

IPAS: An Adaptive Sample Size Method for Weighted Finite Sum Problems with Linear Equality Constraints

Nataša Krejić*, Nataša Krklec Jerinkić †, Sanja Rapajić‡, Luka Rutešić §¶

Abstract

Optimization problems with the objective function in the form of weighted sum and linear equality constraints are considered. Given that the number of local cost functions can be large as well as the number of constraints, a stochastic optimization method is proposed. The method belongs to the class of variable sample size first order methods, where the sample size is adaptive and governed by the additional sampling technique earlier proposed in unconstrained optimization framework. The resulting algorithm may be a mini-batch method, increasing sample size method, or even deterministic in a sense that it eventually reaches the full sample size, depending on the problem and similarity of the local cost functions. Regarding the constraints, the method uses controlled, but inexact projections on the feasible set, yielding possibly infeasible iterates. Almost sure convergence is proved under some standard assumptions for the stochastic framework, without imposing the convexity. Numerical results on relevant problems from CUTEst collection and real-world data sets for logistic regression show the stability and the efficiency of the proposed method when compared to the state-of-the-art methods.

Key words: Constrained Optimization, Projected Gradient Methods, Sample Average Approximation, Adaptive Variable Sample Size Strategies, Nonmonotone Line Search, Additional Sampling.

*Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: natasak@uns.ac.rs

†Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: natasa.krklec@dmf.uns.ac.rs

‡Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: sanja@dmf.uns.ac.rs

§Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Trg Dositeja Obradovića 4, 21000 Novi Sad, Serbia. e-mail: luka.rutesic@dmf.uns.ac.rs

¶Corresponding author

1 Introduction

We consider constrained optimization problem with the objective function in the form of weighted finite sum and linear equality constraints, i.e.,

$$\min_{Ax=b} f(x) := \sum_{i=1}^N w_i f_i(x), \quad (1.1)$$

where $f_i : \mathbb{R}^n \rightarrow \mathbb{R}, i = 1, \dots, N$ are continuously-differentiable functions, w_1, \dots, w_N represent the weights such that

$$\sum_{i=1}^N w_i = 1, \quad w_i \geq 0, \quad i = 1, \dots, N, \quad (1.2)$$

$b \in \mathbb{R}^m$ and $A \in \mathbb{R}^{m \times n}$ is assumed to be a full-rank matrix, $\text{rank}(A) = m \leq n$.

The considered problems come from different fields, mainly including Big Data (BD) problems commonly present in Machine Learning (ML). Notice that the choice of $w_i = 1/N$ for all $i = 1, \dots, N$ yields the standard form of the finite sum objective function $f(x) = \frac{1}{N} \sum_{i=1}^N f_i(x)$. Introduction of possibly different weights is motivated by the so called local regression models (see [19] for instance) in ML where the model parameters are recalculated for any new data point depending on its position in the space of attributes. In such approach, the objective function usually takes into account the distance between the new point and data points from the training data set. The aforementioned distances represent the weights in problem (1.1). On the other hand, the weights w_i can be viewed as probabilities of choosing the corresponding functions f_i and, in that case, the objective function represents mathematical expectation. The motivation for emphasizing weights and observing them separately from the functions f_i comes from stochastic framework. Namely, this allows stochastic algorithms to favor the functions that are more important by giving them better chances to be chosen as explained in the sequel of the paper. Moreover, some of the ML problems also include linear constraints. One of the examples is a Ridge regression which can be stated in the form of constrained optimization problem [19]. Another example would be Markowitz utility function minimization, i.e., the problems of finding optimal portfolio that minimizes the risk and maximizes the return, while the sum of unknowns must be equal to one. The two mentioned examples include only modest number of constraints and the projection on the feasible set is not a big challenge. However, data fitting problems in general (e.g. least squares) may also include a large number of (linear) constraints (see e.g. [36] for further references). In that case, projecting on the feasible set is too expensive and inexact projections may be a better option. Therefore, the problem that we consider assembles different types of problems and generalizes several important classes.

A variety of line search and trust region methods have been proposed to solve nonlinear constrained problems. Various algorithms have been designed to solve deterministic equality-constrained optimization problems (see [6,11] for further references), while recent research has focused on developing stochastic optimization algorithms. There has been a growing interest in adapting line search and trust region methods in stochastic framework for unconstrained optimization problems [1–5, 9, 10, 12, 14, 20, 22, 24, 26–28], but significantly fewer algorithms have been proposed to solve stochastic equality-constrained optimization problems (see [6] for further references and [7, 13, 15, 34, 37]).

Projected gradient methods can be used to solve constrained optimization problems, see [8]. Their upcoming generalization to stochastic framework have been investigated in [23, 25, 33]. A novel class of projected gradient methods for constrained minimization both in deterministic and stochastic settings is proposed in [29]. Large scale problems require inexact projections because of computational cost.

Numerous first order and second order algorithms have been developed for solving unconstrained finite sum minimization [1–4, 14, 26, 28]. These methods are based on nonmonotone line search or trust region technique. The usefulness of line search nonmonotonicity in deterministic case [18, 30] was also demonstrated in a stochastic case. A class of algorithms which uses nonmonotone line search rule fitting a variable sample size scheme at each iteration was proposed in [22].

In large-scale problems, the computation of the objective function and its gradient and additionally its Hessian is expensive, so their approximations are generally used in order to reduce the computational cost. Subsampling is a natural way of computing these approximations, so the large dimension of the problem and specific form of the objective function suggest an adaptive subsampling. Adaptive sample size strategies for finite sum problems are presented in [1–3, 14, 26, 28] and some other types of adaptive subsample approach can also be found in [4, 9, 10, 12, 23].

Additional sampling (two step sampling in each iteration) presented in [14, 26, 28] plays an important role for controlling the sample size. It is used as a control for accepting the step and increasing the sample size if necessary. The additional sampling can be arbitrarily cheap, i.e., even the sample size 1 is sufficient, and hence it does not make the process more expensive. Other additional line search and trust region sample size strategies can be found in [20, 27].

The method we propose here belongs to the family of projected gradient methods, with the step size determined by a nonmonotone line search rule. The specific form of the objective function, in particular the case of large N and n motivates the adaptive subsample approach. Thus we work with approximate objective function (and the corresponding gradient), in other words with random linear models. The sample size that defines the model

in each iteration is computed according to the estimated progress towards minimizer in each iteration. The approximate gradient direction is then projected inexactly to the feasible set, yielding a new iteration that might be infeasible in a controlled way. Inexact projection in fact means that we will solve the corresponding system of linear equations only approximately, using any linear solver. The progress is measured by an additional approximation of the objective function, which can be very rough, in fact even the sample of size 1 will be suitable for this additional objective function approximation. Besides the measure of progress the additional sampling done in each iteration allows us to overcome theoretical difficulties that arise from the fact that the direction and step size are not independent random variables and hence one cannot apply the martingale theory for theoretical analysis. Such difficulties can be overcome with the predetermined step size, for instance $1/k$, which yields a.s. convergence in stochastic gradient descent framework (see e.g. [1] for further references). But in that case the step sizes become very small very fast and hence the method is very slow. Furthermore, the method we propose here is adaptive in the sense that sample size increase is problem dependent. Roughly speaking the similarity of functions f_i governs the process and the iterative procedure might end without reaching the full sample, i.e., with very cheap iterations, or the exact objective function and the gradient are used at the final stages of the iterative procedure.

Random linear models that we use can be more or less similar to the original function f and thus the decrease of the model might not be a decrease for the true objective function. Therefore we use a nonmonotone line search procedure [18,30], that is more relaxed in accepting the step than the classical Armijo rule. The same approach is exploited in several methods with random models, [3, 5, 14, 23, 28]. Putting together the random linear models, nonmonotone line search procedure, additional sampling and inexact projections we end up with an efficient and theoretically sound method that is almost surely (a.s.) converging to a stationary point of (1.1).

To summarize, our main contributions may be listed as follows:

- The additional sampling concept for solving unconstrained finite sum problems [14,26,28] is incorporated into the stochastic method to solve constrained optimization problems with weighted sum objective functions.
- The proposed method relaxes the common assumption of feasible iterates in stochastic projected gradient framework (e.g. [23]) and allows controlled, but inexact projections which can be of great significance for problems with large number of constraints.
- Almost sure convergence is proved under rather standard assumptions for stochastic optimization framework.

- The efficiency of the proposed method is confirmed through a number of tests performed on both academic and real-world data problems.

The paper is organized as follows. Section 2 contains basic definitions and statements known from the literature. It also provides some basic concepts and preliminary results needed for further analysis. The proposed method - IPAS is stated and explained in Section 3, while the convergence analysis is delegated to Section 4. Section 5 is devoted to numerical results, while the main conclusions are derived in final section.

2 Preliminaries

Let us start with explaining the inexact projections we will use in the algorithm. Under the assumption of fully ranked A , one can show that the orthogonal projection $\pi_S(y)$ of a point y on the feasible set $S := \{x \in \mathbb{R}^n \mid Ax = b\}$ is given by

$$\pi_S(y) = y - A^T(AA^T)^{-1}(Ay - b). \quad (2.1)$$

The above equality comes from the fact that $\pi_S(y) = \operatorname{argmin}_{Ax=b} \frac{1}{2} \|y - x\|^2$ and, since the orthogonal projection problem is convex, the solution is determined by the KKT conditions

$$A \pi_S(y) = b, \quad A^T \lambda = y - \pi_S(y),$$

where λ represents the vector of Lagrange multipliers. Multiplying the second equation with A from the left and using the first one, we obtain

$$AA^T \lambda = Ay - b. \quad (2.2)$$

This together with $\pi_S(y) = y - A^T \lambda$ yields (2.1). The important feature for our analysis lies in the fact that the projection operator to the set S has the following property.

