Sampling Theory, Signal Processing, and Data Analysis (2022) 20:4
https://doi.org/10.1007/s43670-022-00021-x

ORIGINAL ARTICLE

Inertial stochastic PALM and applications in machine learning

Johannes Hertrich[1] · **Gabriele Steidl**[1]

Received: 27 January 2022 / Accepted: 18 March 2022 / Published online: 22 April 2022

Abstract

Inertial algorithms for minimizing nonsmooth and nonconvex functions as the inertial proximal alternating linearized minimization algorithm (iPALM) have demonstrated their superiority with respect to computation time over their non inertial variants. In many problems in imaging and machine learning, the objective functions have a special form involving huge data which encourage the application of stochastic algorithms. While algorithms based on stochastic gradient descent are still used in the majority of applications, recently also stochastic algorithms for minimizing nonsmooth and nonconvex functions were proposed. In this paper, we derive an inertial variant of a stochastic PALM algorithm with variance-reduced gradient estimator, called iSPALM, and prove linear convergence of the algorithm under certain assumptions. Our inertial approach can be seen as generalization of momentum methods widely used to speed up and stabilize optimization algorithms, in particular in machine learning, to nonsmooth problems. Numerical experiments for learning the weights of a so-called proximal neural network and the parameters of Student-t mixture models show that our new algorithm outperforms both stochastic PALM and its deterministic counterparts.

Keywords Stochastic PALM · Proximity operator · Variance reduction · Non-convex optimization · Stochastic optimization

Mathematics Subject Classification 65K10 · 65C20 · 60H25 · 49N15 · 60H99

Communicated by Gerlind Plonka.

✉ Johannes Hertrich
j.hertrich@math.tu-berlin.de

Gabriele Steidl
steidl@math.tu-berlin.de

[1] Institute of Mathematics, TU Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany

1 Introduction

Recently, duality concepts were successfully applied for minimizing nonsmooth and nonconvex functions appearing in certain applications in image and data processing. A frequently applied algorithm in this direction is the proximal alternating linearized minimization algorithm (PALM) by Bolte et al. [4] based on results in [1, 2]. Pock and Sabach [36] realized that the convergence speed of PALM can be considerably improved by inserting some nonexpensive inertial steps and called the accelerated algorithm iPALM. In many problems in imaging and machine learning, parts of the objective function can be often written as sum of a huge number of functions sharing the same structure. In general the computation of the gradient of these parts is too time and storage consuming so that stochastic gradient approximations were applied, see, e.g. [5] and the references therein. A combination of the simple stochastic gradient descent (SGD) estimator with PALM was first discussed by Xu and Yin in [46]. The authors refer to their method as block stochastic gradient iteration and do not mention the connection to PALM. Under rather hard assumptions on the objective function F, they proved that the sequence $(x^k)_k$ produced by their algorithm is such that $\mathbb{E}\left(\operatorname{dist}(0, \partial F(x^k)\right)$ converges to zero as $k \to \infty$. Another idea for a stochastic variant of PALM was proposed by Davis et al. [11]. The authors introduce an asynchronous variant of PALM with stochastic noise in the gradient and called it SAPALM. Assuming an explicit bound of the variance of the noise, they proved certain convergence results. Their approach requires an explicit bound on the noise, which is not fulfilled for the gradient estimators considered in this paper. Further, we like to mention that a stochastic variant of the primal-dual algorithm of Chambolle and Pock [9] for solving convex problems was developed in [8].

Replacing the simple stochastic gradient descent estimators by more sophisticated so-called variance-reduced gradient estimators, Driggs et al. [13] could weaken the assumptions on the objective function in [46] and improve the estimates on the convergence rate of a stochastic PALM algorithm. They called the corresponding algorithm SPRING. However, the convergence analysis within [13] is based on the so-called generalized gradient $\mathscr{G}F_{\tau_1,\tau_2}$. Within the first versions of the paper [13], when the preprint of this paper appeared, this generalized gradient was not even well-defined. Even if the definition was fixed over the time, the use of the generalized gradient is not satisfying at all, since it becomes not clear how this generalized gradient is related to the (sub)differential of the objective function in limit processes with varying τ_1 and τ_2. In particular, it is easy to find examples of F and sequences $(\tau_1^k)_k$ and $(\tau_2^k)_k$ such that the generalized gradient $\mathscr{G}F_{\tau_1^k,\tau_2^k}(x_1, x_2)$ is non-zero, but converges to zero for fixed x_1 and x_2. Note that the advantages of variance reduction to accelerate stochastic gradient methods were discussed by several authors, see, e.g. [24, 39].

In this paper, we merge a stochastic PALM algorithm with an inertial procedure to obtain a new iSPALM algorithm. The inertial parameters can also be viewed as a generalization of momentum parameters to nonsmooth problems. Momentum parameters are widely used to speed up and stabilize optimization algorithms based on (stochastic) gradient descent. In particular, for machine learning applications it is known that momentum algorithms [32, 37, 38, 41] as well as their stochastic modifications like the

Adam optimizer [25] perform much better than a plain (stochastic) gradient descent, see e.g. [15, 43]. From this point of view, inertial or momentum parameters are one of the core ingredients for an efficient optimization algorithm to minimize the loss in data driven approaches. We examine the convergence behavior of iSPALM both theoretically and numerically. Under certain assumptions on the parameters of the algorithm which also appear in the iPALM algorithm, we show that iSPALM converges linearly. In particular, we have to modify the definition of variance-reduced gradient estimators to inertial ones. We clearly indicate the few lemmas which are somehow related e.g. to those in [13] and address the necessary technical adaptations in an extended preprint [21]. The proofs given in this paper are completely new. In the numerical part, we focus on two examples, namely (i) MNIST classification with proximal neural networks (PNNs), and (ii) parameter learning for Student-t mixture models (MMs).

PNNs basically replace the standard layer $\sigma(Tx + b)$ of a feed-forward neural network by $T^{\mathrm{T}}\sigma(Tx + b)$ and require that T is an element of the (compact) Stiefel manifold, i.e. has orthonormal columns, see [18, 20]. This implies that PNNs are 1-Lipschitz and hence more stable under adversarial attacks than a neural network of comparable size without the orthogonality constraints. While the PNNs were trained in [18] using a SGD on the Stiefel manifold, we train it in this paper by adding the characteristic function of the feasible weights to the loss for incorporating the orthogonality constraints and use PALM, iPALM, SPRING and iSPALM for the optimization.

Learned MMs provide a powerful tool in data and image processing. While Gaussian MMs are mostly used in the field, more robust methods can be achieved by using heavier tailed distributions, as, e.g. the Student-t distribution. In [44], it was shown that Student-t MMs are superior to Gaussian ones for modeling image patches and the authors proposed an application in image compression. Image denoising based on Student-t models was addressed in [27] and image deblurring in [12, 47]. Further applications include robust image segmentation [3, 34, 42] and superresolution [19] as well as registration [14, 48]. For learning MMs a maximizer of the corresponding log-likelihood has to be computed. Usually an expectation maximization (EM) algorithm [26, 30, 35] or certain of its acceleration [6, 31, 45] are applied for this purpose. However, if the MM has many components and we are given large data, a stochastic optimizat ion approach appears to be more efficient. Indeed, recently, also stochastic variants of the EM algorithm were proposed [7, 10], but show various disadvantages and we are not aware of a circumvent convergence result for these algorithms. In particular, one assumption on the stochastic EM algorithm is that the underlying distribution family is an exponential family, which is not the case for MMs. In this paper, we propose for the first time to use the (inertial) PALM algorithms as well as their stochastic variants for maximizing a modified version of the log-likelihood function.

This paper is organized as follows: In Sect. 2, we provide the notation used throughout the paper. To understand the differences of existing algorithms to our novel one, we discuss PALM and iPALM together with convergence results in Sect. 3. Section 4 introduces our iSPALM algorithm. We discuss the convergence behavior of iSPALM in Sect. 5. Finally, we compare the performance of our iSPALM with (inertial) PALM and stochastic PALM when applied to two nonconvex optimization problems in machine

learning. We provide the code online.[1] Finally, conclusions are drawn and directions of further research are addressed in Sect. 7.

2 Preliminaries

In this section, we introduce the basic notation and results which we will use throughout this paper.

For an proper and lower semi-continuous function $f : \mathbb{R}^d \to (-\infty, \infty]$ and $\tau > 0$ the *proximal mapping* $\operatorname{prox}_\tau^f : \mathbb{R}^d \to \mathcal{P}(\mathbb{R}^d)$ is defined by

$$\operatorname{prox}_\tau^f(x) := \operatorname*{argmin}_{y \in \mathbb{R}^d} \left\{ \tfrac{\tau}{2}\|x - y\|^2 + f(y) \right\},$$

where $\mathcal{P}(\mathbb{R}^d)$ denotes the power set of $\mathbb{R}^d$. The proximal mapping admits the following properties, see e.g. [40].

Proposition 2.1 *Let $f : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ be proper and lower semi-continuous with $\inf_{\mathbb{R}^d} f > -\infty$. Then, the following holds true.*

(i) *The set $\operatorname{prox}_\tau^f(x)$ is nonempty and compact for any $x \in \mathbb{R}^d$ and $\tau > 0$.*
(ii) *If f is convex, then $\operatorname{prox}_\tau^f(x)$ contains exactly one value for any $x \in \mathbb{R}^d$ and $\tau > 0$.*

To describe critical points, we will need the definition of (general) subgradients, see e.g. [40].

Definition 2.2 Let $f : \mathbb{R}^d \to (-\infty, \infty]$ be a proper and lower semi-continuous function and $v \in \mathbb{R}^d$. Then we call

(i) v a *regular subgradient* of f at $\bar{x}$, written $v \in \hat{\partial} f(\bar{x})$, if for all $x \in \mathbb{R}^d$,

$$f(x) \geq f(\bar{x}) + \langle v, x - \bar{x} \rangle + o(\|x - \bar{x}\|).$$

(ii) v a (general) *subgradient* of f at $\bar{x}$, written $v \in \partial f(\bar{x})$, if there are sequences $x^k \to \bar{x}$ and $v^k \in \hat{\partial} f(x^k)$ with $v^k \to v$ as $k \to \infty$.

The following proposition lists useful properties of subgradients.

Proposition 2.3 (Properties of Subgradients) *Let $f : \mathbb{R}^{d_1} \to (-\infty, \infty]$ and $g : \mathbb{R}^{d_2} \to (-\infty, \infty]$ be proper and lower semicontinuous and let $h : \mathbb{R}^{d_1} \to \mathbb{R}$ be continuously differentiable. Then the following holds true.*

1. *For any $x \in \mathbb{R}^{d_1}$, we have $\hat{\partial} f(x) \subseteq \partial f(x)$. If f is additionally convex, we have $\hat{\partial} f(x) = \partial f(x)$.*
2. *For $x \in \mathbb{R}^{d_1}$ with $f(x) < \infty$, it holds*

$$\hat{\partial}(f + h)(x) = \hat{\partial} f(x) + \nabla h(x) \quad \text{and} \quad \partial(f + h)(x) = \partial f(x) + \nabla h(x).$$

[1] https://github.com/johertrich/Inertial-Stochastic-PALM.

3. *If* $\sigma(x_1, x_2) = f_1(x_1) + f_2(x_2)$, *then*

$$\begin{pmatrix} \hat{\partial}_{x_1} f_1(\bar{x}_1) \\ \hat{\partial}_{x_2} f_2(\bar{x}_2) \end{pmatrix} \subseteq \hat{\partial}\sigma(\bar{x}_1, \bar{x}_2) \quad \textit{and} \quad \begin{pmatrix} \partial_{x_1} f_1(\bar{x}_1) \\ \partial_{x_2} f_2(\bar{x}_2) \end{pmatrix} \subseteq \partial\sigma(\bar{x}_1, \bar{x}_2).$$

Proof Part (i) was proved in [40,Theorem 8.6 and Proposition 8.12] and part (ii) in [40,Exercise 8.8]. Concerning part (iii) we have for $v_{x_i} \in \hat{\partial}_{x_i} f(\bar{x}_i)$, $i = 1, 2$ that for all $(x_1, x_2) \in \mathbb{R}^d \times \mathbb{R}^d$ it holds

$$\sigma(x_1, x_2) = f_1(x_1) + f_2(x_2) \geq \sum_{i=1}^{2} f_i(\bar{x}_i) + \langle v_{x_i}, x_i - \bar{x}_i \rangle + o(\|x_i - \bar{x}_i\|).$$

This proves the claim for regular subgradients.

