

Practical Regularized Quasi-Newton Methods with Inexact Function Values (不正確な関数値下における 実用的な正則化準ニュートン法)

浜口広樹

数理情報第5研究室

指導教員: 武田朗子教授

2026/1/26

無制約最適化問題と準ニュートン法の一種 L-BFGS

問題設定: $\min_{x \in \mathbb{R}^n} f(x)$ ($f: \mathbb{R}^n \rightarrow \mathbb{R}$ は連続可微分な非凸関数)

問題を「解く」: $\|\nabla f(x)\|$ が十分小さい値 ϵ_{gtol} 以下になる点の発見

L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shannon): SciPy (著名 OSS) にて高速な手法

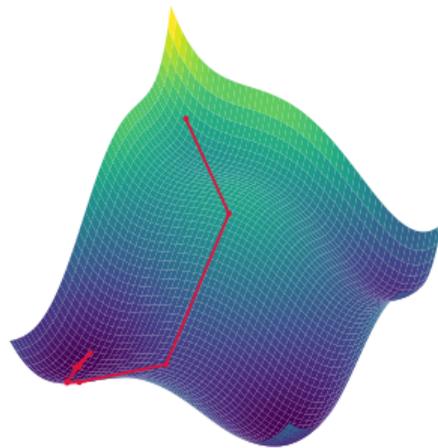


Figure: 非凸最適化の例

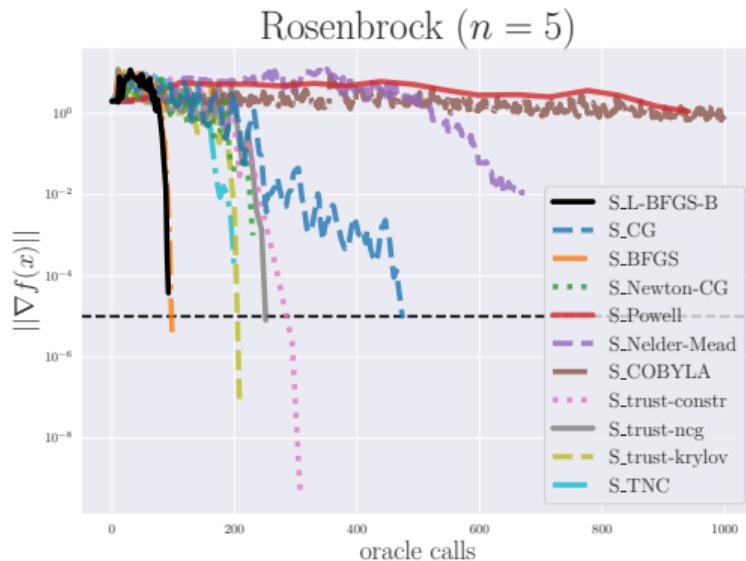


Figure: SciPy の最適化手法の比較 準ニュートン法が最速

問題意識: 数値誤差に対する脆弱性

- 理想的な状況下で、準ニュートン法は実用上最もよく使われる手法の一つ
- 現実の最適化では、計算機上で行うので数値誤差が発生
- 特に、関数値の誤差は準ニュートン法の挙動を著しく不安定化
 - ビッグデータを扱う際の低精度浮動小数点数 (32, 16bit) による計算や、近似値や確率で変動する値のみが計算出来る状況では、不可避な誤差が生じ本質的に解けない問題になる
 - ベンチマーク問題 [Gould et al., 2015] の 141/221 個が、デフォルト設定にて先に関数値誤差が無視できないレベルで大きくなる
- 従来的な理論保証は失われ、小手先の改善では解決しない問題も多い

本研究の主題

実用的な高速性を維持しつつ、特に関数値の数値誤差に頑健な準ニュートン法とは?

