BFGS-GS: A Quasi-Newton Gradient Sampling Algorithm for Nonconvex Nonsmooth Optimization

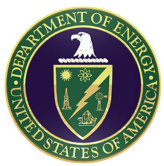
Frank E. Curtis, Lehigh University

joint work with

Xiaocun Que, Lehigh University

International Congress on Industrial and Applied Mathematics (ICIAM)
Beijing, China

13 August 2015



Outline

Outline

Context

Much research today is focused on solving structured optimization problems

- structure often means convex
- seeking sparsity, low matrix rank, low total variation, etc.

This talk focuses on solving unstructured problems

- problems may be nonconvex
- general-purpose algorithms are needed

Context

Much research today is focused on solving structured optimization problems

- structure often means convex
- seeking sparsity, low matrix rank, low total variation, etc.

This talk focuses on solving unstructured problems

- problems may be nonconvex
- general-purpose algorithms are needed

We propose stochastic methods for deterministic optimization

- No gradient info? e.g., simulation-based optimization
- Only some gradient info? e.g., machine learning
- Only some subdifferential info? e.g., (un)structured nonsmooth optimization

Good theory, computational flexibility, etc.

Background

Quasi-Newton methods, e.g., BFGS

- general-purpose for smooth optimization
- "first-order" method, i.e., gradients only
- superlinear convergence

Broyden (1970)
Fletcher (1970)
Goldfarb (1970)
Shanno (1970)

- good performance on nonsmooth problems
- . . . but little in terms of convergence guarantees

Lemaréchal (1981)
Lukšan & Vlček (1999, 2001)
Lewis & Overton (2013)

Gradient sampling (GS)

- general-purpose for nonsmooth optimization
- "first-order" method
- global convergence guarantees (w.p.1)
- good performance in practice
- . . . but expensive! $\mathcal{O}(n)$ gradients per iteration

Burke, Lewis, & Overton (2005)
Kiwiel (2007)

Contributions

New general-purpose methods for nonconvex nonsmooth optimization

- adaptive sampling, $\Omega(1)$ gradients per iteration
- Hessian approximation strategies
- convergence guarantees (w.p.1)
- dramatically reduced per-iteration & overall cost

} Curtis & Que (2013)

- BFGS-based strategy
- adaptive sampling, $\mathcal{O}(1)$ gradients per iteration
- convergence guarantees (w.p.1)
- further empirical improvements
- `BFGS-GS` software (C++)

} Curtis & Que (2015)

Outline

Problem formulation

Consider optimization problems of the form

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

such that the following is satisfied:

Assumption 1

The objective function f is

- *locally Lipschitz in $\mathbb{R}^n$*
- *continuously differentiable in an open, dense subset $\mathcal{D}$ of $\mathbb{R}^n$*

A point x is stationary if

$$0 \in \partial f(x) := \bigcap_{\epsilon > 0} \operatorname{cl} \operatorname{conv} \nabla f(\mathbb{B}_\epsilon(x) \cap \mathcal{D})$$

A point x is ϵ-stationary if

$$0 \in \partial_\epsilon f(x) := \operatorname{cl} \operatorname{conv} \partial f(\mathbb{B}_\epsilon(x))$$

GS idea

At x_k, let $x_{k0} := x_k$ and sample $\{x_{k1}, \ldots, x_{kp}\} \subset \mathbb{B}_{\epsilon_k}(x_k) \cap \mathcal{D}$, yielding:

$$\begin{array}{rcccccccl} X_k & := & \{ & x_{k0}, & x_{k1}, & \cdots, & x_{kp} & \} & \text{(sample points)} \\ G_k & := & [& g_{k0} & g_{k1} & \cdots & g_{kp} &] & \text{(sample gradients)} \end{array}$$

The ϵ_k-subdifferential is approximated by the convex hull of sampled gradients:

$$\begin{aligned} \partial_{\epsilon_k} f(x_k) &= \operatorname{cl}\operatorname{conv} \partial f(\mathbb{B}_{\epsilon_k}(x_k)) \\ &\approx \operatorname{conv}\{g_{k0}, g_{k1}, \ldots, g_{kp}\} \end{aligned}$$

Define the projection of the origin onto the convex hull of sampled gradients:

$$\boxed{g_k := \operatorname{Proj}(0 | \operatorname{conv}\{g_{k0}, g_{k1}, \ldots, g_{kp}\})}$$