Lemma 2.1. *Let $A \in \mathbf{R}^{m \times n}$ with $\operatorname{rank}(A) = m$ and $w_i \geq 0, i = 1, \dots, N$, $\sum_{i=1}^N w_i = 1$. For any set of points $y^i \in \mathbf{R}^n, i = 1, \dots, N$ there holds*

$$\pi_S\left(\sum_{i=1}^N w_i y^i\right) = \sum_{i=1}^N w_i \pi_S(y^i). \quad (2.3)$$

Proof. Using (2.1), there follows

$$\begin{aligned} \pi_S\left(\sum_{i=1}^N w_i y^i\right) &= \sum_{i=1}^N w_i y^i - A^T(AA^T)^{-1}\left(A \sum_{i=1}^N w_i y^i - b\right) \\ &= \sum_{i=1}^N w_i y^i - A^T(AA^T)^{-1}\left(A \sum_{i=1}^N w_i y^i - \sum_{i=1}^N w_i b\right) \\ &= \sum_{i=1}^N w_i (y^i - A^T(AA^T)^{-1}(Ay^i - b)) = \sum_{i=1}^N w_i \pi_S(y^i). \end{aligned}$$

□

To compute the exact projection on the set S one needs to solve the system (2.2) exactly. In some applications, if the number of equalities is modest one can use the closed form (2.1) for any given point y . However, if the dimension of the problem is very large and/or the number of equality constraints is large, finding the exact solution to (2.2) can be impractical. Thus, we will assume that the linear system is solved only approximately. The quality of inexact projection in each iteration will be controlled by the norm of the residual vector defined as follows. Let us denote by $\tilde{\pi}_S(y)$ the inexact projection of point y on feasible set S , more precisely,

$$\tilde{\pi}_S(y) = y - A^T \tilde{\lambda}(y), \quad (2.4)$$

where $\tilde{\lambda}(y)$ is an approximate solution of (2.2). The residual is denoted by

$$r(y) := AA^T \tilde{\lambda}(y) - Ay + b, \quad (2.5)$$

while the feasibility measure of point y is defined as

$$e(y) = \|Ay - b\|. \quad (2.6)$$

We will state the condition for the residual vector in each iteration a bit ahead. The following simple lemma will be used later on.

Lemma 2.2. *Let $z \in \mathbb{R}^n$ be an arbitrary point and $\tilde{\lambda}(z)$ be an approximate solution of (2.2) such that $\|r(z)\| \leq M$ with $M > 0$. Then $e(\tilde{\pi}_S(z)) \leq M$.*

Proof. The condition

$$\|r(z)\| = \|AA^T \tilde{\lambda}(z) - Az + b\| \leq M$$

implies

$$\begin{aligned} e(\tilde{\pi}_S(z)) &= \|A\tilde{\pi}_S(z) - b\| = \|A(z - A^T \tilde{\lambda}(z)) - b\| \\ &= \|Az - b - AA^T((AA^T)^{-1}(Az - b + r(z)))\| \\ &= \|Az - b - Az + b - r(z)\| = \|r(z)\| \leq M, \end{aligned}$$

which completes the proof. □

One can show that the projected gradient direction of the form

$$d(x) = \pi_S(x - \nabla f(x)) - x \quad (2.7)$$

is a descent direction for function f at point $x \in S$ unless x is a stationary point for problem (1.1). More precisely, the following result is known.

Theorem 2.3. [8] *Assume that $f \in C^1(S)$ and $x \in S$. Then the projected gradient direction (2.7) satisfies:*

$$a) \ d(x)^T \nabla f(x) \leq -\|d(x)\|^2.$$

$$b) \ d(x) = 0 \text{ if and only if } x \text{ is a stationary point for problem (1.1).}$$

The method we propose will be based on approximate objective function and the gradient. Let us denote by $f_{\mathcal{N}_k}$ the approximation of function f used in iteration k , i.e.,

$$f_{\mathcal{N}_k}(x) := \frac{1}{N_k} \sum_{i \in \mathcal{N}_k} f_i(x), \quad (2.8)$$

where $N_k := |\mathcal{N}_k|$, $\mathcal{N}_k = \{i_1^k, \dots, i_{N_k}^k\}$, and each $i_j^k \in \mathcal{N}_k$ takes the value $s \in \mathcal{N} := \{1, \dots, N\}$ with probability w_s , i.e.,

$$P(i_j^k = s) = w_s, s = 1, \dots, N, j \in \mathcal{N}_k, k \in \mathbb{N}. \quad (2.9)$$

This way we have an unbiased estimate of f , i.e.,

$$E(f_{\mathcal{N}_k}(x)|x) = E\left(\frac{1}{N_k} \sum_{j=1}^{N_k} f_{i_j^k}(x)|x\right) = \frac{1}{N_k} \sum_{j=1}^{N_k} E(f_{i_j^k}(x)|x) = \frac{1}{N_k} \sum_{j=1}^{N_k} f(x) = f(x).$$

As already stated, in the proposed algorithm we use inexact projections of the approximate gradient $\nabla f_{\mathcal{N}_k}$ which can yield nondescent directions p_k and infeasible points x_k . Moreover, we deal with approximate functions and thus imposing monotone line search is not beneficial in general. Thus, we use nonmonotone Armijo-type line search, [30] to determine the step size t_k

$$f_{\mathcal{N}_k}(x_k + t_k p_k) \leq f_{\mathcal{N}_k}(x_k) + c_1 t_k (\nabla f_{\mathcal{N}_k}(x_k))^T p_k + \varepsilon_k, \quad (2.10)$$

with some $\varepsilon_k > 0$ which satisfies $\sum_{k=0}^{\infty} \varepsilon_k < \infty$.

Clearly, the direction p_k and the step size t_k obtained in (2.10) both depend on \mathcal{N}_k . Therefore we can not rely on the martingale theory commonly used in stochastic gradient methods such as SGD [35] where the predetermined step size sequence is used. To overcome this difficulty and avoid predefined step sizes we extend the idea of additional sampling. This simple and computationally cheap remedy is successfully used in [14, 28]. Other possibilities are given in [20, 27]. The method proposed in [26] also uses additional sampling but in trust-region framework.

Similarly as for $f_{\mathcal{N}_k}$, we form an additional sampling approximation $f_{\mathcal{D}_k}$ by

$$f_{\mathcal{D}_k}(x) := \frac{1}{D_k} \sum_{i \in \mathcal{D}_k} f_i(x), \quad (2.11)$$

where $D_k := |\mathcal{D}_k|$, $\mathcal{D}_k = \{l_1^k, \dots, l_{D_k}^k\}$, and each $l_j^k \in \mathcal{D}_k$ takes the value $s \in \mathcal{N} := \{1, \dots, N\}$ with probability w_s , i.e.,

$$P(l_j^k = s) = w_s, s = 1, \dots, N, j \in \mathcal{D}_k, k \in \mathbb{N}. \quad (2.12)$$

The key point for the efficiency of this approach lies in the fact that the cardinality of \mathcal{D}_k is arbitrary, with only requirement being $D_k \leq N - 1$. Thus one can even take $D_k = 1$ in each iteration. The numerical experiments presented in Section 4 are performed with $D_k = 1$.

The following technical lemmas will be used further on.

Lemma 2.4. *Let $e_{k+1} \leq \theta e_k + \eta_k$ for all $k \geq 0$ with $e_k \geq 0, k = 1, 2, \dots, \theta \in [0, 1)$ and $\{\eta_k\}$ satisfying $\lim_{k \rightarrow \infty} \eta_k = 0$. Then*

$$\lim_{k \rightarrow \infty} e_k = 0.$$

Proof. Applying the induction argument we can show that

$$e_k \leq \theta^k e_0 + s_k,$$

where $s_k = \sum_{j=1}^k \theta^{j-1} \eta_{k-j}$. Under the stated conditions there holds $\lim_{k \rightarrow \infty} s_k = 0$ (see [38, Lemma 3.1, (a)]) and we conclude that $\lim_{k \rightarrow \infty} e_k = 0$.

□

3 The Method

The method we consider will generate an infinite sequence of iterations x_k . As already stated in each iteration we will use a subsample \mathcal{N}_k and the corresponding function $f_{\mathcal{N}_k}(x_k)$ and the gradient $\nabla f_{\mathcal{N}_k}(x_k)$. The inexact projection $\tilde{\pi}_S$ will be used to generate the search direction with the approximation error controlled by a nonincreasing sequence of positive numbers η_k such that

$$\sum_{k=0}^{\infty} \eta_k^2 \leq \bar{\eta} < \infty. \quad (3.1)$$

Thus, for $y_k = x_k - \nabla f_{\mathcal{N}_k}(x_k)$ we will require that the projection residuals satisfy

$$\|r(y_k)\| \leq \eta_k. \quad (3.2)$$

Clearly, the inexactness of the projection will decrease as k increases and eventually we will approach the feasible set. The search direction p_k will be computed as usual, as the difference of inexact projection $\tilde{\pi}_S(y_k)$ and the current approximation x_k . After the computation of p_k we distinguish two cases. If $N_k < N$ we proceed to determine the step size by the nonmonotone rule (2.10) and define $\bar{x}_k = x_k + t_k p_k$. The sequence of parameters ε_k is taken as $\varepsilon_k = \eta_k^2$ in Step 3 of the algorithm, (3.5). Theoretically, we can take another sequence in the line search but this way the inexactness of the projection is taken into account in computation of the step size t_k and we allow more freedom for lower accuracy projection and increase the requirements as k increases. Adding η_k^2 in the decrease condition allow us

to compute a suitable step size even if the direction is nondecreasing and hence the line search rule (3.5) is always well defined, even without the lower bound t_{min} posed in the mini-batch case of Step S3.

In the case $N_k = N$, i.e., if the full precision in the objective function is needed, the projection could still be inaccurate and hence we might search along infeasible direction. In that case we check if the search direction is decreasing. If yes, we proceed to the line search. Else, if the direction is not sufficiently decreasing, we discard the search direction and take a new projection of the current iteration x_k to get x_{k+1} and terminate the iteration. In this case we will call the iteration unsuccessful.

For all iterations where $N_k < N$ in Step 4 of the algorithm we perform additional sampling, taking a new sample \mathcal{D}_k , independently of \mathcal{N}_k . As already mentioned, this step is meant to be a computationally cheap measure of progress as \mathcal{D}_k can be arbitrary small and we will work with $D_k = 1$ in our numerical tests. For this new sample \mathcal{D}_k we compute the direction $u_k = x_k - \nabla f_{\mathcal{D}_k}(x_k)$ and then project it approximately to get $s_k = \tilde{\pi}_S(u_k) - x_k = -\nabla f_{\mathcal{D}_k}(x_k) - A^T \tilde{\lambda}_k(u_k)$. Here we keep the same accuracy of the projection as in Step 2 for p_k . Notice that for $N_k = N$ this additional sampling is not needed as the line search in Step 3 is performed with the true objective function.

