Gradient Sampling Methods with Inexact Subproblem Solves and Gradient Aggregation

Frank E. Curtis, Lehigh University

joint work with

Minhan Li, Lehigh University

presented at

SIAM Conference on Computational Science and Engineering

March 4, 2021
Outline

Motivation

Inexact Subproblem Solutions

Gradient Aggregation

Conclusion
Outline

Motivation

Inexact Subproblem Solutions

Gradient Aggregation

Conclusion
Locally Lipschitz optimization

Consider optimization problems of the form

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is

- locally Lipschitz over $\mathbb{R}^n$;
- continuously differentiable on an open set $\mathcal{D}_f$ that has full measure in $\mathbb{R}^n$.

Our goal is to improve upon the gradient sampling methodology.
Main idea

If $f$ is smooth, then the steepest descent direction at $x_k$ is $-\nabla f(x_k)$ since

$$
\min_{\|d\|_2 \leq 1} f'(x_k, d) = \min_{\|d\|_2 \leq 1} \nabla f(x_k)^T d = -\frac{\nabla f(x_k)}{\|\nabla f(x_k)\|_2}.
$$
Main idea

If $f$ is smooth, then the steepest descent direction at $x_k$ is $-\nabla f(x_k)$ since

$$\min_{\|d\|_2 \leq 1} f'(x_k, d) = \min_{\|d\|_2 \leq 1} \nabla f(x_k)^T d = -\frac{\nabla f(x_k)}{\|\nabla f(x_k)\|_2}.$$

If $f$ is locally Lipschitz, then ideally one can solve

$$\min_{\|d\|_2 \leq 1} f^\circ (x_k, d) = \arg \min_{\|d\|_2 \leq 1} \left( \max_{g \in \partial f(x_k)} g^T d \right).$$
Main idea

If $f$ is smooth, then the steepest descent direction at $x_k$ is $-\nabla f(x_k)$ since

$$\min_{\|d\|_2 \leq 1} f'(x_k, d) = \min_{\|d\|_2 \leq 1} \nabla f(x_k)^T d = -\frac{\nabla f(x_k)}{\|\nabla f(x_k)\|_2}.$$

If $f$ is locally Lipschitz, then ideally one can solve

$$\min_{\|d\|_2 \leq 1} f^\circ (x_k, d) = \arg\min_{\|d\|_2 \leq 1} \left( \max_{g \in \partial f(x_k)} g^T d \right).$$

However, this is intractable, so we approximate:

$$\arg\min_{\|d\|_2 \leq 1} \left( \max_{g \in \partial_{\epsilon_k} f(x_k)} g^T d \right) \approx \arg\min_{\|d\|_2 \leq 1} \left( \max_{g \in G_k} g^T d \right) = -\frac{g_k}{\|g_k\|_2},$$

where $g_k$ is the min-norm element of $G_k := \{\nabla f(x_k), \nabla f(x_k,1), \ldots, \nabla f(x_k,p)\}$. 
Gradient sampling (Burke, Lewis, and Overton)

At a given iterate $x_k \in \mathbb{R}^n$ and with a sampling radius $\epsilon_k \in \mathbb{R}_{>0}$:

- **sample** $p \geq n + 1$ points in $\mathbb{B}(x_k, \epsilon_k)$
- **evaluate** $G_k := \{\nabla f(x_k), \nabla f(x_{k,1}), \ldots, \nabla f(x_{k,p})\}$
- **compute** the minimum norm element of $\text{conv}(G_k)$, call it $g_k$
- **check** $\|g_k\|_2 = O(\epsilon_k)$; if so, then set $\epsilon_{k+1} < \epsilon_k$; else $\epsilon_{k+1} \leftarrow \epsilon_k$
- **perform** a backtracking line search to obtain $x_k - \alpha_k g_k$
- **perturb** $x_k - \alpha_k g_k \approx x_{k+1}$ (if necessary) to ensure $x_{k+1} \in \mathcal{D}_f$

With probability one, either:

(i) $\{f(x_k)\} \downarrow -\infty$

(ii) $\{\epsilon_k\} \downarrow 0$ and every limit point of $\{x_k\}$ is stationary for $f$.
Gradient sampling (Burke, Lewis, and Overton)