The vector $d_k = -g_k$ is an approximate ϵ_k-steepest descent step

GS step computation

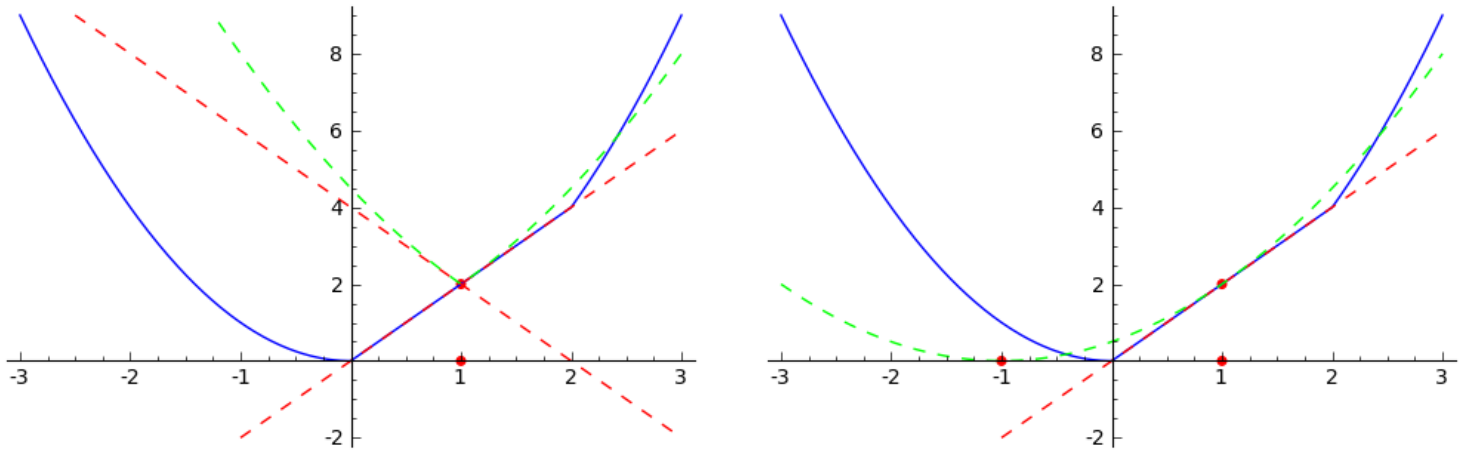
Alternatively, one can view d_k as the minimizer of a piecewise quadratic model:

$$\max_{(z,d)\in\mathbb{R}\times\mathbb{R}^n} \; z + \tfrac{1}{2}\|d\|_2^2 \quad \text{s.t. } f(x_k)e + G_k^T d \leq ze$$

$$\overbrace{\Leftrightarrow}^{\text{primal/dual}}$$

$$\max_{y\in\mathbb{R}^{p+1}} \; f(x_k) - \tfrac{1}{2}\|G_k y\|_2^2 \quad \text{s.t. } e^T y = 1,\; y \geq 0$$

Figure: Sampling yielding a small/zero step (left) vs. nonzero step (right)