Finally, at Step 5 we update the iteration. So, if we have enough decrease in the objective function $f_{\mathcal{D}_k}$, according to (3.7), we update the iteration and keep the same sample size for the next iteration. Roughly speaking we are saying here that $f_{\mathcal{N}_k}$ is a good approximation of the objective function as the decrease condition holds for another (independently sampled) function $f_{\mathcal{D}_k}$. Notice that the condition (3.7) is looser than the step size rule (3.5) as η_k^2 is multiplied with some constant C which can be large.

Algorithm 1: IPAS (Inexact Projection with Additional Sampling)

S0 Initialization. Input: $x_0 \in \mathbb{R}^n, N_0 \in \mathbb{N}, \beta, c, c_1, t_{min} \in (0, 1), C > 0, \{\eta_k\}$ satisfying (3.1), $k := 0$.

S1 Subsampling. If $N_k < N$, choose \mathcal{N}_k via (2.9). Else, set $f_{\mathcal{N}_k} = f$.

S2 Search direction. Compute

$$p_k = \tilde{\pi}_S(y_k) - x_k = -\nabla f_{\mathcal{N}_k}(x_k) - A^T \tilde{\lambda}_k(y_k) \quad (3.3)$$

with $y_k = x_k - \nabla f_{\mathcal{N}_k}(x_k)$, and $\tilde{\lambda}_k(y_k)$ satisfying (3.2).

If $N_k < N$ go to step S3.

If $N_k = N$, and

$$(\nabla f(x_k))^T p_k \leq -c \|p_k\|^2 \quad (3.4)$$

go to step S3.

Else set $x_{k+1} = \tilde{\pi}_S(x_k)$ with $\tilde{\lambda}_k(x_k)$ satisfying $\|r(x_k)\| \leq \eta_k$, set $k = k + 1$ and go to step S1.

S3 Step size. If $N_k = N$, find the smallest $j \in \mathbb{N}_0$ such that $t_k = \beta^j$ satisfies

$$f_{\mathcal{N}_k}(x_k + t_k p_k) \leq f_{\mathcal{N}_k}(x_k) + c_1 t_k (\nabla f_{\mathcal{N}_k}(x_k))^T p_k + \eta_k^2. \quad (3.5)$$

Else, if $N_k < N$, starting with $t_k = 1$, while $t_k \geq t_{min}$ and

$$f_{\mathcal{N}_k}(x_k + t_k p_k) > f_{\mathcal{N}_k}(x_k) + c_1 t_k (\nabla f_{\mathcal{N}_k}(x_k))^T p_k + \eta_k^2,$$

reduce t_k by factor β .

Set $\bar{x}_k = x_k + t_k p_k$.

S4 Additional sampling.

If $N_k = N$, set $x_{k+1} = \bar{x}_k$, $k = k + 1$ and go to step S1.

Else choose \mathcal{D}_k via (2.12) and compute

$$s_k = \tilde{\pi}_S(u_k) - x_k = -\nabla f_{\mathcal{D}_k}(x_k) - A^T \tilde{\lambda}_k(u_k) \quad (3.6)$$

with $u_k = x_k - \nabla f_{\mathcal{D}_k}(x_k)$ and $\tilde{\lambda}_k(u_k)$ satisfying $\|r(u_k)\| \leq \eta_k$.

S5 The update. If

$$f_{\mathcal{D}_k}(\bar{x}_k) \leq f_{\mathcal{D}_k}(x_k) - c\|s_k\|^2 + C\eta_k^2, \quad (3.7)$$

set $x_{k+1} = \bar{x}_k$ and $N_{k+1} = N_k$.

Else set $x_{k+1} = x_k$, choose $N_{k+1} \in \{N_k + 1, \dots, N\}$.

Set $k = k + 1$ and go to step S1.

If the condition (3.7) is not satisfied we take $x_{k+1} = x_k$ and increase the sample size N_{k+1} for the new iteration. Essentially we reason here that the trial iteration should not be accepted because we need more precision and hence we increase the sample size.

Remark. An important point here is to notice that we do not need to specify how to increase the sample size if needed, i.e., we are free to choose any $N_{k+1} > N_k$. Naturally, one can choose to increase the sample size very slowly to keep the iterations computationally cheap, or to increase the sample size faster to achieve a better approximation of the objective function and the gradient, hoping to end up the process in fewer iterations. Clearly, the sample size scheduling is problem dependent.

4 Convergence analysis

Assumption A 1. Each function $f_i, i = 1, \dots, N$ is continuously differentiable with Lipschitz continuous gradient and bounded from below by a constant f_{low} .

This assumption implies that $f(x) \geq f_{\text{low}}$ for all $x \in \mathbb{R}^n$ and it also holds for any approximate function $f_{\mathcal{N}_k}$ and $f_{\mathcal{D}_k}$. Furthermore all approximate gradients are also Lipschitz continuous and without loss of generality we may assume that $L > 0$ is a common Lipschitz constant.

The algorithm we consider generates a set of random iteration $\{x_k\}$. Nevertheless some properties hold for all iterations, independently of the sample we use to generate them. First of all, we can show using the standard arguments and Assumption A1 that the step size t_k generated in Step 3, (3.5) is bounded from below.

Lemma 4.1. *Assume that Assumption A1 holds and that step size t_k is computed in Step 3 of IPAS algorithm. Then $t_k \geq t_{\min}$ provided that $t_{\min} < \min\{1, \frac{2\beta c(1-c_1)}{L}\}$.*

Proof. If the full sample is reached, the line search is performed only if (3.4) holds and it can be proved (see [28] for example) that $t_k \geq \frac{2\beta c(1-c_1)}{L}$. On the other hand, if $N_k < N$, the line search yields $t_k \geq t_{\min}$ by the construction of Step 3 of IPAS algorithm. \square

Furthermore, we can prove that the feasibility of iterations \bar{x}_k is eventually increasing although the projections are inexact. The following statement holds.

Lemma 4.2. *Assume that Assumption A1 holds. Then*

$$e(\bar{x}_k) \leq (1 - t_{\min})e(x_k) + \eta_k. \quad (4.1)$$

Proof. Given that

$$\begin{aligned} \bar{x}_k &= x_k + t_k p_k = x_k + t_k (\tilde{\pi}_S(y_k) - x_k) \\ &= (1 - t_k)x_k + t_k \tilde{\pi}_S(y_k), \end{aligned}$$

and

$$e(\bar{x}_k) = \|A((1 - t_k)x_k + t_k \tilde{\pi}_S(y_k)) - b\| \quad (4.2)$$

$$\leq (1 - t_k)\|Ax_k - b\| + t_k\|A\tilde{\pi}_S(y_k) - b\| \quad (4.3)$$

$$= (1 - t_k)e(x_k) + t_k e(\tilde{\pi}_S(y_k)). \quad (4.4)$$

Since $t_k \geq t_{\min}$ and the residual of $\tilde{\pi}_S(y_k)$ satisfies (3.2), by Lemma 2.2 we have $e(\tilde{\pi}_S(y_k)) \leq \eta_k$. Therefore, the above inequalities imply

$$e(\bar{x}_k) \leq (1 - t_{\min})e(x_k) + \eta_k.$$

\square

Let us denote by \mathcal{D}_k^+ the subset of all possible outcomes of \mathcal{D}_k at iteration k for which the condition (3.7) is satisfied, i.e.,

$$\mathcal{D}_k^+ = \{\mathcal{D}_k \subset \mathcal{N} \mid f_{\mathcal{D}_k}(\bar{x}_k) \leq f_{\mathcal{D}_k}(x_k) - c\|s_k\|^2 + C\eta_k^2\}. \quad (4.5)$$

We denote the complementary subset of outcomes at iteration k by

$$\mathcal{D}_k^- = \{\mathcal{D}_k \subset \mathcal{N} \mid f_{\mathcal{D}_k}(\bar{x}_k) > f_{\mathcal{D}_k}(x_k) - c\|s_k\|^2 + C\eta_k^2\}. \quad (4.6)$$

Although the problem we consider is constrained and the algorithm is quite different from the one in [26, 28] with sampling that is not uniform, the following lemma, similar to the [28, Lemma 1] holds. Essentially, it says that either $N_k = N$ for k large enough or the condition (3.7) is satisfied infinitely many times. We state the proof for completeness.

Lemma 4.3. *Suppose that Assumption A1 holds. If $N_k < N$ for all $k \in \mathbb{N}$, then a.s. there exists $k_1 \in \mathbb{N}$ such that $\mathcal{D}_k^- = \emptyset$ for all $k \geq k_1$.*

Proof. Assume that $N_k < N$ for all $k \in \mathbb{N}$. Since the sample size sequence $\{N_k\}$ in IPAS Algorithm is non-decreasing there exists some $\bar{N} < N$ such that $N_k = \bar{N}$ for all k large enough. Now, let us assume that there is no $k_1 \in \mathbb{N}$ such that $\mathcal{D}_k^- = \emptyset$ for all $k \geq k_1$. Then there exists an infinite sub-sequence of iterations $K \subseteq \mathbb{N}$ such that $\mathcal{D}_k^- \neq \emptyset$ for all $k \in K$. Since \mathcal{D}_k is chosen with finitely many possible outcomes with the same distribution for each k , there exists $q > 0$ such that $\mathbb{P}(\mathcal{D}_k \in \mathcal{D}_k^-) \geq q$ for all $k \in K$. In fact, given (2.12) and the fact that $D_k \leq N - 1$ for each k , we can conclude that $q = (\min_{s \in \{1, 2, \dots, N\}} \{w_s\})^{N-1}$. So, we have

$$\mathbb{P}(\mathcal{D}_k \in \mathcal{D}_k^+, k \in K) \leq \prod_{k \in K} (1 - q) = 0.$$

Therefore we will almost surely encounter an iteration at which the sample size will be increased due to violation of the condition (3.7). This is a contradiction with the condition $N_k = \bar{N}$ for all k large enough and we conclude that the statement holds. \square

As already stated, depending on the problem, IPAS algorithm yields two possibilities - either we generate an infinite sequence $\{x_k\}$ such that $N_k < N$ for all k , or we end up with $N_k = N$ for k large enough. So, the convergence analysis will cover these two cases separately. Let us first consider the mini-batch case, i.e. $N_k < N$ for all $k \in \mathbb{N}$.

The following lemma quantifies the progress in term of the (true) objective function and exact projections in the mini-batch case.

Lemma 4.4. *Suppose that Assumption A1 holds. If $N_k < N$ for all $k \in \mathbb{N}$, then a.s.*

$$f(x_{k+1}) \leq f(x_k) - \frac{c}{2}\|d(x_k)\|^2 + C_L\eta_k^2$$

holds for all $k \geq k_1$ and some constant C_L , where k_1 is as in Lemma 4.3 and $d(x_k)$ is given by (2.7).