本研究の位置づけ

無視できない誤差が f にも ∇f にも存在すると、その最適化問題は **不良設定**

本研究は、特に 関数値 f の誤差が大きい場合 に着目

		f の誤差	
		小	大
∇f の誤差	小	通常最適化 [Liu and Nocedal, 1989, Virtanen et al., 2020]	本研究などの対象
	大	先行研究 [Shi et al., 2022] (いわゆる 0 次法に近い手法など)	(最適化不可能)

より緩い誤差への研究や [Byrd et al., 2012, Paquette and Scheinberg, 2020, Sun and Nocedal, 2023]、信頼領域法における研究 [Bellavia et al., 2021, Sun and Nocedal, 2023] なども存在

→ 本研究は、消すことが出来ない誤差 に対して 実用的な高速性を維持したい

研究概要

本研究の主題

実用的な高速性を維持しつつ、特に関数値の数値誤差に頑健な準ニュートン法とは?

先行研究

直線探索型準ニュートン法 [Liu and Nocedal, 1989, Virtanen et al., 2020] は高速だが脆弱

正則化手法 [Kanzow and Steck, 2023, Ueda and Yamashita, 2014] は代替案として近年注目

しかし、これらは数値誤差を考慮せず、また必ずしも高速ではなく、課題を残す

本研究の貢献

正則化準ニュートン法を用いながら、 f の数値誤差を陽に考慮する

誤差問題を回避するため、 f を使わない関数値非依存最適化を援用

数値誤差に耐性のある高速な準ニュートン法を提案

具体的な問題設定と誤差モデル

現実的な IEEE 754 浮動小数点数を想定した誤差モデル \bar{f} のみが観測可能。

関数値に対するハイブリッド型の誤差モデル

絶対誤差と相対誤差が混在する誤差モデル [Higham, 2002, Izycheva and Darulova, 2017]:

$$\left| f(x) - \bar{f}(x) \right| \leq \epsilon_f \max(1, |f(x)|).$$

停止条件は $\left\| \nabla \bar{f}(x) \right\| \leq \epsilon_{\text{gtol}}$ なので、過程では常に $\left\| \nabla \bar{f}(x) \right\| > \epsilon_{\text{gtol}}$ が成立する。

勾配に対する誤差モデル: 誤差が ϵ_{gtol} より十分小さい

$$\text{(数値実験)} \left\| \nabla f(x) - \nabla \bar{f}(x) \right\| \ll \epsilon_{\text{gtol}}, \quad \text{(理論解析)} \left\| \nabla f(x) - \nabla \bar{f}(x) \right\| = 0.$$

キーワード: 目的関数値非依存最適化 [Gratton and Toint, 2023]

機械学習分野での需要などから近年発展してきた手法。

ADAM や ADA GRAD などの最適化手法は、勾配だけで最適化を行う:

$$\text{(ADAM)} \quad x_{k+1}^{(i)} = x_k^{(i)} - \frac{1}{\theta \sqrt{\varsigma + \sum_{j=0}^k (\nabla f(x_j)^{(i)})^2}} \nabla f(x_k)^{(i)},$$

$$\text{(ADAGRAD)} \quad x_{k+1} = x_k - \frac{1}{\theta \sqrt{\varsigma + \sum_{j=0}^k \|\nabla f(x_j)\|^2}} \nabla f(x_k).$$

(θ は学習率, ς は初期化定数, $^{(i)}$ は第 i 成分)

このような関数値 f を陽に用いない最適化手法の総称

→ 誤差に影響されやすい関数値差分を利用せずに最適化が可能に

提案手法の構成要素 1/2: 緩和 Armijo 条件

通常の Armijo 条件 (従来のエラーが起きる直線探索の終了条件):

$$\bar{f}(x_k) + c\alpha_k \nabla f(x_k)^\top d_k \geq \bar{f}(x_k + \alpha_k d_k).$$

本研究は以下の緩和 Armijo 条件を採用:

$$\begin{cases} \bar{f}(x_k) + c\alpha_k \nabla f(x_k)^\top d_k + \Delta_k \geq \bar{f}(x_k + \alpha_k d_k), \\ \Delta_k = \frac{2\epsilon_f}{1 - \epsilon_f} \max(1, \bar{f}(x_k), -\bar{f}(x_k + \alpha_k d_k)). \end{cases}$$

この Δ_k は (適切な条件下で) 次を満たす、誤差吸収項:

$$\underbrace{|\bar{f}(x_k) - \bar{f}(x_k + \alpha_k d_k)|}_{\text{観測された差分}} - \underbrace{|f(x_k) - f(x_k + \alpha_k d_k)|}_{\text{真の差分}} \leq \Delta_k.$$

