Self-Correcting Variable-Metric Algorithms for Nonsmooth Optimization

Frank E. Curtis, Lehigh University

joint work with

Daniel P. Robinson, Johns Hopkins University

International Conference on Continuous Optimization (ICCOPT)
Tokyo, Japan

8 August 2016

Outline

Outline

Contribution

Self-Correcting Properties of BFGS-type Updating

Proposed Framework

Numerical Experiments

Summary

Nonsmooth optimization

Consider unconstrained optimization problems of the form

$$\min_{x \in \mathbb{R}^n} \; f(x),$$

where f is

- locally Lipschitz in $\mathbb{R}^n$ and
- differentiable in an open, dense subset of $\mathbb{R}^n$,

but

- nonsmooth and (potentially) nonconvex.

Balance between first- and second-order methods

For deterministic, smooth optimization, a nice balance achieved by quasi-Newton:

$$x_{k+1} \leftarrow x_k - \alpha_k W_k g_k,$$

where

- $\alpha_k > 0$ is a stepsize;
- $g_k \leftarrow \nabla f(x_k)$;
- $\{W_k\}$ is updated dynamically.

We all know:

- local rescaling based on iterate/subgradient displacements
- only first-order derivatives required
- no linear system solves required
- global convergence guarantees (say, with line search)
- superlinear local convergence rate

How can we carry these ideas to nonsmooth settings?

What has been done?

Many have observed improved performance with quasi-Newton schemes

"Unadulterated" BFGS

- Lemaréchal (1982)
- Lewis, Overton (2012)

BFGS (with restricted updates)

- Haarala, Miettinen, Mäkelä (2004)
- Curtis, Que (2015)

Issue: global convergence guarantees muddled by

- "Hessian" approximations† tending to singularity
- intertwined $\{x_k\}$, $\{\alpha_k\}$, $\{g_k\}$, and $\{W_k\}$

To our knowledge, none have tried to exploit self-correcting properties of BFGS

† "Hessian" and "inverse Hessian" used loosely in nonsmooth settings

Contribution

Propose a quasi-Newton method for nonsmooth optimization

- unifying framework covering
  - cutting plane / bundle methods (convex only)
  - gradient sampling methods (nonconvex)
- exploit self-correcting properties of BFGS-type updates
  - Powell (1976)
  - Ritter (1979, 1981)
  - Werner (1978)
  - Byrd, Nocedal (1989)
- properties of Hessians offer useful bounds for inverse Hessians
- global convergence guarantees
- improved practical performance

Remember: Forget about superlinear convergence (not relevant here!)

Outline

BFGS-type updates

Inverse Hessian and Hessian approximation updating formulas ($s_k^T v_k > 0$):

$$W_{k+1} \leftarrow \left(I - \frac{v_k s_k^T}{s_k^T v_k}\right)^T W_k \left(I - \frac{v_k s_k^T}{s_k^T v_k}\right) + \frac{s_k s_k^T}{s_k^T v_k}$$

$$H_{k+1} \leftarrow \left(I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k}\right)^T H_k \left(I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k}\right) + \frac{v_k v_k^T}{s_k^T v_k}$$

- These satisfy secant-type equations

$$W_{k+1} v_k = s_k \quad \text{and} \quad H_{k+1} s_k = v_k,$$

  but these are not relevant for this talk.
- Choosing $v_k \leftarrow y_k := g_{k+1} - g_k$ yields standard BFGS, but we consider

$$v_k \leftarrow \beta_k s_k + (1 - \beta_k)\tilde{y}_k \quad \text{for some} \quad \beta_k \in [0, 1] \quad \text{and} \quad \tilde{y}_k \in \mathbb{R}^n.$$

  This scheme is important to preserve self-correcting properties.

Geometric properties of Hessian update: Burke, Lewis, Overton (2007)

Consider the matrices (which only depend on s_k and H_k, not g_k!)

$$P_k := \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \quad \text{and} \quad Q_k := I - P_k.$$

Both H_k-orthogonal projection matrices (i.e., idempotent and H_k-self-adjoint).

