Efficiency of Randomized Coordinate Descent Methods on Minimization Problems with a Composite Objective Function

Martin Takáč
School of Mathematics
The University of Edinburgh
Email: M.Takac@sms.ed.ac.uk

Peter Richtárik
School of Mathematics
The University of Edinburgh
Email: Peter.Richtarik@ed.ac.uk

***Abstract*—We develop a randomized block-coordinate descent method for minimizing the sum of a smooth and a simple nonsmooth block-separable convex function and prove that it obtains an ϵ-accurate solution with probability at least $1-\rho$ in at most $O((4n/\epsilon)\log(1/\epsilon\rho))$ iterations, where n is the dimension of the problem. This extends recent results of Nesterov [2], which cover the smooth case, to composite minimization, and improves the complexity by a factor of 2. In the smooth case we give a much simplified analysis. Finally, we demonstrate numerically that the algorithm is able to solve various ℓ_1-regularized optimization problems with a billion variables.**

I. Introduction

We consider the unconstrained convex optimization problem

$$\min_{x\in\mathbb{R}^N} F(x) \stackrel{\text{def}}{=} f(x)+\Psi(x), \qquad (1)$$

where f is smooth and Ψ is block-separable. By x^* we denote an arbitrary optimal solution of (1) and by F^* the optimal value.

A. Block structure

Let $(\mathcal{U}_1,\dots,\mathcal{U}_n)$ be a block decomposition of (a column permutation of) the $N\times N$ identity matrix, with $\mathcal{U}_i \in \mathbb{R}^{N\times N_i}$ and $\sum_{i=1}^n N_i = N$. Any $x\in\mathbb{R}^N$ can then be represented as $x=\sum_{i=1}^n \mathcal{U}_i x^{(i)}$, where $x^{(i)}\in\mathbb{R}^{N_i}$, and we will write $x=(x^{(1)};\dots;x^{(n)})$. Let $\|\cdot\|_{(i)}, \|\cdot\|_{(i)}^*$ be a pair of conjugate Euclidean norms in $\mathbb{R}^{N_i}$.

Smoothness of f means that the gradient of $t\mapsto f(x+\mathcal{U}_i t)$ is Lipschitz at $t=0$, uniformly in x for all i, with constants $L_i>0$:

$$\|\mathcal{U}_i^T[f'(x+\mathcal{U}_i t)-f'(x)]\|_{(i)}^* \le L_i\|t\|_{(i)},\ x\in\mathbb{R}^N,\ t\in\mathbb{R}^{N_i}. \quad (2)$$

Block separability of Ψ means that $\Psi(x)=\sum_{i=1}^n \Psi_i(x^{(i)})$.

B. Examples of Ψ

- *Unconstrained smooth minimization:* $\Psi(x)\equiv 0$. Iteration complexity analysis in this case was done in [2]. Our results (not in this abstract) are slightly better and analysis much simpler.
- *Block-constrained smooth minimization:* $\Psi_i(x)\equiv$ indicator function of some convex set in $\mathbb{R}^{N_i}$.
- *ℓ_1-regularized minimization:* $\Psi(x)\equiv\lambda\|x\|_1$. In machine learning, this helps to prevent model over-fitting [1] and in compressed sensing this is used to recover sparse signals [3].

II. The Algorithm and its Iteration Complexity

Let us define a norm on $\mathbb{R}^N$ by $\|x\|_L=(\sum_{i=1}^n L_i\|x^{(i)}\|_{(i)}^2)^{\frac{1}{2}}$.

Theorem 1. *Choose $x_0\in\mathbb{R}^N$ and $\epsilon>0$ such that $\mu\equiv\epsilon/\|x^*-x_0\|_L^2\le 2$. Further, pick $\rho\in(0,1)$ and let*

$$k\ge \tfrac{4n\|x^*-x_0\|_L^2}{\epsilon}\log\left(\tfrac{2(F(x_0)-F^*)}{\rho\epsilon}\right).$$

If x_k is the random vector generated by Algorithm 1 when applied to the objective function $F_\mu(x)=F(x)+\frac{\mu}{2}\|x-x_0\|_L^2$, then $\mathbf{Prob}(F(x_k)-F^*\le\epsilon)\ge 1-\rho$.

Algorithm 1 Uniform Coordinate Descent for Composite Functions

for $k=0,1,2,\dots$ **iterate**
  Choose $i_k=i\in\{1,2,\dots,n\}$ with probability $\frac{1}{n}$
  $T^{(i)}=\arg\min_{t\in\mathbb{R}^{N_i}}\langle\nabla f(x_k),\mathcal{U}_i t\rangle+\dfrac{L_i}{2}\|t\|_{(i)}^2+\Psi(x_k+\mathcal{U}_i t)$
  $x_{k+1}=x_k+\mathcal{U}_i T^{(i)}$

III. Numerical results

We will apply Algorithm 1 to random instance of (1) with

$$f(x)=\tfrac{1}{2}\|Ax-b\|_2^2,\qquad \Psi(x)=\|x\|_1, \qquad (3)$$

where $b\in\mathbb{R}^m$, $A\in\mathbb{R}^{m\times n}$, $N=n$.

In the first table below we present duration time (in seconds) of n iterations of Algorithm 1 applied to problem (1), (3) with a sparse solution x^* and random sparse matrix A. By $\|\cdot\|_0$ we denote number of nonzero elements.

$\|x^*\|_0$	$\|A\|_0=10^8$		$\|A\|_0=10^9$	
	$10^7\times10^6$	$10^8\times10^7$	$10^7\times10^6$	$10^8\times10^7$
16×10^2	5.89	11.04	46.28	70.48
16×10^3	5.83	11.59	46.07	59.03
16×10^4	4.28	8.64	46.93	77.44

Let us remark that $n=10^7$ iterations in case when $m=10^8$ and A has a billion nonzeros are executed in about 1 minute. In order to get a solution with accuracy $\epsilon=10^{-5}$, one needs approximately $40\times n$ iterations. In the next table we illustrate, on a random problem with $m=10^7, n=10^6$, $\|A\|_0=10^8$ and $\|x^*\|_0=16\times10^2$, the typical behavior of the method in reducing the gap $F(x_k)-F^*$.

k/n	$F(x_k)-F^*$	$\|x_k\|_0$	time [sec.]
0.0010	$<10^{16}$	857	0.01
15.2320	$<10^{10}$	997944	65.19
20.6150	$<10^{8}$	978761	88.25
25.9120	$<10^{6}$	763314	110.94
30.6620	$<10^{4}$	57991	131.25
35.0520	$<10^{2}$	2538	150.02
38.2650	$<10^{0}$	1633	163.75
40.9880	$<10^{-1}$	1604	175.38
42.7140	$<10^{-4}$	1600	182.77
44.8600	$<10^{-6}$	1600	191.94

References

[1] K.-W. Chang, C.-J. Hsieh, and C.-J. Lin. Coordinate descent method for large-scale l2-loss linear support vector machines. *Journal of Machine Learning Research*, 9:1369–1398, 2008.

[2] Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. CORE Discussion Paper 2010/2.

[3] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo. Sparse reconstruction by separable approximation. *Trans. Sig. Proc.*, 57:2479–2493, July 2009.