Proof. First, we prove that

$$f(\bar{x}_k) \leq f(x_k) - \frac{c}{2} \|d(x_k)\|^2 + C_L \eta_k^2$$

holds a.s. for all $k \geq k_1$, where k_1 is as in Lemma 4.3 and some constant C_L .

Notice that Lemma 4.3 implies that a.s. (3.7) holds for all possible realizations of \mathcal{D}_k and for all $k \geq k_1$. Thus, we conclude that a.s. for every $i = 1, 2, \dots, N$ and every $k \geq k_1$ we have

$$f_i(\bar{x}_k) \leq f_i(x_k) - c \|z_k^i\|^2 + C \eta_k^2, \quad (4.7)$$

where z_k^i denotes the direction obtained for $\mathcal{D}_k = \{i\}$ in Step 4 i.e.,

$$z_k^i = u_k^i - A^T \tilde{\lambda}_k(u_k^i) - x_k \quad (4.8)$$

where $u_k^i = x_k - \nabla f_i(x_k)$ and $\tilde{\lambda}_k(u_k^i)$ satisfies

$$\|AA^T \tilde{\lambda}_k(u_k^i) - Au_k^i + b\| \leq \eta_k. \quad (4.9)$$

Indeed, if there exists $i \in \mathcal{N}$ that violates (4.8), then there would exist at least one possible realization of \mathcal{D}_k (namely, $\mathcal{D}_k = \{i, i, \dots, i\}$) that violates (4.7) and thus we would have $\mathcal{D}_k^- \neq \emptyset$. Let us denote the residual by r_k^i , i.e., we have

$$r_k^i = AA^T \tilde{\lambda}_k(u_k^i) - Au_k^i + b, \quad \|r_k^i\| \leq \eta_k, \quad (4.10)$$

and

$$\tilde{\lambda}_k(u_k^i) = (AA^T)^{-1} r_k^i + (AA^T)^{-1} (Au_k^i - b). \quad (4.11)$$

Moreover,

$$z_k^i = u_k^i - A^T (AA^T)^{-1} r_k^i - A^T (AA^T)^{-1} (Au_k^i - b) - x_k. \quad (4.12)$$

Next, multiplying both sides of (4.7) with w_i satisfying (1.2) and summing over all $i \in \mathcal{N}$ we obtain that a.s. the following holds for all $k \geq k_1$

$$f(\bar{x}_k) \leq f(x_k) - c \sum_{i=1}^N w_i \|z_k^i\|^2 + C \eta_k^2. \quad (4.13)$$

Further, writing $x_k = \sum_{i=1}^N w_i x_k$ and using (2.3) in Lemma 2.1 with $y^i = x_k - \nabla f_i(x_k)$ we obtain that the exact projection direction related to the

original objective function can be represented as follows

$$\begin{aligned}
d(x_k) &= \pi_S(x_k - \nabla f(x_k)) - x_k \\
&= \pi_S\left(\sum_{i=1}^N w_i(x_k - \nabla f_i(x_k))\right) - \sum_{i=1}^N w_i x_k \\
&= \sum_{i=1}^N w_i(\pi_S(x_k - \nabla f_i(x_k)) - x_k) \\
&= \sum_{i=1}^N w_i(z_k^i + \pi_S(x_k - \nabla f_i(x_k)) - x_k - z_k^i) \\
&= \sum_{i=1}^N w_i(z_k^i + \pi_S(u_k^i) - x_k - z_k^i).
\end{aligned} \tag{4.14}$$

Further, using (1.2) and the convexity of $\|\cdot\|^2$, we obtain

$$\begin{aligned}
\|d(x_k)\|^2 &\leq \sum_{i=1}^N w_i \|z_k^i + \pi_S(u_k^i) - x_k - z_k^i\|^2 \\
&\leq 2 \sum_{i=1}^N w_i \|z_k^i\|^2 + 2 \sum_{i=1}^N w_i \|\pi_S(u_k^i) - x_k - z_k^i\|^2.
\end{aligned} \tag{4.15}$$

Now, let us estimate $\|\pi_S(u_k^i) - x_k - z_k^i\|^2$. According to (2.1) and (4.12) we obtain

$$\begin{aligned}
\pi_S(u_k^i) - x_k - z_k^i &= u_k^i - A^T(AA^T)^{-1}(Au_k^i - b) - x_k \\
&= (u_k^i - A^T(AA^T)^{-1}r_k^i - A^T(AA^T)^{-1}(Au_k^i - b) - x_k) \\
&= A^T(AA^T)^{-1}r_k^i.
\end{aligned} \tag{4.16}$$

Thus, for $C_A = \|A^T(AA^T)^{-1}\|$ we get

$$\|\pi_S(u_k^i) - x_k - z_k^i\|^2 \leq \|A^T(AA^T)^{-1}\|^2 \|r_k^i\|^2 \leq C_A^2 \eta_k^2. \tag{4.17}$$

Therefore, from (4.15) we obtain

$$\begin{aligned}
-\sum_{i=1}^N w_i \|z_k^i\|^2 &\leq -\frac{1}{2} \|d(x_k)\|^2 + \sum_{i=1}^N w_i \|\pi_S(u_k^i) - x_k - z_k^i\|^2 \\
&\leq -\frac{1}{2} \|d(x_k)\|^2 + C_A^2 \eta_k^2.
\end{aligned} \tag{4.18}$$

Now, (4.13) implies

$$f(\bar{x}_k) \leq f(x_k) - \frac{c}{2} \|d(x_k)\|^2 + cC_A^2 \eta_k^2 + C\eta_k^2 =: f(x_k) - \frac{c}{2} \|d(x_k)\|^2 + C_L \eta_k^2. \tag{4.19}$$

Next, we show that in the mini-batch scenario we accept the trial point \bar{x}_k for all $k \geq k_1$. Since we know that a.s. $\mathcal{D}_k^- = \emptyset$ for all $k \geq k_1$, i.e., (3.7) is satisfied, therefore, a.s., for all $k \geq k_1$ the trial point is accepted, i.e., $x_{k+1} = \bar{x}_k$, so the statement holds due to (4.19). \square

Now, let us denote by Ω all possible sample paths of the IPAS algorithm. Further, let us denote by $\mathcal{A} \subseteq \Omega$ all possible sample paths that yield mini-batch scenario considered in the previous proposition and by $\bar{\mathcal{A}} \subseteq \Omega$ all possible sample paths that reach the full sample size eventually, i.e., the complement of \mathcal{A} . We use the following notation for the corresponding conditional expectations

$$\mathbb{E}_{\mathcal{A}}(\cdot) := \mathbb{E}(\cdot|\mathcal{A}), \quad \mathbb{E}_{\bar{\mathcal{A}}}(\cdot) := \mathbb{E}(\cdot|\bar{\mathcal{A}}).$$

In order to prove the convergence results, we impose the following assumption similar to one in [28].

Assumption A 2. *There exists a constant $C_{\mathcal{A}}$ such that $\mathbb{E}_{\mathcal{A}}(|f(x_{k_1})|) \leq C_{\mathcal{A}}$, where k_1 is as in Lemma 4.3.*

The following result shows that $d(x_k)$ defined in (2.7) converges to zero a.s. in the mini-batch scenario.

Corollary 4.5. *Suppose that the assumptions of Lemma 4.4 hold together with Assumption A2. Then*

$$P(\lim_{k \rightarrow \infty} d(x_k) = 0 | \mathcal{A}) = 1$$

and every accumulation point of the sequence $\{x_k\}$ is a stationary point of problem (1.1) a. s.

Proof. Lemma 4.4 implies that a.s.

$$f(x_{k+1}) \leq f(x_k) - \frac{c}{2} \|d(x_k)\|^2 + C_L \eta_k^2$$

holds for all $k \geq k_1$. Applying the conditional expectation $\mathbb{E}_{\mathcal{A}}$, by the induction argument we obtain that for each j there holds

$$\mathbb{E}_{\mathcal{A}}(f(x_{k_1+j})) \leq \mathbb{E}_{\mathcal{A}}(f(x_{k_1})) - \frac{c}{2} \sum_{i=0}^{j-1} \mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2) + C_L \sum_{i=0}^{j-1} \eta_{k_1+i}^2.$$

Now, by using Assumptions A1, A2, and (3.1), letting j tend to infinity we get

$$f_{low} \leq C_{\mathcal{A}} - \frac{c}{2} \sum_{i=0}^{\infty} \mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2) + C_L \bar{\eta}$$

and we conclude that

$$\sum_{i=0}^{\infty} \mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2) < \infty.$$

Now, by the extended version of Markov's inequality we have that for any $\epsilon > 0$

$$\mathbb{P}(\|d(x_{k_1+i})\| \geq \epsilon | \mathcal{A}) \leq \frac{\mathbb{E}_{\mathcal{A}}(\|d(x_{k_1+i})\|^2)}{\epsilon^2}$$

which implies

$$\sum_{i=0}^{\infty} \mathbb{P}(\|d(x_{k_1+i})\| \geq \epsilon | \mathcal{A}) < \infty$$

and Borel-Cantelli Lemma [21] implies that $P(\lim_{i \rightarrow \infty} \|d(x_{k_1+i})\| = 0 | \mathcal{A}) = 1$, i.e.,

$$P(\lim_{i \rightarrow \infty} d(x_{k_1+i}) = 0 | \mathcal{A}) = 1. \quad (4.20)$$

Now, recall that e_k is the measure of infeasibility of x_k defined in (2.6), i.e., $e_k := \|Ax_k - b\|$. Lemma 4.2 implies that

$$e_{k+1} \leq (1 - t_{\min})e_k + \eta_k$$

for all $k \geq k_1$. Therefore, due to Lemma 2.4 we obtain $\lim_{k \rightarrow \infty} e_k = 0$. Let us denote by $\tilde{x} = \lim_{k \in K} x_k$ an arbitrary accumulation point of IPAS algorithm in the mini-batch scenario. Then, we have that $\|A\tilde{x} - b\| = \lim_{k \in K} e_k = 0$ and we conclude that \tilde{x} is feasible. Moreover, due to (4.20) and continuity of d , a.s.

$$d(\tilde{x}) = \lim_{k \in K} d(x_k) = 0$$

and we conclude that \tilde{x} is a.s. a stationary point of (1.1) according to Theorem 2.3. \square

\square

Now, we analyze the case where the full sample is reached for some $k_2 \in \mathbb{N}_0$ and therefore $N_k = N$ for $k \geq k_2$. We will distinguish between two types of iterations for $k \geq k_2$, successful and unsuccessful, represented by sets of indices K_{su} and K_{un} . An iteration x_k , $k \geq k_2$ is unsuccessful if the condition (3.4) does not hold, i.e., if the direction p_k is not sufficiently decreasing and hence $x_{k+1} = \tilde{\pi}_S(x_k)$ and $r(x_k) \leq \eta_k$. Otherwise, x_k is successful and the new iteration is determined by the line search rule (3.5) and $x_{k+1} = \bar{x}_k$.

