Nonconvex, Nonsmooth Optimization by Gradient Sampling

Frank E. Curtis, Lehigh University

involving joint work with

Michael L. Overton, New York University
Xiaocun Que, Lehigh University

Johns Hopkins University
Department of Applied Mathematics and Statistics
Research Seminar

April 5, 2012

Outline

Outline

Motivations

Gradient Sampling (GS)

Adaptive GS

SQP-GS

Future Work

Nonlinear/convex optimization research

Emphasis today on solving structured optimization problems.

- In most cases, structure means convex.
- Often goes further, e.g., seeking sparsity, low matrix rank, low total variation.
- Nemirovski, Nesterov, Wright, ...
- d'Aspremont, Lan, Recht, Yin, ...
- Focus on large-scale problems needing only an approximate solution.
- First-order methods, optimal algorithms, regularization, ...

My work

I am interested in algorithms for unstructured nonlinear optimization.

- For one thing, unstructured means nonconvex.
- Other work: Inexact Newton methods for large-scale optimization.
- Other work: Model/data inconsistencies leading to infeasibility and degeneracy.
- This talk: Enhancing practical NLO methods for handling nonsmoothness.

Widespread use of optimization requires accommodating algorithms.

- Accommodating algorithms can be the "go-to" methods for new problems.
- Accommodating algorithms are all we have for very hard problems.

Deterministic optimization methods based on randomized models

Unconstrained minimization of an objective function $f : \mathbb{R}^n \to \mathbb{R}$:

- No gradient info available? e.g., objective values from simulations
- Only some gradient info available? e.g., large-scale machine learning
- Subdifferential not available? e.g., any unstructured nonsmooth problem

Randomized algorithms offer computational flexibility and other benefits.

- DFO: randomization leads to better poised models.
- SO: (batch) stochastic gradient methods have nice practical/theoretical behavior.
- UO: gradient sampling...

Contributions

Gradient sampling is a general-purpose method for nonconvex, nonsmooth problems.

- We dramatically reduce per-iteration and overall computational cost.
- Nothing is lost in terms of global convergence guarantees.
- We extend the methodology and theory to constrained optimization.

Outline

Unconstrained nonconvex, nonsmooth optimization

Consider the unconstrained problem

$$\boxed{\min_x \; f(x)}$$

where f is locally Lipschitz and continuously differentiable in (dense) $\mathcal{D} \subset \mathbb{R}^n$.

Unconstrained nonconvex, nonsmooth optimization

Consider the unconstrained problem

$$\boxed{\min_x \; f(x)}$$

where f is locally Lipschitz and continuously differentiable in (dense) $\mathcal{D} \subset \mathbb{R}^n$.

- Let
$$\mathbb{B}_\epsilon(\overline{x}) := \{x \mid \|x - \overline{x}\| \leq \epsilon\}$$
- $\overline{x}$ is stationary if
$$0 \in \partial f(\overline{x}) := \bigcap_{\epsilon > 0} \operatorname{cl}\operatorname{conv} \nabla f(\mathbb{B}_\epsilon(\overline{x}) \cap \mathcal{D})$$
- $\overline{x}$ is ϵ-stationary if
$$0 \in \partial_\epsilon f(\overline{x}) := \operatorname{cl}\operatorname{conv} \partial f(\mathbb{B}_\epsilon(\overline{x}))$$

Gradient sampling (GS) idea

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

$$\begin{aligned} X_k &:= \{x_{k0}, \quad x_{k1}, \quad \cdots, \quad x_{kp}\} && \text{(sample points)} \\ G_k &:= \begin{bmatrix} g_{k0} & g_{k1} & \cdots & g_{kp} \end{bmatrix} && \text{(sample gradients)} \end{aligned}$$

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

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

Gradient sampling (GS) idea

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

$$\begin{aligned} X_k &:= \{x_{k0}, \quad x_{k1}, \quad \cdots, \quad x_{kp}\} && \text{(sample points)} \\ G_k &:= \begin{bmatrix} g_{k0} & g_{k1} & \cdots & g_{kp} \end{bmatrix} && \text{(sample gradients)} \end{aligned}$$

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

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

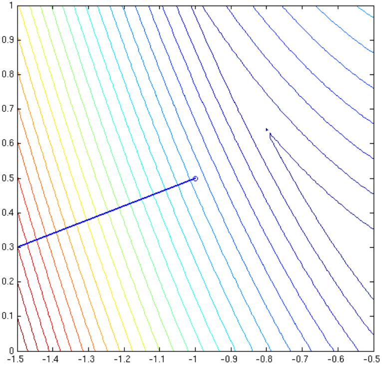
- Compute the projection of 0 onto the convex hull of the sampled gradients:

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

Then, $d_k = -g_k$ is an approximate ϵ-steepest descent step.

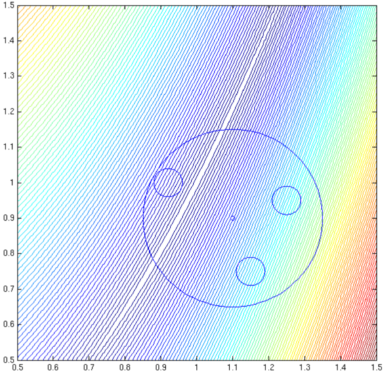
GS illustration

$$\min_x \; 10|x_2 - x_1^2| + (1 - x_1)^2 \;\; \text{at } x_k = (-1, \tfrac{1}{2})$$



GS illustration

$$\min_x \; 10|x_2 - x_1^2| + (1 - x_1)^2 \quad \text{at } x_k = (1.1, 0.9)$$



GS method

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

- Sample $p \geq n+1$ points $\{x_{k1}, \ldots, x_{kp}\} \subset \mathbb{B}_\epsilon(x_k) \cap \mathcal{D}$.
- Compute $d_k \leftarrow -g_k$ by computing the projection

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

- Backtrack from $\alpha_k \leftarrow 1$ to satisfy the sufficient decrease condition

$$f(x_k + \alpha_k d_k) \leq f(x_k) - \eta \alpha_k \|d_k\|^2.$$

- Update $x_{k+1} \approx x_k + \alpha_k d_k$ (to ensure $x_{k+1} \in \mathcal{D}$).
- If $\|d_k\| \leq \epsilon$, then reduce ϵ.