→ 現実的な誤差ありの設定でも、エラーなく最適化を進められる。

提案手法の構成要素 2/2: 正則化を入れた探索方向

探索方向は正則化付き準ニュートン方向を採用:

$$d_k = -(B_k + \mu_k I)^{-1} \nabla f(x_k).$$

(直感: f の勾配情報による近似 B_k + 正則化項 $\mu_k I$ を加えた二次形式の最小解方向)

$\mu_k = 0$: 直線探索型とほぼ同じ挙動を意味、速度の観点からは望ましい

$\mu_k > 0$: 保守的なステップになり、安定性向上、ADAGRAD-NORM に似た正則化:

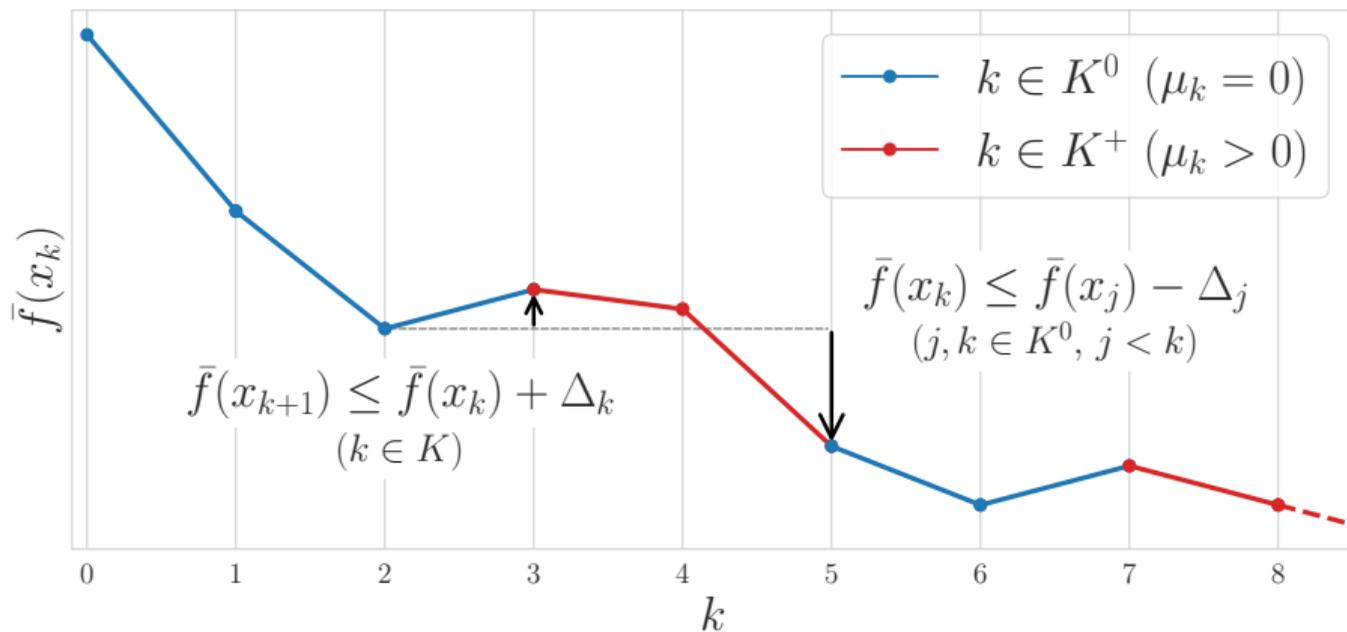
$$\mu_k = \theta_k \sqrt{\varsigma + \sum_{j \in K^+} \|\nabla f(x_j)\|^2}, \quad \theta_k \in [\theta_{\min}, \theta_{\max}].$$

$B_k \approx \nabla^2 f(x_k)$ のとき、 $\mu_k > 0$ は二次近似を壊し、過度に保守的になる恐れがある。
十分に \bar{f} が減少した試行では、 $\mu_k = 0$ を試すことで、速度と安定性の両立を図る。

