52nd IEEE Conference on Decision and Control
December 10-13, 2013. Florence, Italy

Extension of a saddle point mirror descent algorithm with application to robust PageRank*

Andrey Tremba and Alexander Nazin

***Abstract*— The paper is devoted to designing an efficient recursive algorithm for solving the robust PageRank problem recently proposed by Juditsky and Polyak (2012) [4]. To this end, we reformulate the problem to a specific convex-concave saddle point problem $\min_{x\in X}\max_{y\in Y} q(x,y)$ with simple convex sets $X \in \mathbb{R}^N$ and $Y \in \mathbb{R}^N$, i.e., standard simplex and Euclidean unit ball, respectively. Aiming this goal we develop an extension of saddle point mirror descent algorithm where additional parameter sequence is introduced, thus providing more degree of freedom and the refined error bounds. Detailed complexity results of this method applied to the robust PageRank problem are given and discussed. Numerical example illustrates the theoretical results proved.**

I. INTRODUCTION

The ultimate goal is to solve robust page ranking problem on unit simplex $S_N = \{x : \sum_{i=1}^{N} x_i = 1,\ \forall i,\ x_i \geq 0\}$ that is

$$x^* = \operatorname*{argmin}_{x\in S_N}(\|Ax - x\|_2 + \varepsilon\|x\|_2), \tag{1}$$

with a column-stochastic matrix A and a relevant parameter $\varepsilon > 0$. This problem was introduced in [4], motivated that its solution is less-sensitive to disturbances than the original PageRank solution vector[1]. We use game approach proposed in [7] for the original PageRank problem. This approach is very suitable for robust PageRank problem as well due to its nonlinearity[2]. There are good traits of (1) that solution is unique, and it may be reformulated as a minimax (game) problem $\min_{x\in S_N}\max_{y\in B_N} q(x,y)$ with unit Euclidean ball $B_N = \{y \in \mathbb{R}^N : \|y\|_2 \leq 1\}$ and game function

$$q(x,y) = y^T(Ax - x) + \varepsilon\|x\|_2\,. \tag{2}$$

Main issue of solving (1) is that cost of calculation of a (sub)gradient of objective function for huge-dimensional vector x is high, and normally involve laborious matrix-vector multiplications. We exploit and improve mirror descent algorithms, both deterministic and stochastic ones, which only require subgradients (or stochastic subgradients). In the latter method the matrix-vector multiplications are avoided at all.

Notice, that $\varepsilon = 0$ reduces the function in (2) to that of [7], and the algorithms therein may be treated as the consequences of those presented here in the paper. In other words, we extend the approach of [7] to the robust PageRank problem (1). Let's remind that including parameter $\varepsilon > 0$ in (1) and, therefore, (2), is not simply treated as regularization, or penalty, or whatever. In fact, it is a novel approach to solving ranking problem, which has been called robust PageRank [4].

The contribution is two-fold: first, we derive an extension of the stochastic mirror descent algorithm for saddle point problem (as well as deterministic mirror descent algorithm), [9], with extra parameters, and show its non-asymptotic convergence rate. Second, for the specific robust PageRank problem we compare these algorithms, with separate complexity in number of steps and matrix-vector multiplication complexity, thus help choosing appropriate algorithm with given problem size. The results numerically tested on robust PageRank calculation.

The paper organized as follows: in next section preliminary notions and problem formulation are given, then goes a section with deterministic and stochastic (randomized) algorithms themselves. In IV-th section these algorithms are applied to the stated robust PageRank problem, followed by complexity analysis and example. Sections with proofs and future works suggestions conclude the paper.

II. PRELIMINARIES

Let us introduce convex compacts $X \subset \mathbb{R}^N$, $Y \subset \mathbb{R}^M$ equipped with norms $\|\cdot\|_x$, $\|\cdot\|_y$, $x^{(i)}$ denotes i-th coordinate of vector x. Lower bracketed index n means total number of steps, k, t – intermediate step indices, R_n, P_n, Q_n are some shortcuts for complex expressions, hat symbol $\hat{}$ stands for some distinguished point or function, symbol $\asymp$ stands for $\Omega(\cdot)$ "same in order" notation, $\doteq$ means definition of variables/functions.

A. Generic convex-concave saddle point problem

Let function $q : X \times Y \to \mathbb{R}$ be continuous, defined on convex compact sets X, Y, and let it be convex in $x \in X$ and concave in $y \in Y$. Then it has a saddle point $(x^*, y^*) \in X \times Y$, i.e., for all $x \in X$ and $y \in Y$, the following inequalities hold:

$$q(x^*, y) \leq q(x^*, y^*) \leq q(x, y^*). \tag{3}$$

In other words, the saddle point is a solution to minimax problem $q^* \doteq q(x^*, y^*) = \min_{x\in X}\max_{y\in Y} q(x,y) =$

* This work was supported by the Russian Foundation for Basic Research under grant 12-08-01245.

The authors are with the Laboratory for Adaptive and Robust Control Systems, the Institute of Control Sciences RAS, Moscow, Russia, atremba@ipu.ru, nazine@ipu.ru

[1] An eigenvector corresponding to the unit eigenvalue of a column-stochastic matrix $A \in \mathbb{R}^{N\times N}$, i.e. a solution to the equation $Ax = x$ under constraint $x \in S_N$, see [2], [5].

[2] Also for original PageRank problem there exists very simple scheme based on non-expansion contraction of power method iterations $x_{t+1} = Ax_t$ [12]. It has explicit convergence error rate $\varepsilon_t = 2/t$, but unfortunately it is not directly applicable to the robust case.