At a given iterate $x_k \in \mathbb{R}^n$ and with a sampling radius $\epsilon_k \in \mathbb{R}_{>0}$:

- **sample** $p \geq n + 1$ points in $B(x_k, \epsilon_k)$
- **evaluate** $G_k := \{\nabla f(x_k), \nabla f(x_{k,1}), \ldots, \nabla f(x_{k,p})\}$
- **compute** the minimum norm element of $\text{conv}(G_k)$, call it $g_k$
- **check** $\|g_k\|_2 = O(\epsilon_k)$; if so, then set $\epsilon_{k+1} < \epsilon_k$; else $\epsilon_{k+1} \leftarrow \epsilon_k$
- **perform** a backtracking line search to obtain $x_k - \alpha_k g_k$
- **perturb** $x_k - \alpha_k g_k \approx x_{k+1}$ (if necessary) to ensure $x_{k+1} \in \mathcal{D}_f$

With probability one, either:

(i) $\{f(x_k)\} \searrow -\infty$ or

(ii) $\{\epsilon_k\} \searrow 0$ and every limit point of $\{x_k\}$ is stationary for $f$. 
Shortcomings and enhancements

Potential shortcomings of the basic algorithm:

▶ $p \geq n + 1$ gradient evaluations per iteration
▶ no (approximate) second-order information
▶ no exploitation of structure of nonsmoothness

Proposed enhancements:

▶ adaptive sampling (Curtis and Que)
▶ variable-metric variants (Curtis and Que)
▶ manifold sampling (Khan, Larson, Menickelly, Wild, Zhou)
Shortcoming and our contribution

In all of the algorithms mentioned so far:

▶ QP subproblems have potentially many constraints, and
▶ QP subproblems need to be solved exactly in each iteration

Our contributions:

▶ inexact subproblem solves
▶ gradient aggregation to limit subproblem sizes

...all while maintaining convergence guarantees of the basic method.
Outline

Motivation

Inexact Subproblem Solutions

Gradient Aggregation

Conclusion
QP subproblems

Primal-dual form of the gradient sampling QP subproblems:

\[
\begin{align*}
\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} & \quad z + \frac{1}{2} \|d\|^2_{H_k} \\
\text{s.t.} & \quad G_k^T d \leq z \mathbf{1} \\
\max_{y \in \mathbb{R}^{P_k+1}} & \quad -\frac{1}{2} \|G_k y_k\|^2_{W_k} \\
\text{s.t.} & \quad \mathbf{1}^T y = 1, \, y \geq 0
\end{align*}
\]

- \(p_k\) is the number of gradients available (in addition to \(\nabla f(x_k)\))
- \(G_k\) is a matrix with gradients as columns
- \(H_k\) is a Hessian approximation
- \(W_k = H_k^{-1}\) is an inverse Hessian approximation
QP subproblems

Primal-dual form of the gradient sampling QP subproblems:

\[
\begin{align*}
\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} & \quad z + \frac{1}{2} \|d\|^2_{H_k} \\
\text{s.t.} & \quad G_k^T d \leq z1 \\
\max_{y \in \mathbb{R}^{p_k+1}} & \quad -\frac{1}{2} \|G_k y_k\|^2_{W_k} \\
\text{s.t.} & \quad 1^T y = 1, \ y \geq 0
\end{align*}
\]

- \(p_k\) is the number of gradients available (in addition to \(\nabla f(x_k)\))
- \(G_k\) is a matrix with gradients as columns
- \(H_k\) is a Hessian approximation
- \(W_k = H_k^{-1}\) is an inverse Hessian approximation

Given feasible \(y_{k,j}\), a corresponding primal feasible solution:

- \(d_{k,j} \leftarrow -W_k G_k y_{k,j}\)
- \(z_{k,j} \leftarrow \max_{i \in \{0,...,p_k\}} \nabla f(x_k,i)^T d_{k,j}\)

As \(y_{k,j} \to y_{k,*}\), this converges to primal-dual solution.
Primal-dual termination test

Consider the primal and dual objective functions:

\[ q_k(d, z) = z + \frac{1}{2} \|d\|_{H_k}^2 \quad \text{and} \quad \theta_k(y) = -\frac{1}{2} \|G_k y\|_{W_k}^2 \]