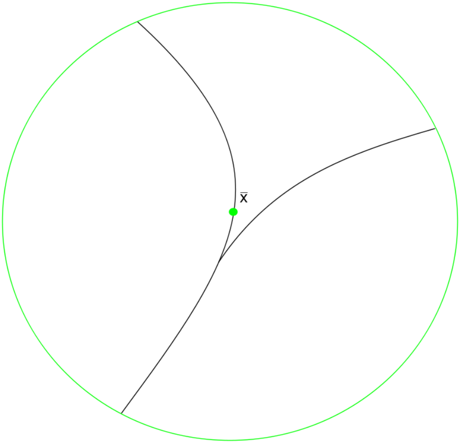
(See Burke, Lewis, and Overton (2005) and Kiwiel (2007).)

Global convergence of GS

Theorem: Let f be locally Lipschitz and continuously differentiable on an open dense $\mathcal{D} \subset \mathbb{R}^n$. Then, w.p.1, $f(x_k) \to -\infty$ or every cluster point of $\{x_k\}$ is stationary for f.

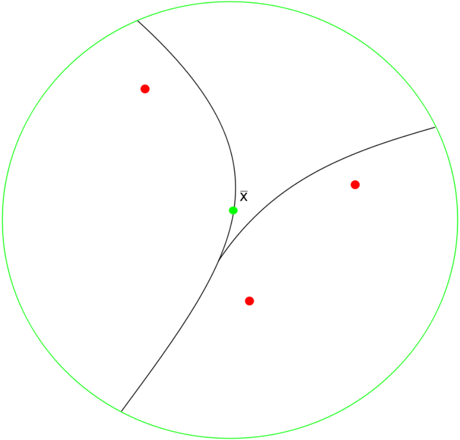
(See Burke, Lewis, and Overton (2005) and Kiwiel (2007).)

Illustration of critical part of proof



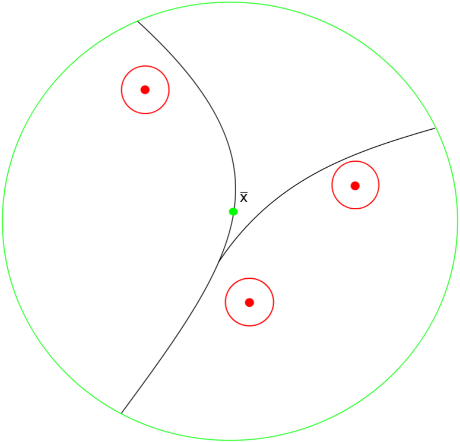
Near $\overline{x}$, the GS algorithm ideally computes $\text{Proj}(0|\partial_\epsilon f(\overline{x}))$.

Illustration of critical part of proof



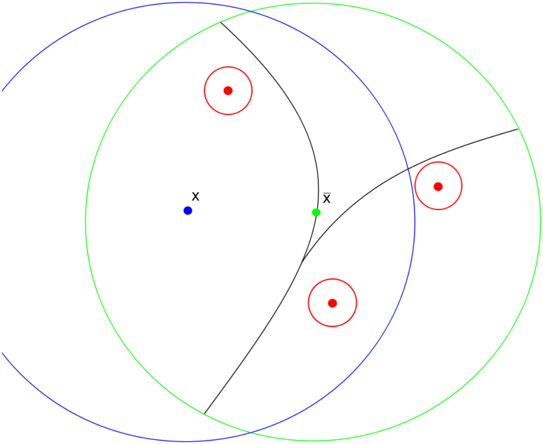
By continuity, there exists $\{y_{ki}\}_{i=1,\ldots,p}$ such that $\text{Proj}(0|\{\nabla f(y_{ki})\}) \approx \text{Proj}(0|\partial_\epsilon f(\overline{x}))$.

Illustration of critical part of proof



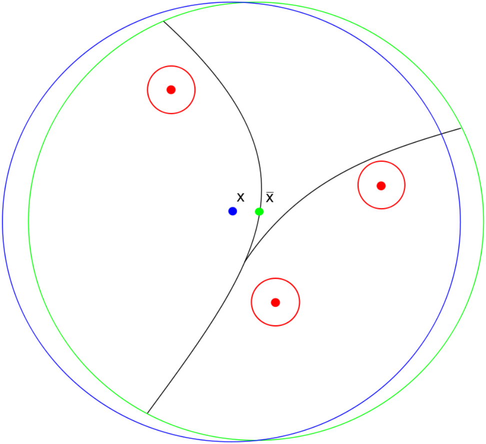
The same holds for sufficiently small neighborhoods about the y_{ki}'s.

Illustration of critical part of proof



Far from $\overline{x}$, the algorithm does not necessarily approximate $\text{Proj}(0|\partial_\epsilon f(\overline{x}))$ well.

Illustration of critical part of proof



However, it can in a sufficiently small neighborhood of $\overline{x}$.