For general subgradients consider $v_{x_i} \in \partial_{x_i} f_i(\bar{x}_i)$, $i = 1, 2$ By definition there exist sequences $x_i^k \to \bar{x}_i$ and $v_{x_i}^k \to v_{x_i}$ with $v_{x_i}^k \in \hat{\partial}_{x_i} f_i(x_i^k)$, $i = 1, 2$. By the statement for regular subgradients we know that $(v_{x_1}^k, v_{x_2}^k) \in \hat{\partial\sigma}(x_1^k, x_2^k)$. Thus, it follows by definition of the general subgradient that $(v_{x_1}, v_{x_2}) \in \partial\sigma(\bar{x}_1, \bar{x}_2)$. □

We call $(x_1, x_2) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$ a *critical point* of F if $0 \in \partial F(x_1, x_2)$. By [40,Theorem 10.1] we have that any local minimizer $\hat{x}$ of a proper and lower semicontinuous function $f : \mathbb{R}^d \to (-\infty, \infty]$ fulfills

$$0 \in \hat{\partial} f(\hat{x}) \subseteq \partial f(\hat{x}).$$

In particular, it is a critical point of f. Further, we have by Proposition 2.3 that $\hat{x} \in \operatorname{prox}_\tau^f(x)$ implies

$$0 \in \tau(\hat{x} - x) + \hat{\partial} f(y) \subseteq \tau(\hat{x} - x) + \partial f(y). \tag{1}$$

In this paper, we consider functions $F : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to (-\infty, \infty]$ of the form

$$F(x_1, x_2) = H(x_1, x_2) + f(x_1) + g(x_2) \tag{2}$$

with proper, lower semicontinuous functions $f : \mathbb{R}^{d_1} \to (-\infty, \infty]$ and $g : \mathbb{R}^{d_2} \to (-\infty, \infty]$ bounded from below and a continuously differentiable function $H : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$. Further, we assume throughout this paper that

$$\underline{F} := \inf_{(x_1, x_2) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}} F(x_1, x_2) > -\infty.$$

By Proposition 2.3 it holds

$$\begin{aligned} \begin{pmatrix} \partial_{x_1} F(x_1, x_2) \\ \partial_{x_2} F(x_1, x_2) \end{pmatrix} &= \nabla H(x_1, x_2) + \begin{pmatrix} \partial_{x_1} f(x_1) \\ \partial_{x_2} g(x_2) \end{pmatrix} \\ &\subseteq \nabla H(x_1, x_2) + \partial(f + g)(x_1, x_2) = \partial F(x_1, x_2). \end{aligned} \tag{3}$$

The *generalized gradient* of $F\colon \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to (-\infty, \infty]$ was defined in [13] as set-valued function

$$\mathscr{G}F_{\tau_1,\tau_2}(x_1, x_2) := \begin{pmatrix} \tau_1(x_1 - \operatorname{prox}^f_{\tau_1}(x_1 - \frac{1}{\tau_1}\nabla_{x_1} H(x_1, x_2))) \\ \tau_2(x_2 - \operatorname{prox}^g_{\tau_2}(x_2 - \frac{1}{\tau_2}\nabla_{x_2} H(x_1, x_2))) \end{pmatrix}.$$

To motivate this definition, note that $0 \in \mathscr{G}F_{\tau_1,\tau_2}(x_1, x_2)$ is a sufficient criterion for (x_1, x_2) being a critical point of F. This can be seen as follows: For $(x_1, x_2) \in \mathscr{G}F_{\tau_1,\tau_2}(x_1, x_2)$ we have

$$x_1 \in \operatorname{prox}^f_{\tau_1}(x_1 - \tfrac{1}{\tau_1}\nabla_{x_1} H(x_1, x_2)).$$

Using (1), this implies

$$0 \in \tau_1(x_1 - x_1 + \tfrac{1}{\tau_1}\nabla_{x_1} H(x_1, x_2)) + \partial f(x_1) = \nabla_{x_1} H(x_1, x_2) + \partial f(x_1).$$

Similarly we get $0 \in \nabla_{x_2} H(x_1, x_2) + \partial g(x_2)$. By (3) we conclude that (x_1, x_2) is a critical point of F.

3 PALM and iPALM

In this section, we review PALM [4] and its inertial version iPALM [36].

3.1 PALM

The following Algorithm 3.1 for minimizing (2) was proposed in [4].

Algorithm 3.1 Proximal Alternating Linearized Minimization (PALM)

Input: $(x_1^0, x_2^0) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$, parameters τ_1^k, τ_2^k for $k \in \mathbb{N}_0$.
for $k = 0, 1, \ldots$ **do**
Set

$$x_1^{k+1} \in \operatorname{prox}^f_{\tau_1^k}\big(x_1^k - \tfrac{1}{\tau_1^k}\nabla_{x_1} H(x_1^k, x_2^k)\big)$$

Set

$$x_2^{k+1} \in \operatorname{prox}^g_{\tau_2^k}\big(x_2^k - \tfrac{1}{\tau_2^k}\nabla_{x_2} H(x_1^{k+1}, x_2^k)\big)$$

To prove convergence of PALM the following additional assumptions on H are needed:

Assumption 3.1 (*Assumptions on* H)

(i) For any $x_1 \in \mathbb{R}^{d_1}$, the function $\nabla_{x_2} H(x_1, \cdot)$ is globally Lipschitz continuous with Lipschitz constant $L_2(x_1)$. Similarly, for any $x_2 \in \mathbb{R}^{d_2}$, the function $\nabla_{x_2} H(\cdot, x_2)$ is globally Lipschitz continuous with Lipschitz constant $L_1(x_2)$.
(ii) There exist $\lambda_1^-, \lambda_2^-, \lambda_1^+, \lambda_2^+ > 0$ such that

$$\inf\{L_1(x_2^k) : k \in \mathbb{N}\} \geq \lambda_1^- \quad \text{and} \quad \inf\{L_2(x_1^k) : k \in \mathbb{N}\} \geq \lambda_2^-,$$
$$\sup\{L_1(x_2^k) : k \in \mathbb{N}\} \leq \lambda_1^+ \quad \text{and} \quad \sup\{L_2(x_1^k) : k \in \mathbb{N}\} \leq \lambda_2^+.$$

Remark 3.2 Assume that $H \in \mathbb{C}^2(\mathbb{R}^{d_1 \times d_2})$ fulfills Assumption 3.1(i). Then, the authors of [4] showed, that there are partial Lipschitz constants $L_1(x_2)$ and $L_2(x_1)$, such that Assumption 3.1(ii) is satisfied. □

Convergence results rely on a Kurdyka-Łojasiewicz property of functions which is defined in Appendix A. The following theorem was proven in [4,Lemma 3, Theorem 1].

Theorem 3.3 (Convergence of PALM) *Let $F : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to (-\infty, \infty]$ be given by (2). Further, assume that it fulfills Assumptions 3.1 and that ∇H is Lipschitz continuous on bounded subsets of $\mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$. Let $(x_1^k, x_2^k)_k$ be the sequence generated by PALM, where the step size parameters fulfill*

$$\tau_1^k \geq \gamma_1 L_1(x_2^k), \quad \tau_2^k \geq \gamma_2 L_2(x_1^{k+1})$$

for some $\gamma_1, \gamma_2 > 1$. Then, for $\eta := \min\{(\gamma_1 - 1)\lambda_1^-, (\gamma_2 - 1)\lambda_2^-\}$, the sequence $(F(x_1^k, x_2^k))_k$ is nonincreasing and

$$\frac{\eta}{2} \left\| (x_1^{k+1}, x_2^{k+1}) - (x_1^k, x_2^k) \right\|_2^2 \leq F(x_1^k, x_2^k) - F(x_1^{k+1}, x_2^{k+1}).$$

If F is in addition a KL function and the sequence $(x_1^k, x_2^k)_k$ is bounded, then it converges to a critical point of F.

3.2 iPALM

To speed up the performance of PALM the inertial variant iPALM in Algorithm 3.2 was suggested in [36].

Remark 3.4 (Relation to Momentum Methods) The inertial parameters in iPALM can be viewed as a generalization of momentum parameters for nonsmooth functions. To see this, note that iPALM with one block, $f = 0$ and $\beta^k = 0$ reads as

$$y^k = x^k + \alpha^k (x^k - x^{k-1}),$$
$$x^{k+1} = y^k - \tfrac{1}{\tau^k} \nabla H(x^k).$$

By introducing $g^k := x^k - x^{k-1}$, this can be rewritten as

$$g^{k+1} = \alpha^k g^k - \tfrac{1}{\tau^k} \nabla H(x^k),$$

Algorithm 3.2 Inertial Proximal Alternating Linearized Minimization (iPALM)

Input: $(x_1^{-1}, x_2^{-1}) = (x_1^0, x_2^0) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$, parameters $\alpha_1^k, \alpha_2^k, \beta_1^k, \beta_2^k, \tau_1^k, \tau_2^k$ for $k \in \mathbb{N}_0$.
for $k = 0, 1, \ldots$ **do**
Set

$$\begin{aligned} y_1^k &= x_1^k + \alpha_1^k (x_1^k - x_1^{k-1}) \\ z_1^k &= x_1^k + \beta_1^k (x_1^k - x_1^{k-1}) \\ x_1^{k+1} &\in \operatorname{prox}^f_{\tau_1^k}\big(y_1^k - \tfrac{1}{\tau_1^k} \nabla_{x_1} H(z_1^k, x_2^k)\big) \end{aligned}$$

Set

$$\begin{aligned} y_2^k &= x_2^k + \alpha_2^k (x_2^k - x_2^{k-1}) \\ z_2^k &= x_2^k + \beta_2^k (x_2^k - x_2^{k-1}) \\ x_2^{k+1} &\in \operatorname{prox}^g_{\tau_2^k}\big(y_2^k - \tfrac{1}{\tau_2^k} \nabla_{x_2} H(x_1^{k+1}, z_2^k)\big) \end{aligned}$$

$$x^{k+1} = x^k + g^{k+1}.$$

This is exactly the momentum method as introduced by Polyak in [37]. Similar, if $f = 0$ and $\alpha^k = \beta^k \neq 0$, iPALM can be rewritten as

$$\begin{aligned} g^{k+1} &= \alpha^k g^k - \tfrac{1}{\tau^k} \nabla H(x^k + \alpha^k g^k), \\ x^{k+1} &= x^k + g^{k+1}, \end{aligned}$$

which is known as Nesterov's Accelerated Gradient (NAG) [32]. Consequently, iPALM can be viewed as a generalization of both the classical momentum method and NAG to the nonsmooth case. Even if there exists no proof of tighter convergence rates for iPALM than for PALM, this motivates that the inertial steps really accelerate PALM, since NAG has tighter convergence rates than a plain gradient descent algorithm provided that the objective function is convex. □

To prove the convergence of iPALM the parameters of the algorithm must be carefully chosen.