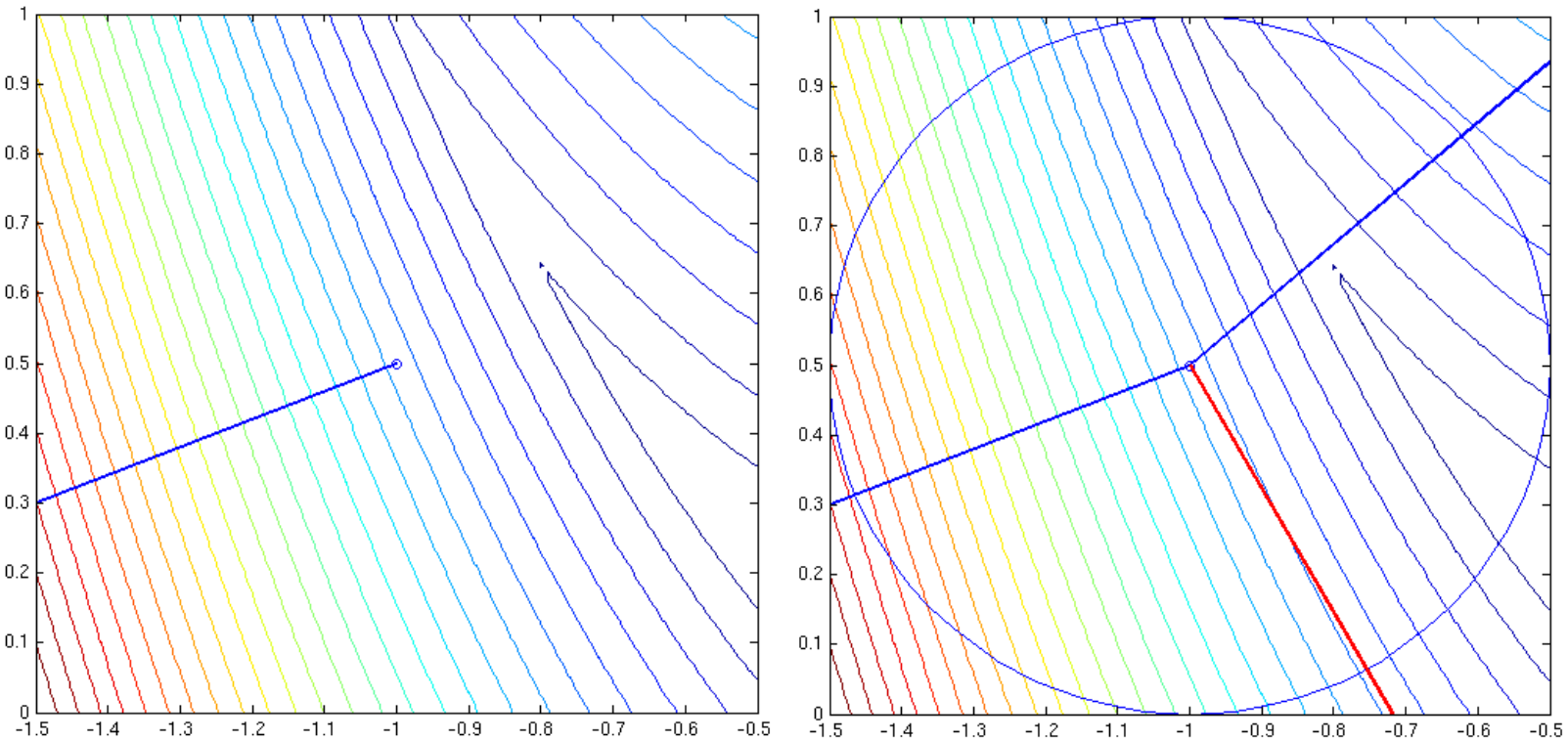


GS illustration

Example: nonsmooth Rosenbrock

$$\min_{x\in\mathbb{R}^2} \; 10|x_{(2)} - x_{(1)}^2| + (1 - x_{(1)})^2 \quad \text{at } x_k = (-1, \tfrac{1}{2})$$

Figure: Without gradient sampling (left) and with gradient sampling (right)



GS algorithm

Algorithm 1 Gradient Sampling (GS) Algorithm

Require:

initial point $x_0 \in \mathbb{R}^n$, initial sampling radius $\epsilon_0 > 0$

sufficient decrease tolerance $\eta_\alpha \in (0,1)$, stationarity tolerance $\eta_\epsilon > 0$

backtracking constant $\gamma_\alpha \in (0,1)$, sampling decrease constant $\gamma_\epsilon \in (0,1)$

sample size $p \geq n+1$

procedure GS

  for $k = 0, 1, 2, \ldots$ **do**

    sample p points $\{x_{k1}, \ldots, x_{kp}\} \subset \mathbb{B}_{\epsilon_k}(x_k) \cap \mathcal{D}$

    compute $d_k = -g_k$ via

$$\boxed{g_k := \text{Proj}(0 | \, \text{conv}\{g_{k0}, g_{k1}, \ldots, g_{kp}\})}$$

    set α_k as the largest element of $\{\gamma_\alpha^0, \gamma_\alpha^1, \gamma_\alpha^2, \ldots\}$ such that

$$f(x_k + \alpha_k d_k) \leq f(x_k) - \eta_\alpha \alpha_k \|d_k\|_2^2$$

    set $x_{k+1} \leftarrow x_k + \alpha_k d_k$ (or perturb to ensure $x_{k+1} \in \mathcal{D}$)

    if $\|d_k\|_2 \leq \eta_\epsilon \epsilon_k$, then set $\epsilon_{k+1} \leftarrow \gamma_\epsilon \epsilon_k$; else, set $\epsilon_{k+1} \leftarrow \epsilon_k$

