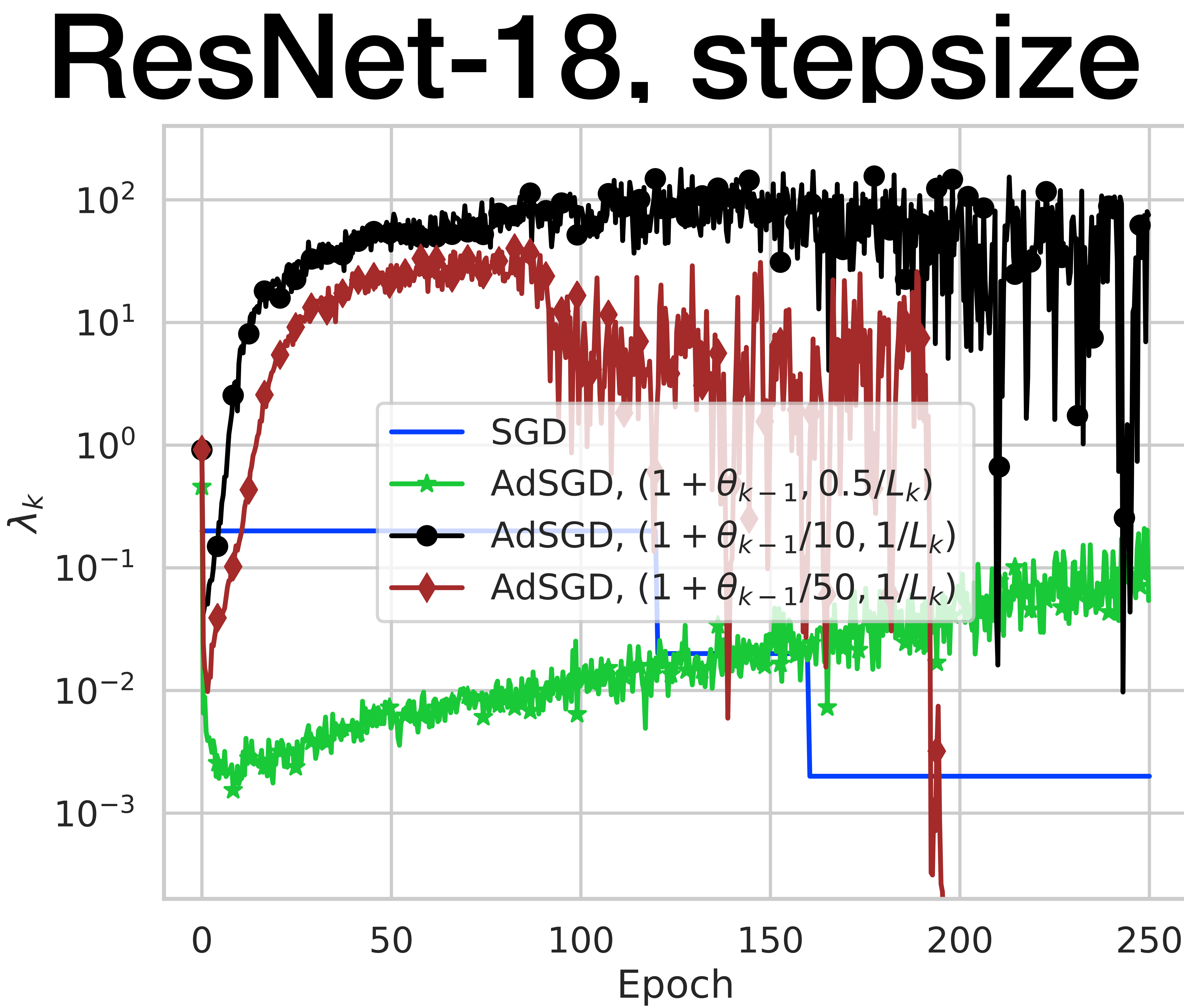
1. **Batch size = 128**
2. **No weight decay**
3. **Architectures for Cifar-10 from https://github.com/kuangliu/pytorch-cifar**
4. $$\lambda_k = \min\left\{\sqrt{1 + 0.02\theta_{k-1}}\lambda_{k-1}, \frac{1}{L_k}\right\}$$
5. **Each epoch is twice more expensive**