The following Lemma states that the sequence $\{x_k\}_{k \geq k_2}$ converges to a feasible point and hence each accumulation point is feasible. We will rely on Lemmas 2.2 and 2.4 from Section 2.

Lemma 4.6. *Assume that Assumption A1 holds and let $\{x_k\}$ be an iterative sequence generated by IPAS algorithm such that $N_k = N$ for $k \geq k_2$. Then $\lim_{k \rightarrow \infty} e(x_k) = 0$ and each accumulation point is feasible.*

Proof. We will distinguish three different cases: 1) all iterations are successful for k large enough; 2) all iterations are unsuccessful for k large enough; and 3) we have an infinite sequence of successful and an infinite sequence of unsuccessful iterations.

Case 1. Assume, without loss of generality, that for all $k \geq k_2$ we have $k \in K_{su}$. In that case we have $x_{k+1} = \bar{x}_k$ and by Lemma 4.2

$$e(x_{k+1}) \leq (1 - t_{\min})e(x_k) + \eta_k,$$

so the statement holds by Lemma 2.4.

Case 2. Without loss of generality assume that $k \in K_{un}$ for all $k \geq k_2$. Then we have

$$x_{k+1} = \tilde{\pi}_S(x_k)$$

for all k and by Lemma 2.2

$$e(x_{k+1}) \leq \eta_k.$$

Given that $\lim_{k \rightarrow \infty} \eta_k = 0$ we obtain $\lim_{k \rightarrow \infty} e(x_k) = 0$.

Case 3. Assume now that both K_{su} and K_{un} are infinite. Let $k-1 \geq k_2$ and assume that $k-1 \in K_{un}$, $k, \dots, k+j-1 \in K_{su}$ and $k+j \in K_{un}$. Then we have

$$e(x_k) = e(\tilde{\pi}_S(x_{k-1})) \leq \eta_{k-1}$$

and for each $i = 1, \dots, j$, by Lemma 4.2 there holds

$$e(x_{k+i}) \leq \theta e(x_{k+i-1}) + \eta_{k+i-1},$$

with $\theta := 1 - t_{\min} \in (0, 1)$ and thus by the induction argument we get

$$e(x_{k+i}) \leq \theta^i \eta_{k-1} + \dots + \theta \eta_{k+i-2} + \eta_{k+i-1}.$$

Since $\{\eta_k\}$ is nonincreasing, for each $i = 1, \dots, j$ there holds

$$e(x_{k+i}) \leq \eta_{k-1} \sum_{t=0}^i \theta^t \leq \eta_{k-1} \sum_{t=0}^{\infty} \theta^t = \eta_{k-1} \frac{1}{1 - \theta} = \eta_{k-1} \frac{1}{t_{\min}}.$$

Thus, we can conclude that for each $k \geq k_2$ there holds

$$e(x_k) \leq \frac{1}{t_{\min}} \eta_{k_{un}},$$

where k_{un} represents the index of last unsuccessful iteration before the iteration k . Letting $k \rightarrow \infty$ we conclude that $\lim_{k \rightarrow \infty} e(x_k) = 0$ in this case as well.

Therefore, in all cases we have $e(x_k) \rightarrow 0$. If \tilde{x} is an arbitrary accumulation point of $\{x_k\}$ then clearly $e(\tilde{x}) = 0$ and hence \tilde{x} is feasible. \square

Lemma 4.7. Assume that Assumption A1 holds and let $\{x_k\}$ be an iterative sequence generated by IPAS algorithm such that $N_k = N$ for $k \geq k_2$ and assume that $\{x_k\}_{k \geq k_2}$ is bounded. If there exists $k_3 \geq k_2$ such that $k \in K_{su}$ for all $k \geq k_3$ then each accumulation point of $\{x_k\}$ is a stationary point of (1.1).

Proof. For each $k \geq k_3$ we have that x_{k+1} is successful and hence we have

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) + c_1 t_k (\nabla f(x_k))^T p_k + \eta_k^2 \\ &\leq f(x_k) - c_1 c t_k \|p_k\|^2 + \eta_k^2 \\ &\leq f(x_k) - c_1 c t_{\min} \|p_k\|^2 + \eta_k^2. \end{aligned}$$

Given that f is continuous and $\{x_k\}_{k \geq k_2}$ is assumed to be bounded, there must exist a constant f_{up} such that $f(x_{k_3}) \leq f_{up}$. Moreover, $f(x_k) \geq f_{low}$ for each k and using the standard arguments we get $\sum_{k=k_3}^{\infty} \|p_k\|^2 < \infty$ and $\lim_{k \rightarrow \infty} \|p_k\| = 0$. Let us denote, as before, the exact projected gradient direction at x_k by $d(x_k)$. Then we have

$$\|d(x_k) - p_k\| \leq \|A^T(AA^T)^{-1}\| \|r(x_k)\| \leq C_A \eta_k$$

and

$$\lim_{k \rightarrow \infty} \|d(x_k)\| \leq \lim_{k \rightarrow \infty} \|p_k\| + \lim_{k \rightarrow \infty} \|d(x_k) - p_k\| \leq \lim_{k \rightarrow \infty} \|p_k\| + C_A \eta_k = 0.$$

So, for arbitrary accumulation point $\tilde{x} = \lim_{k \in K} x_k$ we have $\|d(\tilde{x})\| = \lim_{k \in K} \|d(x_k)\| = 0$. By Lemma 4.6 \tilde{x} is feasible so the statement follows by Lemma 2.3. \square

The following inequality will be used for further analysis.

Lemma 4.8. For each $k \geq k_2$ there holds

$$\begin{aligned} \nabla^T f(x_k) p_k &\leq -\|p_k\|^2 - p_k^T (\pi_S(y_k) - \tilde{\pi}_S(y_k)) \\ &\quad + (p_k + \nabla f(x_k) + \pi_S(y_k) - \tilde{\pi}_S(y_k))^T (\pi_S(x_k) - x_k + \tilde{\pi}_S(y_k) - \pi_S(y_k)). \end{aligned}$$

Proof. For any $y \in \mathbb{R}^n$ we have that $\pi_S(y)$ is a solution of convex problem

$$\min_{Az=b} \frac{1}{2} \|z - y\|^2.$$

Denoting $h(z) = \frac{1}{2} \|z - y\|^2$ we can state the KKT conditions as follows. A point z^* is a solution of $\min_{Az=b} h(z)$ iff $(\nabla h(z^*))^T (z - z^*) \geq 0$ for all z such that $Az = b$. The gradient of h is given by $\nabla h(z) = z - y$ and hence the optimality condition yields

$$(\pi_S(y) - y)^T (z - \pi_S(y)) \geq 0, \text{ for all feasible } z. \quad (4.21)$$

Let us now consider $k \geq k_2$ and take $y = y_k = x_k - \nabla f(x_k)$ and $z = \pi_S(x_k)$ in (4.21). We get

$$(\pi_S(y_k) - y_k \pm \tilde{\pi}_S(y_k))^T (\pi_S(x_k) \pm x_k - \pi_S(y_k) \pm \tilde{\pi}_S(y_k)) \geq 0.$$

Recall that $p_k = \tilde{\pi}_S(y_k) - x_k$. The previous inequality actually states

$$(p_k + \nabla f(x_k) + \pi_S(y_k) - \tilde{\pi}_S(y_k))^T (-p_k + \tilde{\pi}_S(y_k) - \pi_S(y_k) + \pi_S(x_k) - x_k) \geq 0$$

and the statement follows. \square

Lemma 4.9. *Assume that Assumption A1 holds and let $\{x_k\}$ be an iterative sequence generated by IPAS algorithm such that $N_k = N$ for $k \geq k_2$ and assume that $\{x_k\}_{k \geq k_2}$ is bounded. If the sequence of unsuccessful iterations $\{x_k\}_{k \in K_{un}}$ is infinite then there exists an accumulation point \tilde{x} of $\{x_k\}$ such that \tilde{x} is stationary point for the problem (1.1).*

Proof. First of all let us notice that boundedness of $\{x_k\}$ together with A1 implies that $\{p_k\}$ is bounded as well. Namely, for $y_k = x_k - \nabla f(x_k)$ we have

$$\begin{aligned} p_k &= \tilde{\pi}_S(y_k) - x_k = y_k - A^T \tilde{\lambda}(y_k) - x_k \\ &= -\nabla f(x_k) - A^T ((AA^T)^{-1} (Ay_k - b) + (AA^T)^{-1} r(y_k)). \end{aligned}$$

Now, for $C_A = \|A^T (AA^T)^{-1}\|$, having $\|x_k\| \leq C_x$, $\|\nabla f(x_k)\| \leq C_g$, and therefore $\|y_k\| \leq \|x_k\| + \|\nabla f(x_k)\| \leq C_x + C_g$, with $\|r(y_k)\| \leq \eta_k$ by conditions of IPAS algorithm we get

$$\|p_k\| \leq C_g + C_A (\|A\| (C_x + C_g) + \|b\| + \eta_k) := C_p. \quad (4.22)$$

Let us assume that there exists $\varepsilon > 0$ such that $\|p_k\| \geq \varepsilon > 0$ for all $k \geq k_2$. We will consider unsuccessful iterations, i.e. $k \geq k_2, k \in K_{un}$. For these iterations we have $(\nabla f(x_k))^T d_k > -c\|p_k\|^2$. Lemma 4.8 implies

$$\begin{aligned} 0 &< -(1-c)\|p_k\|^2 - p_k^T (\pi_S(y_k) - \tilde{\pi}_S(y_k)) \\ &+ (p_k + \nabla f(x_k) + \pi_S(y_k) - \tilde{\pi}_S(y_k))^T (\tilde{\pi}_S(y_k) - \pi_S(y_k) + \pi_S(x_k) - x_k) \\ &\leq -(1-c)\varepsilon^2 + \|p_k\| \|\pi_S(y_k) - \tilde{\pi}_S(y_k)\| \\ &+ (\|p_k\| + \|\nabla f(x_k)\| + \|\pi_S(y_k) - \tilde{\pi}_S(y_k)\|) \\ &\quad \cdot (\|\tilde{\pi}_S(y_k) - \pi_S(y_k)\| + \|\pi_S(x_k) - x_k\|). \end{aligned} \quad (4.23)$$

