CARPATHIAN J. MATH. 35 (2019), No. 3, 371 - 378
Online version at http://carpathian.ubm.ro
Print Edition: ISSN 1584 - 2851 Online Edition: ISSN 1843 - 4401

Dedicated to Prof. Qamrul Hasan Ansari on the occasion of his 60^{th} anniversary

Numerical experiments on stochastic block proximal-gradient type method for convex constrained optimization involving coordinatewise separable problems

PORNTIP PROMSINCHAI[1] and NARIN PETROT[1,2]

ABSTRACT. In this paper, we consider convex constrained optimization problems with composite objective functions over the set of a minimizer of another function. The main aim is to test numerically a new algorithm, namely a stochastic block coordinate proximal-gradient algorithm with penalization, by comparing both the number of iterations and CPU times between this introduced algorithm and the other well-known types of block coordinate descent algorithm for finding solutions of the randomly generated optimization problems with a ℓ_1-regularization term.

1. INTRODUCTION

Consider the following constrained optimization problem:

$$\begin{aligned} &\text{minimize} \quad F(x) := f(x) + h(x) \\ &\text{subject to} \quad x \in \arg\min g(x), \end{aligned} \tag{1.1}$$

where $f, g : \mathbb{R}^n \to \mathbb{R}$ are convex smooth functions and $h : \mathbb{R}^n \to \mathbb{R}$ is convex lower semi-continuous(possibly non-smooth) coordinatewise separable function. That is, $h(x) = \sum_{i=1}^n h_i(x_i)$, where $h_i : \mathbb{R} \to \mathbb{R}$, for all $i = 1, ..., n$.

The problem (1.1) is called the constrained composite convex optimization problems. A concrete example of the coordinatewise separable function is the ℓ_1-regularization:

$$h(x) = \mu\|x\|_1 \tag{1.2}$$

where $\mu > 0$. It is well-known that the ℓ_1-regularization is a coordinatewise separable function that has attracted a lot of attention in the area of signal processing and data mining, see [9] and references therein.

Moreover, the constrained composite convex optimization problems also arise in many contemporary statistical and signal processing applications including compressive sensing, signal denoising, image processing, support vector machine, traffic equilibrium, and network flow problem, see [7], [9], [10], [23].

The problem (1.1) was initially studied by Attouch and Czarnecki [3] which represented the starting point of a series of articles approaching the minimization of a smooth or nonsmooth objective function subject to the set of minimizers of another function.

Received: 08.06.2019. In revised form: 24.08.2019. Accepted: 01.09.2019
2010 *Mathematics Subject Classification.* 49M27, 90C25, 90C52, 65K05.
Key words and phrases. *constrained optimization problem, composite objective function, block coordinate descent, proximal-gradient algorithm .*
Corresponding author: Narin Petrot; narinp@nu.ac.th

The development of the numerical algorithms for finding the solution of the problem (1.1) motivated many researchers to propose several variant types of problems (1.1) involving hierarchical minimization problems, see [1], [2], [4], [16]. In particular, Attouch et al [4] considered a generalized form of (1.1) and proposed the forward-backward schemes with a penalty for constrained variational inequalities. We notice that, in particular, the proximal-gradient method with a penalization term which is based on full-gradient information has been introduced in [4].

Recently, there has been much interest in the design of algorithms suitable for solving optimization problems concerning the structure of big data. One of the most successful classes of algorithms for solving the big data optimization problem is the coordinate descent methods. These methods were among the first optimization methods studied in literature [8]. The idea of the coordinate descent methods is based on the strategy of updating a single coordinate or a single block coordinate of the vector of variables at each iteration. This often drastically reduces memory requirements as well as the arithmetic complexity of a single iteration, making the methods easily implementable and scalable.