Given a prescribed inexactness parameter \( \sigma_k \in (0, \infty) \):

\[ \theta_k(y_k, \ast) = q_k(d_k, \ast, z_k, \ast) \]

\( \theta_k(y_k, \ast) \quad \text{at} \quad 0 \]

\[ = q_k(d_k, \ast, z_k, \ast) \]
Primal-dual termination test

Consider the primal and dual objective functions:

\[ q_k(d, z) = z + \frac{1}{2} \| d \|^2_{H_k} \quad \text{and} \quad \theta_k(y) = -\frac{1}{2} \| G_k y \|^2_{W_k} \]

Given a prescribed inexactness parameter \( \sigma_k \in (0, \infty) \):

\[ (1 + \sigma_k)^2 \theta_k(y_k,*) = q_k(d_k,*,z_k,*) \]

want to compute \( y_{k,j} \)

with \( \theta_k(y_{k,j}) \) in this range

without knowing \( \theta_k(y_k,*) \)
Termination test 1

If the following condition is satisfied

\[
q_k(d_{k,j}, z_{k,j}) - \theta_k(y_{k,j}) \leq (\sigma_k^2 + 2\sigma_k) \left( 0 - q_k(d_{k,j}, z_{k,j}) \right)
\]

then the desired condition is satisfied.

\[
\begin{align*}
\theta_k(y_{k,j}) & \quad q_k(d_{k,j}, z_{k,j}) & \quad 0 \\
(1 + \sigma_k)^2\theta_k(y_{k,*}) & \quad \theta_k(y_{k,*}) & \quad 0 \\
= q_k(d_{k,*}, z_{k,*})
\end{align*}
\]
Termination test 2

If the following condition is satisfied (for some $\rho \in (0, 1)$)

$$\theta_k(y_{k,j}) - \theta_k(y_{k,0}) \geq \left( \max \left\{ 1 - \frac{\sigma_k^2 + 2\sigma_k}{\theta_k(y_{k,0})/q_k(d_{k,j}, z_{k,j}) - 1}, \rho \right\} \right) (q_k(d_{k,j}, z_{k,j}) - \theta_k(y_{k,0}))$$

then the desired condition is satisfied.

\[ \theta_k(y_{k,0}) \quad \theta_k(y_{k,j}) \quad q_k(d_{k,j}, z_{k,j}) \]

\[ (1 + \sigma_k)^2 \theta_k(y_{k,*}) \quad \theta_k(y_{k,*}) \quad 0 \]

\[ = q_k(d_{k,*}, z_{k,*}) \]
Complete algorithm

The complete algorithm involves

- adaptive sampling (in some iterations, no sampling)
- L-BFGS Hessian approximations
- inexact subproblem solves
Numerical experiments with **NonOpt**