Assumption 3.5 (*Conditions on the Parameters of iPALM*) Let λ_i^+, $i = 1, 2$ and $L_1(x_2^k)$, $L_2(x_1^k)$ be defined by Assumption 3.1. There exists some $\epsilon > 0$ such that for all $k \in \mathbb{N}$ and $i = 1, 2$ the following holds true:

(i) There exist $0 < \bar{\alpha}_i < \frac{1-\epsilon}{2}$ such that $0 \leq \alpha_i^k \leq \bar{\alpha}_i$ and $0 < \bar{\beta}_i \leq 1$ such that $0 \leq \beta_i^k \leq \bar{\beta}_i$.
(ii) The parameters τ_1^k and τ_2^k are given by

$$\tau_1^k := \frac{(1+\epsilon)\delta_1 + (1+\bar{\beta}_1) L_1(x_2^k)}{1 - \alpha_1^k} \quad \text{and} \quad \tau_2^k := \frac{(1+\epsilon)\delta_2 + (1+\bar{\beta}_2) L_2(x_1^{k+1})}{1 - \alpha_2^k},$$

and for $i = 1, 2$,

$$\delta_i := \frac{\bar{\alpha}_i + \bar{\beta}_i}{1 - \epsilon - 2\bar{\alpha}_i} \lambda_i^+.$$

The following theorem was proven in [36,Theorem 4.1].

Theorem 3.6 (Convergence of iPALM) *Let* $F \colon \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to (-\infty, \infty]$ *given by* (2) *be a KL function. Suppose that* H *fulfills the Assumptions* 3.1 *and that* ∇H *is Lipschitz continuous on bounded subsets of* $\mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$. *Further, let the parameters of iPALM fulfill the parameter conditions of Assumption* 3.5. *If the sequence* $(x_1^k, x_2^k)_k$ *generated by iPALM is bounded, then it converges to a critical point of* F.

Remark 3.7 Even though we cited PALM and iPALM just for two blocks (x_1, x_2) of variables, the convergence proofs from [4] and [36] even work with more than two blocks. □

4 iSPALM

In many problems in imaging and machine learning the function $H : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$ in (2) is of the form

$$H(x_1, x_2) = \frac{1}{n} \sum_{i=1}^{n} h_i(x_1, x_2), \tag{4}$$

where n is large. Then the computation of the gradients in PALM and iPALM is very time consuming. The idea to combine stochastic gradient estimators with a PALM scheme was first discussed by Xu and Yin in [46]. The authors replaced the gradient in Algorithm 3.1 by the *stochastic gradient descent* (SGD) *estimator*

$$\tilde{\nabla}_{x_i} H(x_1, x_2) := \frac{1}{b} \sum_{j \in B} \nabla_{x_i} h_j(x_1, x_2),$$

where $B \subset \{1, \ldots, n\}$ is a random subset (mini-batch) of fixed batch size $b = |I|$. This gives Algorithm 4.1 which we call SPALM.

Xu and Yin showed in [46] under rather strong assumptions, in particular f, g have to be Lipschitz continuous and the variance of the SGD estimator has to be bounded, that there exists a subsequence $(x_1^k, x_2^k)_k$ of iterates generated by Algorithm 4.1 such that the sequence $\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^k, x_2^k))\right)$ converges to zero as $k \to \infty$. If F, f and g are strongly convex, the authors proved also convergence of the function values to the infimum of F.

Driggs et al. [13] could weaken the assumptions and improve the convergence rate by replacing the SGD estimator by so-called variance-reduced gradient estimators $\tilde{\nabla}$. They called their method SPRING.

Algorithm 4.1 Stochastic Proximal Alternating Linearized Minimization (SPALM with SGD/SPRING with SARAH estimator $\tilde{\nabla}$)

Input: $(x_1^0, x_2^0) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$, parameters τ_1^k, τ_2^k for $k \in \mathbb{N}_0$.
for $k = 0, 1, \ldots$ **do**
Set

$$x_1^{k+1} \in \operatorname{prox}^f_{\tau_1^k}\left(x_1^k - \tfrac{1}{\tau_1^k}\tilde{\nabla}_{x_1} H(x_1^k, x_2^k)\right)$$

Set

$$x_2^{k+1} \in \operatorname{prox}^g_{\tau_2^k}\left(x_2^k - \tfrac{1}{\tau_2^k}\tilde{\nabla}_{x_2} H(x_1^{k+1}, x_2^k)\right)$$

However, in deep learning with nonsmooth functions, the combination of momentum-like methods and a stochastic gradient estimator turned out to be essential [15, 43]. To this end, we define inertial variance-reduced gradient estimators in a slightly different way as in [13].

Definition 4.1 (*Inertial Variance-Reduced Gradient Estimator*) A gradient estimator $\tilde{\nabla}$ is called *inertial variance-reduced* for a differentiable function $H : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$ with constants $V_1, V_\Upsilon \geq 0$ and $\rho \in (0, 1]$, if for any sequence $(x^k)_k = (x_1^k, x_2^k)_{k\in\mathbb{N}_0}$, $x^{-1} := x^0$ and any $0 \leq \beta_i^k < \bar{\beta}_i$, $i = 1, 2$ there exists a sequence of random variables $(\Upsilon_k)_{k\in\mathbb{N}}$ with $\mathbb{E}(\Upsilon_1) < \infty$ such that following holds true:

(i) For $z_i^k := x_i^k + \beta_i^k(x_i^k - x_i^{k-1})$, $i = 1, 2$, we have

$$\begin{aligned}&\mathbb{E}_k(\|\tilde{\nabla}_{x_1} H(z_1^k, x_2^k) - \nabla_{x_1} H(z_1^k, x_2^k)\|^2 + \|\tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\\ &\quad - \nabla_{x_2} H(x_1^{k+1}, z_2^k)\|^2)\\ &\quad \leq \Upsilon_k + V_1\left(\mathbb{E}_k(\|x^{k+1} - x^k\|^2) + \|x^k - x^{k-1}\|^2 + \|x^{k-1} - x^{k-2}\|^2\right).\end{aligned}$$

(ii) The sequence $(\Upsilon_k)_k$ decays geometrically, that is

$$\begin{aligned}\mathbb{E}_k(\Upsilon_{k+1}) \leq (1-\rho)\Upsilon_k + V_\Upsilon(\mathbb{E}_k(\|x^{k+1} - x^k\|^2) + \|x^k - x^{k-1}\|^2\\ + \|x^{k-1} - x^{k-2}\|^2).\end{aligned}$$

(iii) If $\lim_{k\to\infty} \mathbb{E}(\|x^k - x^{k-1}\|^2) = 0$, then $\mathbb{E}(\Upsilon_k) \to 0$ as $k \to \infty$.

While the SGD estimator is not inertial variance-reduced, we will show that the SARAH [33] estimator has this property.

Definition 4.2 (SARAH Estimator) The SARAH *estimator* reads for $k = 0$ as

$$\tilde{\nabla}_{x_1} H(x_1^0, x_2^0) = \nabla_{x_1} H(x_1^0, x_2^0).$$

For $k = 1, 2, \ldots$ we define random variables $p_i^k \in \{0, 1\}$ with $P(p_i^k = 0) = \frac{1}{p}$ and $P(p_i^k = 1) = 1 - \frac{1}{p}$, where $p \in (1, \infty)$ is a fixed chosen parameter. Further, we

define B_i^k to be random subsets uniformly drawn from $\{1, \ldots, n\}$ of fixed batch size b. Then for $k = 1, 2, \ldots$ the SARAH estimator reads as

$$\tilde{\nabla}_{x_1} H(x_1^k, x_2^k) = \begin{cases} \nabla_{x_1} H(x_1^k, x_2^k), & \text{if } p_1^k = 0, \\ \frac{1}{b} \sum_{i \in B_i^k} \nabla_{x_1} h_i(x_1^k, x_2^k) - \nabla_{x_1} h_i(x_1^{k-1}, x_2^{k-1}) + \tilde{\nabla}_{x_1} H(x_1^{k-1}, x_2^{k-1}), & \text{if } p_1^k = 1, \end{cases}$$

and analogously for $\tilde{\nabla}_{x_2} H$. In the sequel, we assume that the family of the random elements p_i^k, B_i^k for $i = 1, 2$ and $k = 1, 2, \ldots$ is independent.

Indeed, we can show the desired property of the SARAH gradient estimator.

Proposition 4.3 *Let $H : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$ be given by (4) with functions $h_i : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$ having a globally M-Lipschitz continuous gradient. Then the SARAH estimator $\tilde{\nabla}$ is inertial variance-reduced with parameters $\rho = \frac{1}{p}$ and*

$$V_\Upsilon = 3(1 - \tfrac{1}{p}) M^2 \left(1 + \max\left((\bar{\beta}_1)^2, (\bar{\beta}_2)^2\right)\right).$$

Furthermore, we can choose

$$\Upsilon_{k+1} = \|\tilde{\nabla}_{x_1} H(z_1^k, x_2^k) - \nabla_{x_1} H(z_1^k, x_2^k)\|^2 + \|\tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k) - \nabla_{x_2} H(x_1^{k+1}, z_2^k)\|^2.$$

For the proof which follows mainly the path of [13,Proposition 2.2], but must be nevertheless carefully adapted to the inertial setting, we refer to [21].

Finally, we can propose our inertial stochastic PALM (iSPALM) algorithm with SARAH estimator $\tilde{\nabla}$ in Algorithm 4.2.

Algorithm 4.2 Inertial Stochastic Proximal Alternating Linearized Minimization (iSPALM)

Input: $(x_1^{-1}, x_2^{-1}) = (x_1^0, x_2^0) \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$, parameters $\alpha_1^k, \alpha_2^k, \beta_1^k, \beta_2^k, \tau_1^k, \tau_2^k$ for $k \in \mathbb{N}_0$.
for $k = 0, 1, \ldots$ **do**
Set

$$\begin{aligned} y_1^k &= x_1^k + \alpha_1^k (x_1^k - x_1^{k-1}) \\ z_1^k &= x_1^k + \beta_1^k (x_1^k - x_1^{k-1}) \\ x_1^{k+1} &\in \operatorname{prox}_{\tau_1^k}^f \left(y_1^k - \tfrac{1}{\tau_1^k} \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\right). \end{aligned}$$

Set

$$\begin{aligned} y_2^k &= x_2^k + \alpha_2^k (x_2^k - x_2^{k-1}) \\ z_2^k &= x_2^k + \beta_2^k (x_2^k - x_2^{k-1}) \\ x_2^{k+1} &\in \operatorname{prox}_{\tau_2^k}^g \left(y_2^k - \tfrac{1}{\tau_2^k} \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\right). \end{aligned}$$

Remark 4.4 Similarly as in Remark 3.4, iSPALM can be viewed as a generalization of the stochastic versions of the momentum method and NAG to the nonsmooth case. Note, that in the stochastic setting the theoretical error bounds of momentum methods are not tighter than for a plain gradient descent. An overview over these convergence results can be found in [15, 43]. Consequently, we are not able to show tighter convergence rates for iSPALM than for stochastic PALM. Nevertheless, stochastic momentum methods as the momentum SGD and the Adam optimizer [25] are widely used and have shown a better convergence behavior than a plain SGD in a huge number of applications. □

5 Convergence analysis of iSPALM

We assume that the parameters of iSPALM fulfill the following conditions.

Assumption 5.1 (Conditions on the Parameters of iSPALM) Let λ_i^+, $i = 1, 2$ and $L_1(x_2^k), L_2(x_1^k)$ be defined by Assumption 3.1 and ρ, V_1, V_Υ by Definition 4.1. Further, let $H\colon \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$ be given by (4) with functions $h_i\colon \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}$ having a globally M-Lipschitz continuous gradient. There exist $\epsilon, \varepsilon > 0$ such that for all $k \in \mathbb{N}$ and $i = 1, 2$ the following holds true:

(i) There exist $0 < \bar{\alpha}_i < \frac{1-\epsilon}{2}$ such that $0 \le \alpha_i^k \le \bar{\alpha}_i$ and $0 < \bar{\beta}_i \le 1$ such that $0 \le \beta_i^k \le \bar{\beta}_i$

(ii) The parameters τ_i^k, $i = 1, 2$ are given by

$$\tau_1^k := \frac{(1+\epsilon)\delta_1 + M + L_1(x_2^k) + S}{1 - \alpha_1^k}, \quad \text{and} \quad \tau_2^k := \frac{(1+\epsilon)\delta_2 + M + L_2(x_1^{k+1}) + S}{1 - \alpha_2^k},$$

where $S := 4\frac{\rho V_1 + V_\Upsilon}{\rho M} + \varepsilon$ and for $i = 1, 2$,

$$\delta_i := \frac{(M + \lambda_i^+)\bar{\alpha}_i + 2\lambda_i^+ \bar{\beta}_i^2 + S}{1 - 2\bar{\alpha}_i - \epsilon}.$$

To analyze the convergence behavior of iSPALM, we start with two auxiliary lemmas. The first one can be proven analogously to [36,Proposition 4.1].

