An Average Curvature Accelerated Composite Gradient (ACG) Method for Nonconvex Smooth Composite Optimization Problems

Jiaming Liang[1] Renato D.C. Monteiro[1]

[1]School of Industrial and Systems Engineering, Georgia Tech

INFORMS Annual Meeting - October 22, 2019

The main problem:

$$(P) \qquad \min\left\{f(z)+h(z) : z\in\mathbb{R}^n\right\}$$

where

- $h:\mathbb{R}^n\to(-\infty,\infty]$ is a closed proper convex function such that
$$D := \sup\{\|z'-z\| : z,z'\in \operatorname{dom} h\} < \infty$$
- f is differentiable (not necessarily convex) on $\operatorname{dom} h$ and there exist $0<m\le L$ such that for every $z,z'\in\operatorname{dom} h$
$$\|\nabla f(z')-\nabla f(z)\| \le L\|z'-z\|$$
$$f(z')-\ell_f(z';z) \ge -\frac{m}{2}\|z'-z\|^2$$

where $\ell_f(z';z) := f(z)+\langle\nabla f(z), z'-z\rangle$.

A necessary condition for $\bar{z}$ to be a local minimizer of (P) is that

$$0 \in \nabla f(\bar{z}) + \partial h(\bar{z})$$

Goal: for given $\hat{\rho} > 0$, find a $\hat{\rho}$-approximate solution of (P), i.e., a pair $(\hat{z}, \hat{v})$ such that

$$\hat{v} \in \nabla f(\hat{z}) + \partial h(\hat{z}), \quad \|\hat{v}\| \le \hat{\rho}$$

There are a couple of ACG methods which accomplishes the above goal (e.g., Ghadimi-Lan's method). This talk describes a different and novel ACG method for doing that.

Traditional adaptive ACG methods compute the next iterate as

$$z_{k+1} = z_{k+1}(M_k) := \operatorname{argmin}_z \left\{ \ell_f(z;\tilde{x}_k) + h(z) + \frac{M_k}{2}\|z-\tilde{x}_k\|^2 \right\}$$

where $\tilde{x}_k$ is a convex combination of z_k and another auxiliary iterate x_k, and $M_k > 0$ is chosen so as to satisfy

$$M_k \geq \mathcal{C}(z_{k+1};\tilde{x}_k) := \frac{2[f(z_{k+1}) - \ell(z_{k+1};\tilde{x}_k)]}{\|z_{k+1}-\tilde{x}_k\|^2} \quad (*)$$

Choosing M_k as the smallest one satisfying $(*)$ results in faster convergence rate but finding an approximation to this M_k leads to an expensive line search on M_k. A sufficient condition for $(*)$ is to impose the maximum curvature condition

$$M_k \geq \max_{i=0,\ldots,k} \mathcal{C}(z_{i+1};\tilde{x}_i)$$

This strategy leads to a simpler search for M_k but results in a relatively large M_k.

Traditional adaptive ACG methods compute the next iterate as

$$z_{k+1} = z_{k+1}(M_k) := \operatorname{argmin}_z \left\{ \ell_f(z;\tilde{x}_k) + h(z) + \frac{M_k}{2}\|z - \tilde{x}_k\|^2 \right\}$$

where $\tilde{x}_k$ is a convex combination of z_k and another auxiliary iterate x_k, and $M_k > 0$ is chosen so as to satisfy

$$M_k \geq \mathcal{C}(z_{k+1};\tilde{x}_k) := \frac{2[f(z_{k+1}) - \ell(z_{k+1};\tilde{x}_k)]}{\|z_{k+1} - \tilde{x}_k\|^2} \quad (*)$$

Choosing M_k as the smallest one satisfying $(*)$ results in faster convergence rate but finding an approximation to this M_k leads to an expensive line search on M_k. A sufficient condition for $(*)$ is to impose the maximum curvature condition

$$M_k \geq \max_{i=0,\ldots,k} \mathcal{C}(z_{i+1};\tilde{x}_i)$$

This strategy leads to a simpler search for M_k but results in a relatively large M_k.

Traditional adaptive ACG methods compute the next iterate as

$$z_{k+1} = z_{k+1}(M_k) := \mathrm{argmin}_z \left\{ \ell_f(z;\tilde{x}_k) + h(z) + \frac{M_k}{2}\|z-\tilde{x}_k\|^2 \right\}$$

where $\tilde{x}_k$ is a convex combination of z_k and another auxiliary iterate x_k, and $M_k > 0$ is chosen so as to satisfy

$$M_k \geq \mathcal{C}(z_{k+1};\tilde{x}_k) := \frac{2[f(z_{k+1}) - \ell(z_{k+1};\tilde{x}_k)]}{\|z_{k+1}-\tilde{x}_k\|^2} \quad (*)$$

Choosing M_k as the smallest one satisfying $(*)$ results in faster convergence rate but finding an approximation to this M_k leads to an expensive line search on M_k. A sufficient condition for $(*)$ is to impose the maximum curvature condition

$$M_k \geq \max_{i=0,\ldots,k} \mathcal{C}(z_{i+1};\tilde{x}_i)$$

This strategy leads to a simpler search for M_k but results in a relatively large M_k.