The main differences between all variants of coordinate descent methods are the criterion of choosing the coordinate over minimize the objective function at each iteration. The classical criteria are the cyclic coordinate descent method which significantly differs by the number of computations required to choose the appropriate index. The cyclic coordinate descent method was proposed by Bertsekas in [8] and has been applied to solve a composite convex optimization problem with the separable function, see [6], [22]. After that, based on the Gauss-Southwell choice rule, Tseng [24] extended a cyclic block coordinate gradient descent method to solve the composite optimization problem of type (1.1) with linear constraint. Another interesting approach is based on random coordinate descent which randomly chooses one block of the coordinate for updating according to a prescribed probability distribution at each iteration. In particular, Nesterov [15] presented the random coordinate descent methods for smooth convex optimization problem and these methods have been extended to the case of a composite optimization problem with a nonsmooth separable component in [20]. After that, Necoara[13] proposed the random coordinate descent algorithms which randomly choose two block coordinate for minimizing multi-agent convex optimization problems with linearly coupled constraints over networks. In 2014, Necoara et al. [14] extended the random two blocks coordinate descent method for solving a composite optimization problem (1.1) with linearly coupled constraints. Recently, Combettes and Pesquet[11] presented the stochastic block coordinate forward-backward splitting method for solving the monotone inclusion problem. We notice that the stochastic block coordinate forward-backward splitting method can be viewed as a stochastic block coordinate proximal-gradient method for solving unconstrained composite convex optimization problems.

Motivated by the above literature, in this paper, we will introduce a stochastic block coordinate proximal-gradient method with penalization terms (Sto-PGP) for the purpose of solving the problem (1.1). The numerical experiments will be considered for comparing the performance of Sto-PGP algorithm with the other well-known types of block coordinate descent algorithm on the randomly generated optimization problems with a ℓ_1-regularization term, which was proposed in [14].

2. Stochastic block coordinate proximal-gradient type mothods

We begin this section by reminding the notations which will be used in this paper.

Let $x := (x_1, ..., x_n) \in \mathbb{R}^n$ be decomposed into N non-overlapping blocks of variables $\tilde{x}_1, ..., \tilde{x}_N$ with $\tilde{x}_{(i)} \in \mathbb{R}^{n_i}$ such that $\sum_{i=1}^{N} n_i = n$.

For each $i \in \{1, 2, ..., N\}$, we denote by U_i for the blocks of identity matrix, that is

$$I_n := [U_1 \cdots U_N],$$

where $U_i \in \mathbb{R}^{n \times n_i}$, for $i = 1, ..., N$.

Note that for a vector $x \in \mathbb{R}^n$, we have $\tilde{x}_{(i)} = U_i^\top x$ for $i \in \{1, ..., N\}$, where $\top$ is denoted for the transpose of the considered matrix. We also denote by $\nabla_i f(x)$ for the i-th block in the gradient of the function f at x, that is $\nabla_i f(x) := U_i^\top \nabla f(x)$.

Under the above setting, the problem (1.1) can be extended to the following form of the block structure:

$$\begin{aligned} &\text{minimize} \quad F(x) := f(x) + \sum_{i=1}^{N} h_i(\tilde{x}_{(i)}) \\ &\text{subject to} \quad x \in \arg\min g(x), \end{aligned} \tag{2.3}$$

where $x := (\tilde{x}_1, ..., \tilde{x}_N)$ is a block-wise partition such that

$$\sum_{i=1}^{N} n_i = n,$$

and

$$h_i : \mathbb{R}^{n_i} \to \mathbb{R}$$

are all convex lower semi-continuous(possibly non-smooth) functions. A special case the problem (2.3) was considered by Tseng [24] and Necoara et al. [14] with the linear constraint $a^\top x = b$, where $a, x \in \mathbb{R}^n$ and $b \in \mathbb{R}$.

We end this section by recalling the definition of proximal operator. For a function $h : \mathbb{R}^n \to \mathbb{R}$, the proximal operator of h induced by a parameter λ is defined by

$$\text{prox}_{\lambda h}(z) := \arg\min_{x \in \mathbb{R}^n} \left\{ \lambda h(x) + \frac{1}{2}\|x - z\|_2^2 \right\}.$$

For more information on the proximal operator, one may consult the book by Beck [5].

Now, we introduce the stochastic block coordinate proximal-gradient method with penalization terms(Sto-PGP).