Local models in GS

Computing the projection is equivalent to solving the dual subproblem:

$$\max_{\lambda} \; f(x_k) - \tfrac{1}{2}\|G_k\lambda\|^2$$
$$\text{s.t. } e^T\lambda = 1, \;\; \lambda \geq 0.$$

The corresponding primal subproblem is to compute d_k to minimize

$$q(d; X_k) := f(x_k) + \max_{x \in X_k}\{\nabla f(x)^T d\} + \tfrac{1}{2}\|d\|^2.$$

If all gradients about $\overline{x}$ were available, then we would ideally compute $\overline{d}$ minimizing

$$q(d; \mathbb{B}_\epsilon(\overline{x}) \cap \mathcal{D}) = f(\overline{x}) + \max_{x \in \mathbb{B}_\epsilon(\overline{x}) \cap \mathcal{D}}\{\nabla f(x)^T d\} + \tfrac{1}{2}\|d\|^2.$$

Critical lemma

Let the sample space be

$$\mathcal{S}_\epsilon(x_k) := \{x_k\} \times \prod_1^p (\mathbb{B}_\epsilon(x_k) \cap \mathcal{D})$$

and consider the set

$$\mathcal{T}_{\epsilon,\omega}(x_k, \overline{x}) = \{X_k \in \mathcal{S}_\epsilon(x_k) \mid \Delta q(d_k; X_k) \leq \Delta q(\overline{d}; \mathbb{B}_\epsilon(\overline{x}) \cap \mathcal{D}) + \omega\}.$$

Lemma: For any $\omega > 0$, there exists $\zeta > 0$ and a nonempty set $\mathcal{T}$ such that for all $x_k \in \mathbb{B}(\overline{x}, \zeta)$ we have $\mathcal{T} \subset \mathcal{T}_{\epsilon,\omega}(x_k, \overline{x})$.

(That is, in a sufficiently small neighborhood of $\overline{x}$, there exists a sample set revealing $\Delta q(\overline{d}; \mathbb{B}_\epsilon(\overline{x}) \cap \mathcal{D})$ with arbitrarily good, though not necessarily perfect, accuracy.)

Sketch of proof: Follows from Carathéodory's theorem.

Global convergence of GS

Theorem: Let f be locally Lipschitz and continuously differentiable on an open dense $\mathcal{D} \subset \mathbb{R}^n$. Then, w.p.1, $f(x_k) \to -\infty$ or every cluster point of $\{x_k\}$ is stationary for f.

Sketch of proof: If $f(x_k) \nrightarrow -\infty$, then

$$\alpha_k \Delta q(d_k; X_k) \to 0.$$

If $\epsilon \nrightarrow 0$, then for all large k,

$$\Delta q(d_k; X_k) = \tfrac{1}{2}\|d_k\|^2 > \tfrac{1}{2}\epsilon^2, \qquad (\star)$$

and it can be shown that $x_k \to \bar{x}$ and $\alpha_k \to 0$. However, w.p.1, this will not occur:

- If $\bar{x}$ is ϵ-stationary, then w.p.1 we will obtain a sample set in $\mathcal{T}$ yielding $\Delta q(d_k; X_k) \leq \frac{1}{2}\epsilon^2$, contradicting $(\star)$.
- If $\bar{x}$ is not ϵ-stationary, then w.p.1 we will obtain a subsequence with α_k bounded away from zero, contradicting $\alpha_k \to 0$.

Thus, w.p.1, $\epsilon \to 0$ and any cluster point $\bar{x}$ is stationary for f.

Outline

Practical issues

Practical limitations of GS:

- $p \geq n + 1$ gradient evaluations per iteration
- All subproblems solved from scratch
- Behaves like steepest descent(?)

Practical issues

Practical limitations of GS:

- $p \geq n + 1$ gradient evaluations per iteration
- All subproblems solved from scratch
- Behaves like steepest descent(?)

Proposed enhancements:

- Adaptive sampling; only $O(1)$ gradients per iteration (Kiwiel (2010))
- Warm-started subproblem solves
- "Hessian" approximations for quadratic term

Adaptive Gradient Sampling (AGS)

At x_k, we had:

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

At x_{k+1}, we

- maintain sample points still within radius ϵ; (this allows warm-starting!)
- throw out gradients outside of radius;
- sample 1 (or some) new gradients.

How can we maintain global convergence?

- If sample size is at least $n+1$, then proceed as usual; else, truncate line search.

Primal-dual pair of subproblems

Recall the primal-dual pair of GS subproblems:

$$\boxed{\begin{aligned} &\max_{z,d}\ z + \tfrac{1}{2}d^T d \\ &\ \text{s.t. } f(x_k)e + G_k^T d \le ze \end{aligned}} \qquad \boxed{\begin{aligned} &\max_{\lambda}\ f(x_k) - \tfrac{1}{2}\lambda^T G_k^T G_k \lambda \\ &\ \text{s.t. } e^T\lambda = 1,\ \lambda \ge 0 \end{aligned}}$$

Primal-dual pair of subproblems (variable-metric)

Recall the primal-dual pair of GS subproblems:

$$\begin{aligned}\max_{z,d}\ & z + \tfrac{1}{2}d^T d \\ \text{s.t. } & f(x_k)e + G_k^T d \le ze\end{aligned}$$

$$\begin{aligned}\max_{\lambda}\ & f(x_k) - \tfrac{1}{2}\lambda^T G_k^T G_k \lambda \\ \text{s.t. } & e^T\lambda = 1,\ \lambda \ge 0\end{aligned}$$

Introduce second order terms with "Hessian" approximations:

$$\begin{aligned}\max_{z,d}\ & z + \tfrac{1}{2}d^T H_k d \\ \text{s.t. } & f(x_k)e + G_k^T d \le ze\end{aligned}$$

$$\begin{aligned}\max_{\lambda}\ & f(x_k) - \tfrac{1}{2}\lambda^T G_k^T W_k G_k \lambda \\ \text{s.t. } & e^T\lambda = 1,\ \lambda \ge 0\end{aligned}$$

How should H_k (or W_k) be chosen?

Quasi-Newton updating

Consider the model

$$q(d; x_{k+1}, H_{k+1}) = f(x_{k+1}) + \nabla f(x_{k+1})^T d + \tfrac{1}{2} d^T H_{k+1} d.$$

Matching the gradients of f and m_{k+1} at x_k yields the secant equation

$$H_{k+1}(\nabla f(x_{k+1}) - \nabla f(x_k)) = x_{k+1} - x_k.$$

Minimizing changes in $\{H_k\}$ yields the well-known BFGS update.