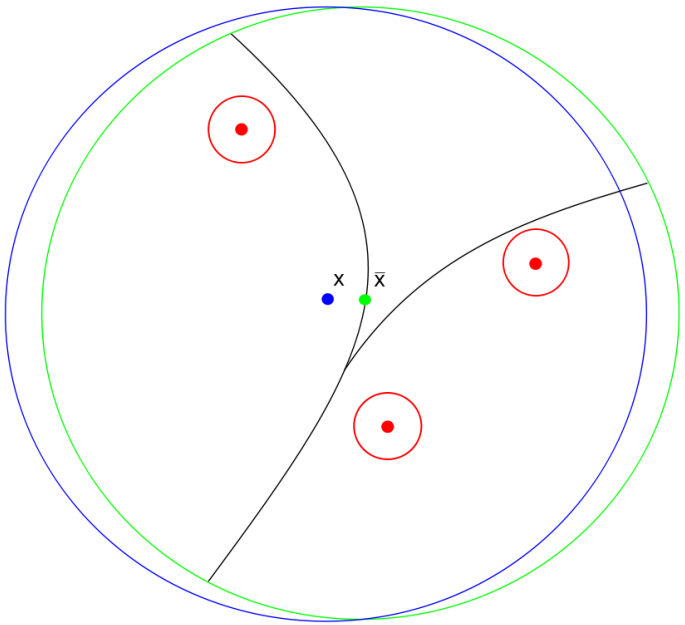
GS global convergence

Theorem 2

If Assumption 1 holds, then w.p.1 either

- $\{f(x_k)\} \to -\infty$, *or*
- *every cluster point of* $\{x_k\}$ *is stationary for* f

Proof idea: At x_k, either a direction of sufficient descent is produced or



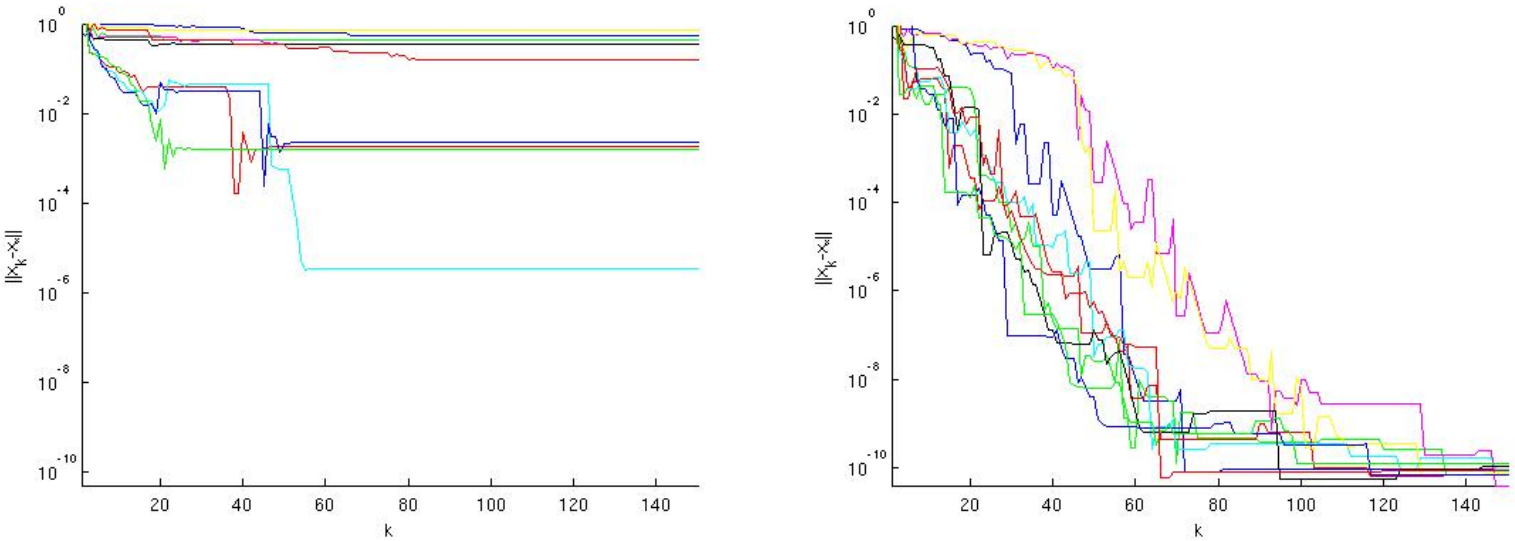
$\exists \{y_{ki}\}_{i=1,\ldots,p}$ and $\delta > 0$ such that $\text{Proj}(0|\{\nabla f(y_{ki} + O(\delta))\}) \approx \text{Proj}(0|\partial_{\epsilon_k} f(\overline{x}))$

GS illustration

Example: nonsmooth Rosenbrock

$$\min_{x\in\mathbb{R}^2} \; 10|x_{(2)} - x_{(1)}^2| + (1 - x_{(1)})^2 \quad \text{at } x_k = (-1, \tfrac{1}{2})$$

Figure: Without gradient sampling (left) and with gradient sampling (right)



GS issues

Practical limitations:

- $p \geq n + 1$ gradient evaluations per iteration
- subproblems distinct; solved from scratch
- "steepest descent" method