Algorithm: Stochastic block coordinate proximal-gradient type method (Sto-PGP)

Let $(\lambda_k)_{k\in\mathbb{N}}$ and $(\beta_k)_{k\in\mathbb{N}}$ be positive sequences of real numbers. Let $(\varepsilon_k)_{k\in\mathbb{N}}$ be identically distribution $\mathcal{D}$-valued random variable, where $\mathcal{D} = \{0, 1\}^N \backslash \{\mathbf{0}\}^N$. Let $x_0 \in \mathbb{R}^n$.

for $k = 0, 1, ...$

for $i = 1, ..., N$

$$x_{i,k+1} = x_{i,k} + \varepsilon_{i,k}\left(\text{prox}_{\lambda_k h_i}\left[x_{i,k} - \lambda_k \nabla_i f(x_k) - \lambda_k \beta_k \nabla_i g(x_k)\right] - x_{i,k}\right). \tag{2.4}$$

Remark 2.1. (1) It is worth to point out that the Sto-PGP is calculated only the block coordinate with $\varepsilon_i = 1$ for updating at each iteration, according to the distribution of the random variable ε_i at each iteration.

(2) If $\varepsilon_j = 1$ for only $j \in \{1, ..., N\}$ with a cyclic choice rule for updating the algorithm at each iteration, the Sto-PGP is the cyclic block coordinate proximal-gradient method with penalization term(Cyclic-PGP):
for $k = 0, 1, ...$
1: Choose coordinate $j \in \{1, ..., N\}$ by cyclic search.

2: Update x_{k+1} by
for $i = 1, ..., N$
set $x_{i,k+1} = x_{i,k}, \forall i \neq j$

$$x_{i,k+1} = \text{prox}_{\lambda_k h_j}\bigg[x_k - \lambda_k \nabla_j f(x_k) - \lambda_k \beta_k \nabla_j g(x_k)\bigg]. \tag{2.5}$$

(3) If there is only one block coordinate $j \in \{1, ..., N\}$ such that $\varepsilon_j = 1$ for updating the algorithm at each iteration, the Sto-PGP is the random single block coordinate proximal-gradient method with penalization term(Single-PGP):
for $k = 0, 1, ...$
1: Choose randomly only one coordinates $j \in \{1, ..., N\}$.
2: Update x_{k+1} by
for $i = 1, ..., N$
set $x_{i,k+1} = x_{i,k}, \forall i \neq j$

$$x_{j,k+1} = \text{prox}_{\lambda_k h_j}\bigg[x_k - \lambda_k \nabla_j f(x_k) - \lambda_k \beta_k \nabla_j g(x_k)\bigg]. \tag{2.6}$$

(4) In the case that there are only two block coordinate $j, l \in \{1, ..., N\}$ such that $\varepsilon_j = 1 = \varepsilon_l$ for updating the algorithm at each iteration, the Sto-PGP is the random two-block coordinate proximal-gradient method with penalization term(Two-PGP):
for $k = 0, 1, ...$
1: Choose randomly two coordinates $(j, l) \in \{1, ..., N\}$ with $j \neq l$.
2: Update x_{k+1} by
for $i = 1, ..., N$
set $x_{i,k+1} = x_{i,l}, \forall i \notin (j, l)$

$$x_{j,k+1} = \text{prox}_{\lambda_k h_j}\bigg[x_k - \lambda_k \nabla_j f(x_k) - \lambda_k \beta_k \nabla_j g(x_k)\bigg].$$

$$x_{l,k+1} = \text{prox}_{\lambda_k h_l}\bigg[x_k - \lambda_k \nabla_l f(x_k) - \lambda_k \beta_k \nabla_l g(x_k)\bigg]. \tag{2.7}$$

(5) If $\varepsilon_i = 1$, for all $i \in \{1, ..., N\}$, for updating the algorithm at each iteration, the Sto-PGP is the proximal-gradient method with penalization term(Full-PGP) in [4]:
for $k = 0, 1, ...$

$$x_{k+1} = \text{prox}_{\lambda_k h}\bigg[x_k - \lambda_k \nabla f(x_k) - \lambda_k \beta_k \nabla g(x_k)\bigg]. \tag{2.8}$$

3. Numerical experiment

In this section, we present the experiments for testing the performance of the Sto-PGP algorithm to the optimization problem (1.1). We consider the Sto-PGP algorithm for randomly generated problems with a ℓ_1-regularization term, which was considered in [14].