Questions:

- Is BFGS effective within GS?
- Are we making the best use of info?
- Ill-conditioning: Bad or good?

Quasi-Newton updating (AGS-LBFGS)

Consider BFGS, but instead of updating between iterations, update during them.

- For each k, initialize $H_k \leftarrow \mu_k I$.
- Imagine moving along each $d_{ki} = x_{ki} - x_k$ and apply BFGS update.

With at most p points in the sample set, this is an L-BFGS-type approach.

Overestimation (AGS-over)

Suppose we also have function values at the sample points.

- Try to choose H_k so that the following model overestimates f:

$$q(d; X_k, H_k) = f(x_k) + \max_{x \in X_k} \{\nabla f(x)^T d\} + \tfrac{1}{2} d^T H_k d.$$

- If $q(d_{ki}; X_k, H_k) < f(x_{ki})$, then "lift" $d_{ki}^T H_k d_{ki}$ so that $q(d_{ki}; X_k, H_k) = f(x_{ki})$.
- Updates we use have the form $H_k \leftarrow M_{ki}^T H_k M_{ki}$ where

$$M_{ki} = \left(I + \frac{\gamma}{d_{ki}^T d_{ki}} d_{ki} d_{ki}^T \right).$$

Global convergence of AGS

Theorem: Let $\sigma, \gamma > 0$ be user-defined constants. Then, for any k, after all updates have been performed for AGS-LBFGS for sample points 1 through $p_k \leq p$, the following holds for any $d \in \mathbb{R}^n$:

$$\left(2^p\left(1+\frac{\sigma}{\gamma^2}\right)^p \mu_k + \frac{1}{\gamma}\left(\frac{2^p\left(1+\frac{\sigma}{\gamma^2}\right)^p - 1}{2\left(1+\frac{\sigma}{\gamma^2}\right)-1}\right)\right)^{-1} \|d\|^2 \leq d^T H_k d \leq \left(\mu_k + \frac{p\sigma}{\gamma}\right)\|d\|^2.$$

Theorem: Let $\rho \geq 1/2$ be a user-defined constant. Then, for any k, after all updates have been performed for AGS-over for sample points 1 through $p_k \leq p$, the following holds for any $d \in \mathbb{R}^n$:

$$\mu_k \|d\|^2 \leq d^T H_k d \leq \mu_k (2\rho)^p \|d\|^2.$$

Theorem: Let f be locally Lipschitz and continuously differentiable on an open dense $\mathcal{D} \subset \mathbb{R}^n$. Then, w.p.1, $f(x_k) \to -\infty$ or every cluster point of $\{x_k\}$ is stationary for f.

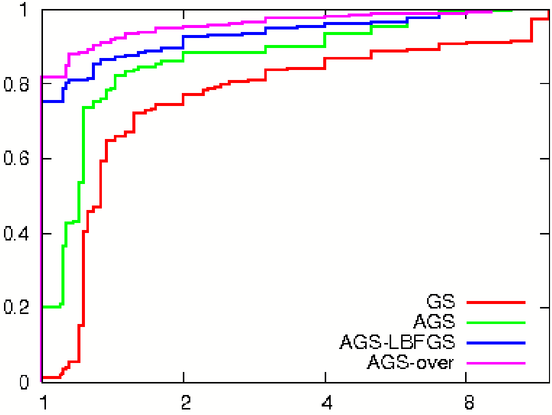
(See Curtis and Que (2011).)

Implementation and test details

- Matlab implementation
- QO solver adapted from Kiwiel (1986)
- 26 test problems from Haarala (2004) with $n = 50$
- Each problem run with 10 random starting points
- GS: $p = 2n$ gradients per iteration
- AGS: $p = 2n$ required for full line search, but only 5 gradients per iteration

Performance profile for final ϵ

Limit of 5000 gradient evaluations: GS, 49 iters.; AGS, 833 iters.



Final $\epsilon \in \{10^{-1}, \ldots, 10^{-12}\}$; performance profile for $\log_{10} \epsilon + 13$.

Outline

Nonlinear constrained optimization

Consider constrained optimization problems of the form:

$$\begin{aligned} \min_x \; & f(x) && \text{(smooth)} \\ \text{s.t. } & c_{\mathcal{E}}(x) = 0 && \text{(smooth)} \\ & c_{\mathcal{I}}(x) \leq 0 && \text{(smooth)} \end{aligned}$$

- Decades worth of algorithmic development.
- SQP, IPM, etc., with countless variations.
- Strong global and local convergence guarantees.
- Multiple popular, successful software packages.

Nonlinear constrained optimization with nonsmoothness

Consider constrained optimization problems of the form:

$$
\begin{aligned}
\min_{x}\ & f(x) && \text{((non)smooth)} \\
\text{s.t. } & c_{\mathcal{E}}(x) = 0 && \text{(smooth)} \\
& c_{\tilde{\mathcal{E}}}(x) = 0 && \text{(nonsmooth)} \\
& c_{\mathcal{I}}(x) \le 0 && \text{(smooth)} \\
& c_{\tilde{\mathcal{I}}}(x) \le 0 && \text{(nonsmooth)}
\end{aligned}
$$

- Algorithms for smooth problems no longer effective theoretically/practically.
- However, so much of the structure is the same as before.
- Can we adapt nonlinear optimization technology to handle nonsmoothness?