As

$$\pi_S(y_k) - \tilde{\pi}_S(y_k) = A^T (AA^T)^{-1} r(y_k),$$

we have

$$\|\tilde{\pi}_S(y_k) - \pi_S(y_k)\| \leq C_A \eta_k. \quad (4.24)$$

Furthermore,

$$\|\pi_S(x_k) - x_k\| = \|x_k - A^T \lambda(x_k) - x_k\| = \|A^T(AA^T)^{-1}(Ax_k - b)\| \leq C_A e(x_k), \quad (4.25)$$

and $e(x_k) \rightarrow 0$ by Lemma 4.6. Putting together (4.22), (4.23)-(4.25) we get

$$0 < -(1 - c)\varepsilon^2 + C_p C_A \eta_k + (C_p + C_g + C_A \eta_k)(C_A \eta_k + C_A e(x_k)).$$

Taking the limit for $k \in K_{un}, k \rightarrow \infty$ in the above inequality we get

$$0 \leq -(1 - c)\varepsilon^2$$

which can not be true. Thus there is no $\varepsilon > 0$ such that $\|p_k\| \geq \varepsilon > 0$ for all $k \geq k_2$. Therefore there exists an infinite $K \subset \mathbb{N}$ such that $\lim_{k \in K} p_k = 0$ and therefore $\lim_{k \in K} d(x_k) = 0$, as in the proof of Lemma 4.7. As $\{x_k\}_{k \geq k_2}$ is bounded then there exists $\tilde{K} \subset K$ such that $\lim_{k \in \tilde{K}} x_k = \tilde{x}$ and thus $d(\tilde{x}) = 0$. By Lemma 4.6 $\tilde{x} \in S$ and hence the statement follows by Lemma 2.3.

Theorem 4.10. *Let Assumption A1 holds and assume that $\{x_k\}$ generated by IPAS algorithm is bounded. Then a.s. there exists an accumulation point of $\{x_k\}$ which is a stationary point of (1.1).*

Proof. If $N_k < N$ for all k we have that the sequence $\{x_k\}$ is bounded so Assumption A2 holds and there exists at least one accumulation point of $\{x_k\}$. That point is stationary by Corollary 4.5. In the case of $N_k = N$ for k large enough the statement follows by Lemma 4.7 and Lemma 4.9. \square

5 Numerical results

In this section we will demonstrate the efficiency and effectiveness of the IPAS algorithm through a series of experiments. The evaluation is conducted in two parts, each focusing on different problems. In the first part, we apply IPAS to some of equality constrained problems from CUTEst collection [16], a standard set of optimization problems. This part of the evaluation explores how different parameters affect the performance of IPAS algorithm. By varying parameters, we gain some insight into the algorithms adaptability and robustness under diverse settings. The results provide our view on parameter tuning and how the balance between the inexactness of projections and subsampling size influences the algorithm's behavior. In the second part, we apply the IPAS algorithm to solve an equality constrained logistic regression problem. Logistic regression is a widely used ML model for binary classification, and by adding equality constraints the complexity of the problem rises. To evaluate IPAS, we compare its performance with two other notable methods, namely the Stochastic Sequential Quadratic Programming method (Stochastic SQP) [6], and Fully Stochastic Trust-region

Sequential Quadratic Programming method (TrStochasticSQP) [15]. Both of these algorithms are designed to deal with equality constrained optimization problems, making them suitable benchmarks for comparison. For all the tests, we use the additional sample size equal to 1, i.e., $|\mathcal{D}_k| = 1$.

5.1 CUTEst

We implemented a few optimization problems from the CUTEst [16] collection that include equality constraints to evaluate the performance of IPAS algorithm. To adapt these CUTEst problems to IPAS, we transformed them into finite sum problems. The transformation involved introducing the Gaussian noise to simulate the stochastic nature commonly encountered in large-scale optimization task. This is done by multiplying the noise with the quadratic term in order to preserve the randomness in the derivatives of the objective function, thereby mimicking the behavior of stochastic problem while maintaining the structure of the problem. The transformed problems have the following form

$$\min_{Ax=b} f_N(x) := f(x) + \sum_{i=1}^N \varepsilon_i^2 \|x\|^2, \quad (5.1)$$

where $f(x)$ is the objective of the CUTEst problem, and ε_i is drawn from the $\mathcal{N}(0, \sigma^2)$ distribution. The noise is squared to preserve lower boundedness of the function. The gradient is then calculated in the following way

$$\nabla f_N(x) = \nabla f(x) + \sum_{i=1}^N 2\varepsilon_i^2 x. \quad (5.2)$$

We control the inexactness of projections by setting the linear system solver tolerance to $\eta_k = \frac{1}{k^s}$, where k is the number of iteration, and $s > 0.5$ for the convergence of the sum (3.1). In Step 5 of the algorithm, in the case the trial point is not accepted, we choose $N_{k+1} \in \{N_k + 1, \dots, N\}$, i.e., we increase the subsample size by $dN_k \in \{1, \dots, N - N_k\}$. Finding the balance between the inexactness of projections and the subsampling updates turned out to be one of the key issues influencing the behavior of the algorithm.

To evaluate the performance, we observe the norm of the exact projected direction d_k (2.7). The true projected direction is calculated for the sake of plots only, and it serves as a measure of distance to the stationary point. The computational cost is modeled by the total number of scalar products. When calculating the number of scalar products, at each iteration we have also taken into account the cost of making inexact projections. We used Conjugate Gradient method to solve the linear systems inexactly at each iteration. Hence, every time we needed to use the solver, we accounted $(m+4) \cdot \text{iter}$ number of scalar products, where, iter is the number of iterations

Conjugate Gradient method made to achieve the tolerance η_k . Function evaluations also contribute to number of scalar products depending on the subsampling size.

Results for HUESTIS and DTOC1L CUTEst problems [17] are showcased. The dimension of the HUESTIS problem is $n = 10, m = 2$, whereas the dimension of the DTOC1L is $n = 58, m = 36$. For both problems the total number of elements in the sum is $N = 10000$. For each problem, we conducted 10 independent simulations, systematically varying standard deviation σ for the Gaussian noise, tolerance parameter s and subsample size increments dN_k . The number of iterations was fixed $k_{max} = 500$ for all simulations. In order to emphasize the importance of choosing suitable parameters for the optimization process, we present our results in two different ways. The Box plots capture the variability in performance across different setups. Visualization provides insights into parameter identification that lead to optimal performance and computational efficiency. We show the resulting Box plots regarding the stationarity measure $\|d(x_k)\|$ and computational costs, i.e., budgets needed to achieve the relevant vicinity of the solution.

In Figure 1, for $\sigma = 3$ we can see that choosing $s = 3$ and $dN_k = 100$, i.e., pushing η_k faster to zero and making larger subsample size increments, the resulting point is the closest to the stationary point compared to the rest of the configurations. The aforementioned configuration requires sig-

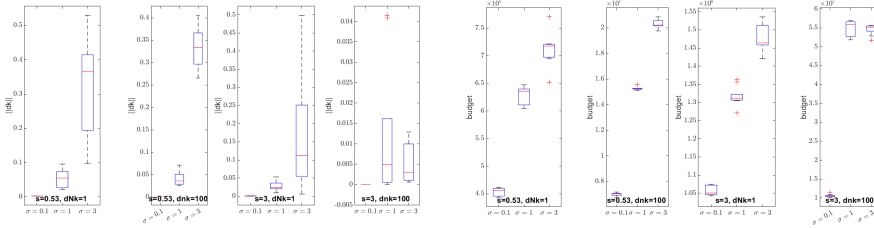


Figure 1: DTOC1L problem. Different configurations of IPAS algorithm regarding inexact projections and sample size increment. Box plot of optimality measure $\|d_k\|$ and scalar products (budget) based on 10 simulations for different levels of variance (σ). The remaining parameters: $c = 10^{-4}, c_1 = 10^{-2}, \beta = 0.1, C = 10^{-2}, k_{max} = 500$.

nificantly more computations, as seen in the same Figure 1, whereas other parameter configurations are magnitude of orders cheaper. The choice of the parameters is highly dependent on the dimension of the problem, i.e. choosing the right balance between the inexactness of projections and the accuracy of the approximations is problem dependent.

The following plots in Figure 2 show the mean of $\|d_k\|$ and its confidence interval for 10 simulations in terms of the average cost per iteration, for $\sigma \in \{0.1, 1, 3\}$. In each subplot four parameter configuration are compared.

The algorithm behaves differently depending on the configuration, however in this example, $s = 3$ and $dN_k = 100$ gave us the closest point to stationarity in 10^5 scalar products.

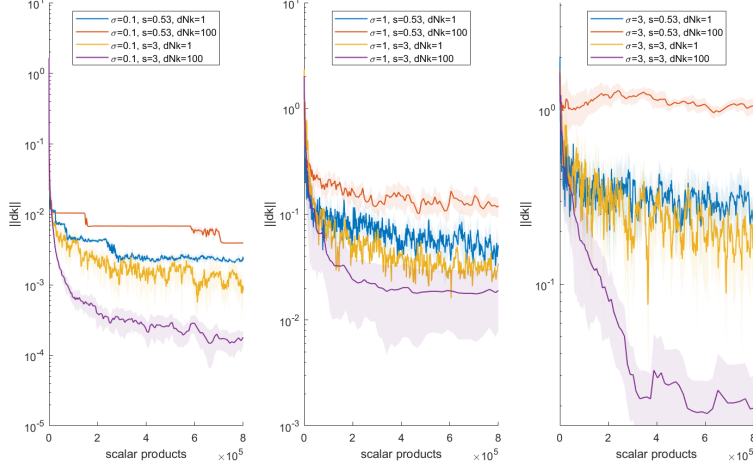


Figure 2: DTOC1L problem. Different configurations of IPAS algorithm regarding inexact projections and sample size increment. Mean optimality measure $\|d_k\|$ with the confidence interval based on 10 simulations presented in log scale. Plots correspond to different levels of variance: left $\sigma = 0.1$; middle $\sigma = 1$; right $\sigma = 3$. The remaining parameters: $c = 10^{-4}$, $c_1 = 10^{-2}$, $\beta = 0.1$, $C = 10^{-2}$, $k_{max} = 500$.