In [14], the authors presented the random two-coordinate descent method on the randomly generated problems with a ℓ_1-regularization term and compared the experiments to the full-gradient information algorithm and the cyclic coordinate descent methods. They gave the conclusion that the random two-coordinate descent method with block size 1 performs up to 100 CPU times better than the full-gradient information algorithm and the cyclic coordinate descent method.

Here, the major experiments are the comparison of the iteration numbers and CPU times of the Sto-PGP with some variant numbers of block sizes, Two-PGP with the block size 1, and Single-PGP with the block size 2. The experiment is performed under MATLAB 9.2 (R2017a) running on a laptop with 2.59 GHz Intel Core i7 and 4 GB RAM.

We consider the randomly generated problems with ℓ_1-regularization term in the following form:

$$\text{(3.9)} \qquad \begin{aligned} &\text{minimize} \quad \frac{1}{2}x^\top Z^\top Zx + q^\top x + \left(\mu\sum_{i=1}^{n}|x_i| + 1_{[l,u]}(x)\right) \\ &\text{subject to} \quad a^\top x = b, \end{aligned}$$

where $x, q \in \mathbb{R}^n$, $Z \in \mathbb{R}^{m\times n}$, $1_{[l,u]}$ is the indicator function for the box constranit set $[l,u]^n$ and the parameter $\mu > 0$. Notice that many applications from signal processing and data mining can be formulated into the optimization problem (3.9), see [9], [19].

The problem (3.9) can be written in the form of the problem (1.1) as

$$\text{(3.10)} \qquad \begin{aligned} &\text{minimize} \quad \frac{1}{2}x^\top Z^\top Zx + q^\top x + \left(\mu\sum_{i=1}^{n}|x_i| + 1_{[l,u]}(x)\right) \\ &\text{subject to} \quad x \in \arg\min g(\cdot) := \frac{1}{2}\|a^\top x - b\|_2^2. \end{aligned}$$

We observe that the problem (3.10) can be presented as the block structure problem (2.3) by the following setting:

$$f(x) = \frac{1}{2}x^\top Z^\top Z + q^\top x,$$

$$g(x) = \frac{1}{2}\|a^\top x - b\|_2^2,$$

for all $x \in \mathbb{R}^n$, and

$$h_i(x_{(i)}) = \mu|x_{(i)}| + 1_{[l_{(i)},u_{(i)}]}(x_{(i)}),$$

for $i = 1, ..., N$, where $x_{(i)} \in \mathbb{R}^{n_i}$ is the N non-overlapping blocks of $x \in \mathbb{R}^n$.

Moreover, under these setting we can check that $f, g : \mathbb{R}^n \to \mathbb{R}$ are convex smooth functions and $h_i : \mathbb{R}^{n_i} \to \mathbb{R}$ are convex lower semi-continuous functions, for all $i = 1, ..., N$.

We generate the problems under the setting dimensions of $n = 5000, 8000, 10000$ and $m = 10$, and generated randomly the matrix $Z \in \mathbb{R}^{m\times n}$ and $q \in \mathbb{R}^n$, by using uniform distribution. Further, the parameters are chosen as follows: $a = e, b = 1$, with $e = [1...1]^\top \in \mathbb{R}^n$ and $-l = u = 1$. We test the Sto-PGP algorithm with $\mu = 10$ and using the parameter $\lambda_k = 1/(L_h + k)$ and the penalization parameter $\beta_k = (L_h + k)/2L_g$, where $L_h = \gamma_{\max}(\sum_{i=1}^{m} Z_i^\top Z_i)$ and $L_g = \gamma_{\max}(a^\top a)$, which $\gamma_{\max}(\cdot)$ denotes the largest eigenvalue of a matrix. We perform the algorithms with the starting point $x_0 = e$.