Constrained optimization with smooth functions

Consider constrained optimization problems of the form:

$$\begin{aligned} \min_x \ & f(x) && \text{(smooth)} \\ \text{s.t. } & c(x) \le 0 && \text{(smooth)} \end{aligned}$$

At x_k, solve the SQP subproblem

$$\begin{aligned} \min_d \ & f(x_k) + \nabla f(x_k)^T d + \tfrac{1}{2} d^T H_k d \\ \text{s.t. } & c(x_k) + \nabla c(x_k)^T d \le 0 \end{aligned}$$

to compute the search direction d_k.

Inconsistent linearizations of the constraints

The linearized constraints may be inconsistent, but we can relax the problem to

$$\begin{aligned} \min_{d,s} \ & \rho(f(x_k) + \nabla f(x_k)^T d) + e^T s + \tfrac{1}{2} d^T H_k d \\ \text{s.t. } & c(x_k) + \nabla c(x_k)^T d \le s, \quad s \ge 0, \end{aligned}$$

Solving the (P)SQP subproblem is equivalent to minimizing

$$q_\rho(d; x_k, H_k) := \rho(f(x_k) + \nabla f(x_k)^T d) + \sum \max\{c^i(x_k) + \nabla c^i(x_k)^T d, 0\} + \tfrac{1}{2} d^T H_k d.$$

We perform a line search on the exact penalty function

$$\phi_\rho(x) \triangleq \rho f(x) + \sum \max\{c^i(x), 0\}$$

to promote global convergence.

SQP method

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

- Solve the SQP subproblem

$$\min_{d,s} \ \rho(f(x_k) + \nabla f(x_k)^T d) + e^T s + \tfrac{1}{2} d^T H_k d$$
$$\text{s.t. } c(x_k) + \nabla c(x_k)^T d \leq s, \quad s \geq 0$$

  to compute d_k.
- Backtrack from $\alpha_k \leftarrow 1$ to satisfy the sufficient decrease condition

$$\phi_\rho(x_k + \alpha_k d_k) \leq \phi_\rho(x_k) - \eta \alpha_k \Delta q_\rho(d_k; x_k, H_k).$$

- Update $x_{k+1} \leftarrow x_k + \alpha_k d_k$.

Constrained optimization of nonsmooth functions

Consider constrained optimization problems of the form

$$\begin{aligned} \min_x \ & f(x) && \text{(nonsmooth, locally Lipschitz)} \\ \text{s.t. } & c(x) \leq 0 && \text{(nonsmooth, locally Lipschitz)} \end{aligned}$$

We may consider applying an unconstrained technique (e.g., AGS) directly to

$$\min_x \ \phi_\rho(x),$$

but can we do better by maintaining the framework of SQP?

SQP and GS

- The SQP subproblem (for a smooth constrained problem) is

$$\begin{aligned}\min_{z,d,s}\ & \rho z + e^T s + \tfrac{1}{2} d^T H_k d \\ \text{s.t. } & f(x_k) + \nabla f(x_k)^T d \le z \\ & c(x_k) + \nabla c(x_k)^T d \le s,\ \ s \ge 0.\end{aligned}$$

- The AGS subproblem (for a nonsmooth objective) is

$$\begin{aligned}\min_{z,d}\ & z + \tfrac{1}{2} d^T H_k d \\ \text{s.t. } & f(x_k) + \nabla f(x)^T d \le z,\ \text{ for } x \in X_k.\end{aligned}$$

SQP and GS

- The SQP subproblem (for a smooth constrained problem) is

$$\begin{aligned}\min_{z,d,s}\ & \rho z + e^T s + \tfrac{1}{2} d^T H_k d \\ \text{s.t.}\ & f(x_k) + \nabla f(x_k)^T d \le z \\ & c(x_k) + \nabla c(x_k)^T d \le s,\ \ s \ge 0.\end{aligned}$$

- The AGS subproblem (for a nonsmooth objective) is

$$\begin{aligned}\min_{z,d}\ & z + \tfrac{1}{2} d^T H_k d \\ \text{s.t.}\ & f(x_k) + \nabla f(x)^T d \le z,\ \text{ for } x \in X_k.\end{aligned}$$

- The SQP-GS subproblem (for a nonsmooth constrained problem) is

$$\begin{aligned}\min_{z,d,s}\ & \rho z + e^T s + \tfrac{1}{2} d^T H_k d \\ \text{s.t.}\ & f(x_k) + \nabla f(x)^T d \le z,\ \text{ for } x \in X_k^f \\ & c^i(x_k) + \nabla c^i(x)^T d \le s^i,\ \ s^i \ge 0,\ \text{ for } x \in X_k^{c^i},\ \ i = 1, \ldots, m\end{aligned}$$

SQP-GS in more detail

- The SQP-GS subproblem is

$$\begin{aligned}
\min_{z,d,s}\ & \rho z + e^T s + \tfrac{1}{2} d^T H_k d \\
\text{s.t. } & f(x_k) + \nabla f(x)^T d \le z, \ \text{ for } x \in X_k^f \\
& c^i(x_k) + \nabla c^i(x)^T d \le s^i, \ s^i \ge 0, \ \text{ for } x \in X_k^{c^i}, \ i = 1, \dots, m
\end{aligned}$$

where X_k is composed of

$$\begin{aligned}
X_k^f &= \{x_k, x_{k1}^f, \dots, x_{kp}^f\} \subset \mathbb{B}_\epsilon(x_k) \cap \mathcal{D}^f \\
\text{and } X_k^{c^i} &= \{x_k, x_{k1}^{c^i}, \dots, x_{kp}^{c^i}\} \subset \mathbb{B}_\epsilon(x_k) \cap \mathcal{D}^{c^i} \text{ for } i = 1, \dots, m.
\end{aligned}$$