ResNet-18, train loss

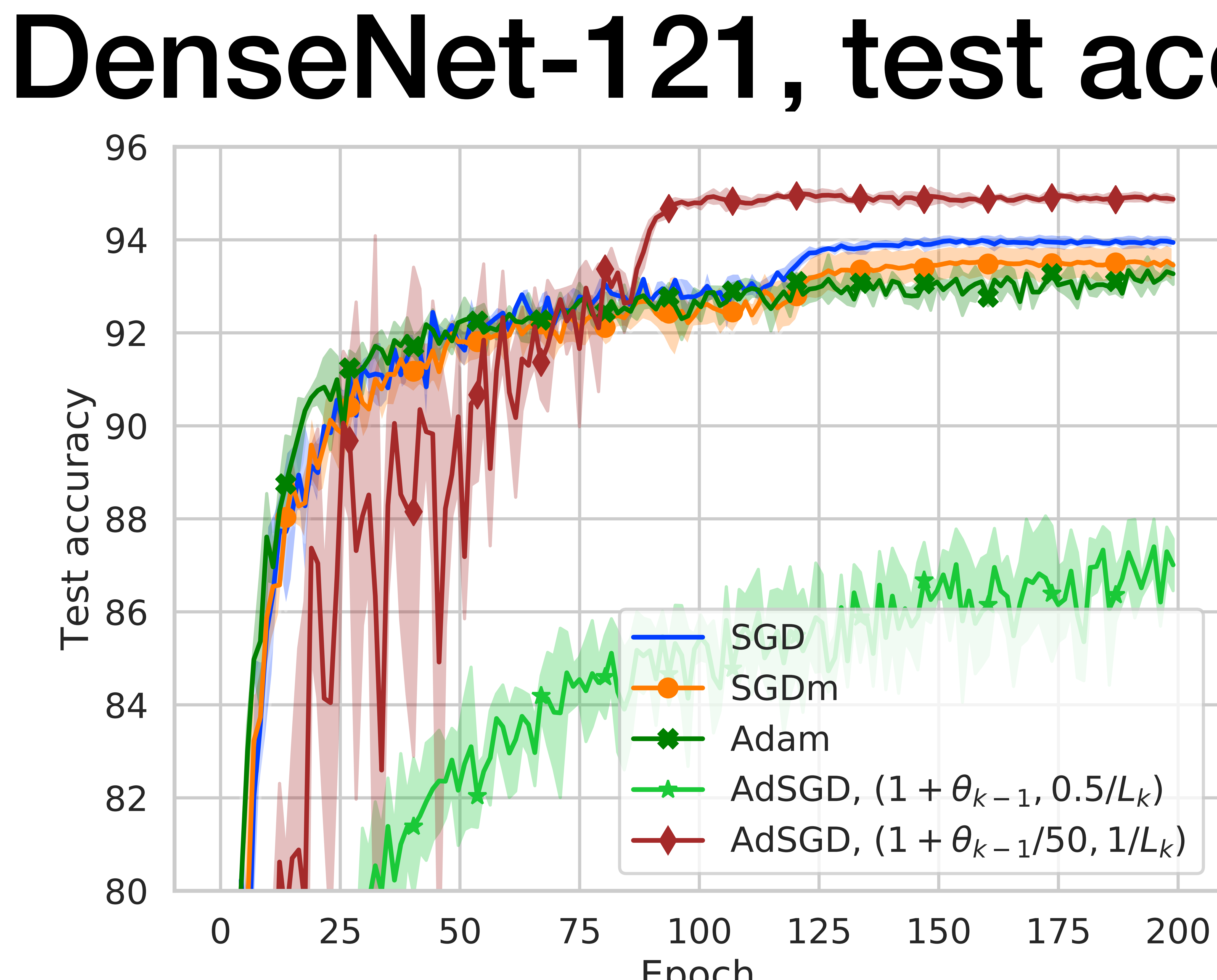




ResNet-18, test acc
Test accuracy
96
94
92
90
88
86
84
82
80
0
50
100
150
200
250
Epoch
SGD
SGDm
Adam
AdSGD, $(1+\theta_{k-1}, 0.5/L_k)$
AdSGD, $(1+\theta_{k-1}/50, 1/L_k)$



ResNet-18, stepsize
10^2
10^1
10^0
10^{-1}
10^{-2}
10^{-3}
λ_k
0
50
100
150
200
250
Epoch
SGD
AdSGD, $(1+\theta_{k-1}, 0.5/L_k)$
AdSGD, $(1+\theta_{k-1}/10, 1/L_k)$
AdSGD, $(1+\theta_{k-1}/50, 1/L_k)$


DenseNet-121, test acc
Test accuracy
Epoch
96
94
92
90
88
86
84
82
80
0
25
50
75
100
125
150
175
200
SGD
SGDm
Adam
AdSGD, $(1+\theta_{k-1}, 0.5/L_k)$
AdSGD, $(1+\theta_{k-1}/50, 1/L_k)$

More things in the paper

1. **Analysis for SGD**
2. **Discussion of estimating strong convexity**
3. **Experiments on matrix factorization problem**

arxiv:1910.09529