- P_k yields H_k-orthogonal projection onto span(s_k).
- Q_k yields H_k-orthogonal projection onto span$(s_k)^{\perp_{H_k}}$.

Geometric properties of Hessian update: Burke, Lewis, Overton (2007)

Consider the matrices (which only depend on s_k and H_k, not g_k!)

$$P_k := \frac{s_k s_k^T H_k}{s_k^T H_k s_k} \quad \text{and} \quad Q_k := I - P_k.$$

Both H_k-orthogonal projection matrices (i.e., idempotent and H_k-self-adjoint).

- P_k yields H_k-orthogonal projection onto $\text{span}(s_k)$.
- Q_k yields H_k-orthogonal projection onto $\text{span}(s_k)^{\perp_{H_k}}$.

Returning to the Hessian update:

$$H_{k+1} \leftarrow \underbrace{\left(I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k}\right)^T H_k \left(I - \frac{s_k s_k^T H_k}{s_k^T H_k s_k}\right)}_{\text{rank } n-1} + \underbrace{\frac{v_k v_k^T}{s_k^T v_k}}_{\text{rank } 1}$$

- Curvature projected out along $\text{span}(s_k)$
- Curvature corrected by $\frac{v_k v_k^T}{s_k^T v_k} = \left(\frac{v_k v_k^T}{\|v_k\|_2^2}\right)\left(\frac{\|v_k\|_2^2}{v_k^T W_{k+1} v_k}\right)$ (inverse Rayleigh).

Self-correcting properties of Hessian update

Since curvature is constantly projected out, what happens after many updates?

Self-correcting properties of Hessian update

Since curvature is constantly projected out, what happens after many updates?

Theorem 1 (Byrd, Nocedal (1989))

Suppose that, for all k, there exists $\{\eta, \theta\} \subset \mathbb{R}_{++}$ such that

$$\eta \leq \frac{s_k^T v_k}{\|s_k\|_2^2} \quad \textit{and} \quad \frac{\|v_k\|_2^2}{s_k^T v_k} \leq \theta. \tag{$\star$}$$

Then, for any $p \in (0,1)$, there exist constants $\{\iota, \kappa, \lambda\} \subset \mathbb{R}_{++}$ such that, for any $K \geq 2$, the following relations hold for at least $\lceil pK \rceil$ values of $k \in \{1, \dots, K\}$:

$$\iota \leq \frac{s_k^T H_k s_k}{\|s_k\|_2 \|H_k s_k\|_2} \quad \textit{and} \quad \kappa \leq \frac{\|H_k s_k\|_2}{\|s_k\|_2} \leq \lambda.$$

Proof technique.

Building on work of Powell (1976), involves bounding growth of

$$\gamma(H_k) = \mathrm{tr}(H_k) - \ln(\det(H_k)).$$

Self-correcting properties of inverse Hessian update

Rather than focus on superlinear convergence results, we care about the following.

Corollary 2

Suppose the conditions of Theorem 1 hold. Then, for any $p \in (0,1)$, there exist constants $\{\mu, \nu\} \subset \mathbb{R}_{++}$ such that, for any $K \geq 2$, the following relations hold for at least $\lceil pK \rceil$ values of $k \in \{1, \ldots, K\}$:

$$\mu \|\bar{g}_k\|_2^2 \leq \bar{g}_k^T W_k \bar{g}_k \quad \textit{and} \quad \|W_k \bar{g}_k\|_2^2 \leq \nu \|\bar{g}_k\|_2^2$$

Here $\bar{g}_k$ is the vector such that the iterate displacement is

$$x_{k+1} - x_k = s_k = -W_k \bar{g}_k$$

Proof sketch.

Follows simply after algebraic manipulations from the result of Theorem 1, using the facts that $s_k = -W_k \bar{g}_k$ and $W_k = H_k^{-1}$ for all k.