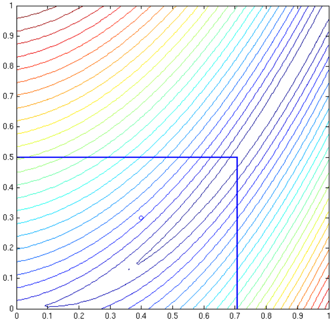
- This is equivalent to minimizing

$$q_\rho(d; X_k, H_k) :=$$

$$\rho \max_{x \in X_k^f} (f(x_k) + \nabla f(x)^T d) + \sum \max_{x \in X_k^{c^i}} \max\{c^i(x_k) + \nabla c^i(x)^T d, 0\} + \tfrac{1}{2} d^T H_k d.$$

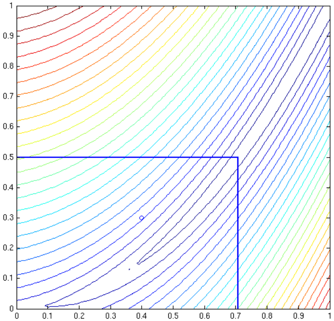
SQP-GS illustration

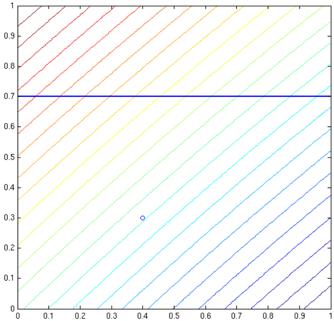
$$\min_x \; 10|x_2 - x_1^2| + (1 - x_1)^2 \quad \text{s.t.} \quad \max\{\sqrt{2}x_1, 2x_2\} - 1 \le 0 \quad \text{at } x_k = (\tfrac{2}{5}, \tfrac{3}{10}).$$



SQP-GS illustration

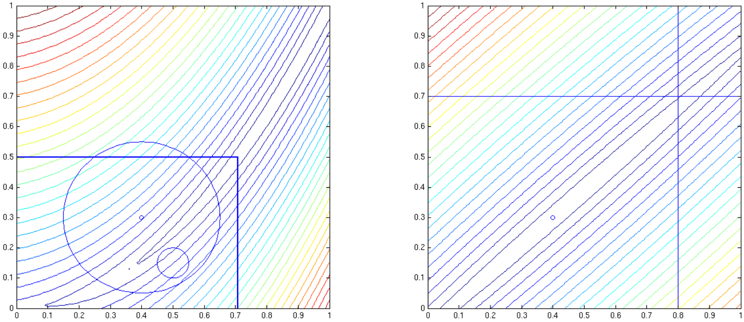
$$\min_x \; 10|x_2 - x_1^2| + (1-x_1)^2 \quad \text{s.t.} \quad \max\{\sqrt{2}x_1, 2x_2\} - 1 \leq 0 \quad \text{at } x_k = (\tfrac{2}{5}, \tfrac{3}{10}).$$





SQP-GS illustration

$$\min_{x} \; 10|x_2 - x_1^2| + (1 - x_1)^2 \quad \text{s.t.} \; \max\{\sqrt{2}x_1, 2x_2\} - 1 \leq 0 \quad \text{at } x_k = (\tfrac{2}{5}, \tfrac{3}{10}).$$



SQP-GS method

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

- Sample $p \geq n+1$ points for each function to generate $X_k = \{X_k^f, X_k^{c^1}, \ldots, X_k^{c^m}\}$.
- Compute d_k by solving the SQP-GS subproblem

$$\begin{aligned}
\min_{z,d,s}\ & \rho z + e^T s + \tfrac{1}{2} d^T H_k d \\
\text{s.t. } & f(x_k) + \nabla f(x)^T d \leq z, \text{ for } x \in X_k^f \\
& c^i(x_k) + \nabla c^i(x)^T d \leq s^i,\ s^i \geq 0, \text{ for } x \in X_k^{c^i},\ i = 1, \ldots, m
\end{aligned}$$

- Backtrack from $\alpha_k \leftarrow 1$ to satisfy the sufficient decrease condition

$$\phi_\rho(x_k + \alpha_k d_k) \leq \phi_\rho(x_k) - \eta \alpha_k \Delta q_\rho(d_k; X_k, H_k).$$

- Update $x_{k+1} \approx x_k + \alpha_k d_k$ (to ensure $x_{k+1} \in \mathcal{D}^f \cap \mathcal{D}^{c^1} \cap \cdots \cap \mathcal{D}^{c^m}$)
- If $\Delta q_\rho(d_k; X_k, H_k) \leq \frac{1}{2}\epsilon^2$, then reduce ϵ.
- If ϵ has been reduced and x_k is not sufficiently feasible, then reduce ρ.

Convergence theory for SQP-GS

Theorem: Suppose the following conditions hold:

- f and c^i, $i = 1, \ldots, m$, are locally Lipschitz and continuously differentiable on open dense subsets of $\mathbb{R}^n$.
- $\{x_k\}$ and all generated sample points are contained in a convex set over which f and c^i, $i = 1, \ldots, m$, and their first derivatives are bounded.
- $\{H_k\}$ are symmetric positive definite, bounded above in norm, and bounded away from singularity.

Then, w.p.1, one of the following holds true:

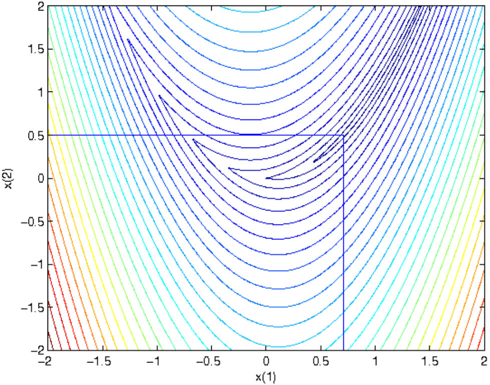
- $\rho = \rho_* > 0$ for all large k and every cluster point of $\{x_k\}$ is stationary for ϕ_{ρ_*}. Moreover, with K defined as the infinite subsequence of iterates during which ϵ is decreased, all cluster points of $\{x_k\}_{k \in K}$ are feasible for the optimization problem.
- $\rho \to 0$ and every cluster point of $\{x_k\}$ is stationary for ϕ_0.