Lemma 5.2 *Let $(x_1^k, x_2^k)_k$ be an arbitrary sequence and $\alpha_i^k, \beta_i^k \in \mathbb{R}$, $i = 1, 2$. Further define*

$$y_i^k := x_i^k + \alpha_i^k(x_i^k - x_i^{k-1}), \quad z_i^k := x_i^k + \beta_i^k(x_i^k - x_i^{k-1}), \quad i = 1, 2,$$

and

$$\Delta_i^k := \tfrac{1}{2}\|x_i^k - x_i^{k-1}\|^2, \qquad i = 1, 2.$$

Then, for any $k \in \mathbb{N}$ and $i = 1, 2$, we have

(i) $\|x_i^k - y_i^k\|^2 = 2(\alpha_i^k)^2 \Delta_i^k$,
(ii) $\|x_i^k - z_i^k\|^2 = 2(\beta_i^k)^2 \Delta_i^k$,
(iii) $\|x_i^{k+1} - y_i^k\|^2 \geq 2(1 - \alpha_i^k \Delta_i^{k+1} + 2\alpha_i^k)(\alpha_i^k - 1)\Delta_i^k$.

The second auxiliary lemma can be proven analogously to [36,Lemma 3.2].

Lemma 5.3 *Let $\psi = \sigma + h$, where $h\colon \mathbb{R}^d \to \mathbb{R}$ is a continuously differentiable function with L_h-Lipschitz continuous gradient, and $\sigma\colon \mathbb{R}^d \to (-\infty, \infty]$ is proper and lower semicontinuous with $\inf_{\mathbb{R}^d} \sigma > -\infty$. Then it holds for any $u, v, w \in \mathbb{R}^d$ and any $u^+ \in \mathbb{R}^d$ defined by*

$$u^+ \in \operatorname{prox}_t^\sigma (v - \tfrac{1}{t}\tilde{\nabla} h(w)), \quad t > 0$$

that

$$\begin{aligned}\psi(u^+) \leq \psi(u) + \langle u^+ - u, \nabla h(u) - \tilde{\nabla} h(w)\rangle + \frac{{L_h}^2}{2}\|u - u^+\|^2 \\ + \frac{t}{2}\|u - v\|^2 - \frac{t}{2}\|u^+ - v\|^2.\end{aligned}$$

Now we can establish a result on the expectation of squared subsequent iterates. Note that equivalent results were shown for PALM, iPALM and SPRING. Here we use a function Ψ, which not only contains the current function value, but also the distance of the iterates to the previous ones. A similar idea was used in the convergence proof of iPALM [36]. Nevertheless, incorporating the stochastic gradient estimator here makes the proof much more involved.

Theorem 5.4 *Let $F\colon \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to (-\infty, \infty]$ be given by (2) and fulfill Assumption 3.1. Let $(x_1^k, x_2^k)_k$ be generated by iSPALM with parameters fulfilling Assumption 5.1, where we use an inertial variance-reduced gradient estimator $\tilde{\nabla}$. Then it holds for $\Psi\colon (\mathbb{R}^{d_1} \times \mathbb{R}^{d_2})^3 \to \mathbb{R}$ defined for $u = (u_{11}, u_{12}, u_{21}, u_{22}, u_{31}, u_{32}) \in (\mathbb{R}^{d_1} \times \mathbb{R}^{d_2})^3$ by*

$$\begin{aligned}\Psi(u) &:= F(u_{11}, u_{12}) + \tfrac{\delta_1}{2}\|u_{11} - u_{21}\|^2 + \tfrac{\delta_2}{2}\|u_{12} - u_{22}\|^2 \\ &\quad + \tfrac{S}{4}\left(\|u_{21} - u_{31}\|^2 + \|u_{22} - u_{32}\|^2\right)\end{aligned}$$

that there exists $\gamma > 0$ such that

$$\Psi(u^1) - \inf_{u \in (\mathbb{R}^{d_1} \times \mathbb{R}^{d_2})^2} \Psi(u) + \tfrac{1}{M\rho}\mathbb{E}(\Upsilon_1) \geq \gamma \sum_{k=0}^{T} \mathbb{E}(\|u^{k+1} - u^k\|^2),$$

where $u^k := (x_1^k, x_2^k, x_1^{k-1}, x_2^{k-1}, x_1^{k-2}, x_2^{k-2})$. In particular, we have

$$\sum_{k=0}^{\infty} \mathbb{E}(\|u^{k+1} - u^k\|^2) < \infty.$$

Proof By Lemma 5.3 with $\psi := H(\cdot, x_2) + f$, we obtain

$$\begin{aligned} H(x_1^{k+1}, x_2^k) + f(x_1^{k+1}) &\le H(x_1^k, x_2^k) + f(x_1^k) \\ &\quad + \left\langle x_1^{k+1} - x_1^k, \nabla_{x_1} H(x_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k) \right\rangle \\ &\quad + \tfrac{L_1(x_2^k)}{2} \|x_1^{k+1} - x_1^k\|^2 + \tfrac{\tau_1^k}{2} \|x_1^k - y_1^k\|^2 - \tfrac{\tau_1^k}{2} \|x_1^{k+1} - y_1^k\|^2. \end{aligned} \tag{5}$$

Using $ab \le \frac{s}{2} a^2 + \frac{1}{2s} b^2$ for $s > 0$ and $\|a - c\|^2 \le 2\|a - b\|^2 + 2\|b - c\|^2$ the inner product is smaller or equal than

$$\begin{aligned} &\tfrac{s_1^k}{2} \|x_1^{k+1} - x_1^k\|^2 + \tfrac{1}{2s_1^k} \|\nabla_{x_1} H(x_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 \\ &\le \tfrac{s_1^k}{2} \|x_1^{k+1} - x_1^k\|^2 + \tfrac{1}{s_1^k} \|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 \\ &\quad + \tfrac{1}{s_1^k} \|\nabla_{x_1} H(x_1^k, x_2^k) - \nabla_{x_1} H(z_1^k, x_2^k)\|^2 \\ &= \tfrac{s_1^k}{2} \|x_1^{k+1} - x_1^k\|^2 + \tfrac{1}{s_1^k} \|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 + \tfrac{L_1(x_2^k)^2}{s_1^k} \|x_1^k - z_1^k\|^2. \end{aligned}$$

Combined with (5) this becomes

$$\begin{aligned} &H(x_1^{k+1}, x_2^k) + f(x_1^{k+1}) \\ &\le H(x_1^k, x_2^k) + f(x_1^k) + \tfrac{L_1(x_2^k)}{2} \|x_1^{k+1} - x_1^k\|^2 + \tfrac{\tau_1^k}{2} \|x_1^k - y_1^k\|^2 - \tfrac{\tau_1^k}{2} \|x_1^{k+1} - y_1^k\|^2 \\ &\quad + \tfrac{s_1^k}{2} \|x_1^{k+1} - x_1^k\|^2 + \tfrac{1}{s_1^k} \|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 + \tfrac{L_1(x_2^k)^2}{s_1^k} \|x_1^k - z_1^k\|^2. \end{aligned}$$

Using Lemma 5.2 we get

$$\begin{aligned} &H(x_1^{k+1}, x_2^k) + f(x_1^{k+1}) \\ &\le H(x_1^k, x_2^k) + f(x_1^k) + \left(L_1(x_2^k) + s_1^k - \tau_1^k (1 - \alpha_1^k) \right) \Delta_1^{k+1} \\ &\quad + \tfrac{1}{s_1^k} \left(2L_1(x_2^k)^2 (\beta_1^k)^2 + s_1^k \tau_1^k \alpha_1^k \right) \Delta_1^k + \tfrac{1}{s_1^k} \|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2. \end{aligned}$$

Analogously we conclude for $\psi := H(x_1, \cdot) + g$ that

$$\begin{aligned} &H(x_1^{k+1}, x_2^{k+1}) + g(x_2^{k+1}) \\ &\le H(x_1^{k+1}, x_2^k) + g(x_2^k) + \left(L_2(x_1^{k+1}) + s_2^k - \tau_2^k (1 - \alpha_2^k) \right) \Delta_2^{k+1} \\ &\quad + \tfrac{1}{s_2^k} \left(2L_2(x_1^{k+1})^2 (\beta_2^k)^2 + s_2^k \tau_2^k \alpha_2^k \right) \Delta_2^k + \tfrac{1}{s_2^k} \|\nabla_{x_2} H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\|^2. \end{aligned}$$

Adding the last two inequalities and using the abbreviation $L_1^k := L_1(x_2^k)$ and $L_2^k := L_2(x_1^{k+1})$, we obtain

$$\begin{aligned} F(x_1^{k+1}, x_2^{k+1}) &\leq F(x_1^k, x_2^k) \\ &+ \sum_{i=1}^{2} \left(\left(L_i^k + s_i^k - \tau_i^k(1-\alpha_i^k) \right) \Delta_i^{k+1} + \tfrac{1}{s_i^k} \left(2(L_i^k)^2(\beta_i^k)^2 + s_i^k \tau_i^k \alpha_i^k \right) \Delta_i^k \right) \\ &+ \tfrac{1}{s_1^k} \|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 + \tfrac{1}{s_2^k} \|\nabla_{x_2} H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\|^2. \end{aligned} \tag{6}$$

Reformulating (6) in terms of

$$\Psi(u^k) = F(x_1^k, x_2^k) + \delta_1 \Delta_1^k + \delta_2 \Delta_2^k + \tfrac{S}{2}(\Delta_1^{k-1} + \Delta_2^{k-1}) \tag{7}$$

leads to

$$\begin{aligned} \Psi(u^k) - \Psi(u^{k+1}) &= F(x_1^k, x_2^k) - F(x_1^{k+1}, x_2^{k+1}) + \delta_1 \Delta_1^k + \delta_2 \Delta_2^k - \delta_1 \Delta_1^{k+1} - \delta_2 \Delta_2^{k+1} \\ &+ \tfrac{S}{2} \left(\Delta_1^{k-1} + \Delta_2^{k-1} - \Delta_1^k - \Delta_2^k \right) \\ &\geq \sum_{i=1}^{2} \left(\left(\tau_i^k(1-\alpha_i^k) - s_i^k - L_i^k - \delta_i \right) \Delta_i^{k+1} \right) + \sum_{i=1}^{2} \left(\left(\delta_i - \tfrac{2}{s_i^k}(L_i^k)^2(\beta_i^k)^2 - \tau_i^k \alpha_i^k \right) \Delta_i^k \right) \\ &- \tfrac{1}{s_1^k} \|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 - \tfrac{1}{s_2^k} \|\nabla_{x_2} H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\|^2 \\ &+ \tfrac{S}{2} \left(\Delta_1^{k-1} + \Delta_2^{k-1} - \Delta_1^k - \Delta_2^k \right). \end{aligned} \tag{8}$$