We proceed the experiment as follows: firstly, we computed F^* with the algorithm Two-PGP by using relative stopping criteria

$$\text{(3.11)} \qquad \max\left\{\frac{|F(x_k) - F(x_{k+1})|}{|F(x_k)|}, \frac{|g(x_k) - g(x_{k+1})|}{|g(x_k)|}\right\} \le \epsilon,$$

where $\epsilon = 10^{-5}$. Secondly, we used the pre-computed optimal value of F^* to test the other algorithms with a termination criterion

$$\text{(3.12)} \qquad \max\left\{\frac{|F(x_k) - F^*|}{|F(x_k)|}, \frac{|g(x_k) - g(x_{k+1})|}{|g(x_k)|}\right\} \le \epsilon,$$

where $\epsilon = 10^{-5}$ to obtain the number of iterations and CPU times, where the CPU times are considered in the second unit.

The results are presented in Table 1. We see that when the block size $n_i = 1$, the Two-PGP shows better performance than the Sto-PGP. However, when we consider the block

TABLE 1. The comparison of the number of iterations and CPU time for the Sto-PGP, Two-PGP, and Single-PGP on randomly generated problems with ℓ_1-regularization term.

Algorithms	n_i	n	Number of iterations	Terminated value	CPU time(s)
Two-PGP	1	5000	302	8.8694	70.71
		8000	476	8.8836	286.39
		10000	364	9.3033	357.03
Single-PGP	2	5000	280	8.9582	65.74
		8000	426	8.9733	253.89
		10000	316	9.3987	305.17
Sto-PGP	1	5000	549	8.9577	129.46
		8000	864	8.9733	518.65
		10000	630	9.3974	614.62
	2	5000	278	8.9556	64.86
		8000	436	8.9712	260.43
		10000	315	9.3991	308.23
	10	5000	58	8.8838	13.06
		8000	89	8.9678	52.32
		10000	75	9.2689	72.97

size n_i by 2 and 10, the Sto-PGP provides better performance both CPU times and the number of iterations than those of the Two-PGP. Furthermore, we observe that the bigger block size of Sto-PGP provides better performance than the smaller block size both the CPU times and the number of iterations for all dimensional cases.

Moreover, one can see that the Single-PGP with the block size $n_i = 2$ provides the better performance both CPU times and the number of iterations than those of Two-PGP with the block size $n_i = 1$ for all dimensional cases. This may suggest that updating with a block method, by packing and fix each two coordinates into one block, should provide better performance than randomly the independent two coordinates method. In fact, we notice that in [14] the authors considered the Two-PGP in the sense of coordinate descent algorithm, while in this work we consider the Sto-PGP in the sense of block descent algorithm. These observations will be the topic for further researches both to consider a suitable partition of block structure for improving the performance of the algorithms and proving its convergence theorems.

4. CONCLUSIONS

The main purpose of this paper is to discuss the numerical experiments of the algorithms for finding a solution of convex constrained optimization problems when the objective function is the sum of two functions over the set of a minimizer of another function. We propose a stochastic block coordinate proximal-gradient algorithm with penalization (Sto-PGP) and present the preliminary experiments to evaluate the performance of the Sto-PGP algorithm and compare its performance with the other well-known types of block coordinate descent algorithm. The presented Sto-PGP algorithm and its numerical experiments in this paper can be viewed as the guiding concept of the theoretical convergence theorems in the future researches.

Acknowledgements. P. Promsinchai is supported by the Thailand Research Fund through the Royal Golden Jubilee Ph.D. Program (Grant No. PHD/0023/2555) and Naresuan University.

REFERENCES

[1] Ansari, Q. H., Ceng, L.-C. and Gupta, H. *Triple heirarchical variational inequalities, in Nonlinear Analysis: Approximation Theory, Optimization and Applications* , Edited by Q. H. Ansari, Springer, 2014

[2] Ansari, Q. H. and Rehan, A., *An iterative method for split hierarchical monotone variational inclusions*, Fixed Point Theory Appl., **2015**, 2015:121, 10 pp.