978-1-4673-5716-6/13/$31.00 ©2013 IEEE

$\max_{y\in Y}\min_{x\in X} q(x,y)$. Quality of a candidate point $(\widehat{x},\widehat{y}) \in X\times Y$ can be naturally expressed in terms of the error (or gap):

$$\begin{aligned}\Delta(\widehat{x},\widehat{y}) &\doteq (\max_{y\in Y} q(\widehat{x},y) - q^*)\\ &\quad + (q^* - \min_{x\in X} q(x,\widehat{y}))\\ &= \max_{y\in Y} q(\widehat{x},y) - \min_{x\in X} q(x,\widehat{y}).\end{aligned} \tag{4}$$

Note that inequality $q(\widehat{x},y)-q(x,\widehat{y}) \le c$, $\forall\,(x,y)\in X\times Y$, implies the bound $\Delta(\widehat{x},\widehat{y}) \le c$, which leads to the inequality $|q(\widehat{x},\widehat{y}) - q^*| \le c$.

B. Proxy functions

Here we exploit the properties of convex compacts X and Y. Let's consider strongly convex modulus[3] α_x and α_y proxy functions $V : X \to \mathbb{R}_+$ and $T : Y \to \mathbb{R}_+$ (with respect to the norms $\|\cdot\|_x$ and $\|\cdot\|_y$). For example, let X be a unit simplex S_N, and Y be a unit ball B_M, equipped with ℓ_1-norm and Euclidean one, respectively. Then proxy functions may be chosen as[4]

$$\begin{aligned}&\widehat{V}(x) = \ln N + \textstyle\sum_{i=1}^N x^{(i)} \ln x^{(i)}, && x\in S_N,\\ &\widehat{T}(y) = \tfrac{1}{2}\|y\|_2^2, && y\in B_M,\end{aligned}$$

with the modulus' $\alpha_x = 1$ and $\alpha_y = 1$. We remind that the choice of these proxy-functions made in purpose to exploit simple (and explicit) calculation of Legendre-Fenchel transform according to X and Y (following functions W and U), and acquiring its maximizers:

$$\begin{aligned}&W_\beta(\zeta) \doteq \max_{x\in X}(-\zeta^T x - \beta V(x)), && \zeta\in\mathbb{R}^N,\\ &U_\delta(\eta) \doteq \max_{y\in Y}(-\eta^T y - \delta T(y)), && \eta\in\mathbb{R}^M,\end{aligned}$$

with parameters $\beta,\delta > 0$, and properties of their gradients, including Lipschitz ones [13]:

$$\begin{aligned}&-\nabla W_\beta(\zeta) = \operatorname{argmin}_{x\in X}(\zeta^T x + \beta V(x)) \in X,\\ &-\nabla U_\delta(\eta) = \operatorname{argmin}_{y\in Y}(\eta^T y + \delta T(y)) \in Y,\\ &\|\nabla W_\beta(\zeta_1) - \nabla W_\beta(\zeta_2)\|_x \le \tfrac{1}{\alpha_x\beta}\|\zeta_1-\zeta_2\|_{*,x},\\ &\|\nabla U_\delta(\eta_1) - \nabla U_\delta(\eta_2)\|_y \le \tfrac{1}{\alpha_y\delta}\|\eta_1-\eta_2\|_{*,y}.\end{aligned}$$

Used dual norms are defined as usual through the primal ones, i.e. $\|\zeta\|_{*,x} \doteq \max_{\|x\|_x\le 1} x^T\zeta$.

For $\widehat{V}$ on S_N and $\widehat{T}$ on B_M, we have componentwise

$$\begin{aligned}&\frac{\partial \widehat{W}_\beta(\zeta)}{\partial \zeta^{(i)}} = -e^{-\zeta^{(i)}/\beta}\left(\textstyle\sum_{k=1}^N e^{-\zeta^{(k)}/\beta}\right)^{-1},\ i=1,\dots,N,\\ &\frac{\partial \widehat{U}_\delta(\eta)}{\partial \eta^{(j)}} = \begin{cases}\frac{1}{\delta}\eta^{(j)} & \text{if } \|\eta\|_2\le\delta,\\ \eta^{(j)}/\|\eta\|_2 & \text{otherwise,}\end{cases}\quad j=1,\dots,M.\end{aligned}$$

It is a matter of choice to consider these gradient maps as non-linear projections onto S_N and B_N (X and Y in general case) [1]. Finally, let's define extremal values

$$\begin{aligned}&V_* \doteq \min_{x\in X} V(x), && V^* \doteq \max_{x\in X} V(x),\\ &T_* \doteq \min_{y\in Y} T(y), && T^* \doteq \max_{y\in Y} T(y).\end{aligned}$$

For the particular choice $\widehat{V},\widehat{T}$ for $X = S_N, Y = B_M$ we have $\widehat{V}_* = \widehat{T}_* = 0$, $\widehat{V}^* = \ln N$, $\widehat{T}^* = 1/2$.

C. Subgradients

Let function $q(x,y)$ be convex on $x\in X$ (for a given y). Its (partial) subdifferential at point (x_1,y) is a set of all vectors $q_x\in\mathbb{R}^N$ satisfying

$$q(x_2,y) \ge q(x_1,y) + (x_2-x_1)^T q_x,\ \forall x_2\in X, y\in Y.$$

This set is nonempty for convex functions and contain exactly one element coinciding with partial derivative for the smooth convex functions. Sometimes it is denoted as $\frac{\partial}{\partial x}q(x,y)$. We denote as partial subgradient $q_x(x,y)$ one somehow deterministically distinguished element of the subdifferential set at the point (x,y). The "partial" prefix thereafter is omitted for shortcut.