Implementation

- Matlab implementation
- QO subproblems solved with MOSEK
- BFGS approximations of Hessian of $\phi_\rho(x)$ (as in AGS-LBFGS)
- $p = 2n$ gradients per iteration

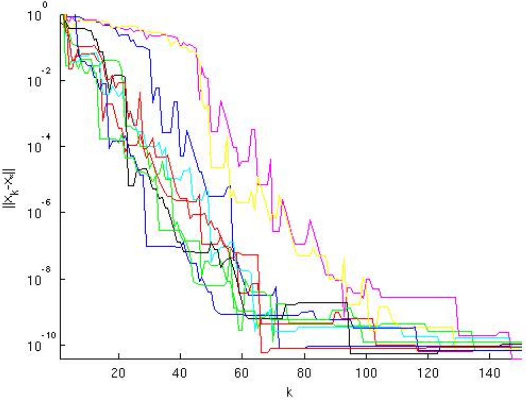
Example 1: Nonsmooth Rosenbrock

$$\min_{x} \; 10|x_1^2 - x_2| + (1 - x_1)^2 \quad \text{s.t.} \;\; \max\{\sqrt{2}x_1, 2x_2\} \leq 1.$$



Example 1: Nonsmooth Rosenbrock

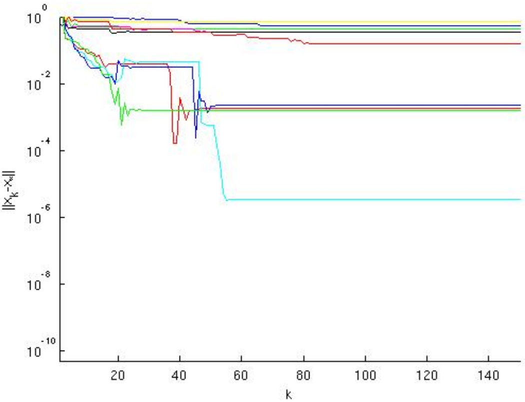
$$\min_{x} \; 10|x_1^2 - x_2| + (1 - x_1)^2 \quad \text{s.t.} \;\; \max\{\sqrt{2}x_1, 2x_2\} \leq 1.$$



Plot of distance to solution

Example 1: Nonsmooth Rosenbrock

$$\min_x \; 10|x_1^2 - x_2| + (1 - x_1)^2 \quad \text{s.t.} \quad \max\{\sqrt{2}x_1, 2x_2\} \leq 1.$$



Plot of distance to solution (no sampling)

Example 2: Entropy minimization

Find a $N \times N$ matrix X that solves

$$\min_X \ \ln \left(\prod_{j=1}^{K} \lambda_j (A \circ X^T X) \right)$$
$$\text{s.t. } \|X_j\| = 1, \ j = 1, \ldots, N$$

where $\lambda_j(M)$ denotes the jth largest eigenvalue of M, A is a real symmetric $N \times N$ matrix, $\circ$ denotes the Hadamard matrix product, and X_j denotes the jth column of X.

Example 2: Entropy minimization

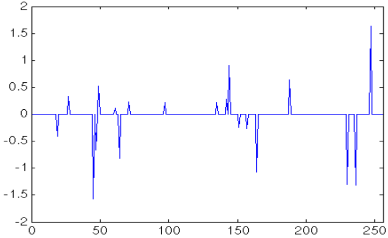
N	K	n	Objective	Infeasibility	Final ϵ	Opt. error
2	1	4	1.0000e+00	3.1752e-14	5.9605e-09	7.6722e-12
4	2	16	7.4630e-01	2.8441e-07	4.8828e-05	1.1938e-04
6	3	36	6.3359e-01	2.1149e-06	9.7656e-05	8.7263e-02
8	4	64	5.5832e-01	2.0492e-05	9.7656e-05	2.7521e-03
10	5	100	2.1841e-01	9.8364e-06	7.8125e-04	9.6041e-03
12	6	144	1.2265e-01	1.8341e-04	7.8125e-04	6.0492e-03
14	7	196	8.4650e-02	1.6692e-04	7.8125e-04	7.1461e-03
16	8	256	6.5051e-02	6.4628e-04	1.5625e-03	3.1596e-03

Example 3: $\ell_{0.5}$ norm minimization

Recover a sparse signal by solving

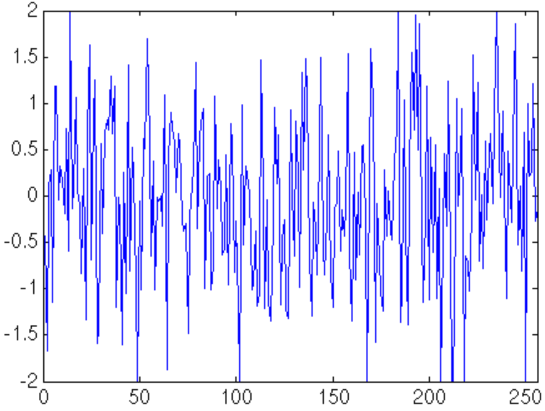
$$\begin{aligned} \min_x \ & \|x\|_{0.5} \\ \text{s.t. } & Ax = b \end{aligned}$$

where A is a 64×256 submatrix of a discrete cosine transform (DCT) matrix.



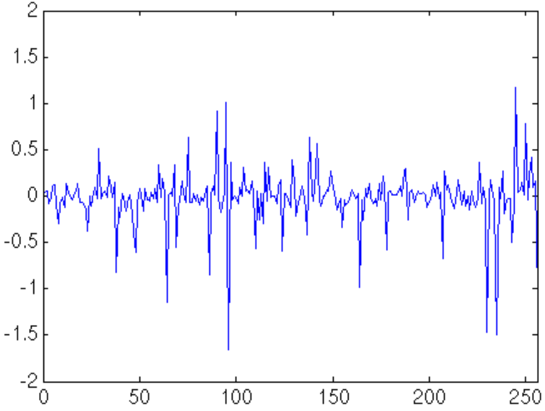
(Use $\ell_{0.5}$ norm as ℓ_1 does not recover sparse solution.)