本質的な設計思想の図示

直線探索で \bar{f} の上昇を許し、エラーを回避する (図左側)

$\mu_k = 0$ を優先的に試すが、十分に \bar{f} が減少した試行に限ることで、安定化 (図右側)



$$K^0 := \{k \in K \mid \mu_k = 0\}, \quad K^+ := \{k \in K \mid \mu_k > 0\}.$$

理論解析における仮定

以下では、提案手法が適切な仮定の下で収束することを示す。

最適化問題は、次の非常に標準的な2つの仮定を満たすとする:

仮定 1: 目的関数 f の下側に関する有界性

$$f(x) \geq f^* := \inf_{x \in \mathbb{R}^n} f(x) > -\infty.$$

仮定 2: 目的関数 f の L -smooth 性

$$\exists 0 < L < \infty \quad \text{s.t.} \quad f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{L}{2} \|y - x\|^2.$$

この設定の最適な反復計算量 $\min\{k \mid \|\nabla f(x_k)\| \leq \epsilon_{\text{gtol}}\}$ は $\Omega(\epsilon_{\text{gtol}}^{-2})$ である。

収束解析の概要: コアとなる不等式

$K^0 = \{k \in K \mid \mu_k = 0\}$, $K^+ = \{k \in K \mid \mu_k > 0\}$, (Armijo 定数) c , (smooth 性) L .

勾配ノルム和に対する複合的な上界評価

仮定 1,2 の下で、 $F = \mathcal{O}(f(x_0) - f^*)$, 定数 C_0, C_1, C_2 を用いて

$$F \geq C_0 \sum_{k \in K^0} \|\nabla f(x_k)\|^2 + C_1 \sqrt{c + \sum_{k \in K^+} \|\nabla f(x_k)\|^2} - C_2 \ln\left(c + \sum_{k \in K^+} \|\nabla f(x_k)\|^2\right).$$

K^0 : 正則化のない通常ステップ

古典的な勾配法型の解析結果と一致:

$$\sum_{k \in K^0} \|\nabla f(x_k)\|^2 \lesssim \frac{1}{C_0} F.$$

K^+ : 正則化が入る保守的ステップ

$\sqrt{\cdot} \gg \ln \cdot$ より、 $\sqrt{\cdot}$ の項が支配的:

$$\sum_{k \in K^+} \|\nabla f(x_k)\|^2 \lesssim \left(\frac{2F}{C_1}\right)^2.$$

大域収束率の結論

主結果

仮定 1,2 の下で、任意の反復回数 k_{\max} に対し、提案手法は以下を満たす:

$$\sum_{j=0}^{k_{\max}} \|\nabla f(x_j)\|^2 \leq \text{const.} \implies \frac{1}{k_{\max} + 1} \sum_{j=0}^{k_{\max}} \|\nabla f(x_j)\|^2 = \mathcal{O}\left(\frac{1}{k_{\max}}\right).$$

提案手法が達成する反復計算量

$$\min\{k \mid \|\nabla f(x_k)\| \leq \epsilon_{\text{gtol}}\} = \mathcal{O}\left(\epsilon_{\text{gtol}}^{-2}\right).$$

誤差がある設定でも、非凸問題に対する最適な反復計算量とオーダーレベルで一致
具体的には、古典的な勾配法型の解析結果と ADA GRAD-NORM 型の解析結果
のハイブリッドになっている

実験設定

比較手法

Ours: 提案手法 (Ours-MS は一種の改良版)

Line: Wolfe 条件による直線探索型 (Line-MS も同様の変種)

SciPy: 著名 OSS である SciPy に実装された L-BFGS-B 法

Reg: 近年の正則化 L-BFGS 法 [[Kanzow and Steck, 2023](#)]

NTQN: 冒頭で述べた 0 次法に類似したノイズ耐性準ニュートン法 [[Shi et al., 2022](#)]