Similarly we denote subgradient $q_y\in\mathbb{R}^M$ at point (x,y_1) for concave function, e.g. for $q(x,y_1)$ being concave on y, the following property holds

$$q(x,y_2) \le q(x,y_1) + (y_2-y_1)^T q_y(x,y_1),\ x\in X,\ \forall y_2\in Y.$$

III. Mirror descent algorithm (MDA)

The mirror descent method goes back to [10]. There is a connection of MDA with nonlinear projected subgradient method [1]. Deterministic algorithm with averaging was presented in early 2000-th as primal-dual subgradient method [11], including variation for saddle point problems. There were two parameter sequences introduced. The advanced stochastic MDA versions were developed in [3]. For game problem the stochastic MDA is also known as saddle point mirror stochastic approximation algorithm [9].

In this section we enhance the saddle point MDA with one more parameter sequence δ_k and show that the algorithm's core is the same for both deterministic and stochastic algorithms producing similar bounds. General properties of the algorithms are followed by specific choice of parameters.

A. Deterministic MDA: Algorithm 1

Algorithm 1 (which is much similar to that of described in [7]) has two different parameter sequences (β_k and δ_k) on the second (projection) step instead of the same sequence. Including weighting parameter sequence γ_k there are as many as three control sequences in total.

1) Specify positive sequence $\gamma_k, k\ge 1$ and two non-decreasing positive sequences $\beta_k,\delta_k, k\ge 0$. Fix initial value as zero vector for joint variables $(\zeta_0,\eta_0) = 0 \in \mathbb{R}^{N+M}$ and calculate[5] initial values $(x_0 = -\nabla W_{\beta_0}(\zeta_0), y_0 = -\nabla U_{\delta_0}(\eta_0)) \in X\times Y$ for the primal vector variables.
2) For each $k=1,\dots,n$, given primal pair (x_{k-1},y_{k-1}) and dual one (ζ_{k-1},η_{k-1}), update variables

$$\begin{aligned}&\begin{bmatrix}\zeta_k\\ \eta_k\end{bmatrix} = \begin{bmatrix}\zeta_{k-1}\\ \eta_{k-1}\end{bmatrix} + \gamma_k\begin{bmatrix}q_x(x_{k-1},y_{k-1})\\ -q_y(x_{k-1},y_{k-1})\end{bmatrix},\\ &\begin{bmatrix}x_k\\ y_k\end{bmatrix} = \begin{bmatrix}-\nabla W_{\beta_k}(\zeta_k)\\ -\nabla U_{\delta_k}(\eta_k)\end{bmatrix},\end{aligned} \tag{5}$$

using subgradients $q_x(\cdot,\cdot)$, $q_y(\cdot,\cdot)$.

[3] E.g. for the norm $\|\cdot\|_x, \forall x_1,x_2\in X$, it holds $V(\tau x_1 + (1-\tau)x_2) \le \tau V(x_1) + (1-\tau)V(x_2) - \frac{\alpha_x}{2}\tau(1-\tau)\|x_1-x_2\|_x^2$, $\forall\, 0\le\tau\le 1$.

[4] For $\widehat{V}(x)$ (entropy function) we put $0\ln 0 \equiv 0$.

[5] These values should not be chosen arbitrary as supposed in [3].

3) Output the averaged vectors

$$\begin{array}{l} \overline{x}_n = \frac{1}{\sum_{k=1}^n \gamma_k} \sum_{k=1}^n \gamma_k x_{k-1}, \\ \overline{y}_n = \frac{1}{\sum_{k=1}^n \gamma_k} \sum_{k=1}^n \gamma_k y_{k-1}. \end{array} \tag{6}$$

Theorem 1 (deterministic MDA): Given averaged estimates (6) generated by Algorithm 1 at n-th step, the error is bounded as follows:

$$\Delta(\overline{x}_n, \overline{y}_n) \leq \frac{1}{\sum_{k=1}^n \gamma_k} \left(R_n + \sum_{k=1}^n \frac{\gamma_k^2}{2} \left(\frac{L_{*,x}^2}{\alpha_x \beta_{k-1}} + \frac{L_{*,y}^2}{\alpha_y \delta_{k-1}} \right) \right)$$

where $L_{*,x} \doteq \max_{(x,y)\in X\times Y} \|q_x(x,y)\|_{*,x}$,
$L_{*,y} \doteq \max_{(x,y)\in X\times Y} \|q_y(x,y)\|_{*,y}$,
$R_n \doteq \beta_n V^* - \beta_0 V_* + \delta_n T^* - \delta_0 T_*$. □

All proofs are given in section V.

B. Stochastic (or randomized) MDA: Algorithm 2

Algorithm 2 is formally the same as deterministic one, but on the second step in (5) instead of exact subgradients $q_x(x_k, y_k)$ and $q_y(x_k, y_k)$ the stochastic subgradients ϕ_k, ψ_k are used with the following properties:

$$\begin{array}{ll} \mathbb{E}\{\phi_k | (x_t, y_t)_{t<k}\} & = q_x(x_{k-1}, y_{k-1}), \\ \mathbb{E}\{\psi_k | (x_t, y_t)_{t<k}\} & = q_y(x_{k-1}, y_{k-1}), \end{array} \tag{7}$$

$$\begin{array}{l} \mathbb{E}(\|\phi_k\|_{*,x}^2) \leq \mathbb{L}_{*,x}^2, \\ \mathbb{E}(\|\psi_k\|_{*,y}^2) \leq \mathbb{L}_{*,y}^2. \end{array} \tag{8}$$

Theorem 2 (stochastic MDA): Under assumptions of Algorithm 2 and properties (7), (8), the bound holds

$$\mathbb{E}\Delta(\overline{x}_n, \overline{y}_n) \leq \frac{1}{\sum_{k=1}^n \gamma_k} \left(R_n + \sum_{k=1}^n \frac{\gamma_k^2}{2} \left(\frac{\mathbb{L}_{*,x}^2}{\alpha_x \beta_{k-1}} + \frac{\mathbb{L}_{*,y}^2}{\alpha_y \delta_{k-1}} \right) \right),$$

$R_n = \beta_n V^* - \beta_0 V_* + \delta_n T^* - \delta_0 T_*$. □

In application to PageRank (Section IV), we'll introduce auxiliary random variables and define stochastic gradients based on these variables. To express artificial nature of randomization such MDAs may be called as randomized MDAs.