### Table: Results for GS-exact.

<table>
<thead>
<tr>
<th>name</th>
<th>obj</th>
<th>its</th>
<th>f evs</th>
<th>g evs</th>
<th>qp its</th>
<th>CPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaxQ</td>
<td>3.050E-07</td>
<td>3717</td>
<td>14216</td>
<td>5912</td>
<td>6859</td>
<td>30.55</td>
</tr>
<tr>
<td>MxHilb</td>
<td>1.820E-05</td>
<td>526</td>
<td>5597</td>
<td>4416</td>
<td>2006</td>
<td>40.79</td>
</tr>
<tr>
<td>ChainedLQ</td>
<td>-3.946E+02</td>
<td>268</td>
<td>4397</td>
<td>6631</td>
<td>60789</td>
<td>35.91</td>
</tr>
<tr>
<td>ChainedCB3_1</td>
<td>6.180E+02</td>
<td>337</td>
<td>4858</td>
<td>6046</td>
<td>45035</td>
<td>36.59</td>
</tr>
<tr>
<td>ChainedCB3_2</td>
<td>4.818E+03</td>
<td>95</td>
<td>591</td>
<td>292</td>
<td>598</td>
<td>33.88</td>
</tr>
<tr>
<td>ActiveFaces</td>
<td>3.083E-02</td>
<td>21</td>
<td>669</td>
<td>619</td>
<td>6173</td>
<td>37.15</td>
</tr>
<tr>
<td>BrownFunction_2</td>
<td>3.347E-03</td>
<td>233</td>
<td>3647</td>
<td>4843</td>
<td>31333</td>
<td>31.82</td>
</tr>
<tr>
<td>ChainedMifflin_2</td>
<td>-1.550E+02</td>
<td>482</td>
<td>9991</td>
<td>18229</td>
<td>160818</td>
<td>62.86</td>
</tr>
<tr>
<td>ChainedCrescent_1</td>
<td>5.197E-03</td>
<td>33</td>
<td>252</td>
<td>201</td>
<td>128</td>
<td>33.33</td>
</tr>
<tr>
<td>ChainedCrescent_2</td>
<td>1.258E-03</td>
<td>397</td>
<td>6441</td>
<td>9462</td>
<td>77608</td>
<td>47.18</td>
</tr>
<tr>
<td>Test29_2</td>
<td>4.840E-05</td>
<td>966</td>
<td>9390</td>
<td>7096</td>
<td>18357</td>
<td>38.39</td>
</tr>
<tr>
<td>Test29_5</td>
<td>9.194E-05</td>
<td>508</td>
<td>4311</td>
<td>2373</td>
<td>3736</td>
<td>39.07</td>
</tr>
<tr>
<td>Test29_6</td>
<td>2.263E-04</td>
<td>706</td>
<td>9305</td>
<td>9479</td>
<td>40356</td>
<td>33.13</td>
</tr>
<tr>
<td>Test29_11</td>
<td>1.913E+03</td>
<td>347</td>
<td>5216</td>
<td>7693</td>
<td>66261</td>
<td>35.65</td>
</tr>
<tr>
<td>Test29_13</td>
<td>1.747E+02</td>
<td>338</td>
<td>6313</td>
<td>10516</td>
<td>66438</td>
<td>41.17</td>
</tr>
<tr>
<td>Test29_17</td>
<td>3.961E-05</td>
<td>408</td>
<td>5341</td>
<td>4296</td>
<td>14739</td>
<td>42.63</td>
</tr>
<tr>
<td>Test29_19</td>
<td>6.247E-08</td>
<td>644</td>
<td>7561</td>
<td>9105</td>
<td>45696</td>
<td>43.67</td>
</tr>
<tr>
<td>Test29_20</td>
<td>1.339E-04</td>
<td>1777</td>
<td>21947</td>
<td>23897</td>
<td>110077</td>
<td>40.45</td>
</tr>
<tr>
<td>Test29_22</td>
<td>4.539E-05</td>
<td>40</td>
<td>574</td>
<td>377</td>
<td>10453</td>
<td>66.28</td>
</tr>
<tr>
<td>Test29_24</td>
<td>5.562E-05</td>
<td>2708</td>
<td>49258</td>
<td>89275</td>
<td>349192</td>
<td>76.25</td>
</tr>
</tbody>
</table>
Numerical experiments with NonOpt