問題設定

CUTEst 無制約問題

非凸問題を含む有名ベンチマーク

最大 220 問題

評価手法: Performance Profile

どれだけ安定かつ高速に解けたかを評価

x 軸方向に左であればあるほど高速

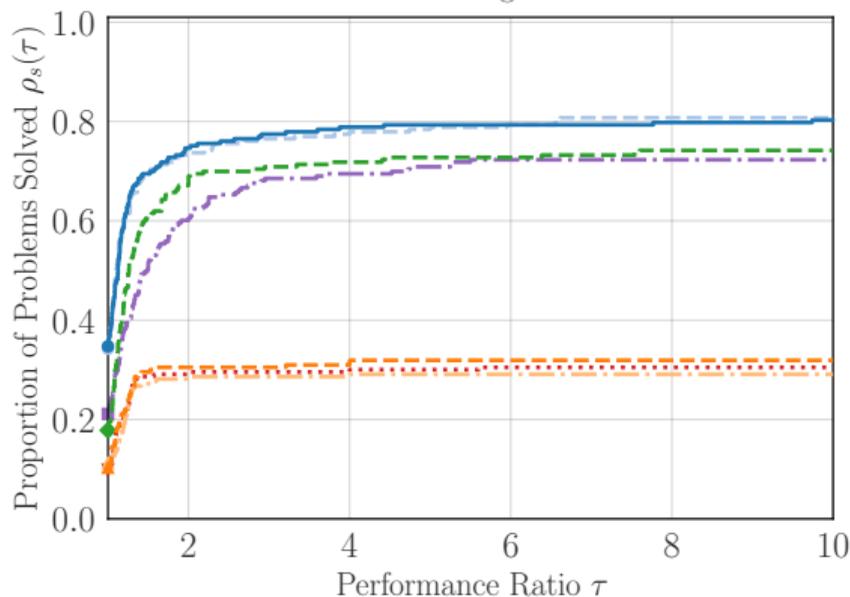
y 軸方向に上であればあるほど安定

人工ノイズ下での性能

ノイズモデル [Shi et al., 2022] (unif は一様分布ノイズ)

$$\bar{f}(x) = f(x) + \text{unif}(-10^{-3}, 10^{-3}), \quad \nabla \bar{f}(x) = \nabla f(x) + \text{unif}(-10^{-3}, 10^{-3})\mathbb{1}$$

noise= 10^{-3} , $\epsilon_{\text{gtol}} = 10^{-2}$



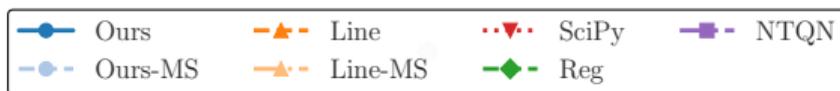
提案法が最も高い成功率 (安定性)

最速で解いた数 ($\tau = 1$) も最多 (高速性)