C. Parameter's choice

As we see, three parameter sequences $\gamma_k, \beta_k, \delta_k$ are to be prefixed in both algorithms. While γ_k have sense of weighting factors for data in given points, and they actually cannot be chosen different in advance[6], it is being simply put as $\gamma_k = 1, k = 1, \ldots, n$, with recurrent calculation of $\overline{x}_n, \overline{y}_n$:

$$\begin{array}{l} \overline{x}_n = \left(1 - \frac{1}{n}\right) \overline{x}_{n-1} + \frac{1}{n} x_{n-1}, \ \ \overline{x}_0 \doteq 0, \\ \overline{y}_n = \left(1 - \frac{1}{n}\right) \overline{y}_{n-1} + \frac{1}{n} y_{n-1}, \ \ \overline{y}_0 \doteq 0. \end{array} \tag{9}$$

So we are left only with β_k, δ_k for minimization of the upper bounds. It may be shown (cf. [3]) that good choice is

$$\begin{array}{l} \beta_k = \beta_0 \sqrt{k+1}, \ \ \beta_0 = \frac{L_{*,x}}{\sqrt{\alpha_x V^*}}, \\ \delta_k = \delta_0 \sqrt{k+1}, \ \ \delta_0 = \frac{L_{*,y}}{\sqrt{\alpha_y T^*}}, \end{array}$$

which leads to the error bound as small as

$$\Delta(\overline{x}_n, \overline{y}_n) \leq \frac{\sqrt{n+1}}{n} \left(L_{*,x} \sqrt{\frac{V^*}{\alpha_x}} + L_{*,y} \sqrt{\frac{T^*}{\alpha_y}} \right). \tag{10}$$

This bound has the same order on number of iterations n but is better in constant than the bound for two-sequences case with $\delta_k = \beta_k$, cf. [7][7]. For stochastic MDA, from Theorem 2 the same bound with constants $\mathbb{L}_{*,x}$ and $\mathbb{L}_{*,y}$ follows instead of $L_{*,x}$ and $L_{*,y}$, respectively. Another advantage of having three control sequences is that the adaptive step-size may be chosen for each of β_k and δ_k separately, though it would need proof modification like in [8].

IV. Application to the Robust PageRank

In this section we apply the both algorithms for the specific problem (1), for minimizing function

$$f(x) = \|Ax - x\|_2 + \varepsilon \|x\|_2$$

on unit simplex S_N with column-stochastic[8] matrix A. The solution x^* may be thought as x-part of saddle point problem for function $q(x, y)$ (2) and sets $X = S_N, Y = B_N$.

In application of MDAs for this problem it is naturally to use proxy-functions $\widehat{V}, \widehat{T}$, ℓ_1-norm as $\|\cdot\|_x$ norm and Euclidean $\|\cdot\|_2$ norm as $\|\cdot\|_y$, with conjugate ℓ_∞-norm as $\|\cdot\|_{*,x}$ norm and Euclidean $\|\cdot\|_2$ norm as $\|\cdot\|_{*,y}$.

Note that since saddle point problem for robust PageRank came from minimization problem, it is possible to use error bound on $\Delta(\overline{x}_n, \overline{y}_n)$ for estimating function residual

$$f(\overline{x}_n) - f(x^*) \leq \Delta(\overline{x}_n, \overline{y}_n).$$

Partial subgradients for (2) are gradients

$$\begin{array}{l} q_x(x, y) = A^T y - y + \varepsilon \frac{x}{\|x\|_2}, \\ q_y(x, y) = Ax - x, \end{array} \tag{11}$$

and, on sets $X = S_N$ and $Y = B_N$, they lead to the evident uniform bounds on column-stochastic matrices A:

$$\begin{array}{l} \max_{x\in S_N, y\in B_N} \|A^T y - y + \varepsilon x / \|x\|_2\|_\infty \\ \leq \max_{y\in B_N} \|A^T y - y\|_\infty + \varepsilon \max_{x\in S_N} \|x\|_\infty / \|x\|_2 \\ \leq 2 + \varepsilon = L_{x,*} \, , \\ \max_{x\in S_N} \|Ax - x\|_2 \leq 2 = L_{y,*} \, . \end{array}$$

Final bound for deterministic MDA follows from (10) with appropriate constants

$$f(\overline{x}_n) - f(x^*) \leq \frac{\sqrt{n+1}}{n} \left((2+\varepsilon)\sqrt{\ln N} + \sqrt{2} \right) \asymp (2+\varepsilon) \sqrt{\frac{\ln N}{n}} \, .$$

[6] An online choice still possible and may be studied as well, c.f. [11].

[7] It is interesting that almost the same bound with a factor around 2 was obtained in [11], c.f. Section 4, for two-sequences case and entropy proxy-function T by using norm variability and different β_k choice.

[8] All matrix elements $a^{(i,j)}$ are nonnegative, and sums in each column $\sum_{i=1}^N a^{(i,j)}$, $j = 1, \ldots, N$ are equal to 1.

A. Randomization technique

Here we briefly describe randomization, which simplifies calculation for huge-dimensional problem, for details see [7]. The key point is that we introduce randomization artificially, generating some random variables, which allow to fallback from matrix-vector multiplications to sole vectors. Using such technique allows solving problems with dense matrices A, being out of reach of sparse optimization problem statement.