We will exploit the novel idea of choosing M_k as

$$M_k = \frac{\sum_{i=0}^{k-1} \mathcal{C}(z_{i+1}; \tilde{x}_i)}{k\,\alpha}$$

where $\alpha \in (0, 1)$

Note: No search for M_k is involved here!

Average Curvature ACG (AC-ACG) Method

0. Let $\alpha, \gamma \in (0,1)$, tolerance $\hat{\rho} > 0$ and initial point $z_0 \in \operatorname{dom} h$ be given; set $A_0 = 0$, $x_0 = z_0$, $M_0 = \gamma L$ and $k = 0$
1. compute

$$a_k = \frac{1 + \sqrt{1 + 4M_k A_k}}{2M_k} \qquad A_{k+1} = A_k + a_k \qquad \tilde{x}_k = \frac{A_k z_k + a_k x_k}{A_{k+1}}$$

2. compute

$$x_{k+1} = \operatorname{argmin}_u \left\{ a_k \left(\ell_f(u; \tilde{x}_k) + h(u) \right) + \frac{1}{2} \|u - x_k\|^2 \right\}$$

$$z_{k+1}^g = \operatorname{argmin}_u \left\{ \ell_f(u; \tilde{x}_k) + h(u) + \frac{M_k}{2} \|u - \tilde{x}_k\|^2 \right\}$$

$$v_{k+1} = M_k(\tilde{x}_k - z_{k+1}^g) + \nabla f(z_{k+1}^g) - \nabla f(\tilde{x}_k)$$

3. if $\|v_{k+1}\| \leq \hat{\rho}$ then output $(\hat{z}, \hat{v}) = (z^g_{k+1}, v_{k+1})$ and **stop**; otherwise, compute

$$C_k = \max\left\{ \frac{2\left[f(z^g_{k+1}) - \ell_f(z^g_{k+1}; \tilde{x}_k)\right]}{\|z^g_{k+1} - \tilde{x}_k\|^2}, \frac{\|\nabla f(z^g_{k+1}) - \nabla f(\tilde{x}_k)\|}{\|z^g_{k+1} - \tilde{x}_k\|} \right\}$$

$$C_k^{avg} = \frac{1}{k+1} \sum_{j=0}^{k} C_j$$

$$M_{k+1} = \max\left\{ \frac{1}{\alpha} C_k^{avg}, \gamma L \right\}$$

4. set

$$z_{k+1} = \begin{cases} z^g_{k+1} & \text{if } C_k \leq 0.9 M_k \quad \text{(good iteration)} \\ \frac{A_k z_k + a_k x_{k+1}}{A_{k+1}} & \text{otherwise} \quad \text{(bad iteration)} \end{cases}$$

and $k \leftarrow k+1$, and go to step 1

Remarks:

- both good and bad iterations perform well-known types of acceleration steps
- if
$$\alpha \le \frac{0.9}{8}\left(1+\frac{1}{0.9\gamma}\right)^{-1}$$
then it can be shown that the proportion of good iterations is at least $2/3$
- in practice, α can be much larger, i.e., $\Omega(1)$ instead of $\Omega(\gamma)$
- our implementation sets $\alpha = 0.5$ or 0.7 or 1

1. The Main Problem
   - Assumptions
   - Approximate solutions
2. Average Curvature ACG Method
   - Motivation
   - AC-ACG method
   - Convergence rate and iteration-complexity
   - Proof techniques
3. Computational Results
4. Implementation and Concluding Remarks

Theorem

The following statements hold:

(a) *for every* $k \geq 1$, *we have* $v_k \in \nabla f(z_k) + \partial h(z_k)$

(b) *for every* $k \geq 12$, *we have*

$$\min_{1\leq i\leq k} \|v_i\|^2 \leq \mathcal{O}\left(\frac{M_k^2 D^2}{\gamma k^2} + \frac{\theta_k m M_k D^2}{k}\right)$$

where

$$\theta_k := \max\left\{\frac{M_k}{M_i} : 0 \leq i \leq k\right\} \geq 1.$$

The facts that $\theta_k = \mathcal{O}(1)$ and $M_k/L = \mathcal{O}(1)$ imply that the iteration-complexity bound for AC-ACG to obtain $\hat{\rho}$-approx. sol. is

$$\mathcal{O}\left(\frac{LD}{\hat{\rho}} + \frac{mLD^2}{\hat{\rho}^2} + 1\right)$$

Define

$$\mathcal{G} := \{k \geq 0 : C_k \leq 0.9M_k\}, \quad \mathcal{B} := \{k \geq 0 : C_k > 0.9M_k\},$$

and

$$\mathcal{G}_k = \{i \in \mathcal{G} : i \leq k-1\}, \quad \mathcal{B}_k := \{i \in \mathcal{B} : i \leq k-1\}.$$

The following lemma is the key to the proof of the main theorem.