Now, we set $s_1^k = s_2^k := M$ use that $L_i^k \leq M$, take the conditional expectation $\mathbb{E}_k$ in (8) and use that $\tilde{\nabla}$ is an inertial variance-reduced estimator to get

$$\begin{aligned} &\Psi(u^k) - \mathbb{E}_k(\Psi(u^{k+1})) \\ &\geq \sum_{i=1}^{2} \left(\left(\tau_i^k(1-\alpha_i^k) - M - L_i^k - \delta_i \right) \mathbb{E}_k(\Delta_i^{k+1}) + \left(\delta_i - \tfrac{2}{M}(L_i^k)^2(\beta_i^k)^2 - \tau_i^k \alpha_i^k \right) \Delta_i^k \right) \\ &\quad - \tfrac{2V_1}{M} \sum_{i=1}^{2} \left(\mathbb{E}_k(\Delta_i^{k+1}) + \Delta_i^k \right) - \tfrac{1}{M} \Upsilon_k + \tfrac{S}{2} \left(\Delta_1^{k-1} + \Delta_2^{k-1} - \Delta_1^k - \Delta_2^k \right) \\ &\geq \sum_{i=1}^{2} \left(\left(\tau_i^k(1-\alpha_i^k) - M - L_i^k - \delta_i - \tfrac{2V_1}{M} \right) \mathbb{E}_k(\Delta_i^{k+1}) \right) \\ &\quad + \sum_{i=1}^{2} \left(\left(\delta_i - 2L_i^k(\beta_i^k)^2 - \tau_i^k \alpha_i^k - \tfrac{2V_1}{M} \right) \Delta_i^k \right) \\ &\quad - \tfrac{1}{M} \Upsilon_k + \tfrac{S}{2} \left(\Delta_1^{k-1} + \Delta_2^{k-1} - \Delta_1^k - \Delta_2^k \right). \end{aligned} \tag{9}$$

Since $\tilde{\nabla}$ is inertial variance-reduced, we know from Definition 4.1 (ii) that

$$\rho \Upsilon_k \leq \Upsilon_k - \mathbb{E}_k(\Upsilon_{k+1}) + 2V_\Upsilon \sum_{i=1}^{2} \left(\mathbb{E}_k(\Delta_i^{k+1}) + \Delta_i^k + \Delta_i^{k-1} \right). \tag{10}$$

Inserting this in (9) and using the definition of S yields

$$\begin{aligned}
\Psi(u^k) - \mathbb{E}_k\left(\Psi(u^{k+1})\right) &\geq \sum_{i=1}^{2} \left(\left(\tau_i^k (1-\alpha_i^k) - M - L_i^k \right) - \delta_i - \tfrac{S}{2} \right) \mathbb{E}_k(\Delta_i^{k+1}) \Big) \\
&\quad + \sum_{i=1}^{2} \left(\left(\delta_i - 2L_i^k (\beta_i^k)^2 - \tau_i^k \alpha_i^k - \tfrac{S}{2} \right) \Delta_i^k \right) \\
&\quad - \frac{2V_\Upsilon}{\rho M} (\Delta_1^{k-1} + \Delta_2^{k-1}) + \tfrac{1}{M\rho} \left(\mathbb{E}_k(\Upsilon_{k+1}) - \Upsilon_k \right) + \tfrac{S}{2} \left(\Delta_1^{k-1} + \Delta_2^{k-1} - \Delta_1^k - \Delta_2^k \right) \\
&\geq \sum_{i=1}^{2} \Big(\underbrace{\left(\tau_i^k (1-\alpha_i^k) - M - L_i^k - \delta_i - S \right)}_{a_i^k} \mathbb{E}_k(\Delta_i^{k+1}) \Big) \\
&\quad + \sum_{i=1}^{2} \Big(\underbrace{\left(\delta_i - 2L_i^k (\beta_i^k)^2 - \tau_i^k \alpha_i^k - S \right)}_{b_i^k} \Delta_i^k \Big) \\
&\quad + \tfrac{1}{M\rho} \left(\mathbb{E}_k(\Upsilon_{k+1}) - \Upsilon_k \right) + \left(\tfrac{S}{2} - \tfrac{2V_\Upsilon}{\rho M} \right) \left(\Delta_1^{k-1} + \Delta_2^{k-1} \right).
\end{aligned} \tag{11}$$

Choosing $\tau_i^k, \delta_i, i = 1, 2$ and ϵ as in Assumption 5.1(ii), we obtain by straightforward computation for $i = 1, 2$ and all $k \in \mathbb{N}$ that $a_i^k = \epsilon \delta_i$ and

$$\begin{aligned}
b_i^k &= \tfrac{1}{1-\alpha_i^k} \left((1 - \epsilon - 2\alpha_i^k)\delta_i - \alpha_i^k M - S - L_i^k \left(2(\beta_i^k)^2 (1-\alpha_i^k) + \alpha_i^k \right) \right) + \epsilon \delta_i \\
&\geq \tfrac{1}{1-\alpha_i^k} \left((1 - \epsilon - 2\bar{\alpha}_i)\delta_i - \bar{\alpha}_i M - S - \lambda_i^+ \left(2(\bar{\beta}_i)^2 (1-\alpha_i^k) + \bar{\alpha}_i \right) \right) + \epsilon \delta_i \\
&= \epsilon \delta_i + 2 \frac{2\lambda_i^+ \alpha_i^k (\bar{\beta}_i)^2}{1-\alpha_i^k} \geq \epsilon \delta_i .
\end{aligned}$$

Applying this in (11), we get

$$\begin{aligned}
\Psi(u^k) - \mathbb{E}_k\left(\Psi(u^{k+1})\right) &\geq \epsilon \min(\delta_1, \delta_2) \sum_{i=1}^{2} (\mathbb{E}_k(\Delta_i^{k+1}) + \Delta_i^k) \\
&\quad + \tfrac{1}{M\rho} (\mathbb{E}_k(\Upsilon_{k+1}) - \Upsilon_k) + \left(\tfrac{S}{2} - \tfrac{2V_\Upsilon}{\rho M} \right) \left(\Delta_1^{k-1} + \Delta_2^{k-1} \right).
\end{aligned}$$

By definition of S it holds $\left(\frac{2V_\Upsilon}{\rho M} - \frac{S}{2}\right) \geq \varepsilon$. Thus, we get for $\gamma := \frac{1}{2}\min(\epsilon\delta_1, \epsilon\delta_2, \varepsilon)$ that

$$\Psi(u^k) - \mathbb{E}_k\left(\Psi(u^{k+1})\right) \geq 2\gamma \sum_{i=1}^{2} \left(\mathbb{E}_k(\Delta_i^{k+1}) + \Delta_i^k + \Delta_i^{k-1}\right) + \tfrac{1}{M\rho}(\mathbb{E}_k(\Upsilon_{k+1}) - \Upsilon_k).$$

Taking the full expectation yields

$$\mathbb{E}(\Psi(u^k) - \Psi(u^{k+1})) \geq \gamma \mathbb{E}(\|u^{k+1} - u^k\|^2) + \tfrac{1}{M\rho}\mathbb{E}(\Upsilon_{k+1} - \Upsilon_k), \tag{12}$$

and summing up for $k = 1, \ldots, T$,

$$\mathbb{E}(\Psi(u^1) - \Psi(u^{T+1})) \geq \gamma \sum_{k=0}^{T} \mathbb{E}(\|u^{k+1} - u^k\|^2) + \tfrac{1}{M\rho}\mathbb{E}(\Upsilon_{T+1} - \Upsilon_1).$$

Since $\Upsilon_k \geq 0$, this yields

$$\gamma \sum_{k=0}^{T} \mathbb{E}(\|u^{k+1} - u^k\|^2) \leq \Psi(u^1) - \underbrace{\inf_{u \in (\mathbb{R}^{d_1} \times \mathbb{R}^{d_2})^2} \Psi(u)}_{>-\infty} + \underbrace{\tfrac{1}{M\rho}\mathbb{E}(\Upsilon_1)}_{<\infty} < \infty.$$

This finishes the proof. □

Next, we want relate the sequence of iterates generated by iSPALM to the subgradient of the objective function. Such a relation was also established for the (inertial) PALM algorithm. However, due to the stochastic gradient estimator the proof differs significantly from its deterministic counterparts. Note that the convergence analysis of SPRING in [13] does not use the subdifferential but the so-called generalized gradient $\mathscr{G}F_{\tau_1,\tau_2}$. This is not satisfying at all, since it becomes not clear how this generalized gradient is related to the (sub)differential of the objective function in limit processes with varying τ_1 and τ_2. In particular, it is easy to find examples of F and sequences $(\tau_1^k)_k$ and $(\tau_2^k)_k$ such that the generalized gradient $\mathscr{G}F_{\tau_1^k,\tau_2^k}(x_1, x_2)$ is non-zero, but converges to zero for fixed x_1 and x_2.

Theorem 5.5 *Under the assumptions of Theorem 5.4 there exists some $C > 0$ such that*

$$\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2\right) \leq C\mathbb{E}(\|u^{k+1} - u^k\|^2) + 3\mathbb{E}(\Upsilon_k).$$

In particular, it holds

$$\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2\right) \to 0 \quad \text{as } k \to \infty.$$

Proof By definition of x_1^{k+1}, and (1) as well as Proposition 2.3 it holds

$$0 \in \tau_1^k (x_1^{k+1} - y_1^k) + \tilde{\nabla}_{x_1} H(z_1^k, x_2^k) + \partial f(x_1^{k+1}).$$

This is equivalent to

$$\begin{aligned}\tau_1^k (y_1^k - x_1^{k+1}) + \nabla_{x_1} H(x_1^{k+1}, x_2^{k+1}) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k) \\ \in \nabla_{x_1} H(x_1^{k+1}, x_2^{k+1}) + \partial f(x_1^{k+1}) \in \partial_{x_1} F(x_1^{k+1}, x_2^{k+1}).\end{aligned}$$

Analogously we get that

$$\begin{aligned}\tau_2^k (y_2^k - x_2^{k+1}) + \nabla_{x_2} H(x_1^{k+1}, x_2^{k+1}) - \tilde{\nabla}_{x_1} H(x_1^{k+1}, z_2^k) \\ \in \nabla_{x_2} H(x_1^{k+1}, x_2^{k+1}) + \partial g(x_2^{k+1}) \in \partial_{x_2} F(x_1^{k+1}, x_2^{k+1}).\end{aligned}$$

Then we obtain by Proposition 2.3 that

$$v := \begin{pmatrix} \tau_1^k (y_1^k - x_1^{k+1}) + \nabla_{x_1} H(x_1^{k+1}, x_2^{k+1}) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k) \\ \tau_2^k (y_2^k - x_2^{k+1}) + \nabla_{x_2} H(x_1^{k+1}, x_2^{k+1}) - \tilde{\nabla}_{x_1} H(x_1^{k+1}, z_2^k) \end{pmatrix} \in \partial F(x_1^{k+1}, x_2^{k+1}),$$

and it remains to show that the squared norm of v is in expectation bounded by $C\mathbb{E}(\|u^{k+1} - u^k\|^2) + 3\mathbb{E}(\Upsilon_k)$ for some $C > 0$. Using $(a+b+c)^2 \le 3(a^2+b^2+c^2)$ we estimate

$$\begin{aligned}\|v\|^2 &= \|\tau_1^k (y_1^k - x_1^{k+1}) + \nabla_{x_1} H(x_1^{k+1}, x_2^{k+1}) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 \\ &\quad + \|\tau_2^k (y_2^k - x_2^{k+1}) + \nabla_{x_2} H(x_1^{k+1}, x_2^{k+1}) - \tilde{\nabla}_{x_1} H(x_1^{k+1}, z_2^k)\|^2 \\ &\le 3(\tau_1^k)^2 \|y_1^k - x_1^{k+1}\|^2 + 3\|\nabla_{x_1} H(x_1^{k+1}, x_2^{k+1}) - \nabla_{x_1} H(z_1^k, x_2^k)\|^2 \\ &\quad + 3\|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 + 3(\tau_2^k)^2 \|y_2^k - x_2^{k+1}\|^2 \\ &\quad + 3\|\nabla_{x_2} H(x_1^{k+1}, x_2^{k+1}) - \nabla_{x_2} H(x_1^{k+1}, z_2^k)\|^2 \\ &\quad + 3\|\nabla_{x_2} H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\|^2.\end{aligned}$$