In each k-th step of stochastic (or randomized) MDA we are to obtain stochastic gradients. Let's introduce two discrete independently distributed random variables (indices) $\omega_k, \chi_k \in \{1, \ldots, N\}$, which are generated by the following conditional distributions:

$$\begin{aligned} &\mathbb{P}(\chi_k = j | (x_t, y_t)_{t \le k}) = \tfrac{1}{N}, \qquad j = 1, \ldots, N, \\ &\mathbb{P}(\omega_k = i | (x_t, y_t)_{t \le k}) = x_k^{(i)}, \qquad i = 1, \ldots, N, \end{aligned}$$

and choose (random as well) vectors

$$\begin{aligned} \phi_{k+1} &= N(A^T)_{(\chi_k)} y_k^{(\chi_k)} - y_k + \varepsilon \frac{x_k}{\|x_k\|_2}, \\ \psi_{k+1} &= A_{(\omega_k)} - x_k, \end{aligned}$$

where $A_{(i)}$ represents i-th column of matrix A. Comparison with (11) reveals that for these vectors property (7) holds so we can use them as stochastic subgradients. This randomization is slightly different from the one in [7], here used vectors themselves when possible.

Let us shortly denote conditional expectation $\mathbb{E}_k\{\cdot\} \doteq \mathbb{E}\{\cdot | (x_t, y_k), t \le k\}$. Bounds $\mathbb{L}^2_{*,x}, \mathbb{L}^2_{*,y}$ are as follows:

$$\begin{aligned} \mathbb{E}_k\{\|\phi_k\|^2_\infty\} = &\sum_{j=1}^N \left\| N(A^T)_{(j)} y_k^{(j)} - y_k + \varepsilon \tfrac{x_k}{\|x_k\|_2} \right\|^2_\infty \tfrac{1}{N} \\ &\le 2N \textstyle\sum_{j=1}^N \|(A^T)_{(j)}\|^2_\infty |y_k^{(j)}|^2 + 2 \left\| y_k - \varepsilon \tfrac{x_k}{\|x_k\|_2} \right\|^2_\infty \\ &\le 2N \max_{j=1,\ldots,N} \|(A^T)_{(j)}\|^2_\infty \textstyle\sum_{i=1}^N |y_k^{(i)}|^2 + 2(1+\varepsilon)^2 \\ &\le 2N + 4 + 4\varepsilon^2 = \mathbb{L}^2_{*,x}, \\ \mathbb{E}_k\{\|\psi_k\|^2_2\} = &\textstyle\sum_{i=1}^N \|A_{(i)} - x_k\|^2_2 x_k^{(i)} \\ &\le \max_{i=1,\ldots,N} \|A_{(i)} - x_k\|^2_2 \le 4 = \mathbb{L}^2_{*,y}. \end{aligned} \tag{12}$$

Though the first bound is not very satisfying due to N appearance, it gives final estimate

$$\begin{aligned} \mathbb{E} f(\overline{x}_n) - f(x^*) &\le \tfrac{\sqrt{n+1}}{n} \left(\sqrt{2(N + 2 + 2\varepsilon^2) \ln N} + \sqrt{2} \right) \\ &\asymp \sqrt{2 \tfrac{N \ln N}{n}} \end{aligned}$$

where ε^2 is supposed to be much less than N. This bound says that at least approximately N iterations should be made to achieve $O(1)$ error. It appears that in practice this algorithm performs faster, because any specific matrix A has smaller value $\mathbb{L}_{*,x}$ and seems that bound is overrated for the problem as it will be shown in the example. There was different approach of eliminating N factor in (12) by using matrix $H\tilde{A}$ instead of A, where H is Hadamard matrix and $\tilde{A}$ is extended up to needed dimension matrix A [7]. Unfortunately, there high complexity is implicit either in the premultiplation, either appears on runtime randomization step. Namely, when using column randomization, getting random column of matrix $(H\tilde{A})^T$ in the expression of vector ϕ in (12) still requires multiplication of matrix A to a vector.

B. Calculation complexity

Here we introduce calculation complexity in terms of the two components: matrix-vector multiplication (MVM) should be thought separately[9] from a vector-vector operation (VVO) like sum of two vectors, vector to scalar multiplication or calculation of vector norm, etc. There is intuitive meaning of VVO as "(sequential) access to all elements of a vector".

For deterministic algorithm, partial derivatives (11) require[10] 2 MVMs and 5 VVOs (finding norm of vector x, negating result of MVM $A^T y$ with y, multiplying x on $\varepsilon / \|x\|_2$, summing two previous results; negating result of MVM Ax with x).

Thus step (5) of Algorithm 1 require 2 MVMs and 13 VVOs (including 4 VVOs in gradients of W, U), plus averaging (9) takes 6 VVOs. For whole algorithm up to n-th step there are $2n$ MVMs and $19n$ VVOs in total.

For the randomized algorithm, it takes 5 VVO to calculate vectors ϕ_k, ψ_k, additionally there are $2N + 2\log_2 N$ elementary operations[11] for random value generation of index ω_{k-1}. The algorithm up to n-th step uses $19n$ VVOs and $2n(N + \log_2 N)$ elementary operations in total.

To make comparison between deterministic and stochastic bounds we are to define operation cost of each VVO and MVM. Cost of a VVO is a constant number of N elementary operations, but a MVM cost depends on how sparse matrix A is. Let's put a parameter ν as sparsity coefficient, with meaning that matrix A contains νN nonzero elements. Then one MVM cost is νN elementary operations.

Ergo for fixed n deterministic algorithm have complexity of $(19 + 2\nu)nN$ and stochastic algorithm have complexity of $(21 + 2\frac{\log_2 N}{N})nN$ elementary operations. Given that on error estimates are the same (for the matrix A of interest), the prior algorithm should be chosen with knowledge of matrix A structure, namely its sparsity.