Lemma

For every $k \geq 1$, $|\mathcal{B}_k| \leq k/4 + 1$. As a consequence, $|\mathcal{B}_k| \leq k/3$ for every $k \geq 12$.

Computational Results

The variant of AC-ACG described above was benchmarked against

- AG method by Ghadimi and Lan (known Lipschitz constant)
- nmAPG method by Li and Lin (known Lipschitz constant)
- UPFAG method by Ghadimi, Lan and Zhang (backtracking)

on **five** classes of problems.

All methods stop with a pair (z, v) satisfying

$$v \in \nabla f(z) + \partial h(z), \qquad \frac{\|v\|}{\|\nabla f(z_0)\| + 1} \leq \hat{\rho}$$

1st Problem (Nonconvex QP):

$$\min \left\{ f(Z) := -\frac{\xi}{2}\|D\mathcal{B}(Z)\|^2 + \frac{\tau}{2}\|\mathcal{A}(Z) - b\|^2 : z \in P_n \right\}$$

where P_n is the unit spectraplex, i.e.,

$$P_n := \left\{ Z \in S^n_+ : \mathrm{tr}(Z) = 1 \right\}$$

$\mathcal{A} : S^n_+ \to \mathbb{R}^\ell$ and $\mathcal{B} : S^n_+ \to \mathbb{R}^p$ are linear operators, $D \in \mathbb{R}^{p\times p}$ is a positive diagonal matrix, and $b \in \mathbb{R}^\ell$ is a vector.

(L,m)	Iteration Count / Running Time (s)				Curvature		Good
	AG	APG	UPFAG	AC	Max	Avg	
$(10^6,10^6)$	69 22.0	117 26.4	13 8.3	8 **3.5**	1.28E5	1.70E4	88%
$(10^6,10^5)$	277 119.0	502 117.7	9 5.7	7 **3.1**	1.80E4	2.84E3	86%
$(10^6,10^4)$	491 173.3	1030 245.5	13 9.1	11 **4.6**	3.26E4	3.89E3	91%
$(10^6,10^3)$	531 168.9	1144 259.3	13 9.1	12 **6.8**	3.41E4	3.73E3	92%
$(10^6,10^2)$	535 171.8	1156 260.2	13 8.6	12 **5.5**	3.42E4	3.75E3	92%
$(10^6,10^1)$	536 172.1	1157 266.1	13 8.3	12 **5.2**	3.43E4	3.75E3	92%

Table: QP — $(l,p,n)=(50,800,1000)$, 0.1% sparse ($\alpha=1$ and $\hat{\rho}=10^{-7}$)

2nd Problem (SVM):

$$\min_{z\in\mathbb{R}^n} \frac{1}{p}\sum_{i=1}^{p} \ell(x_i, y_i; z) + \frac{\lambda}{2}\|z\|^2 + I_{B_r}(z)$$

for some $\lambda, r > 0$, where $x_i \in \mathbb{R}^n$ is a feature vector, $y_i \in \{1, -1\}$ denotes the corresponding label, $\ell(x_i, y_i; \cdot) = 1 - \tanh(y_i\langle\cdot, x_i\rangle)$ is a nonconvex sigmoid loss function and $I_{B_r}(\cdot)$ is the indicator function of $B_r := \{z \in \mathbb{R}^n : \|z\| \leq r\}$.

L	Iteration Count / Running Time (s)				Curvature		Good
	AG	APG	UPFAG	AC	Max	Avg	
13	37384 639	42532 649	130 **8**	546 12	0.25	0.05	67%
25	112562 4419	123551 4486	278 **39**	1131 60	0.47	0.06	65%
38	155503 12636	163197 12101	401 97	1032 **95**	0.34	0.07	63%
50	79752 4406	79064 5264	247 44	615 **39**	0.18	0.07	71%

Table: SVM — $(\lambda, r) = (1/p, 50)$ ($\alpha = 0.5$ and $\hat{\rho} = 10^{-7}$)