For HUESTIS problem, in Figure 3 we see that choosing low increments for subsample size yields highly efficient results for larger σ . The cost for such computation is two order of magnitude less than the computation using larger increments. The configuration comparison plots of mean $\|d_k\|$ are

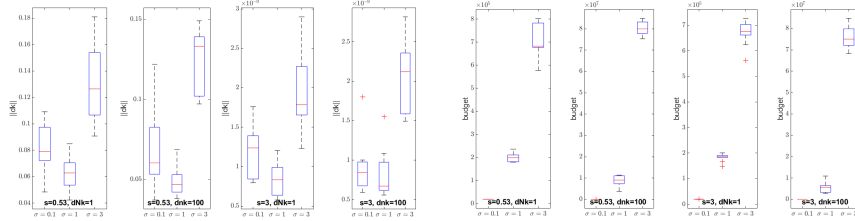


Figure 3: HUESTIS problem. Different configurations of IPAS algorithm regarding inexact projections and sample size increment. Box plot of optimality measure $\|d_k\|$ and scalar products (budget) based on 10 simulations for different levels of variance (σ). The remaining parameters: $c = 10^{-4}$, $c_1 = 10^{-4}$, $\beta = 0.7$, $C = 1$, $k_{max} = 500$.

also shown for the HUESTIS problem (Figure 4). Considering this problem, $s = 3$ and $dN_k = 1$ yielded the best result.

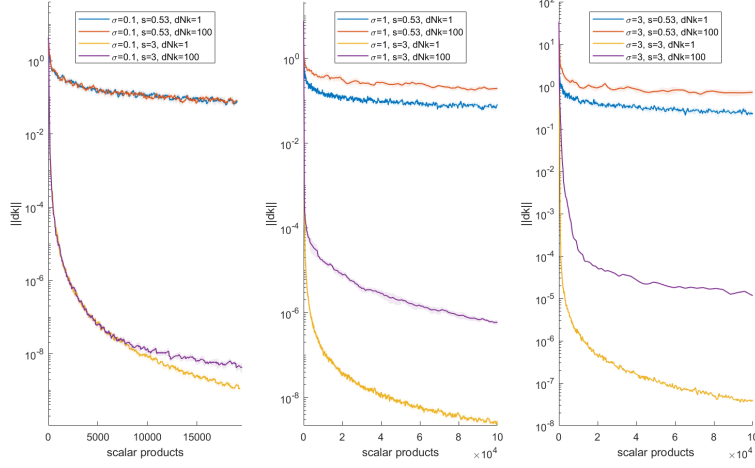


Figure 4: HUESTIS problem. Different configurations of IPAS algorithm regarding inexact projections and sample size increment. Mean optimality measure $\|d_k\|$ with the confidence interval based on 10 simulations presented in log scale. Plots correspond to different levels of variance: left $\sigma = 0.1$; middle $\sigma = 1$; right $\sigma = 3$. The remaining parameters: $c = 10^{-4}$, $c_1 = 10^{-4}$, $\beta = 0.7$, $C = 1$, $k_{max} = 500$.

5.2 Equality constrained logistic regression

We consider the following finite sum problem

$$\min_{Ax=b} f(x) := \frac{1}{N} \sum_{i=1}^N \log(1 + e^{(-y_i(x^T z_i))}), \quad (5.3)$$

where N is the number of samples, $z_i \in \mathbb{R}^n$ are sample attributes, $y_i \in \{-1, 1\}$ respective labels, and $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ define equality constraint system. The algorithm has been implemented and evaluated on several datasets from LIBSVM repository [32]. In this discussion, we present results for Mushroom, MNIST and Diabetes dataset [31]. The performance analysis has been done by comparing algorithms in terms of stationarity measure $\|d(x_k)\|$ plotted against computational cost modeled by the number of scalar products needed to reach iteration k . The equality constraints are created in a way that $A \in \mathbb{R}^{m,n}$ is a full rank matrix. Number of rows of A , i.e., the number of equality constraints, is set to $m = 0.5n$. This way we create robust problems, with number of equalities proportional to the dimension of the problem. As in the CUTest experiment, we set linear equation solver tolerance to $\eta_k = \frac{1}{k^s}$, $s > 0.5$, in accordance with the convergence requirements outlined in (3.1).

The following Figures 5, 6 and 7 show the comparison of IPAS with StochasticSQP [6] and TrStochasticSQP [15], all applied to solve the prob-

lem (5.3) using Mushroom, Diabetes and MNIST dataset respectively. The plots on the left show how the norm of the true projected direction $d(x_k)$ (2.7) changes in terms of the number of scalar products. The plots on the right show how the subsample size changes for the first 1000 iterations. Notice that some of the configurations do not reach the full sample size for the length of the simulation.

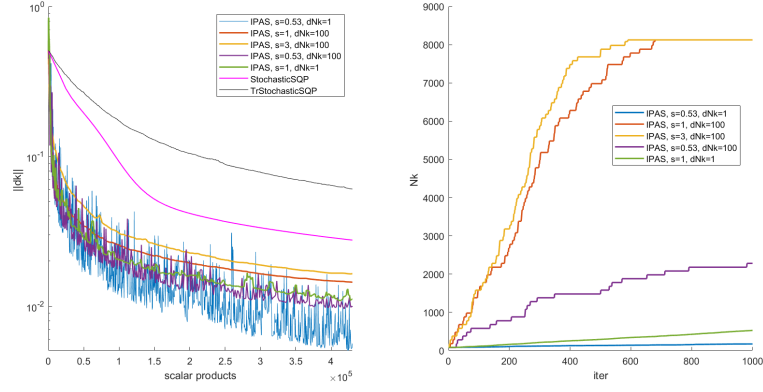


Figure 5: Mushroom dataset, $N = 8124, n = 112$. Stationarity measure against computational costs (left) and the sample size behavior (right). Parameters: $x_0 = A^T((AA^T)^{-1}b)$, $N_0 = 0.01N$, $c = 10^{-4}$, $c_1 = 10^{-4}$, $\beta = 0.7$, $C = 1$, $k_{max} = 1000$.

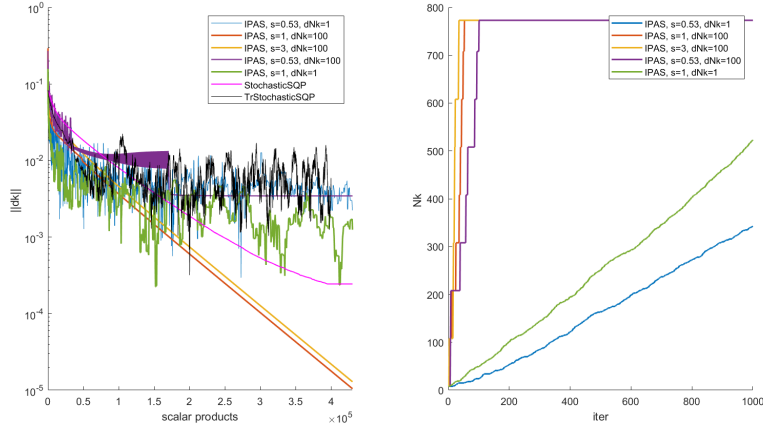


Figure 6: Diabetes dataset, $N = 773, n = 8$. Stationarity measure against computational costs (left) and the sample size behavior (right). Parameters: $x_0 = A^T((AA^T)^{-1}b)$, $N_0 = 0.01N$, $c = 10^{-4}$, $c_1 = 10^{-4}$, $\beta = 0.1$, $C = 10^{-4}$, $k_{max} = 1000$.

We tested IPAS algorithm with different parameter configurations. When applied to Mushrooms dataset (Figure 5), all the IPAS variants outper-

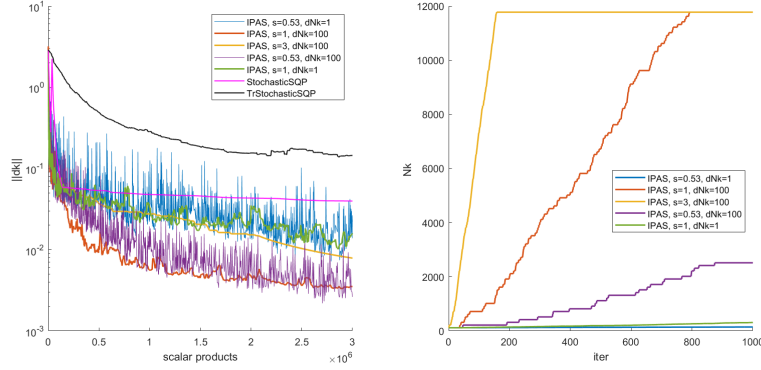


Figure 7: MNIST dataset, $N = 11774, n = 780$. Stationarity measure against computational costs (left) and the sample size behavior (right). Parameters: $x_0 = A^T((AA^T)^{-1}b)$, $N_0 = 0.01N$, $c = 10^{-4}$, $c_1 = 10^{-4}$, $\beta = 0.7$, $C = 1$, $k_{max} = 1000$.

formed their state-of-the-art counterparts. Notice that higher increase rate $dN_k = 100$ yields smoother behavior, especially when combined with lower inexactness such as the one governed by $s = 1$ or $s = 3$. However the best stationarity was reached with IPAS $s = 0.53, dN_k = 1$, although the presence of noise is evident through oscillations. Notice that the ad-hoc choice of $s = 1, dN_k = 1$ also yields satisfactory results. For MNIST dataset (Figure 7) the situation is similar, with a difference that the best performing combination seems to be $s = 1, dN_k = 100$. Diabetes dataset (Figure 6) is smaller both in sample size (N) and number of variables (n). The best performing algorithm seems to be IPAS $s = 1, dN_k = 100$, together with IPAS $s = 3, dN_k = 100$. Both of these variants reach the full sample quickly. The choice of IPAS $s = 1, dN_k = 1$ seems to work well at the beginning of the optimization process, but exhibits erratic behavior caused by the noise coming from smaller sample sizes. Since the problem is relatively small as well as the number of constraints, it is likely that the savings of IPAS were not big enough to show the advantages of all the configurations with respect to StochasticSQP method.