C. Example

We take same graph of movies-related web sites as in [4] with $N = 4754$ vertices and put $\varepsilon = 1$. Reference solution was computed by MOSEK toolbox in Matlab [6]. In Fig. 1 there are $q(\overline{x}_k, \overline{y}_k)$ values of deterministic and randomized MDAs for $n = 10N$ iterations. Also plotted averaged (10 runs) iterations of randomized MDA. The error well inside derived bounds, these are even out of plot. It is seen that even with worse error bound the randomized MDA outperforms the deterministic one.

[9] We'll later use the fact that normally sparse matrices A are used.

[10] These numbers may be decreased in small factor due to optimization, i.e. operation $x_1 + ax_2$ may be done in 1 VVO.

[11] Such as accessing one element of a vector, multiplying or adding two scalars, etc.

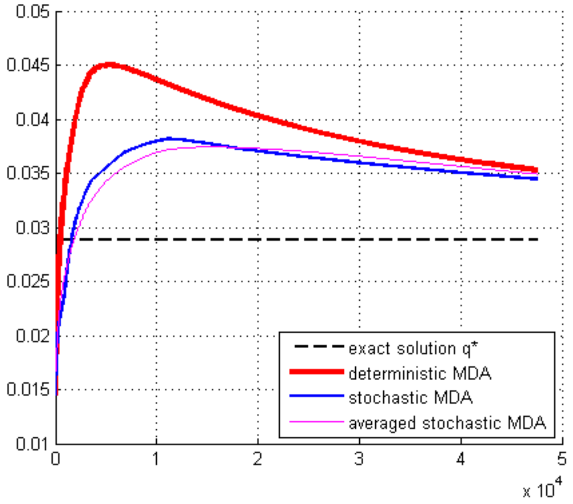

Fig. 1. $q(\overline{x}_k, \overline{y}_k)$ vs $k = 1, \dots, n$ for different MDAs.

V. PROOFS

First there is given technical lemma's proof with bounds on weighted sums of some scalar products for vectors produced by algorithm (5). This lemma uses Bregman distance properties. Then Theorems 1 and 2 with general bounds for subgradient and stochastic subgradients cases are proved.

Lemma 1 (cf. [3], with initial conditions refined): Let set $X \subset \mathbb{R}^N$ be compact and convex. Given arbitrary sequence $\phi_k \in \mathbb{R}^N, k = 1, \dots, n$, positive sequence $\gamma_k, k = 1, \dots, n$, positive non-decreasing sequence $\beta_k, k = 0, \dots, n-1$, proxy-function $V(\cdot)$ and its β-conjugate $W_\beta(\cdot)$, initial points $\zeta_0 = 0 \in \mathbb{R}^N, x_0 = -\nabla W_{\beta_0}(\zeta_0) \in X$, and process

$$\begin{array}{l} \zeta_k = \zeta_{k-1} + \gamma_k \phi_k, \\ x_k = -\nabla W_{\beta_k}(\zeta_k), \end{array} \quad k = 1, \dots, n,$$

the following inequality holds

$$\begin{aligned} &\textstyle\sum_{k=1}^n \gamma_k (x_{k-1} - x)^T \phi_k \le \\ &\quad \beta_n V^* - \beta_0 V_* + \textstyle\sum_{k=1}^n \frac{\gamma_k^2}{2\alpha_x \beta_{k-1}} \|\phi_k\|_{*,x}^2 \end{aligned}$$

for all $x \in X$. □

Proof of Lemma 1: Function $W_{\beta_{k-1}}(\cdot)$ is continuously differentiable and thus may be written in integral form

$$\begin{aligned} W_{\beta_{k-1}}(\zeta_k) = W_{\beta_{k-1}}(\zeta_{k-1}) &+ \gamma_k \phi_k^T \nabla W_{\beta_{k-1}}(\zeta_{k-1}) \\ &+ \gamma_k \textstyle\int_0^1 \phi_k^T \left[\nabla W_{\beta_{k-1}}(\tau \zeta_k + (1-\tau)\zeta_{k-1}) \right. \\ &\qquad \left. - \nabla W_{\beta_{k-1}}(\zeta_{k-1}) \right] d\tau \\ \le W_{\beta_{k-1}}(\zeta_{k-1}) &- \gamma_k \phi_k^T x_{k-1} + \tfrac{\gamma_k^2}{2\alpha_x \beta_{k-1}} \|\phi_k\|_{*,x}^2, \end{aligned}$$

where we used definition of x_{k-1}, equality $\zeta_k - \zeta_{k-1} = \gamma_k \phi_k$, inequality based on definition of dual norm $a^T b \le \|a\|_{*,x} \|b\|_x$, and Lipschitz property on ∇W_β. Noting that $W_{\beta_k}(\zeta_k) \le W_{\beta_{k-1}}(\zeta_k)$ due to monotonic decrease of W_β by β, and $\beta_i \ge \beta_{i-1}$, we can derive

$$\gamma_k x_{k-1}^T \phi_k \le W_{\beta_{k-1}}(\zeta_{k-1}) - W_{\beta_k}(\zeta_k) + \frac{\gamma_k^2 \|\phi_k\|_{*,x}^2}{2\alpha_x \beta_{k-1}}.$$

Summing over $k = 1, \dots, n$ and subtracting $x^T \zeta_n = x^T \sum_{k=1}^n \gamma_k \phi_k$ from both sides we get

$$\begin{aligned} \textstyle\sum_{k=1}^n \gamma_k (x_{k-1} - x)^T \phi_k \le W_{\beta_0}(\zeta_0) - W_{\beta_n}(\zeta_n) \\ - x^T \zeta_n + \textstyle\sum_{k=1}^n \frac{\gamma_k^2}{2\alpha_x \beta_{k-1}} \|\phi_k\|_{*,x}^2. \end{aligned}$$