<table>
<thead>
<tr>
<th>name</th>
<th>obj</th>
<th>its</th>
<th>f evs</th>
<th>g evs</th>
<th>qp its</th>
<th>CPU</th>
<th>CPU diff</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaxQ</td>
<td>2.870E-07</td>
<td>3863</td>
<td>14676</td>
<td>6083</td>
<td>6121</td>
<td>27.75</td>
<td>-9.18%</td>
</tr>
<tr>
<td>MxHilb</td>
<td>2.000E-05</td>
<td>464</td>
<td>5872</td>
<td>4410</td>
<td>1835</td>
<td>40.50</td>
<td>-0.69%</td>
</tr>
<tr>
<td>ChainedLQ</td>
<td>-3.946E+02</td>
<td>247</td>
<td>3855</td>
<td>5510</td>
<td>54310</td>
<td>34.55</td>
<td>-3.81%</td>
</tr>
<tr>
<td>ChainedCB3_1</td>
<td>6.180E+02</td>
<td>336</td>
<td>4712</td>
<td>5444</td>
<td>31627</td>
<td>22.33</td>
<td>-38.98%</td>
</tr>
<tr>
<td>ChainedCB3_2</td>
<td>4.818E+03</td>
<td>94</td>
<td>614</td>
<td>213</td>
<td>387</td>
<td>22.94</td>
<td>-32.28%</td>
</tr>
<tr>
<td>ActiveFaces</td>
<td>3.083E-02</td>
<td>21</td>
<td>669</td>
<td>619</td>
<td>23</td>
<td>1.27</td>
<td>-96.58%</td>
</tr>
<tr>
<td>BrownFunction_2</td>
<td>3.131E-03</td>
<td>256</td>
<td>3917</td>
<td>4861</td>
<td>32461</td>
<td>32.34</td>
<td>1.65%</td>
</tr>
<tr>
<td>ChainedMifflin_2</td>
<td>-1.550E+02</td>
<td>442</td>
<td>9214</td>
<td>16351</td>
<td>127904</td>
<td>50.16</td>
<td>-20.21%</td>
</tr>
<tr>
<td>ChainedCrescent_1</td>
<td>4.627E-03</td>
<td>34</td>
<td>256</td>
<td>202</td>
<td>123</td>
<td>32.01</td>
<td>-3.96%</td>
</tr>
<tr>
<td>ChainedCrescent_2</td>
<td>1.065E-03</td>
<td>332</td>
<td>5735</td>
<td>8969</td>
<td>68458</td>
<td>44.08</td>
<td>-6.58%</td>
</tr>
<tr>
<td>Test29_2</td>
<td>4.961E-05</td>
<td>942</td>
<td>9384</td>
<td>6877</td>
<td>16719</td>
<td>38.35</td>
<td>-0.11%</td>
</tr>
<tr>
<td>Test29_5</td>
<td>6.824E-04</td>
<td>216</td>
<td>1455</td>
<td>800</td>
<td>1316</td>
<td>13.80</td>
<td>-64.67%</td>
</tr>
<tr>
<td>Test29_6</td>
<td>2.034E-04</td>
<td>703</td>
<td>9297</td>
<td>9493</td>
<td>35815</td>
<td>29.99</td>
<td>-9.48%</td>
</tr>
<tr>
<td>Test29_11</td>
<td>1.913E+03</td>
<td>433</td>
<td>5686</td>
<td>7806</td>
<td>98615</td>
<td>46.45</td>
<td>30.28%</td>
</tr>
<tr>
<td>Test29_13</td>
<td>1.747E+02</td>
<td>363</td>
<td>7356</td>
<td>12937</td>
<td>71641</td>
<td>46.94</td>
<td>14.03%</td>
</tr>
<tr>
<td>Test29_17</td>
<td>4.549E-05</td>
<td>410</td>
<td>5337</td>
<td>4237</td>
<td>13976</td>
<td>41.63</td>
<td>-2.34%</td>
</tr>
<tr>
<td>Test29_19</td>
<td>3.507E-08</td>
<td>580</td>
<td>6804</td>
<td>8239</td>
<td>37114</td>
<td>35.94</td>
<td>-17.71%</td>
</tr>
<tr>
<td>Test29_20</td>
<td>1.158E+04</td>
<td>425</td>
<td>6222</td>
<td>9015</td>
<td>42825</td>
<td>16.48</td>
<td>-59.25%</td>
</tr>
<tr>
<td>Test29_22</td>
<td>6.458E-05</td>
<td>36</td>
<td>526</td>
<td>347</td>
<td>355</td>
<td>2.91</td>
<td>-95.61%</td>
</tr>
<tr>
<td>Test29_24</td>
<td>4.351E-05</td>
<td>2197</td>
<td>34581</td>
<td>59774</td>
<td>287530</td>
<td>55.83</td>
<td>-26.77%</td>
</tr>
</tbody>
</table>
Outline

Motivation

Inexact Subproblem Solutions

Gradient Aggregation

Conclusion
Subgradient aggregation

Subgradient aggregation is a well-known technique for bundle methods.
  ▶ It has not previously been used in gradient sampling,
  ▶ and generally is harder to employ in nonconvex settings.