GS issues AGS solutions

Practical limitations:

- $p \geq n+1$ gradient evaluations per iteration
- subproblems distinct; solved from scratch
- "steepest descent" method

Adaptive GS: Curtis & Que (2013)

- adaptive sampling: Kiwiel (2010)
- $\Theta(1)$ gradients per iteration
- maintain sample points within ϵ-ball
- warm/hot-started subproblem solves
- quasi-Newton or over-estimation "Hessian" approximations ($W_k = H_k^{-1}$)

$$\max_{(z,d)\in\mathbb{R}\times\mathbb{R}^n} \; z + \tfrac{1}{2}\|d\|^2_{H_k} \quad \text{s.t. } f(x_k)e + G_k^T d \leq ze \qquad \Leftrightarrow \qquad \max_{y\in\mathbb{R}^{p+1}} \; f(x_k) - \tfrac{1}{2}\|G_k y\|^2_{W_k} \quad \text{s.t. } e^T y = 1,\; y \geq 0$$

Outline

Motivation

Why merge BFGS and GS?

BFGS:

- fast, cheap
- no automatic stationarity condition
- limited convergence guarantees
- . . . difficult to obtain as Hessians "blow up"

GS:

- expensive
- automatic stationarity condition
- convergence guarantees w.p.1

Idea: BFGS iteration, employing GS only when it appears needed

Search direction computation

At x_k, given an inverse Hessian approximation $W_k \succ 0$:

$$\boxed{d_k \leftarrow -W_k g_k}$$

On the other hand, if we have

$$\begin{array}{lcccccccl} X_k & := & \{ & x_{k0}, & x_{k1}, & \cdots, & x_{kp_k} & \} & \text{(sample points)} \\ G_k & := & [& g_{k0} & g_{k1} & \cdots & g_{kp_k} &] & \text{(sample gradients)} \end{array}$$

and a Hessian approximation H_k or inverse approximation W_k, then:

$$\boxed{\begin{array}{l} \max\limits_{z,d} \; z + \frac{1}{2}\|d\|^2_{H_k} \\ \text{s.t. } f(x_k)e + G_k^T d \leq ze \end{array}} \quad \Leftrightarrow \quad \boxed{\begin{array}{l} \max\limits_{y} \; f(x_k) - \frac{1}{2}\|G_k y\|^2_{W_k} \\ \text{s.t. } e^T y = 1, \; y \geq 0 \end{array}}$$

With $p_k = 0$, we recover the BFGS step $d_k \leftarrow -W_k g_k$

Line search and iterate update

In a BFGS method, to avoid damping or skipping, the line search would ideally yield a step size satisfying the Wolfe conditions

- Forward/backtracking line search to satisfy the Wolfe conditions

$$f(x_k) - f(x_k + \alpha_k d_k) > \underline{\eta}\alpha_k \|d_k\|_{H_k}^2 \qquad // \ \underline{\eta} \in (0,1)$$
$$v^T d_k \geq \overline{\eta} \nabla f(x_k)^T d_k, \text{ where } v \in \partial f(x_k + \alpha d_k) \qquad // \ \overline{\eta} \in (\underline{\eta}, 1)$$

- Curvature condition is abandoned after finite number of forward/backtracks (Motivation: Finite termination if $f_k(\alpha) := f(x_k + \alpha d_k) - f(x_k)$ is weakly lower semismooth: Lewis, Overton (2012); Mifflin (1977); Lemaréchal (1981))
- Line search abandoned ($\alpha_k \leftarrow 0$) if unsuccessful after finite number of forward/backtracks and sample size is not sufficiently large ($p_k \geq n+1$)

If necessary, perturb $x_k + \alpha_k d_k$ to find $x_{k+1} \in \mathcal{D}$ satisfying

$$f(x_k) - f(x_{k+1}) > \underline{\eta}\alpha_k \|d_k\|_{H_k}^2$$
$$\nabla f(x_{k+1})^T d_k \geq \overline{\eta} \nabla f(x_k)^T d_k$$
$$\|x_k + \alpha_k d_k - x_{k+1}\|_2 \leq \min\{\alpha_k, \epsilon_k\} \|d_k\|_2$$

Sample radius update

Reduce the sampling radius (i.e., choose $\epsilon_{k+1} \leftarrow \gamma_\epsilon \epsilon_k$) if

$$\begin{aligned} \|d_k\|_{H_k}^2 &\leq \eta_\epsilon \epsilon_k && // \ \eta_\epsilon > 0 \\ \|d_k\|_{H_k}^2 &\geq \underline{\xi} \epsilon_k \|d_k\|_2 && // \ \underline{\xi} \in (0,1) \\ \alpha_k &> 0 \end{aligned}$$