先行研究 (NTQN)[Shi et al., 2022] は勾配ノイズに特に強いアルゴリズム

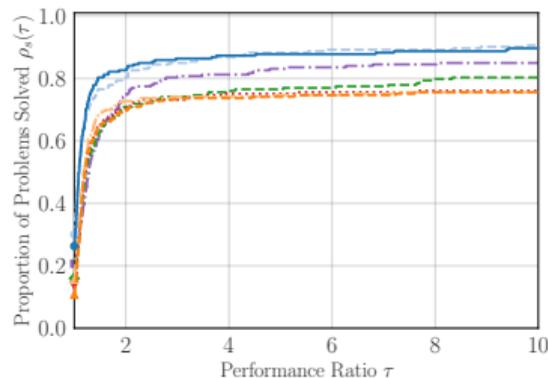
我々の手法も勾配ノイズに耐性

ノイズレベルを変えても同様の傾向

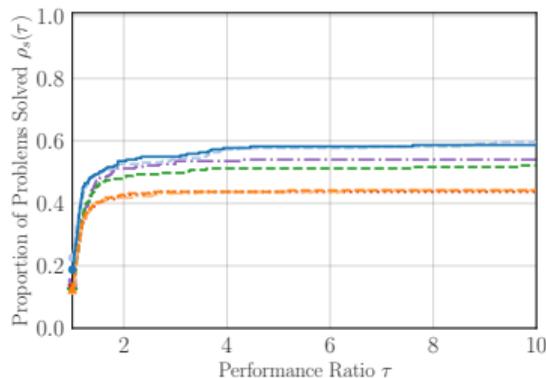


有限精度計算に対する頑健性

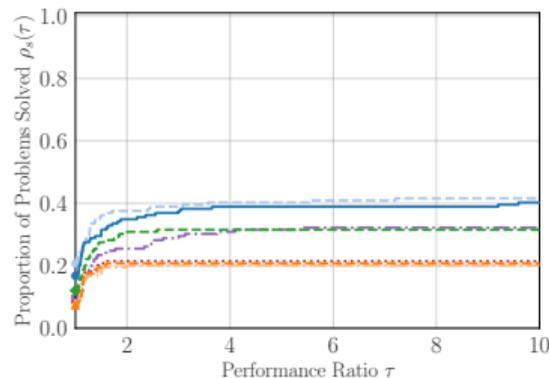
64-bit $\epsilon_{\text{gtol}} = 10^{-5}$



32-bit $\epsilon_{\text{gtol}} = 10^{-4}$



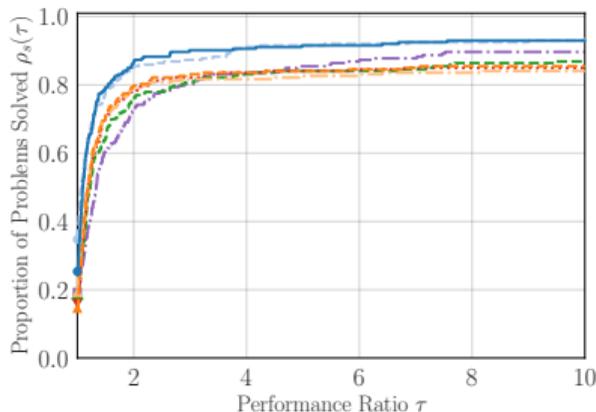
16-bit $\epsilon_{\text{gtol}} = 10^{-3}$



精度低下に伴い既存法は不安定化 提案手法 (Ours) は一貫して高い成功率

誤差なし設定からみる実用性

誤差の殆ど関係ない設定:
64-bit $\epsilon_{\text{gtol}} = 10^{-3}$



最速で解けた問題数が、
2倍近くに達している

提案手法は誤差なし設定でも高速
(最速で解けている問題数が SciPy の約 2 倍)

既存の理論的に良い正則化に類似点 ($c_1, c_2, \delta > 0$):

$$\mu_k = c_1 \max(0, -\lambda_{\min}(B_k)) + c_2 \|\nabla f(x_k)\|^\delta.$$

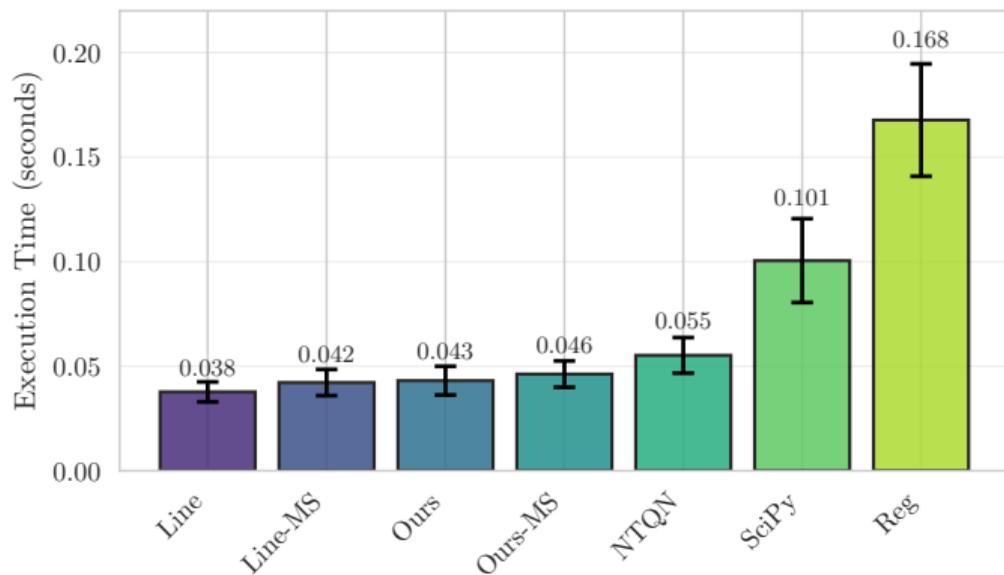
近似ヘッセ行列 B_k に既存の工夫も活用:

$$\text{BFGS}(\{(s_{k_i}, \bar{y}_{k_i} := \theta_i^{\text{damp}} y_{k_i} + (1 - \theta_i^{\text{damp}}) B_{k-1} s_{k_i})\}).$$