6 Conclusions

The proposed algorithm represents a novel approach for solving weighted sum problems with possibly large number of linear equality constraints. It adapts additional sampling approach originally constructed for finite sum unconstrained problems to more general problems of the form (1.1). This yields adaptive sample size algorithm suitable for the considered constrained framework. Moreover, inexact projections are allowed, but controlled by a common parameter that also plays role in sample size update. Allowing inex-

act projections shows to be very important in terms of computational costs, especially when the number of constraints is large. The almost sure convergence of the proposed method is proved under a set of standard assumptions for the stochastic framework, without the convexity assumption. Preliminary numerical results on both academic and real-world data show that IPAS is competitive with the relevant state-of-the-art methods in this field. Possible future work may include fine-tuning for some special subclasses of the considered problems, analysis of nonlinear and inequality constraints, as well as nondifferential objective functions.

7 Funding

This research was supported by the Science Fund of the Republic of Serbia, GRANT No 7359, Project title - LASCADO.

References

- [1] S. BELLAVIA, T. BIANCONCINI, N. KREJIĆ, B. MORINI, Subsampled first-order optimization methods with applications in imaging, *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging*, (2021).
- [2] S. BELLAVIA, N. KREJIĆ, N. KRKLEC JERINKIĆ, Subsampled Inexact Newton methods for minimizing large sums of convex function, *IMA Journal of Numerical Analysis* 40(4) (2020), pp. 2309-2341, <https://doi.org/10.1093/imanum/drz027>.
- [3] S. BELLAVIA, N. KREJIĆ, N. KRKLEC JERINKIĆ, M. RAYDAN, SLiSeS: Subsampled Line Search Spectral Gradient Method for Finite Sums, *Optimization Methods and Software*, (2024), pp. 1–26, <https://doi.org/10.1080/10556788.2024.2426620>
- [4] S. BELLAVIA, N. KREJIĆ, B. MORINI, S. REBEGOLDI, A stochastic first-order trust-region method with inexact restoration for finite-sum minimization, *Computational Optimization and Applications* 84(1), pp. 53–84 (2023), <https://doi.org/10.1007/s10589-022-00430-7>
- [5] S. BELLAVIA, N. KRKLEC JERINKIĆ, G. MALASPINA, Subsampled nonmonotone spectral gradient methods, *Communications in Applied and Industrial Mathematics* 11(1) (2020), pp. 19-34, DOI: 10.2478/caim-2020-0002.
- [6] A.S. BERAHAS, F. E. CURTIS, D. ROBINSON, B. ZHOU, Sequential Quadratic Optimization for Nonlinear Equality Constrained Stochastic Optimization, *SIAM Journal on Optimization* 31 (2021), pp. 1352-1379
- [7] A.S. BERAHAS, M. XIE, B. ZHOU, A sequential quadratic programming method with high probability complexity bounds for nonlinear equality constrained stochastic optimization, *arXiv preprint arXiv:2301.00477* (2024), <https://doi.org/10.48550/arXiv.2301.00477>
- [8] E.G. BIRGIN, J.M. MARTÍNEZ, M. RAYDAN, Nonmonotone Spectral Projected Gradients on Convex Sets, *SIAM Journal on Optimization* 10 (2000) pp. 1196-1211, <https://doi.org/10.1137/S1052623497330963>.
- [9] J. BLANCHET, C. CARTIS, M. MENICKELLY, K. SCHEINBERG, Convergence rate analysis of a stochastic trust-region method via supermartingales, *INFORMS journal on optimization* 1(2) (2019), pp. 92–119, <https://doi.org/10.1287/ijoo.2019.0016>

- [10] R. BOLLAPRAGADA, J. NOCEDAL, D. MUDIGERE, H.J. SHI, P.T.P. TANG, A progressive batching L-BFGS method for machine learning, *In: International Conference on Machine Learning (2018)*, pp. 620–629.
- [11] J.J. BRUST, R.F. MARCIA, C.G. PETRA, M.A. SAUNDERS, Large-scale Optimization with Linear Equality Constraints using Reduced Compact Representation, *arXiv:2101.11048 (2021)*, <https://doi.org/10.48550/arXiv.2101.11048>
- [12] R. CHEN, M. MENICKELLY, K. SCHEINBERG, Stochastic optimization using a trust region method and random models, *Mathematical Programming* 169(2) (2018), pp. 447–487, <https://doi.org/10.1007/s10107-017-1141-8>
- [13] F.E. CURTIS, D.P. ROBINSON, B. ZHOU, A stochastic inexact sequential quadratic optimization algorithm for nonlinear equality-constrained optimization, *INFORMS Journal on Optimization* 6 (3-4)(2024)
- [14] D. DI SERAFINO, N. KREJIĆ, N. KRKLEC JERINKIĆ, M. VIOLA, LSOS: Line-search Second-Order Stochastic optimization methods for nonconvex finite sums, *arXiv preprint arXiv:2007.15966 (2020)*.
- [15] Y.FANG, S. NA, M. MAHONEY, M. KOLAR, Fully Stochastic Trust-Region Sequential Quadratic Programming for Equality-Constrained Optimization Problems *SIAM Journal on Optimization* 34 (2024), pp. 2007 - 2037
- [16] N. I. M. GOULD, D. ORBAN, AND P. L. TOINT. CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization. *Computational Optimization and Applications* 60(3) (2014), pp. 545-557.
- [17] S. GRATTON, PH.L.TOINT S2MPJ and CUTEst optimization problems for Matlab, Python and Julia *arXiv:2407.07812*
- [18] L. GRIPPO, F. LAMPARIELLO, S. LUCIDI, A nonmonotone line search technique for Newton’s method, *SIAM Journal on Numerical Analysis* 23(4) (1986), pp. 707-716, <https://doi.org/10.1137/0723046>.
- [19] T. HASTIE, R. TIBSHIRANI, J. FRIEDMAN, Elements of Statistical Learning, *Springer*, 2009.
- [20] A. N. IUSEM, A. JOFRÉ, R. I. OLIVEIRA, P. THOMPSON, Variance-based extragradient methods with line search for stochastic variational inequalities, *SIAM Journal on Optimization* 29(1) (2019), pp. 175–206, <https://doi.org/10.1137/17M1144799>.

- [21] A. KLENKE, Probability Theory: A Comprehensive Course, third edition, 2020 *Springer Nature link*, eBook ISBN 978-3-030-56402-5, <https://link.springer.com/book/10.1007/978-3-030-56402-5>
- [22] N. KREJIĆ, N. KRKLEC JERINKIĆ, Nonmonotone line search methods with variable sample size, *Numer. Algorithms* 68(4) (2015), pp. 711-739, <https://doi.org/10.1007/s11075-014-9869-1>.
- [23] N. KREJIĆ, N. KRKLEC JERINKIĆ, Spectral projected gradient method for stochastic optimization, *Journal of Global Optimization* 73 (2018), pp. 59–81, <https://doi.org/10.1007/s10898-018-0682-6>.
- [24] N. KREJIĆ, N. KRKLEC JERINKIĆ, T. OSTOJIĆ, An inexact restoration-nonsmooth algorithm with variable accuracy for stochastic nonsmooth convex optimization problems in machine learning and stochastic linear complementarity problems, *Journal of Computational and Applied Mathematics* (2022), pp. <https://doi.org/10.1016/j.cam.2022.114943>.
- [25] N. KREJIĆ, N. KRKLEC JERINKIĆ, T. OSTOJIĆ, Spectral Projected Subgradient Method for Nonsmooth Convex Optimization Problems, *Numerical Algorithms*, (2022), pp. DOI : 10.1007/s11075-022-01419-3.
- [26] N. KREJIĆ, N. KRKLEC JERINKIĆ, MARTINEZ, A., YOUSEFI, M., A non-monotone trust-region method with noisy oracles and additional sampling, *Computational Optimization and Applications* (2024) 89, pp. 247–278, <https://doi.org/10.1007/s10589-024-00580-w>
- [27] N. KREJIĆ, Z. LUŽANIN, Z. OVCIN, I. STOJKOVSKA, Descent direction method with line search for unconstrained optimization in noisy environment, *Optimization Methods and Software* 30(6) (2015), pp. 1164-1184, <https://doi.org/10.1080/10556788.2015.1025403>.
- [28] N. KRKLEC JERINKIĆ, V. RUGGIERO, I. TROMBINI, Spectral Stochastic Gradient Method with Additional Sampling for Finite and Infinite Sums, *Computational Optimization and Applications*, (2025), <https://doi.org/10.1007/s10589-025-00664-1>
- [29] G. LAN, T. LI, Y. XU, Projected gradient methods for nonconvex and stochastic optimization: new complexities and auto-conditioned stepsizes, *arXiv:2412.14291* (2024), <https://doi.org/10.48550/arXiv.2412.14291>
- [30] D.H. LI, M. FUKUSHIMA, A derivative-free line search and global convergence of Broyden-like method for nonlinear equations, *Opt. Methods Software* 13 (2000), pp. 181-201, DOI:10.1080/10556780008805782.

- [31] M. LICHMAN, UCI machine learning repository (2013), <https://archive.ics.uci.edu/ml/index.php>
- [32] LIBSVM Data: Classification (Binary Class), <https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html>
- [33] M. LORETO, A. CREMA, Convergence analysis for the modified spectral projected subgradient method, *Optimization Letters* 9(5) (2015), pp. 915-929, <https://doi.org/10.1007/s11590-014-0792-0>.
- [34] F. OZTOPRAK, R. BYRD, J. NOCEDAL, Constrained optimization in the presence of noise, *SIAM Journal on Optimization* 33(3) (2023), pp. 2118–2136
- [35] H. ROBBINS, S. MONRO, A stochastic approximation method, *The annals of mathematical statistics*, pp. 400-407, (1951).
- [36] J. SCOTT, M. TUMA, Solving large linear least squares problems with linear equality constraints, *Bit Numer Math* 62 (2022), pp. 1765–1787
- [37] S. SUN, J. NOCEDAL, A Trust-Region Algorithm for Noisy Equality Constrained Optimization, *arXiv:2411.02665* (2024), <https://doi.org/10.48550/arXiv.2411.02665>
- [38] S. SUNDHAR RAM, A. NEDIĆ, V.V. VEERAVALLI, Distributed Stochastic Subgradient Projection Algorithms for Convex Optimization, *J. Optim. Theory Appl.* (2010) 147: 516–545