Sample point generation

At x_k, suppose we had

$$\begin{array}{lcccccccc} X_k & := & \{ & x_{k0}, & x_{k1}, & \cdots, & x_{kp_k} & \} & \text{(sample points)} \\ G_k & := & [& g_{k0} & g_{k1} & \cdots & g_{kp_k} &] & \text{(sample gradients)} \end{array}$$

If curvature is bounded and step-size sufficiently large in that

$$\begin{aligned} \underline{\xi}\epsilon_k\|d_k\|_2^2 \leq \|d_k\|_{H_k}^2 \leq \overline{\xi}\epsilon_k^{-1}\|d_k\|_2^2 \quad & // \ 0 < \underline{\xi} < \overline{\xi} \\ \underline{\alpha} \leq \alpha_k \quad & // \ 0 < \underline{\alpha} \end{aligned}$$

then erase sample set (i.e., $X_{k+1} \leftarrow \{x_{k+1}\}$ and $p_{k+1} \leftarrow 0$); else,

- discard gradients outside of radius ϵ_{k+1} about x_{k+1}
- maintain sample points within radius; warm/hot-starting
- sample $\Theta(1)$ new gradient(s)
- discard "old gradients" so $p_{k+1} \leq n+1$

Overall,

$$X_{k+1} \leftarrow (X_k \cap \mathbb{B}_{\epsilon_{k+1}}(x_{k+1})) \cup \{x_{k+1}\} \cup \overline{X}_{k+1}$$

where $\overline{X}_{k+1} \subset \mathbb{B}_{\epsilon_{k+1}}(x_{k+1}) \cap \mathcal{D}$

Quasi-Newton updating

If curvature is bounded and step-size sufficiently large in that

$$\begin{aligned} \underline{\xi}\epsilon_k\|d_k\|_2^2 \leq \|d_k\|_{H_k}^2 \leq \overline{\xi}\epsilon_k^{-1}\|d_k\|_2^2 \quad & // \ 0 < \underline{\xi} < \overline{\xi} \\ \underline{\alpha} \leq \alpha_k \quad & // \ 0 < \underline{\alpha} \end{aligned}$$

then standard BFGS update; else, L-BFGS update with pairs satisfying

$$\begin{aligned} \max\{\|s_j\|_2^2, \|y_j\|_2^2\} \leq \sigma \quad & // \ \sigma > 0 \\ s_j^T y_j \geq \gamma \quad & // \ \gamma > 0 \end{aligned}$$

Theorem 3

Initializing $H_{k+1} \leftarrow \mu_k I \succ 0$, *after* m *updates we have for any* $d \in \mathbb{R}^n$ *that*

$$\left(\frac{2^m}{\mu_k}\left(1+\frac{\sigma^2}{\gamma^2}\right)^m + \frac{\sigma}{\gamma}\left(\frac{2^m\left(1+\frac{\sigma^2}{\gamma^2}\right)^m - 1}{2\left(1+\frac{\sigma^2}{\gamma^2}\right) - 1}\right)\right)^{-1} \|d\|_2^2 \leq \|d\|_{H_{k+1}}^2 \leq \left(\mu_k + \frac{m\sigma}{\gamma}\right)\|d\|_2^2$$

BFGS-GS method

Algorithm 2 BFGS Gradient Sampling (BFGS-GS) Algorithm

Require:

initial point $x_0 \in \mathbb{R}^n$, initial sampling radius $\epsilon_0 > 0$, initial $W_0 \succ 0$

procedure BFGS-GS
  for $k = 0, 1, 2, \ldots$ **do**
    compute y_k from (dual) subproblem QP
    compute $d_k = -W_k G_k y_k$
    forward/backtrack Armijo/Wolfe line search to obtain α_k
    perturb (if necessary) to obtain $x_{k+1} \in \mathcal{D}$
    set sampling radius $\epsilon_{k+1} \leq \epsilon_k$
    set sample set X_{k+1}
    set (L-)BFGS inverse Hessian approximation W_{k+1}

Theorem 4

If Assumption 1 holds, then w.p.1 either

- $\{f(x_k)\} \to -\infty$, *or*
- *every cluster point of* $\{x_k\}$ *is stationary for* f

BFGS-GS

Implemented in C++

- implemented QP solver, adapted from Kiwiel (1985)
- 26 test problems, 10 random initial points each

Comparisons with:

- `HANSO-BFGS`: BFGS method, Overton et al.
- `HANSO-DEFAULT`: BFGS then GS, Overton et al.
- `LMBM`: limited memory bundle method, Haarala et al.