(BFGS(\cdot) は更新則の関数、 s_{k_i}, y_{k_i} は差分ベクトル)

実行時間からみる実用性

Execution Time Comparison ($n = 10000$), 100 Runs



左図は大まかにアルゴリズム自体 (f や ∇f を除く) の計算時間。
正則化型の手法

$$d_k = -(B_k + \mu_k I)^{-1} \nabla f(x_k)$$

は行列計算に時間を要する。
提案は既存の近似手法を採用。
それでも性能は落ちない。

(近似ヘッセ行列)

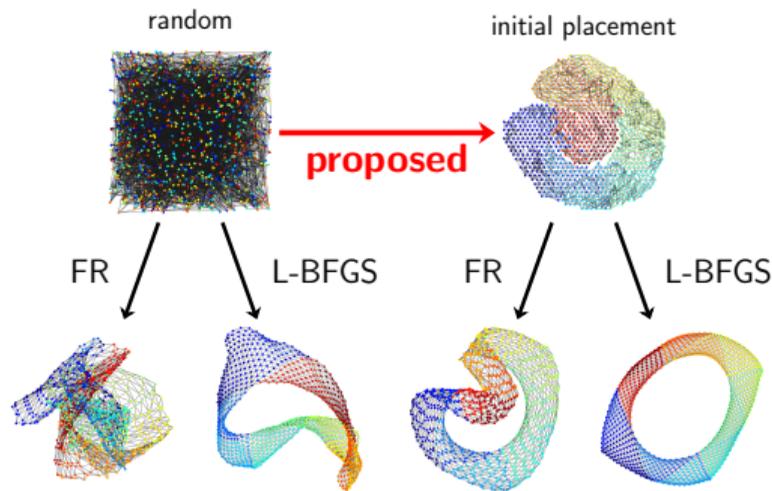
$$B_k \approx \text{BFGS}(\{s_{k_i}, \bar{y}_{k_i} + \mu_k s_{k_i}\}_{i=0}^{p-1}) - \mu_k I,$$

(探索方向)

$$d_k = -(B_k + \mu_k I)^{-1} \nabla f(x_k) = -\text{BFGS}(\dots)^{-1} \nabla f(x_k).$$

余談: 修士1年での研究内容との関連

研究概要図



NetworkX へ果たした貢献

Spring Layout

Draw graphs using the three different spring layout algorithms.

The spring layout is typically generated by the Fruchterman–Reingold force-directed algorithm. This algorithm treats edges as springs holding nodes close while treating nodes as repelling objects and simulates the system until it reaches equilibrium. The algorithm also terminates when it reaches a maximum number of iterations.

NetworkX offers mainly three different kinds of methods based on the same theoretical foundation:

- `nx.spring_layout(G, method="force")`
 - The default for graphs with fewer than 500 nodes in `nx.spring_layout`.
 - Direct implementation of the Fruchterman–Reingold force-directed algorithm.
 - Can handle negative edge weights as they are.
- `nx.spring_layout(G, method="energy")`
 - The default for graphs with more than or equal to 500 nodes in `nx.spring_layout`.
 - It solves an energy-based optimization problem, taking the absolute value of negative edge weights.
 - Uses gravitational forces acting on each connected component to prevent divergence.
- `nx.nx_agraph.graphviz_layout(G, prog="fdp")`
 - Uses `fdp` from GraphViz to compute the layout.
 - Employs a multilevel approach for faster force-directed graph drawing.
 - Requires separate installation of GraphViz. For details, see `networkx.drawing.nx_agraph`