Since ∇H is M-Lipschitz continuous and $(a+b)^2 \le 2(a^2+b^2)$, we get further

$$\begin{aligned}\|v\|^2 &\le 12(\tau_1^k)^2 \Delta_1^{k+1} + 6(\tau_1^k)^2 \|y_1^k - x_1^k\|^2 + 12(\tau_2^k)^2 \Delta_2^{k+1} + 6(\tau_2^k)^2 \|y_2^k - x_2^k\|^2 \\ &\quad + 3M^2 \|x_1^{k+1} - z_1^k\|^2 + 6M^2 \Delta_2^{k+1} + 3M^2 \|x_2^{k+1} - z_2^k\|^2 \\ &\quad + 3\left(\|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 + \|\nabla_{x_2} H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\|^2\right).\end{aligned}$$

Using Lemma 5.2 and the fact that $\tilde{\nabla}$ is inertial variance-reduced, this implies

$$\begin{aligned}\|v\|^2 &\le 12(\tau_1^k)^2 \Delta_1^{k+1} + 12(\tau_1^k)^2 (\alpha_1^k)^2 \Delta_1^k + 12(\tau_2^k)^2 \Delta_2^{k+1} + 12(\tau_2^k)^2 (\alpha_2^k)^2 \Delta_2^k \\ &\quad + 12M^2 \Delta_1^{k+1} + 6M^2 \|x_1^k - z_1^k\|^2 + 6M^2 \Delta_2^{k+1} + 12M^2 \Delta_2^{k+1} + 6M^2 \|x_2^k - z_2^k\|^2 \\ &\quad + 3\left(\|\nabla_{x_1} H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1} H(z_1^k, x_2^k)\|^2 + \|\nabla_{x_2} H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2} H(x_1^{k+1}, z_2^k)\|^2\right)\end{aligned}$$

$$
\begin{aligned}
&\leq 12\left((\tau_1^k)^2 + M^2\right)\Delta_1^{k+1} + 12\left((\tau_1^k)^2(\alpha_1^k)^2 + M^2(\beta_1^k)^2\right)\Delta_1^k \\
&\quad + \left(12(\tau_2^k)^2 + 18M^2\right)\Delta_2^{k+1} + 12\left((\tau_2^k)^2(\alpha_2^k)^2 + M^2(\beta_2^k)^2\right)\Delta_2^k \\
&\quad + 3\left(\|\nabla_{x_1}H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1}H(z_1^k, x_2^k)\|^2 + \|\nabla_{x_2}H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2}H(x_1^{k+1}, z_2^k)\|^2\right) \\
&\leq C_0\|u^{k+1} - u^k\|^2 \\
&\quad + 3(\|\nabla_{x_1}H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1}H(z_1^k, x_2^k)\|^2) \\
&\quad + 3(\|\nabla_{x_2}H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2}H(x_1^{k+1}, z_2^k)\|^2),
\end{aligned}
$$

where

$$
C_0 = 12\max\left((\tau_1^k)^2 + M^2, (\tau_1^k)^2(\alpha_1^k)^2 + M^2(\beta_1^k)^2, (\tau_2^k)^2 + \frac{3}{2}M^2, (\tau_2^k)^2(\alpha_2^k)^2 + M^2(\beta_2^k)^2\right).
$$

Noting that $\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2 \leq \|v\|^2$, taking the conditional expectation $\mathbb{E}_k$ and using that $\tilde{\nabla}$ is inertial variance-reduced, we conclude

$$
\begin{aligned}
&\mathbb{E}_k\left(\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2\right) \\
&\leq \mathbb{E}_k\left(C_0\|u^{k+1} - u^k\|^2\right) \\
&\quad + 3\mathbb{E}_k\left(\|\nabla_{x_1}H(z_1^k, x_2^k) - \tilde{\nabla}_{x_1}H(z_1^k, x_2^k)\|^2 + \|\nabla_{x_2}H(x_1^{k+1}, z_2^k) - \tilde{\nabla}_{x_2}H(x_1^{k+1}, z_2^k)\|^2\right) \\
&\leq \mathbb{E}_k\left((C_0 + 3V_1)\|u^{k+1} - u^k\|^2\right) + 3\Upsilon_k.
\end{aligned}
$$

Taking the full expectation on both sides and setting $C := C_0 + 3V_1$ proves the claim. □

Using Theorem 5.5, we can show the sub-linear decay of the expected squared distance of the subgradient to 0.

Theorem 5.6 (Convergence of iSPALM) *Under the assumptions of Theorem 5.4 it holds for t drawn uniformly from $\{2, \ldots, T+1\}$ that there exists some $0 < \sigma < \gamma$ such that*

$$
\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^t, x_2^t))^2\right) \leq \frac{C}{T(\gamma - \sigma)}\left(\Psi(u^1) - \inf_{u \in \mathbb{R}^{d_1} \times \mathbb{R}^{d_2}} \Psi(u) + \left(\frac{3(\gamma-\sigma)}{\rho C} + \frac{1}{M\rho}\right)\mathbb{E}(\Upsilon_1)\right).
$$

Proof By (10), Theorem 5.5 and (12) it holds for $0 < \sigma < \gamma$ that

$$
\begin{aligned}
\mathbb{E}\left(\Psi(u^k) - \Psi(u^{k+1})\right) &\geq \gamma\mathbb{E}(\|u^{k+1} - u^k\|^2) + \tfrac{1}{M\rho}\mathbb{E}\left(\Upsilon_{k+1} - \Upsilon_k\right) \\
&\geq \sigma\mathbb{E}(\|u^{k+1} - u^k\|^2) + \tfrac{\gamma-\sigma}{C}\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2\right) \\
&\quad - \tfrac{3(\gamma-\sigma)}{C}\mathbb{E}(\Upsilon_k) + \tfrac{1}{M\rho}\mathbb{E}\left(\Upsilon_{k+1} - \Upsilon_k\right) \\
&\geq \sigma\mathbb{E}(\|u^{k+1} - u^k\|^2) + \tfrac{\gamma-\sigma}{C}\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2\right) \\
&\quad + \left(\tfrac{3(\gamma-\sigma)}{\rho C} + \tfrac{1}{M\rho}\right)\mathbb{E}\left(\Upsilon_{k+1} - \Upsilon_k\right) - \tfrac{3(\gamma-\sigma)V_\Upsilon}{C\rho}\mathbb{E}(\|u^{k+1} - u^k\|^2).
\end{aligned}
$$

Choosing $\sigma := \frac{3(\gamma-\sigma)V_\Upsilon}{C\rho}$ yields

$$\mathbb{E}\left(\Psi(u^k)-\Psi(u^{k+1})\right) \geq \tfrac{\gamma-\sigma}{C}\mathbb{E}\left(\operatorname{dist}(0,\partial F(x_1^{k+1},x_2^{k+1}))^2\right)-\left(\tfrac{3(\gamma-\sigma)}{\rho C}+\tfrac{1}{M\rho}\right)\mathbb{E}\left(\Upsilon_{k+1}-\Upsilon_k\right).$$

Adding this up for $k = 1, \dots, T$ we get

$$\begin{aligned}\mathbb{E}\left(\Psi(u^1)-\Psi(u^T)\right) &\geq \tfrac{\gamma-\sigma}{C}\sum_{k=1}^{T}\mathbb{E}\left(\operatorname{dist}(0,\partial F(x_1^{k+1},x_2^{k+1}))^2\right)\\ &\quad+\left(\tfrac{3(\gamma-\sigma)}{\rho C}+\tfrac{1}{M\rho}\right)\mathbb{E}\left(\Upsilon_T-\Upsilon_1\right).\end{aligned}$$

Since $\Upsilon_T \geq 0$ this yields for t drawn randomly from $\{2, \dots, T+1\}$ that

$$\begin{aligned}\mathbb{E}\left(\operatorname{dist}(0,\partial F(x_1^t,x_2^t))^2\right) &= \tfrac{1}{T}\sum_{k=1}^{T}\mathbb{E}\left(\operatorname{dist}(0,\partial F(x_1^{k+1},x_2^{k+1}))^2\right)\\ &\leq \tfrac{C}{T(\gamma-\sigma)}\left(\Psi(u^1)-\inf_{u\in(\mathbb{R}^{d_1}\times\mathbb{R}^{d_2})^2}\Psi(u)+\left(\tfrac{3(\gamma-\sigma)}{\rho C}+\tfrac{1}{M\rho}\right)\mathbb{E}(\Upsilon_1)\right).\end{aligned}$$

This finishes the proof. □

In [13] the authors proved global convergence of the objective function evaluated at the iterates of SPRING in expectation if the global error bound

$$F(x_1,x_2)-\underline{F} \leq \mu \operatorname{dist}(0,\partial F(x_1,x_2))^2, \quad \text{for all } x_1\in\mathbb{R}^{d_1}, x_2\in\mathbb{R}^{d_2} \tag{13}$$

is fulfilled for some $\mu > 0$. Using this error bound, we can also prove global convergence of iSPALM in expectation with a linear convergence rate. Note that the authors of [13] used the generalized gradient instead of the subgradient also for this error bound. Similar as before this seems to be unsuitable due to the heavy dependence on of the generalized gradient on the step size parameters.

Theorem 5.7 (Convergence of iSPALM) *Let the assumptions of Theorem 5.4 hold true. If in addition (13) is fulfilled, then there exists some $\Theta_0 \in (0,1)$ and $\Theta_1 > 0$ such that*

$$\mathbb{E}\left(F(x_1^{T+1},x_2^{T+1})-\underline{F}\right) \leq (\Theta_0)^T\left(\Psi(u^1)-\underline{F}+\Theta_1\mathbb{E}(\Upsilon_1)\right).$$

In particular, it holds $\lim_{T\to\infty}\mathbb{E}(F(x_1^T,x_2^T)-\underline{F})=0$.

Proof By (12) and Theorem 5.5, we obtain for $0 < d < \min(\gamma, \frac{C\rho\mu}{1-\rho})$ that

$$\begin{aligned}\mathbb{E}\left(\Psi(u^{k+1})-\underline{F}+\tfrac{1}{M\rho}\Upsilon_{k+1}\right) &\leq \mathbb{E}\left(\Psi(u^k)-\underline{F}+\tfrac{1}{M\rho}\Upsilon_k\right)-\gamma\mathbb{E}(\|u^{k+1}-u^k\|^2)\\ &\leq \mathbb{E}\left(\Psi(u^k)-\underline{F}+\tfrac{1}{M\rho}\Upsilon_k\right)\end{aligned}$$

$$- \tfrac{d}{C}\mathbb{E}\left(\operatorname{dist}(0, \partial F(x_1^{k+1}, x_2^{k+1}))^2\right) + \tfrac{3d}{C}\mathbb{E}(\Upsilon_k)$$
$$- (\gamma - d)\,\mathbb{E}(\|u^{k+1} - u^k\|^2).$$

Using (10) in combination with the global error bound (13), we get

$$\mathbb{E}\left(\Psi(u^{k+1}) - \underline{F} + \left(\tfrac{3d}{\rho C} + \tfrac{1}{M\rho}\right)\Upsilon_{k+1}\right) \le \mathbb{E}\left(\Psi(u^k) - \underline{F} + \left(\tfrac{3d}{\rho C} + \tfrac{1}{M\rho}\right)\Upsilon_k\right)$$
$$- \tfrac{d}{C\mu}\mathbb{E}\left(F(x_1^{k+1}, x_2^{k+1}) - \underline{F}\right) - \left(\gamma - d - \tfrac{3dV_\Upsilon}{\rho C}\right)\mathbb{E}(\|u^{k+1} - u^k\|^2).$$