Termination flags:

(1) stationarity tolerance satisfied

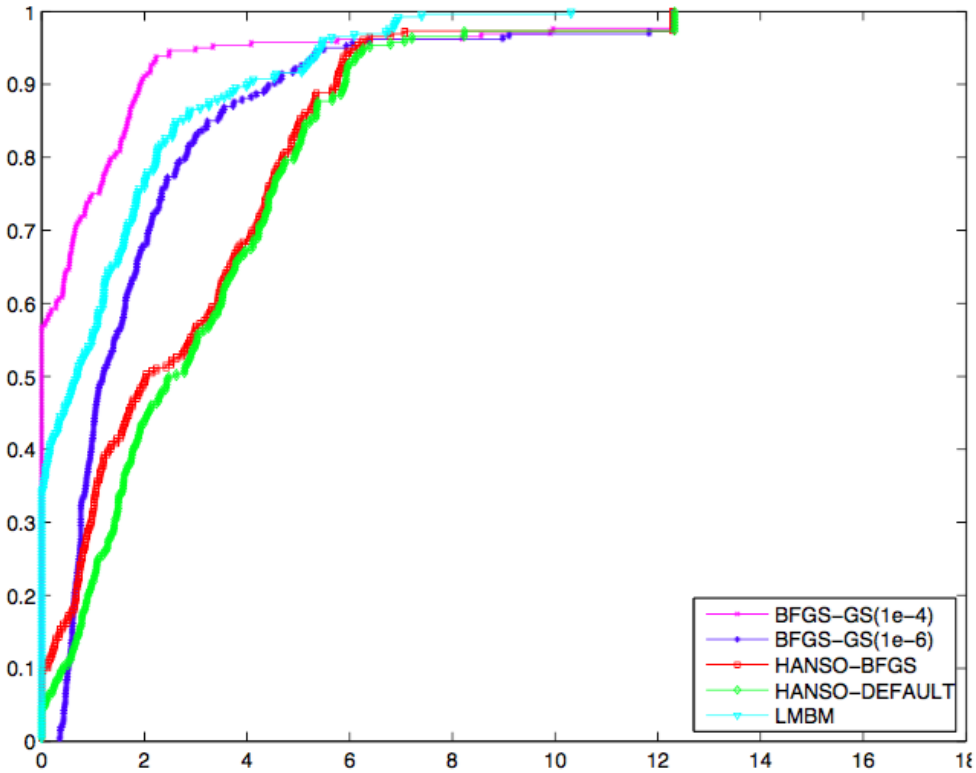
(2) maximum iteration limit reached

(3) other

flag	BFGS-GS(10^{-4})	BFGS-GS(10^{-6})	HANSO-BFGS	HANSO-DEFAULT	LMBM
(1)	253	229	68	68	20
(2)	7	31	31	19	0
(3)	0	0	161	173	240