連続最適化を用いたグラフ描画の研究:

“Initial Placement for Fruchterman–Reingold Force Model with Coordinate Newton Direction”

L-BFGS 法は、そのグラフ描画においても高速で有用な手法。

NetworkX に関連した実装を行い、`nx.draw` 関数を改善。

SciPy にも似たような貢献が出来ればと考えている。

まとめと今後の課題

まとめ

数値誤差を陽に考慮した正則化付き準ニュートン法を提案

緩和 Armijo 条件により関数値誤差を吸収し安全に更新

正則化パラメータ $\mu_k = 0$ と $\mu_k > 0$ の切替で高速性と安定性を両立

標準仮定の下で、非凸における最適レートと同じオーダーの収束率 $\mathcal{O}(\epsilon^{-2})$ を証明

既存の人工設定 + 有限精度環境の CUTEst データセットで安定性と高速性を確認

今後の課題

勾配にも無視できない誤差がある場合との統合、理論解析の強化

Reference I

- Stefania Bellavia, Gianmarco Gurioli, Benedetta Morini, and Philippe L. Toint. The Impact of Noise on Evaluation Complexity: The Deterministic Trust-Region Case, April 2021.
- Richard H. Byrd, Gillian M. Chin, Jorge Nocedal, and Yuchen Wu. Sample size selection in optimization methods for machine learning. *Mathematical Programming*, 134(1):127–155, August 2012. ISSN 1436-4646. doi: 10.1007/s10107-012-0572-5.
- Nicholas I. Gould, Dominique Orban, and Philippe L. Toint. CUTEst: A Constrained and Unconstrained Testing Environment with safe threads for mathematical optimization. *Comput. Optim. Appl.*, 60(3):545–557, April 2015. ISSN 0926-6003. doi: 10.1007/s10589-014-9687-3.
- S. Gratton and Ph L. Toint. OFFO minimization algorithms for second-order optimality and their complexity. *Computational Optimization and Applications*, 84(2):573–607, March 2023. ISSN 1573-2894. doi: 10.1007/s10589-022-00435-2.
- Nicholas J. Higham. *Accuracy and Stability of Numerical Algorithms*. Society for Industrial and Applied Mathematics, January 2002. ISBN 978-0-89871-521-7. doi: 10.1137/1.9780898718027.
- Anastasiia Izycheva and Eva Darulova. On sound relative error bounds for floating-point arithmetic. In *Proceedings of the 17th Conference on Formal Methods in Computer-Aided Design*, Austin, Texas, 2017. FMCAD Inc. ISBN 978-0-9835678-7-5.
- Christian Kanzow and Daniel Steck. Regularization of limited memory quasi-Newton methods for large-scale nonconvex minimization. *Mathematical Programming Computation*, 15(3):417–444, September 2023. ISSN 1867-2957. doi: 10.1007/s12532-023-00238-4.

Reference II

- Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1):503–528, August 1989. ISSN 1436-4646. doi: 10.1007/BF01589116.
- Courtney Paquette and Katya Scheinberg. A Stochastic Line Search Method with Expected Complexity Analysis. *SIAM Journal on Optimization*, 30(1):349–376, January 2020. ISSN 1052-6234. doi: 10.1137/18M1216250.
- Hao-Jun M. Shi, Yuchen Xie, Richard Byrd, and Jorge Nocedal. A Noise-Tolerant Quasi-Newton Algorithm for Unconstrained Optimization. *SIAM Journal on Optimization*, 32(1):29–55, March 2022. ISSN 1052-6234. doi: 10.1137/20M1373190.
- Shigeng Sun and Jorge Nocedal. A trust region method for noisy unconstrained optimization. *Mathematical Programming*, 202(1):445–472, November 2023. ISSN 1436-4646. doi: 10.1007/s10107-023-01941-9.
- Kenji Ueda and Nobuo Yamashita. A regularized Newton method without line search for unconstrained optimization. *Computational Optimization and Applications*, 59(1):321–351, October 2014. ISSN 1573-2894. doi: 10.1007/s10589-014-9656-x.
- Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, Ilhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods*, 17: 261–272, 2020. doi: 10.1038/s41592-019-0686-2.