Setting $C_\Upsilon := \left(\frac{3d}{\rho C} + \frac{1}{M\rho}\right)$ and applying the definition (7) of Ψ, this implies

$$\left(1 + \tfrac{d}{C\mu}\right)\mathbb{E}\left(\Psi(u^{k+1}) - \underline{F}\right) - \tfrac{d}{C\mu}\mathbb{E}(\delta_1\Delta_1^{k+1} + \delta_2\Delta_2^{k+1}) + C_\Upsilon\mathbb{E}(\Upsilon_{k+1})$$
$$\le \mathbb{E}\left(\Psi(u^k) - \underline{F}\right) + C_\Upsilon\mathbb{E}(\Upsilon_k) - \left(\gamma - d - \tfrac{3dV_\Upsilon}{\rho C}\right)\mathbb{E}(\|u^{k+1} - u^k\|^2).$$

With $\delta := \max(\delta_1, \delta_2)$ and $\Delta_1^{k+1} + \Delta_2^{k+1} \le \frac{1}{2}\|u^{k+1} - u^k\|^2$ we get

$$\left(1 + \tfrac{d}{C\mu}\right)\mathbb{E}\left(\Psi(u^{k+1}) - \underline{F}\right) + C_\Upsilon\mathbb{E}(\Upsilon_{k+1})$$
$$\le \mathbb{E}\left(\Psi(u^k) - \underline{F}\right) + C_\Upsilon\mathbb{E}(\Upsilon_k) - \left(\gamma - d - \tfrac{3dV_\Upsilon}{\rho C} - \tfrac{d\delta}{2C\mu}\right)\mathbb{E}(\|u^{k+1} - u^k\|^2).$$

Multiplying by $C_d := \frac{1}{1+\frac{d}{C\mu}} = \frac{C\mu}{C\mu+d}$ this becomes

$$\mathbb{E}\left(\Psi(u^{k+1}) - \underline{F}\right) + C_\Upsilon C_d\mathbb{E}(\Upsilon_{k+1}) \le \tfrac{C\mu}{C\mu+d}\mathbb{E}\left(\Psi(u^k) - \underline{F}\right) + C_\Upsilon C_d\mathbb{E}(\Upsilon_k)$$
$$- \tfrac{C\mu}{C\mu+d}\left(\gamma - d - \tfrac{3dV_\Upsilon}{\rho C} - \tfrac{d\delta}{2C\mu}\right)\mathbb{E}(\|u^{k+1} - u^k\|^2). \tag{14}$$

Since $d < \frac{C\rho\mu}{1-\rho}$ we know that $s := \frac{1-C_d}{C_d+\rho-1} = \frac{d}{\rho C\mu+(\rho-1)d} > 0$. Thus, adding $sC_\Upsilon C_d$ times equation Definition 4.1 (ii) to (14) gives

$$\mathbb{E}\left(\Psi(u^{k+1}) - \underline{F}\right) + (1+s)C_\Upsilon C_d\mathbb{E}(\Upsilon_{k+1}) \le C_d\mathbb{E}\left(\Psi(u^k) - \underline{F} + (1+s)C_\Upsilon C_d\mathbb{E}(\Upsilon_k)\right)$$
$$+ C_d\underbrace{\left(V_\Upsilon sC_\Upsilon - \left(\gamma - d - \tfrac{3dV_\Upsilon}{\rho C} - \tfrac{d\delta}{2C\mu}\right)\right)}_{=:h(d)}\mathbb{E}(\|u^{k+1} - u^k\|^2),$$

where we have used that $1 + (1-\rho)s = C_d(1+s)$. Since s converges to 0 as $d \to 0$ we have that $\lim_{d\to 0} h(d) = -\gamma$. Thus we can choose $d > 0$ small enough, such that $h(d) < 0$. Then we get

$$\mathbb{E}\left(\Psi(u^{k+1}) - \underline{F}\right) + (1+s)C_\Upsilon C_d\mathbb{E}(\Upsilon_{k+1}) \le C_d\mathbb{E}\left(\Psi(u^k) - \underline{F} + (1+s)C_\Upsilon C_d\mathbb{E}(\Upsilon_k)\right).$$

Finally, setting $\Theta_0 := C_d$ and $\Theta_1 := (1+s)C_\Upsilon C_d$ and applying the last equation iteratively, we obtain

$$\mathbb{E}\left(\Psi(u^{T+1}) - \underline{F} + \Theta_1 \Upsilon_{T+1}\right) \leq (\Theta_0)^T \mathbb{E}\left(\Psi(u^1) - \underline{F} + \Theta_1 \Upsilon_1\right).$$

Note that $\Psi(u^{T+1}) \geq F(x_1^{T+1}, x_2^{T+1})$ and that $\Upsilon_{T+1} \geq 0$. This yields

$$\mathbb{E}\left(F(x_1^{T+1}, x_2^{T+1}) - \underline{F}\right) \leq (\Theta_0)^T \mathbb{E}\left(\Psi(u^1) - \underline{F} + \Theta_1 \Upsilon_1\right),$$

and we are done. □

6 Numerical results

In this section, we demonstrate the performance of iSPALM for two different applications, namely for learning (i) the parameters of Student-t MMs, and (ii) the weights of PNNs. In comparison with PALM, iPALM and SPRING, we will see that our algorithm increases the stability of SPRING and iSPALM if we enforce the evaluation of the full gradient at the beginning of each epoch. We will exclusively use the SARAH estimator.

We run all our experiments on a Lenovo ThinkStation with Intel i7-8700 processor, 32GB RAM and a NVIDIA GeForce GTX 2060 Super GPU. For the implementation we use Python and Tensorflow.

6.1 Parameter choice and implementation aspects

On the one hand, the algorithms based on PALM have many parameters which enables a high adaptivity of the algorithms to the specific problems. On the other hand, it is often hard to fit these parameters to ensure the optimal performance of the algorithms.

Based on approximations $\tilde{L}_1(x_2^k)$ and $\tilde{L}_2(x_1^{k+1})$ of the partial Lipschitz constants $L_1(x_2^k)$ and $L_2(x_1^{k+1})$ outlined below, we use the following step size parameters τ_i^k, $i = 1, 2$:

- For **PALM** and **iPALM**, we choose $\tau_1^k = \tilde{L}_1(x_1^k, x_2^k)$ and $\tau_2^k = \tilde{L}_2(x_1^{k+1}, x_2^k)$ which was also suggested in [4, 36].
- For **SPRING** and **iSPALM**, we choose $\tau_1^k = s_1 \tilde{L}_1(x_1^k, x_2^k)$ and $\tau_2^k = s_1 \tilde{L}_2(x_1^{k+1}, x_2^k)$, where the manually chosen scalar $s_1 > 0$ depends on the application. Note that the authors in [13] propose to take $s_1 = 2$ which was not optimal in our examples.

Computation of Gradients and Approximative Lipschitz Constants Since the global and partial Lipschitz constants of the block-wise gradients of H are usually unknown, we estimate them locally using the second order derivative of H which exists in our examples. If H acts on a high dimensional space, it is often computationally to costly to compute the full Hessian matrix. Thus we compute a local Lipschitz constant only

in the gradient direction, i.e. we compute

$$\tilde{L}_i(x_1, x_2) := \|\nabla^2_{x_i} H(x_1, x_2) g\|, \quad g := \frac{\nabla_{x_i} H(x_1, x_2)}{\|\nabla_{x_i} H(x_1, x_2)\|} \tag{15}$$

For the stochastic algorithms we replace H by the approximated function $\tilde{H}(x_1, x_2) := \frac{1}{b}\sum_{i \in B_i^k} h_i(x_1, x_2)$, where B_i^k is the current mini-batch. The analytical computation of $\tilde{L}_i$ in (15) is still hard. Even computing the gradient of a complicated function H can be error prone and laborious. Therefore, we compute the (partial) gradients of H or $\tilde{H}$, respectively, using the reverse mode of algorithmic differentiation (also called backpropagation), see e.g. [16]. To this end, note that the chain rule yields that

$$\begin{aligned}\left\|\nabla_{x_i}\left(\|\nabla_{x_i} H(x_1, x_2)\|^2\right)\right\| &= \left\|2\|\nabla_{x_i} H(x_1, x_2)\|\nabla^2_{x_i} H(x_1, x_2)\nabla_{x_i} H(x_1, x_2)\right\| \\ &= 2\|\nabla_{x_i} H(x_1, x_2)\|^2 \tilde{L}_i(x_1, x_2).\end{aligned}$$

Thus, we can compute $\tilde{L}_i(x_1, x_2)$ by applying two times the reverse mode. If we neglect the taping, the execution time of this procedure can provably be bounded by a constant times the execution time of H, see [16,Sect. 5.4]. Therefore, this procedure gives us an accurate and computationally very efficient estimation of the local partial Lipschitz constant.

Inertial Parameters For the iPALM and iSPALM we have to choose the inertial parameters $\alpha_i^k \geq 0$ and $\beta_i^k \geq 0$. With respect to our convergence results we have to assume that there exist $\alpha_i^k \leq \bar{\alpha}_i < \frac{1}{2}$ and $\beta_i^k \leq \bar{\beta}_i < 1$, $i = 1, 2$. Note that for convex functions f and g, the authors in [36] proved that the assumption on the α's can be lowered to $\alpha_i^k \leq \bar{\alpha}_i < 1$ and suggested to use $\alpha_i^k = \beta_i^k = \frac{k-1}{k+2}$. Unfortunately, we cannot show this for iSPALM and indeed we observe instability and divergence in iSPALM, if we choose $\alpha_i^k > \frac{1}{2}$. Therefore, we choose for iSPALM the parameters

$$\alpha_i^k = \beta_i^k = s_2 \frac{k-1}{k+2},$$

where the scalar $0 < s_2 < 1$ is manually chosen depending on the application.

Implementation We provide a general framework for implementing PALM, iPALM, SPRING and iSPALM[2] on a GPU. Using this framework, it suffice to provide an implementation for the functions H and $\text{prox}_{\tau_i}^{f_i}$ in order to use one of the above algorithms for the function $F(x_1, \ldots, x_K) = H(x_1, \ldots, x_K) + \sum_{i=1}^K f_i(x_i)$. We provide also the code of our numerical examples below on this website.

[2] https://github.com/johertrich/Inertial-Stochastic-PALM.

6.2 Student-*t* mixture models

First, we apply the various PALM algorithms for estimating the parameters of d-dimensional Student-t MMs with K components. More precisely, we aim to find $\alpha = (\alpha_1, \dots, \alpha_K) \in \Delta_K := \{(\alpha_k)_{k=1}^K : \sum_{k=1}^K \alpha_k = 1,\ \alpha_k \geq 0\}$, $\nu = (\nu_1, \dots, \nu_K) \in \mathbb{R}^K_{>0}$, $\mu = (\mu_1, \dots, \mu_K) \in \mathbb{R}^K$, and $\Sigma = (\Sigma_1, \dots, \Sigma_K) \in \mathrm{SPD}(d)^K$ in the probability density function

$$p(x) = \sum_{k=1}^K \alpha_k f(x|\nu_k, \mu_k, \Sigma_k).$$

Here $\mathrm{SPD}(d)$ denotes the symmetric positive definite $d \times d$ matrices, and f is the density function of the Student-t distribution with $\nu > 0$ degrees of freedom, *location* parameter $\mu \in \mathbb{R}^d$ and *scatter matrix* $\Sigma \in \mathrm{SPD}(d)$ given by

$$f(x|\nu, \mu, \Sigma) = \frac{\Gamma\left(\frac{d+\nu}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right) \nu^{\frac{d}{2}} \pi^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \frac{1}{\left(1 + \frac{1}{\nu}(x-\mu)^\mathrm{T} \Sigma^{-1}(x-\mu)\right)^{\frac{d+\nu}{2}}}$$

with the Gamma function Γ.