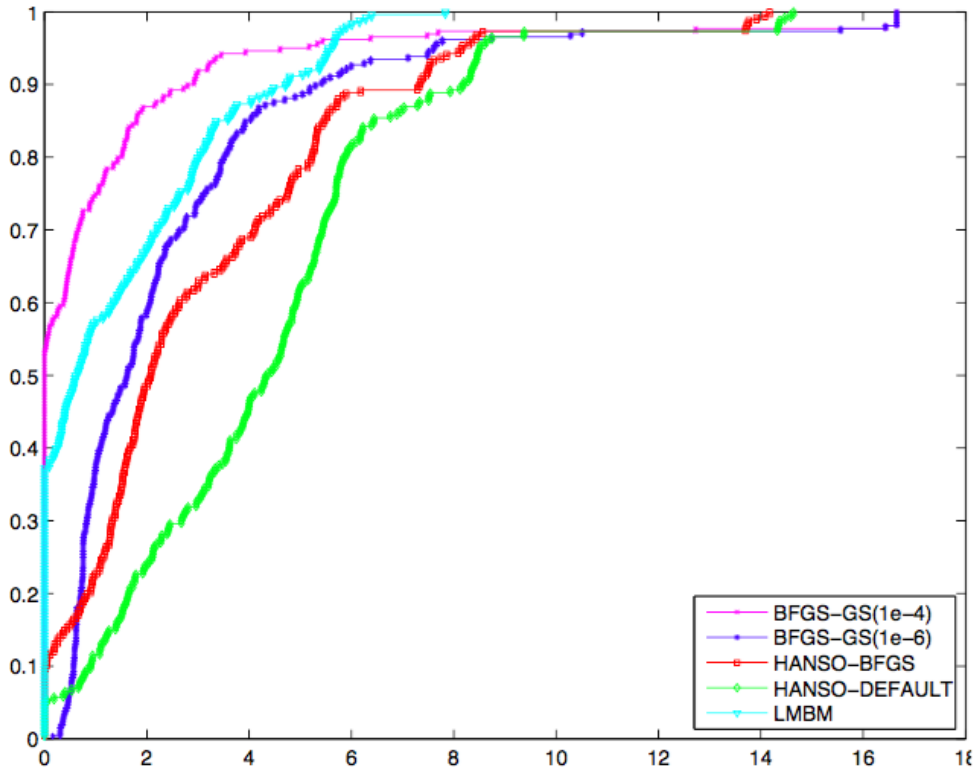
Performance profile: Iterations

Figure: Performance profile for iterations



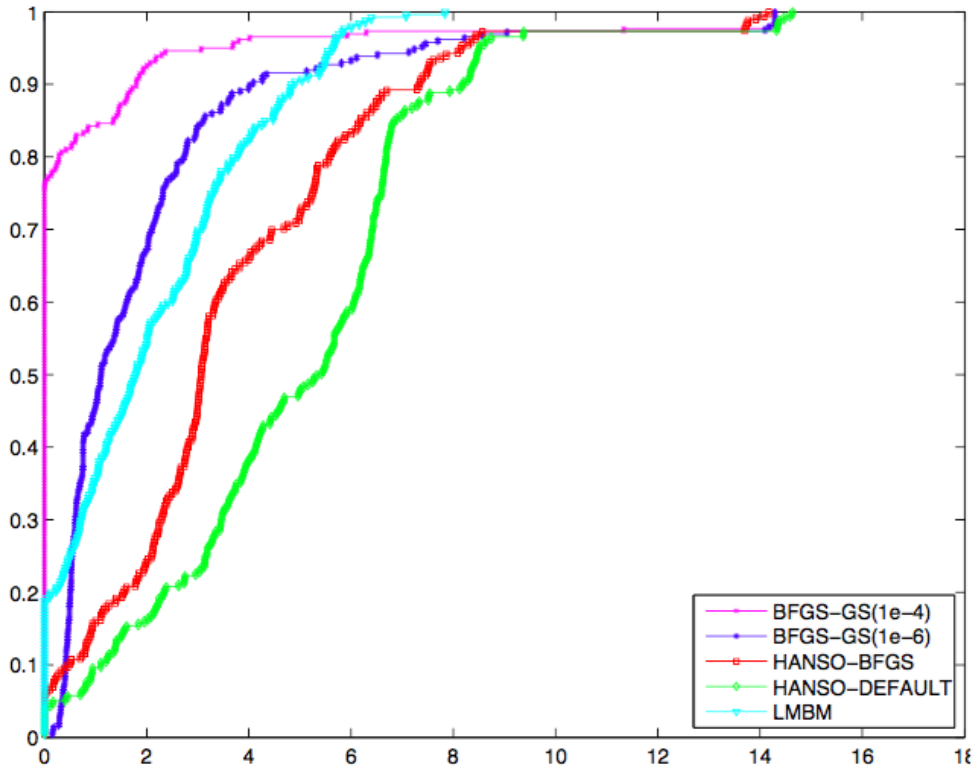
Performance profile: Function evaluations

Figure: Performance profile for function evaluations



Performance profile: Gradient evaluations

Figure: Performance profile for gradient evaluations



Overall, to obtain solutions of similar quality (see paper):

- `BFGS-GS`(10^{-4}) more efficient than `LMBM`
- `BFGS-GS`(10^{-6}) at least competitive with `HANSO-BFGS` and `HANSO`

Outline

Contributions

New general-purpose methods for nonconvex nonsmooth optimization

- adaptive sampling, $\Theta(1)$ gradients per iteration
- Hessian approximation strategies
- convergence guarantees (w.p.1)
- dramatically reduced per-iteration & overall cost

} Curtis & Que (2013)

- BFGS-based strategy
- adaptive sampling, $\mathcal{O}(1)$ gradients per iteration
- convergence guarantees (w.p.1)
- further empirical improvements
- `BFGS-GS` software (C++)

} Curtis & Que (2015)

⋆ F. E. Curtis and X. Que.
An Adaptive Gradient Sampling Algorithm for Nonsmooth Optimization.
Optimization Methods and Software, 28(6):1302–1324, 2013.

⋆ F. E. Curtis and X. Que.
A Quasi-Newton Algorithm for Nonconvex, Nonsmooth Optimization with Global Convergence Guarantees.
Mathematical Programming Computation, DOI: 10.1007/s12532-015-0086-2, 2015.