[3] Attouch, H. and Czarnecki, M.-O., *Asymptotic behavior of coupled dynamical systems with multiscale aspects*, J. Differ. Equ., **248** (2010), 1315–1344

[4] Attouch, H., Czarnecki, M.-O. and Peypouquet, J., *Coupling forward-backward with penalty schemes and parallel splitting for constrained variational inequalities*, SIAM J. Optim., **21** (2011), 1251–1274

[5] Beck, A., *First-Order Methods in Optimization*, Society for Industrial and Applied Mathematics; Philadelphia: Mathematical Optimization Society, 2017

[6] Beck, A. and Tetruashvili, L., *On the convergence of block coordinate descent type methods*, SIAM J. Optim., **23** (2013), No. 4, 2037–2060

[7] Bertsekas, D. P., *Parallel and Distributed Computation: Numerical Methods. Athena Scientific*, Nashua, 2003

[8] Bertsekas, D. P., *Nonlinear Programming*, Athena Scientific, Nashua, 1999

[9] Candes, E., Romberg, J. and Tao, T., *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inf. Theory, **52**, (2006), 489–509

[10] Chen, S., Donoho, D. and Saunders, M., *Atomic decomposition by basis pursuit*, SIAM Rev., **43** (2001), 129–159

[11] Combettes, P. L. and Pesquet J.-C., *Stochastic quasi-Fejér block-coordinate fixed point iterations with random sweeping*, SIAM J. Optimiz., **25** (2015), No. 2, 1221–1248

[12] Elad, M. and Aharon, M., *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Trans. Image Process., **15** (2006), No. 12, 3736–3745

[13] Necoara, I., *Random coordinate descent algorithms for multi-agent convex optimization over networks*, IEEE Trans. Autom. Control, **58** (2013), No. 7, 1–12

[14] Necoara, I. and Patrascu, A., *A random coordinate descent algorithm for optimization problems with composite objective function and linear coupled constraints*, Comp. Optimiz. Applicat., **57** (2014), No. 2, 307–337

[15] Nesterov, Y., *Efficiency of coordinate descent methods on huge-scale optimization problems*, SIAM J. Optim., **22**(2012), No. 2, 341–362

[16] Nimana, N. and Petrot, N., *Generalized forward-backward splitting with penalization for monotone inclusion problems*, J. Global Optim., **73** (2019), No. 4, 825–847

[17] Peypouquet, J., *Coupling the gradient method with a general exterior penalization scheme for convex minimization*, J. Optim. Theory Appl., **153** (2012), No. 1, 123–138

[18] Pique-Regi, R., Monso-Varona, J., Ortega, A., Seeger, R. C., Triche, T. J. and Asgharzadeh, S., *Sparse representation and bayesian detection of genome copy number alterations from microarray data*, Bioinformatics, **24** (2008), No. 3, 309–318

[19] Qin, Z., Scheinberg, K. and Goldfarb, D., *Efficient block-coordinate descent algorithms for the group Lasso*, Math. Prog. Comp., **5** (2013), No. 2, 143–169

[20] Richtárik, P. and Takáč, M., *Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function*, Math. Program., Ser. A, **144** (2014), No. 1-2, 1–38

[21] Richtárik, P. and Takáč, M., *Parallel coordinate descent methods for big data optimization*, Technical report (2012), arXiv:1212.0873

[22] Saha, A. and Tewari, A., *On the finite time convergence of cyclic coordinate descent methods*, SIAM J. Optim., **23** (2013), No. 1, 576–601

[23] Tseng, P. and Yun, S., *A coordinate gradient descent method for linearly constrained smooth optimization and support vector machines training*, Comput. Optim. Appl., **47** (2010), 179–206

[24] Tseng, P. and Yun, S., *A block-coordinate gradient descent method for linearly constrained nonsmooth separable optimization*, J. Optim. Theory Appl., **140** (2009), 513–535

[25] Wright, J., Ma, Y., Mairal, J., Sapiro, G., Huang, T. S., and Yan, S., *Sparse representation for computer vision and pattern recognition*, Proceedings of the IEEE, **98** (2010), No. 6, 1031–1044

[26] Yang, M. and Zhang, L., *Gabor feature based sparse representation for face recognition with gabor occlusion dictionary*, In European conference on computer vision, pages 448–461, Springer, 2010

[1]Department of Mathematics
Naresuan University
Faculty of Science
Phitsanulok, 65000, Thailand
E-mail address: petoypsc@gmail.com
E-mail address: narinp@nu.ac.th

[2]Center of Excellence in Nonlinear Analysis and Optimizations
Faculty of Science
Phitsanulok, 65000, Thailand
E-mail address: narinp@nu.ac.th