The final lemma statement follows from the fact that $W_{\beta_0}(\zeta_0) = W_{\beta_0}(0) = -\beta_0 V_*$, and $-W_{\beta_n}(\zeta_n) - x^T \zeta_n \le \max_{\zeta \in \mathbb{R}^N} (-W_{\beta_n}(\zeta) - x^T \zeta) = \beta_n V(x) \le \beta_n V^*$ by duality of $W_\beta(\cdot)$ and $\beta V(\cdot)$. ■

Corollary 1: For the process

$$\begin{array}{l} \eta_k = \eta_{k-1} - \gamma_k \psi_k, \\ y_k = -\nabla U_{\delta_k}(\eta_k), \end{array} \quad k = 1, \dots, n,$$

holds

$$\begin{aligned} &\textstyle\sum_{k=1}^n \gamma_k (y - y_{k-1})^T \psi_k \le \\ &\quad \delta_n T^* - \delta_0 T_* + \textstyle\sum_{k=1}^n \frac{\gamma_k^2}{2\alpha_y \delta_{k-1}} \|\psi_k\|_{*,y}^2 \,. \end{aligned}$$

□

The corollary proof is the same as that of Lemma 1 with $y, \eta, \psi, T, U, \alpha_y, \|\cdot\|_{*,y}$ instead of $x, \zeta, \phi, V, W, \alpha_x, \|\cdot\|_{*,x}$. The only difference is that minus sign in formulae for η_k results in different negation order of the left side expression.

Proof of Theorem 1: For brevity, let's define

$$P_n \doteq R_n + \sum_{k=1}^n \frac{\gamma_k^2}{2} \left(\frac{L_{*,x}^2}{\alpha_x \beta_{k-1}} + \frac{L_{*,y}^2}{\alpha_y \delta_{k-1}} \right).$$

From convexity-concavity of function q and the property of subgradients we have for all $(x, y) \in X \times Y, k = 0, \dots, n-1$,

$$\begin{array}{l} q(x_k, y_k) - q(x, y_k) \le (x_k - x)^T q_x(x_k, y_k), \\ q(x_k, y) - q(x_k, y_k) \le (y - y_k)^T q_y(x_k, y_k). \end{array} \tag{13}$$

By summing both hand sides in (13), multiplying by γ_k, and applying the bounds obtained in Lemma 1 and Corollary 1 for the case $\phi_k = q_x(x_{k-1}, y_{k-1}), \psi_k = q_y(x_{k-1}, y_{k-1})$ we have

$$\sum_{k=1}^n \gamma_k (q(x_{k-1}, y) - q(x, y_{k-1})) \le P_n.$$

Using Jensen's inequality with convexity (concavity) over arguments with nonnegative γ_k we get

$$\begin{aligned} &q(\overline{x}_n, y) = q\left(\tfrac{1}{\sum_{k=1}^n \gamma_k} \textstyle\sum_{k=1}^n \gamma_k x_k, y \right) \le \\ &\tfrac{1}{\sum_{k=1}^n \gamma_k} \textstyle\sum_{k=1}^n \gamma_k q(x_k, y), \\ &-q(x, \overline{y}_n) = -q\left(x, \tfrac{1}{\sum_{k=1}^n \gamma_k} \textstyle\sum_{k=1}^n \gamma_k y_k \right) \le \\ &-\tfrac{1}{\sum_{k=1}^n \gamma_k} \textstyle\sum_{k=1}^n \gamma_k q(x, y_k), \end{aligned} \tag{14}$$

and conclude that $q(\overline{x}_n, y) - q(x, \overline{y}_n) \le \frac{1}{\sum_{k=1}^n \gamma_k} P_n$.

Taking a maximizer in Y and minimizer in X on the left-side difference, we arrive at the theorem's stated bound on error $\Delta(\overline{x}_n, \overline{y}_n)$ (4). ■

Now we are to prove stochastic case. Its proof follows the one of Theorem 1 but steps are taken in different order.

Proof of Theorem 2: Similarly, denote

$$Q_n \doteq R_n + \sum_{k=1}^n \frac{\gamma_k^2}{2} \left(\frac{\mathbb{L}_{*,x}^2}{\alpha_x \beta_{k-1}} + \frac{\mathbb{L}_{*,y}^2}{\alpha_y \delta_{k-1}} \right).$$

Sum the bounds proved in Lemma 1 and Corollary 1, then divide on $\sum_{i=1}^n \gamma_k$, apply mathematical expectation with bounds (8), and get

$$\begin{aligned}&\mathbb{E}\left(\frac{1}{\sum_{k=1}^n \gamma_k}\sum_{k=1}^n \gamma_k (x_{k-1}-x)^T\phi_k\right)\\ &+\mathbb{E}\left(\frac{1}{\sum_{k=1}^n \gamma_k}\sum_{k=1}^n \gamma_k (y-y_{k-1})^T\psi_k\right) \leq \quad \frac{Q_n}{\sum_{k=1}^n \gamma_k}.\end{aligned}$$

Consider only x-related terms like $(x_{k-1}-x)^T\phi_k$. As soon each next point (x_k, y_k) depends on previous stochastic gradients $\phi_t, \psi_t, t < k$, for each term we should sequentially take conditional expectation. I.e. for $\mathbb{E}(\phi_1|x_0, y_0) = q_x(x_0, y_0)$ it is evident as initial conditions are fixed. Next, $\mathbb{E}(x_1|x_0, y_0) = x_1$ became deterministic[12] function only on x_0, y_0, and do not depend on random ϕ_1, ψ_1. Same applied to y-related terms $(y - y_{k-1})^T\psi_k$. Step-by-step following timeline further with property (7) we get

$$\begin{aligned}&\frac{1}{\sum_{k=1}^n \gamma_k}\sum_{k=1}^n \gamma_k \mathbb{E}((x_{k-1}-x)^T q_x(x_{k-1}, y_{k-1}))\\ &+\frac{1}{\sum_{k=1}^n \gamma_k}\sum_{k=1}^n \gamma_k \mathbb{E}((y-y_{k-1})^T q_y(x_{k-1}, y_{k-1}))\\ &\qquad\qquad\qquad\qquad \leq \frac{1}{\sum_{k=1}^n \gamma_k}Q_n.\end{aligned}$$