However, since it can *drastically* reduce the size of subproblems, it’s worth a try.
Gradient aggregation

Recall the primal-dual form of the gradient sampling QP subproblems:

\[
\begin{aligned}
&\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} \quad z + \frac{1}{2} \|d\|_{H_k}^2 \\
&\text{s.t. } G_k^T d \leq z \mathbf{1}
\end{aligned}
\]

\[
\begin{aligned}
&\max_{y \in \mathbb{R}^{p_k+1}} \quad -\frac{1}{2} \|G_k y_k\|_{W_k}^2 \\
&\text{s.t. } \mathbf{1}^T y = 1, \quad y \geq 0
\end{aligned}
\]

At the primal-dual optimal solution:

\[
d_{k,*} = -W_k G_k y_k,*
\]
Gradient aggregation

Recall the primal-dual form of the gradient sampling QP subproblems:

\[
\begin{align*}
\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} & \quad z + \frac{1}{2} \|d\|^2_{H_k} \\
\text{s.t.} & \quad G_k^T d \leq z 1
\end{align*}
\]

\[
\begin{align*}
\max_{y \in \mathbb{R}^{p_k+1} \setminus \mathbb{R}^n} & \quad -\frac{1}{2} \|G_k y_k\|^2_{W_k} \\
\text{s.t.} & \quad 1^T y = 1, \quad y \geq 0
\end{align*}
\]

At the primal-dual optimal solution:

\[d_k,^* = -W_k G_k y_k,^*\]

Hence, the primal optimal solution is also a solution to

\[
\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} z + \frac{1}{2} \|d\|^2_{H_k}
\]

\[
\text{s.t.} \quad (G_k y_k,^*)^T d \leq z \quad \leftarrow \text{single constraint!}
\]
Gradient aggregation

Recall the primal-dual form of the gradient sampling QP subproblems:

\[
\begin{align*}
\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} & \quad z + \frac{1}{2} \|d\|_H^2 \\
\text{s.t.} & \quad G_k^T d \leq z \mathbf{1}
\end{align*}
\]

\[
\begin{align*}
\max_{y \in \mathbb{R}^{p_k+1}} & \quad - \frac{1}{2} \|G_k y_k\|_{W_k}^2 \\
\text{s.t.} & \quad \mathbf{1}^T y = 1, \ y \geq 0
\end{align*}
\]

At the primal-dual optimal solution:

\[d_k,^* = -W_k G_k y_k,^*\]

Hence, the primal optimal solution is also a solution to

\[
\begin{align*}
\min_{(z,d) \in \mathbb{R} \times \mathbb{R}^n} & \quad z + \frac{1}{2} \|d\|_H^2 \\
\text{s.t.} & \quad (G_k y_k,^*)^T d \leq z \quad \leftarrow \text{single constraint!}
\end{align*}
\]

If the adaptive sampling strategy is to augment \(G_k\), then replace:

\[
\underbrace{G_k}_{p_k + 1 \text{ columns}} \quad \text{with} \quad \underbrace{G_k y_k,^*}_{1 \text{ column}}
\]
Numerical experiments with NonOpt

Table: Results for GS-inexact-agg.