Example 3: $\ell_{0.5}$ norm minimization



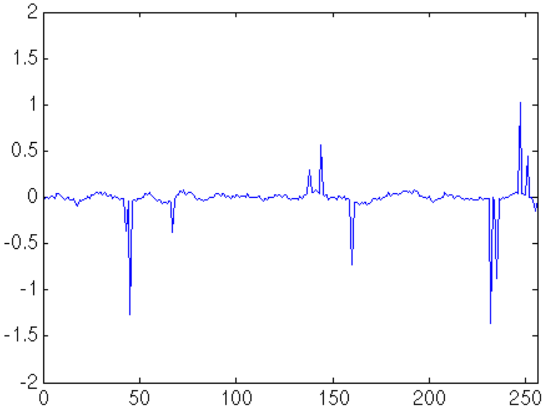
$k = 1$

Example 3: $\ell_{0.5}$ norm minimization



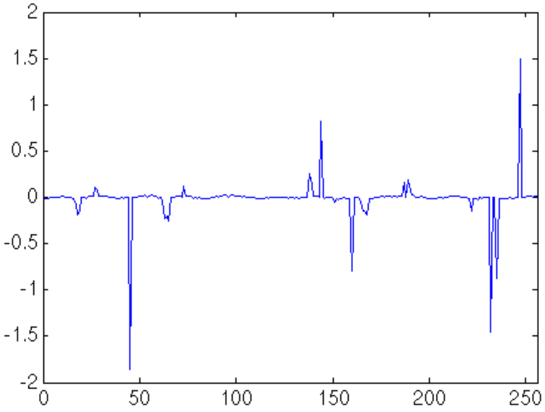
$k = 10$

Example 3: $\ell_{0.5}$ norm minimization



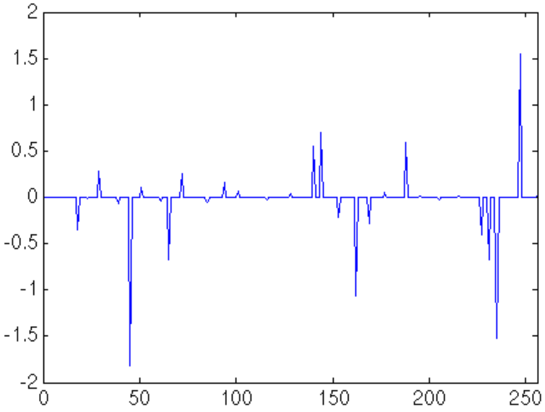
$k = 25$

Example 3: $\ell_{0.5}$ norm minimization



$k = 50$

Example 3: $\ell_{0.5}$ norm minimization



$k = 200$

Example 4: Robust optimization

Find the robust minimizer of a linear objective s.t. an uncertain quadratic constraint:

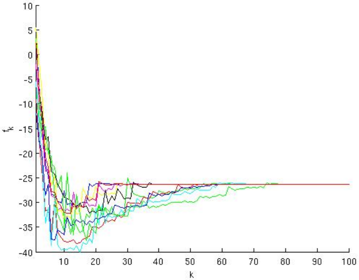
$$\min_x \; f^T x \quad \text{s.t.} \; x^T A x + b^T x + c \leq 0, \; \forall (A, b, c) \in \mathcal{U},$$

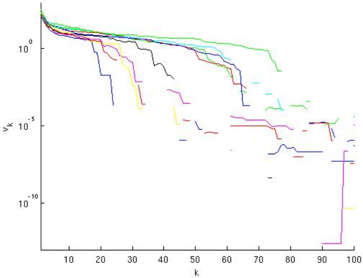
where $f \in \mathbb{R}^n$ and for each (A, b, c) in the uncertainty set

$$\mathcal{U} := \left\{ (A, b, c) : (A, b, c) = (A^{(0)}, b^{(0)}, c^{(0)}) + \sum_{i=1}^{10} u^i (A^{(i)}, b^{(i)}, c^{(i)}), \; u^T u \leq 1 \right\}$$

$A \in \mathbb{R}^{n \times n}$ is positive semidefinite, $b \in \mathbb{R}^n$, and $c \in \mathbb{R}$.

Example 4: Robust optimization





Plot of function values (left) and constraint violation values (right)

Outline

Summary

We set out to improve the practicality and enhance GS methods.

- We aimed to reduce overall gradient evaluations.
- We aimed to reduce the cost of the subproblem solves.
- We aimed to maintain convergence guarantees.
- We aimed to extend the methodology to constrained optimization.

The first goals can be achieved with adaptive sampling and Hessian approximations:

- $O(1)$ gradient evaluations required per iteration
- Subproblem solver warm-started effectively
- Hessian updating schemes improve performance
- Global convergence guarantees maintained

Last goal can be achieved in a SQP-GS framework with constraint gradient sampling:

- Subproblem solve is still a QO per iteration
- Global convergence guarantees maintained

Future work

- C++ implementation
- Tailored QO solver for constrained case
- Adaptive sampling in constrained case
- Special handling of partly smooth functions
- Merge with bundle techniques for convex problems

Thanks!

- F. E. Curtis and X. Que, "An Adaptive Gradient Sampling Algorithm for Nonsmooth Optimization," in 2^{nd} review for *Optimization Methods and Software*.
- F. E. Curtis and M. L. Overton, "A Sequential Quadratic Programming Method for Nonconvex, Nonsmooth Constrained Optimization," to appear in *SIAM Journal on Optimization*.