where expectation is over variables $x_k, y_k, k = 1, \ldots, n-1$ only. Next following subgradient inequalities (13) with Jensen's ones (14), and taking expectation we get

$$\begin{aligned}&\mathbb{E}(q(\overline{x}_n, y) - q(x, \overline{y}_n))\\ &\leq \frac{1}{\sum_{k=1}^n \gamma_k}\sum_{k=1}^n \gamma_k \mathbb{E}((x_{k-1}-x)^T q_x(x_{k-1}, y_{k-1}))\\ &\quad + \frac{1}{\sum_{k=1}^n \gamma_k}\sum_{k=1}^n \gamma_k \mathbb{E}((y-y_{k-1})^T q_y(x_{k-1}, y_{k-1})),\end{aligned}$$

which is valid for all $x \in X, y \in Y$. Combining two previous inequalities (for any fixed n) and taking maximum on $y \in Y$ and minimum on $x \in X$ leads to the theorem's result. ■

VI. CONCLUSIONS AND FUTURE WORKS

A. Conclusions

In the paper extension of mirror descent algorithm for convex-concave saddle point problems is derived by introducing extra parameter sequence. This approach allows to obtain refined error bounds, explained proofs provided. The explored mirror descent algorithms (both deterministic and randomized) were applied to robust PageRank problem, followed by complexity analysis and comparison of these algorithms. Also minor fix in initial condition setup made for the algorithms' grounding lemma.

B. Future Works

There are quite a number of possible improvements for this direction. First, in complexity analysis for robust PageRank calculation above no storage related and access-time related issues were touched, which would also affect complexity bounds, especially cost of MVM operations.

Some bound improvement could be reached by using adaptive alteration of algorithm's parameters γ_k, β_k, δ_k through iterations versus using a priori setup.

Another thought is that in contrast with deterministic algorithm stochastic one guarantees only estimate on expectation, with sampled random output and its quality (e.g. standard deviation) should be studied in future works as well. Still should be kept in mind that for huge dimensional problems small numeric calculation errors tend to accumulate, so deterministic algorithm may behave like stochastic one. For sparse problems a comparison with specific sparse optimization methods can be studied.

Finally, it is interesting to apply MDA for slightly different version of robust PageRank, using another norms, e.g. minimizing

$$f_1(x) = \|Ax - x\|_\infty + \varepsilon\|x\|_\infty.$$

While optimization with such norms could hardly be done directly under high dimension, corresponding saddle point problem may be solved efficiently. The authors assume that constants (12) in this case won't depend on N at all.

VII. ACKNOWLEDGMENTS

The authors thank Boris Polyak for encouraging discussion of robust PageRank, and anonymous referees for valuable notes and suggestions.

REFERENCES

[1] A. Beck, M. Teboulle, Mirror Descent and Nonlinear Projected Subgradient Methods for Convex Optimization, *Operations Research Letters*, vol. 31, 2003, pp. 167–175.

[2] S. Brin and L. Page, The anatomy of a large-scale hypertextual Web search engine, *in Proceedings of the seventh international conference on World Wide Web 7*, Brisbane, Australia, 1998, pp. 107–117.

[3] A. B. Juditsky, A. V. Nazin, A. B. Tsybakov, and N. Vayatis, Recursive Aggregation of Estimators by the Mirror Descent Algorithm with Averaging, *Problems of Information Transmission*, Vol. 41, No. 4, 2005, pp. 368–384.

[4] A. Juditsky and B. Polyak, Robust Eigenvector of a Stochastic Matrix with Application to PageRank, *in 51st IEEE Conference on Decision and Control, CDC 2012*, Maui, Hawaii, USA, 2012. pp. 3171–3176.

[5] A. N. Langville and C. D. Meyer, *Google's PageRank and Beyond: The Science of Search Engine Rankings*, Princeton University Press; 2009.

[6] E. Andersen, D. Andersen, MOSEK Optimization Tools Manual, http://docs.mosek.com/6.0/tools/index.html

[7] A. V. Nazin, Estimating the principal eigenvector of a stochastic matrix: Mirror Descent Algorithms via game approach with application to PageRank problem, *in 49th IEEE Conference on Decision and Control, CDC 2010*, Atlanta, Georgia, USA, 2010, pp. 792–797.

[8] A. Nazin and B. Polyak, Adaptive Randomized Algorithm for Finding Eigenvector of Stochastic Matrix with Application to PageRank, *in 48th Conference on Decision and Control, CDC 2009*, Shanghai, P.R. China, 2009, pp. 127–132.

[9] A. Nemirovski, A. Juditsky, G. Lan and A. Shapiro, Robust stochastic approximation approach to stochastic programming, *SIAM J. Optim.*, vol. 19, No. 4, 2009, pp. 1574–1609.

[10] A. Nemirovski and D. Yudin, *Problem complexity and method efficiency in optimization*, J. Wiley & Sons, 1983.

[11] Yu. Nesterov, Primal-dual subgradient methods for convex problems, *Math. Program., Ser. B*, vol. 120, iss. 1, 2009, pp. 221-259.

[12] B. T. Polyak and A. A. Tremba, Regularization-based solution of the PageRank problem for large matrices, Automation and Remote Control, 2012, vol. 73, issue 11, pp. 1877–1894.

[13] R.T. Rockafellar and R.J.B. Wets, *Variational Analysis*, New-York, Springer; 1998.

[12] We remind that there are used deterministic pick-up of subgradient element $q_x(x_k, y_k)$ in cases of non-one-element subgradient sets.