Outline

Subproblems in nonsmooth optimization algorithms

With sets of points, scalars, and (sub)gradients

$$\{x_{k,j}\}_{j=1}^{m}, \quad \{f_{k,j}\}_{j=1}^{m}, \quad \{g_{k,j}\}_{j=1}^{m},$$

nonsmooth optimization methods involve the primal subproblem

$$\begin{aligned} \min_{x \in \mathbb{R}^n} \ & \left(\max_{j \in \{1,\dots,m\}} \{f_{k,j} + g_{k,j}^T(x - x_{k,j})\} + \tfrac{1}{2}(x - x_k)^T H_k (x - x_k) \right) \\ \text{s.t. } & \|x - x_k\| \leq \delta_k, \end{aligned} \tag{P}$$

but, with $G_k \leftarrow [g_{k,1} \ \cdots \ g_{k,m}]$, it is typically more efficient to solve the dual

$$\begin{aligned} \sup_{(\omega,\gamma) \in \mathbb{R}_+^m \times \mathbb{R}^n} \ & -\tfrac{1}{2}(G_k\omega + \gamma)^T W_k (G_k\omega + \gamma) + b_k^T \omega - \delta_k \|\gamma\|_* \\ \text{s.t. } & \mathbb{1}_m^T \omega = 1. \end{aligned} \tag{D}$$

The primal solution can then be recovered by

$$x_k^* \leftarrow x_k - W_k \underbrace{(G_k\omega_k + \gamma_k)}_{\tilde{g}_k}.$$

Algorithm Self-Correcting BFGS for Nonsmooth Optimization

Choose $x_1 \in \mathbb{R}^n$.
Choose a symmetric positive definite $W_1 \in \mathbb{R}^{n \times n}$.
Choose $\alpha \in (0, 1)$
for $k = 1, 2, \ldots$ **do**
  Solve (P)–(D) such that setting

$$\begin{aligned} G_k &\leftarrow \begin{bmatrix} g_{k,1} & \cdots & g_{k,m} \end{bmatrix}, \\ s_k &\leftarrow -W_k(G_k\omega_k + \gamma_k), \\ \text{and} \quad x_{k+1} &\leftarrow x_k + s_k \end{aligned}$$

  yields

$$f(x_{k+1}) \leq f(x_k) - \tfrac{1}{2}\alpha(G_k\omega_k + \gamma_k)^T W_k (G_k\omega_k + \gamma_k).$$

  Choose $\tilde{y}_k \in \mathbb{R}^n$.
  Set $\beta_k \leftarrow \min\{\beta \in [0, 1] : v(\beta) := \beta s_k + (1 - \beta)\tilde{y}_k$ satisfies $(\star)\}$.
  Set $v_k \leftarrow v(\beta_k)$.
  Set

$$W_{k+1} \leftarrow \left(I - \frac{v_k s_k^T}{s_k^T v_k}\right)^T W_k \left(I - \frac{v_k s_k^T}{s_k^T v_k}\right) + \frac{s_k s_k^T}{s_k^T v_k}.$$

end for

Instances of the framework

Cutting plane / bundle methods

- Points added incrementally until sufficient decrease obtained
- Finite number of additions until accepted step

Gradient sampling methods

- Points added randomly / incrementally until sufficient decrease obtained
- Sufficient number of iterations with "good" steps

In any case: convergence guarantees require $\{W_k\}$ to be uniformly positive definite and bounded *on a sufficient number of accepted steps*

Outline

Matlab implementation

Random instances of max-of-affine plus strongly convex quadratic, i.e.,

$$f(x) = \max_{i \in \{1,\dots,m\}} \{a_i^T x + b_i\} + c^T x + \tfrac{1}{2} x^T Q x$$

with $n = m = 100$; varying numbers of “active” affine functions at $x_* = 0$

Algorithms:

BFGS	:	BFGS w/ Wolfe line search	
B	:	Bundle method	(guarantees)
B-SC	:	. . . w/ self-correcting BFGS	(guarantees)
B-free	:	. . . w/ unadulterated BFGS	
GS	:	Gradient sampling	(guarantees)
GS-SC	:	. . . w/ self-correcting BFGS	(guarantees)
GS-free	:	. . . w/ unadulterated BFGS	

Relative performance measures: $\kappa(Q) = 100$

function evaluations:

# act.	BFGS	B	B-SC	B-free	GS	GS-SC	GS-free
4	1	2.7861	1.6154	0.6976	79.111	1.0801	1.0801
8	1	1.9192	1.2771	1.0580	158.698	1.0149	1.0127
12	1	1.4433	1.0293	1.0462	218.103	1.0975	1.0975
16	1	0.9760	0.7573	0.9222	241.187	1.0042	1.0042

gradient evaluations:

# act.	BFGS	B	B-SC	B-free	GS	GS-SC	GS-free
4	1	3.4729	2.0136	0.8695	16.001	1.0858	1.0858
8	1	3.0148	2.0063	1.6620	32.704	1.0406	1.0375
12	1	2.6174	1.8667	1.8973	47.674	1.1433	1.1433
16	1	1.9266	1.4950	1.8205	54.882	1.0098	1.0098

Relative performance measures: $\kappa(Q) = 100$

function evaluations:

# act.	BFGS	B	B-SC	B-free	GS	GS-SC	GS-free
4	1	2.7861	1.6154	0.6976	79.111	1.0801	1.0801
8	1	1.9192	1.2771	1.0580	158.698	1.0149	1.0127
12	1	1.4433	1.0293	1.0462	218.103	1.0975	1.0975
16	1	0.9760	0.7573	0.9222	241.187	1.0042	1.0042

gradient evaluations:

# act.	BFGS	B	B-SC	B-free	GS	GS-SC	GS-free
4	1	3.4729	2.0136	0.8695	16.001	1.0858	1.0858
8	1	3.0148	2.0063	1.6620	32.704	1.0406	1.0375
12	1	2.6174	1.8667	1.8973	47.674	1.1433	1.1433
16	1	1.9266	1.4950	1.8205	54.882	1.0098	1.0098

- **GS** very poor, but adding BFGS yields great improvements
- **B-SC** and **B-free** better than **B**
- self-correcting BFGS improves both bundle and gradient sampling methods

Relative performance measures: $\kappa(Q) = 1000$

function evaluations:

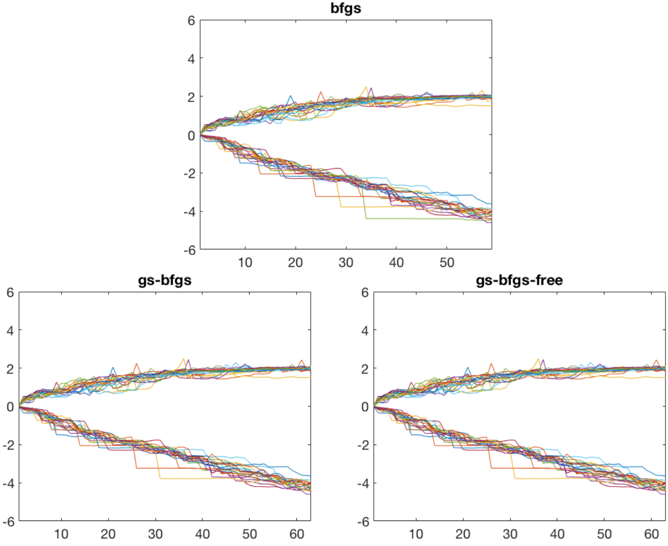
# act.	BFGS	B	B-SC	B-free	GS	GS-SC	GS-free
4	1	5.9193	5.5070	0.4741 (3)	111.425	0.9806	0.9831
8	1	3.8184	3.6010	0.5912 (2)	158.768	1.0490	1.0494
12	1	3.2655	3.0035	1.0220 (0)	193.947	1.0008	1.0235
16	1	2.9943	2.8077	1.4598 (6)	303.429	0.9943	0.9943

gradient evaluations:

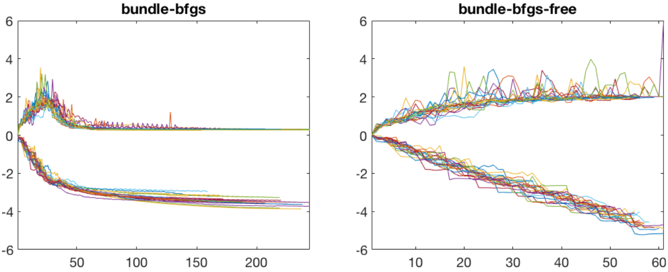
# act.	BFGS	B	B-SC	B-free	GS	GS-SC	GS-free
4	1	6.9029	6.4220	0.5529 (3)	27.890	0.9924	0.9945
8	1	4.7267	4.4575	0.7318 (2)	39.922	1.0424	1.0398
12	1	4.3938	4.0412	1.3751 (0)	47.516	1.0026	1.0277
16	1	4.4746	4.1958	2.1814 (6)	72.748	0.9930	0.9930

- similar conclusions, but **B-free** now unreliable (11 failures of 80 problems)

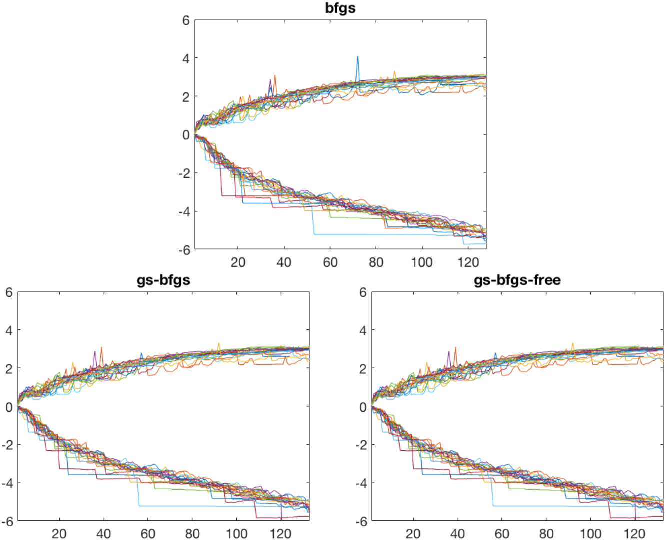
Minimum and maximum eigenvalues



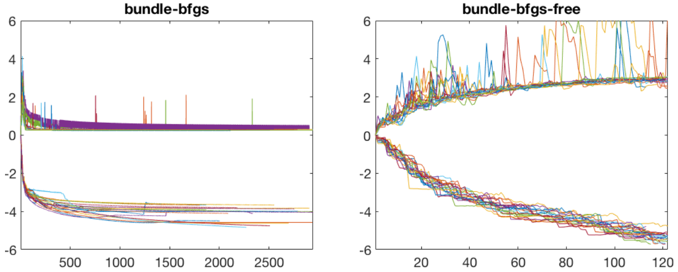
Minimum and maximum eigenvalues



Minimum and maximum eigenvalues



Minimum and maximum eigenvalues



Outline

Contributions

Proposed a quasi-Newton method for nonsmooth optimization

- unifying framework covering
  - cutting plane / bundle methods (convex only)
  - gradient sampling methods (nonconvex)
- exploit self-correcting properties of BFGS-type updates
- properties of Hessians offer useful bounds for inverse Hessians
- global convergence guarantees
- improved practical performance
  - different effects in cutting plane / bundle vs. gradient sampling...
  - worthwhile to explore this further...

Paper forthcoming...