3rd Problem (Sparse PCA):

$$\min \langle -\hat{\Sigma}, X\rangle_F + \frac{\mu}{2}\|X\|_F^2 + Q_{\lambda,b}(Y) + \lambda\|Y\|_1 + \frac{\beta}{2}\|X - Y\|_F^2 + I_{\mathcal{F}^k}(X)$$
$$s.t. \quad X, Y \in \mathbb{R}^{p\times p}$$

where $\hat{\Sigma} \in \mathbb{R}^{p\times p}$ is an empirical covariance matrix, μ, λ, β, b are positive scalars,

$$\|Y\|_1 := \sum_{i,j=1}^{p} |Y_{ij}|, \quad Q_{\lambda,b}(X) := \sum_{ij=1}^{p} q_{\lambda,b}(X_{ij})$$

where

$$q_{\lambda,b}(t) := \begin{cases} -\frac{t^2}{2b}, & \text{if } |t| \le b\lambda; \\ \frac{b\lambda^2}{2} - \lambda|t|, & \text{otherwise} \end{cases}$$

and $I_{\mathcal{F}^k}(\cdot)$ is the indicator function of the Fantope

$$\mathcal{F}^k := \{X \in S^n : 0 \preceq X \preceq I \text{ and } \operatorname{tr}(X) = k\}.$$

L	Iteration Count / Running Time (s)				Curvature		Good
	AG	APG	UPFAG	AC	Max	Avg	
2.33	21	18	7	15	2.00	0.72	67%
	8.63	**4.96**	6.71	7.33			
4	7	9	8	7	3.67	3.41	71%
	10.08	**2.73**	7.55	3.94			
63	32	43	18	27	44.41	31.12	89%
	19.91	12.06	17.61	**12.04**			
60.67	35	46	17	31	0.18	0.07	94%
	19.01	14.28	16.97	**12.51**			

Table: Sparse PCA ($\alpha = 0.5$ and $\hat{\rho} = 10^{-7}$)

4th Problem (Constrained matrix completion):

$$\min_{X \in \mathbb{R}^{m \times n}} \left\{ \frac{1}{2} \|\Pi_\Omega(X - O)\|_F^2 + \mu \sum_{i=1}^{r} p(\sigma_i(X)) : \|X\|_F \leq R \right\}$$

where $O \in \mathbb{R}^\Omega$ is an incomplete observed matrix, $\mu > 0$ is a parameter, $r := \min\{m, n\}$, $\sigma_i(X)$ is the i-th singular value of X and

$$p(t) = p_{\beta,\theta}(t) := \beta \log \left(1 + \frac{|t|}{\theta}\right)$$

L	Function Value ×1000 / Iteration Count				Running Time ×1000 seconds				Curvature		Good
	AG	APG	UPFAG	AC	AG	APG	UPFAG	AC	Max	Avg	
4	2.26	**1.81**	2.60	2.29	4.6	1.0	2.6	**0.9**	1.00	0.31	96%
	3856	1036	521	765							
9	3.89	**3.36**	4.26	3.88	10.3	1.6	4.3	**1.2**	1.00	0.28	94%
	9158	1617	576	968							
20	4.28	**3.64**	4.64	4.27	29.2	2.8	4.6	**1.2**	0.99	0.25	91%
	22902	2875	676	1079							
30	5.97	**5.24**	6.75	5.97	41.7	4.2	6.8	**1.3**	0.97	0.23	89%
	37032	3717	606	1085							

Table: MC — 100K MovieLens dataset ($\alpha = 0.5$ and $\hat{\rho} = 5 \times 10^{-4}$)

5th Problem: (Nonnegative matrix factorization)

$$\min \left\{ f(V,W) := \frac{1}{2}\|X - VW\|_F^2 : V \in \mathbb{R}_+^{n\times p}, W \in \mathbb{R}_+^{p\times \ell} \right\}$$

based on a facial image dataset provided by AT&T Laboratories Cambridge

$$n = 10,304 \quad \ell = 400 \quad p = 20$$

Method	Function Value	Iteration Count	Running time(s)
AG	2.80E+09	786	73.03
APG	2.80E+09	87	14.91
UPFAG	2.80E+09	37	11.12
AC	2.80E+09	37	**4.70**

Table: NMF ($\alpha = 0.7$ and $\hat{\rho} = 10^{-7}$)

Implementation Remarks

- We can choose α to control the percentage of good iterations.
- We have been able to solve problems for which $\mathrm{dom}\, h$ is unbounded but sometimes unboundness of $\mathrm{dom}\, h$ can cause difficulty.

Concluding Remarks

- We have presented AC-ACG that is an ACG method based on the average of the previously observed curvatures.
- AC-ACG does not require any line search for M_k.
- We have argued that AC-ACG is quite promising computationally.
- We have established a convergence rate bound for AC-ACG in terms of the average observed curvatures (novel result).
- We have shown that AC-ACG has an iteration-complexity bound that is similar to the ones for other ACG methods (e.g., Lan and Ghadimi's AG method).

THE END

Thanks!