<table>
<thead>
<tr>
<th>name</th>
<th>obj</th>
<th>its</th>
<th>f evs</th>
<th>g evs</th>
<th>qp its</th>
<th>CPU</th>
<th>CPU diff</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaxQ</td>
<td>2.460E-07</td>
<td>3539</td>
<td>13171</td>
<td>4967</td>
<td>5387</td>
<td>26.01</td>
<td>-14.87%</td>
</tr>
<tr>
<td>MxHilb</td>
<td>1.115E-04</td>
<td>429</td>
<td>4184</td>
<td>2696</td>
<td>1826</td>
<td>31.90</td>
<td>-21.80%</td>
</tr>
<tr>
<td>ChainedLQ</td>
<td>-3.946E+02</td>
<td>229</td>
<td>5286</td>
<td>6861</td>
<td>58896</td>
<td>38.35</td>
<td>6.79%</td>
</tr>
<tr>
<td>ChainedCB3_1</td>
<td>6.180E+02</td>
<td>285</td>
<td>5698</td>
<td>6630</td>
<td>22419</td>
<td>16.57</td>
<td>-54.71%</td>
</tr>
<tr>
<td>ChainedCB3_2</td>
<td>4.818E+03</td>
<td>89</td>
<td>561</td>
<td>238</td>
<td>483</td>
<td>27.96</td>
<td>-17.48%</td>
</tr>
<tr>
<td>ActiveFaces</td>
<td>3.083E-02</td>
<td>21</td>
<td>669</td>
<td>619</td>
<td>23</td>
<td>1.22</td>
<td>-96.72%</td>
</tr>
<tr>
<td>BrownFunction_2</td>
<td>1.843E-03</td>
<td>238</td>
<td>4533</td>
<td>4872</td>
<td>19376</td>
<td>17.60</td>
<td>-44.68%</td>
</tr>
<tr>
<td>ChainedMifflin_2</td>
<td>-1.550E+02</td>
<td>516</td>
<td>12762</td>
<td>16994</td>
<td>187575</td>
<td>69.35</td>
<td>10.32%</td>
</tr>
<tr>
<td>ChainedCrescent_1</td>
<td>2.795E-03</td>
<td>24</td>
<td>141</td>
<td>66</td>
<td>71</td>
<td>18.90</td>
<td>-43.29%</td>
</tr>
<tr>
<td>ChainedCrescent_2</td>
<td>9.704E-04</td>
<td>315</td>
<td>6123</td>
<td>6851</td>
<td>29742</td>
<td>22.27</td>
<td>-52.80%</td>
</tr>
<tr>
<td>Test29_2</td>
<td>5.104E-05</td>
<td>1108</td>
<td>11368</td>
<td>6307</td>
<td>14573</td>
<td>30.91</td>
<td>-19.49%</td>
</tr>
<tr>
<td>Test29_5</td>
<td>7.822E-05</td>
<td>414</td>
<td>3768</td>
<td>1825</td>
<td>3340</td>
<td>33.84</td>
<td>-13.39%</td>
</tr>
<tr>
<td>Test29_6</td>
<td>2.326E-04</td>
<td>886</td>
<td>12819</td>
<td>10279</td>
<td>33892</td>
<td>28.40</td>
<td>-14.27%</td>
</tr>
<tr>
<td>Test29_11</td>
<td>1.913E+03</td>
<td>324</td>
<td>5563</td>
<td>5714</td>
<td>30196</td>
<td>8.62</td>
<td>-75.82%</td>
</tr>
<tr>
<td>Test29_13</td>
<td>1.747E+02</td>
<td>253</td>
<td>6419</td>
<td>8598</td>
<td>34949</td>
<td>27.47</td>
<td>-33.27%</td>
</tr>
<tr>
<td>Test29_17</td>
<td>5.141E-05</td>
<td>425</td>
<td>5662</td>
<td>3469</td>
<td>9700</td>
<td>27.78</td>
<td>-34.83%</td>
</tr>
<tr>
<td>Test29_19</td>
<td>1.050E-01</td>
<td>492</td>
<td>7941</td>
<td>8170</td>
<td>17696</td>
<td>15.14</td>
<td>-65.33%</td>
</tr>
<tr>
<td>Test29_20</td>
<td>1.329E-04</td>
<td>532</td>
<td>7971</td>
<td>7889</td>
<td>54621</td>
<td>18.85</td>
<td>-53.39%</td>
</tr>
<tr>
<td>Test29_22</td>
<td>4.850E-05</td>
<td>41</td>
<td>631</td>
<td>364</td>
<td>10405</td>
<td>45.62</td>
<td>-31.17%</td>
</tr>
<tr>
<td>Test29_24</td>
<td>3.826E-05</td>
<td>2413</td>
<td>42786</td>
<td>54909</td>
<td>337121</td>
<td>55.02</td>
<td>-27.84%</td>
</tr>
</tbody>
</table>
Outline

Motivation

Inexact Subproblem Solutions

Gradient Aggregation

Conclusion
Summary

Shortcomings of gradient sampling methods to date:
- QP subproblems have potentially many constraints, and
- QP subproblems need to be solved exactly in each iteration

Our contributions:
- *inexact* subproblem solves
- *gradient aggregation* to limit subproblem sizes
  ...all while maintaining convergence guarantees of the basic method.