For samples $\mathcal{X} = (x_1, \dots, x_n) \in \mathbb{R}^{d \times n}$, we want to minimize the negative log-likelihood function

$$\mathcal{L}(\alpha, \nu, \mu, \Sigma|\mathcal{X}) = -\frac{1}{n} \sum_{i=1}^n \log\left(\sum_{k=1}^K \alpha_k f(x_i|\nu_k, \mu_k, \Sigma_k)\right)$$

subject to the parameter constraints. A first idea to rewrite this problem in the form (2) looks as

$$F(\alpha, \nu, \mu, \Sigma) = H(\alpha, \nu, \mu, \Sigma) + f_1(\alpha) + f_2(\nu) + f_3(\mu) + f_4(\Sigma), \tag{16}$$

where $H := \mathcal{L}$, $f_1 := \iota_{\Delta_K}$, $f_2 := \iota_{\mathbb{R}^K_{>0}}$, $f_3 := 0$, $f_4 := \iota_{\mathrm{SPD}(d)^K}$, and $\iota_\mathcal{S}$ denotes the *indicator function* of the set $\mathcal{S}$ defined by $\iota_\mathcal{S}(x) := 0$ if $x \in \mathcal{S}$ and $\iota_\mathcal{S}(x) := \infty$ otherwise. Indeed one of the authors has applied PALM and iPALM to such a setting without any convergence guarantee in [19]. The problem is that $\mathcal{L}$ is not defined on the whole Euclidean space and since $\mathcal{L}(\alpha, \nu, \mu, \Sigma) \to \infty$ as $\Sigma_k \to 0$ for some k, the function can also not continuously extended to the whole $\mathbb{R}^K \times \mathbb{R}^K \times \mathbb{R}^{d \times K} \times \mathrm{Sym}(d)^K$, where $\mathrm{Sym}(d)$ denotes the space of symmetric $d \times d$ matrices. Furthermore, the functions f_2 and f_4 are not lower semi-continuous. Consequently, the function (16) does not fulfill the assumptions required for the convergence of PALM and iPALM. Therefore we modify the above model as follows: Let $\mathrm{SPD}_\epsilon(d) := \{\Sigma \in \mathrm{SPD}(d) : \Sigma \succeq \epsilon I_d\}$. Then we use the surjective mappings $\varphi_1 \colon \mathbb{R}^K \to \Delta_K$, $\varphi_2 \colon \mathbb{R}^K \to \mathbb{R}^K_{\geq \epsilon}$

and $\varphi_3 \colon \mathrm{Sym}(d)^K \to \mathrm{SPD}_\varepsilon(d)^K$ defined by

$$\varphi_1(\alpha) := \frac{\exp(\alpha)}{\sum_{j=1}^K \exp(\alpha_j)}, \quad \varphi_2(v) := v^2 + \epsilon, \quad \varphi_3(\Sigma) := \left(\Sigma_k^T \Sigma_k + \epsilon I_d\right)_{k=1}^K$$

to reshape problem (16) as the unconstrained optimization problem

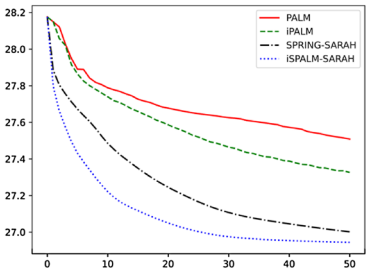
$$\underset{\alpha \in \mathbb{R}^K, v \in \mathbb{R}^K, \mu \in \mathbb{R}^{d \times K}, \Sigma \in \mathrm{Sym}(d)^K}{\operatorname{argmin}} H(\alpha, v, \mu, \Sigma) := \mathcal{L}(\varphi_1(\alpha), \varphi_2(v), \mu, \varphi_3(\Sigma) | \mathcal{X}). \tag{17}$$

For this problem, PALM and iPALM reduce basically to block gradient descent algorithms. In Appendix B, we verify that the above function H is indeed a KL function which is bounded from below, and satisfies the Assumption 3.1(i). Since $H \in C^2(\mathbb{R}^K \times \mathbb{R}^K \times \mathbb{R}^{d \times K} \times \mathrm{Sym}(d)^K)$, we know by Remark 3.2 that Assumption 3.1(ii) is also fulfilled. Further, ∇H is continuous on bounded sets. Then, choosing the parameters of PALM, resp. iPALM as required by Theorem 3.3 resp. 3.6, we conclude that the sequences generated by both algorithms converge to a critical point of H supposed that they are bounded. Similarly, if we assume in addition that the stochastic gradient estimators are inertial variance-reduced, we can conclude that the iSPALM sequence converges as in Theorems 5.6 and 5.7, if the corresponding requirements on the parameters are fulfilled.

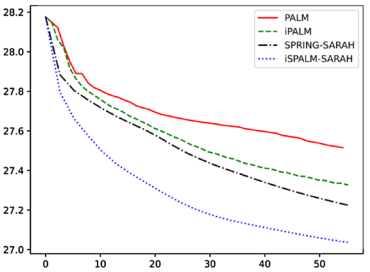
In our numerical examples, we generate the data by sampling from the Student-t MM, where the parameters of the ground truth MM are generate as follows:

- We generate $\alpha = \frac{\bar{\alpha}^2+1}{\|\bar{\alpha}^2+1\|_1}$, where the entries of $\bar{\alpha} \in \mathbb{R}^K$ are drawn independently from the standard normal distribution.
- We generate $v_i = \min(\bar{v}_i^2 + 1, 100)$, where $\bar{v}_i$, $i = 1, \ldots, n$ is drawn from a normal distribution with mean 0 and standard deviation 10.
- The entries of $\mu \in \mathbb{R}^{d \times K}$ are drawn independently from a normal distribution with mean 0 and standard deviation 2.
- We generate $\Sigma_i = \bar{\Sigma}_i^T \bar{\Sigma}_i + I$, where the entries of $\bar{\Sigma}_i \in \mathbb{R}^{d \times d}$ are drawn independently from the standard normal distribution.

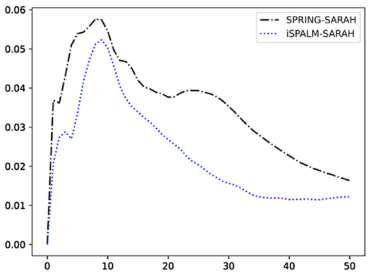
For the initialization of the algorithms, we assign to each sample x_i randomly a class $c_i \in \{1, \ldots, K\}$. Then we initialize the parameters (v_k, μ_k, Σ_k) by estimating the parameters of a Student-t distribution of all samples with $c_i = k$ using a faster alternative of the EM algorithm called multivariate myriad filter, see [17]. Further we initialize α by $\alpha_k = \frac{|\{i \in \{1,\ldots,N\} : c_i = k\}|}{N}$. We run the algorithm for $n = 200{,}000$ data points of dimension $d = 10$ and $K = 30$ components. We use a batch size of $b = 20{,}000$. To represent the randomness in SPRING and iSPALM, we repeat the experiment 10 times with the same samples and the same initialization. The resulting mean and standard deviation of the negative log-likelihood values versus the number of epochs and the execution times, respectively, are given in Fig. 1. Further, we visualize the mean squared norm of the gradient after each epoch. One epoch contains for SPRING and iSPALM 10 steps and for PALM and iPALM 1 step. We see that in terms of the number of epochs as well as in terms of the execution time the iSPALM is the fastest algorithm.



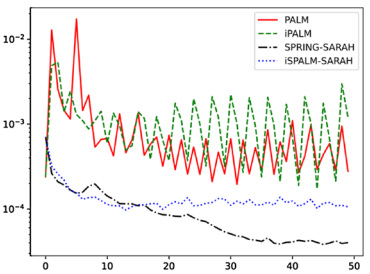
(a) Average objective versus epochs



(b) Average objective versus execution time



(c) Standard deviation of the objective versus epochs



(d) Average squared norm of the gradient versus epochs

Fig. 1 Objective function versus number of epochs and versus execution time for estimating the parameters of Student-t MMs

6.3 Proximal neural networks (PNNs)

PNNs for MNIST classification In this example, we train a Proximal Neural Network as introduced in [18] for classification on the MNIST data set.[3] The training data consists of $N = 60{,}000$ images $x_i \in \mathbb{R}^d$ of size $d = 28^2$ and labels $y_i \in \{0, 1\}^{10}$, where the jth entry of y_i is 1 if and only if x_i has the label j. A PNN with $K - 1$ layers and activation function σ is defined by

$$T_{K-1}^{\mathrm{T}} \sigma (T_{K-1} \cdots T_1^{\mathrm{T}} \sigma (T_1 x + b_1) \cdots + b_{K-1}),$$

where the T_i are contained in the (compact) Stiefel manifold $\mathrm{St}(d, n_i)$ and $b_i \in \mathbb{R}^{n_i}$ for $i = 1, \ldots, K - 1$. To get 10 output elements in (0, 1), we add similar as in [18] an additional layer

$$g(T_K x), \quad T_K \in [-10, 10]^{10,d}, b_K \in \mathbb{R}^{10}$$

[3] http://yann.lecun.com/exdb/mnist.

with the activation function $g(x) := \frac{1}{1+\exp(-x)}$. Thus the full network is given by

$$\Psi(x,u) = g(T_K T_{K-1}^{\mathrm{T}} \sigma(T_{K-1} \cdots T_1^{\mathrm{T}} \sigma(T_1 x + b_1) + \cdots + b_{K-1}) + b_K),$$
$$u = (T_1, \ldots, T_K, b_1, \ldots, b_K).$$

It was demonstrated in [18] that this kind of network is more stable under adversarial attacks than the same network without the orthogonality constraints.

Training PNNs with iSPALM Now, we want to train a PNN with $K - 1 = 3$ layers and $n_1 = 784, n_2 = 400$ and $n_3 = 200$ for MNIST classification. In order of applying our theory, we use the exponential linear unit (ELU)

$$\sigma(x) = \begin{cases} \exp(x) - 1, & \text{if } x < 0, \\ x & \text{if } x \geq 0, \end{cases}$$

as activation function, which is differentiable with a 1-Lipschitz gradient. Then, the loss function is given by

$$F(u) = H(u) + f(u), \quad u = (T_1, \ldots, T_4, b_1, \ldots, b_4)$$

where $T_i \in \mathbb{R}^{d,n_i}$, $b_i \in \mathbb{R}^{n_i}$, $i = 1, 2, 3$, and $T_4 \in [-10, 10]^{10,d}$, $b_4 \in \mathbb{R}^{10}$, and $f(u) = \iota_{\mathcal{U}}$ with

$$\mathcal{U} := \{(T_1, \ldots, T_4, b_1, \ldots, b_4) : T_i \in \mathrm{St}(d, n_i), i = 1, 2, 3, T_4 \in [-10, 10]^{10,d}\}.$$

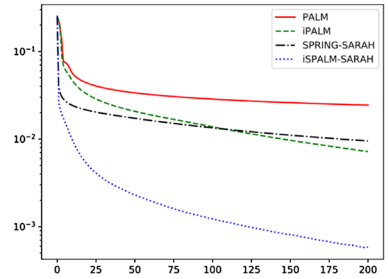
and

$$H(u) := \frac{1}{N} \sum_{i=1}^{N} \|\Psi(x_i, u) - y_i\|^2.$$

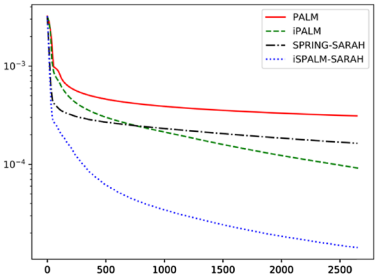
Since H is unfortunately not Lipschitz continuous, we propose a slight modification. Note that for any $u = (T_1, \ldots, T_4, b_1, \ldots, b_4)$ which appears as x^k, y^k or z^k in PALM, iPALM, SPRING or iSPALM we have that there exist $v, w \in \mathcal{U}$ such that $u = v + w$. In particular, we have that $\|T_i\|_F \leq 2\sqrt{d}$, $i = 1, 2, 3$ and $\|T_4\|_F \leq 20\sqrt{10d}$ Therefore, we can replace H by

$$\tilde{H}(u) = \Pi_{i=1}^{4} \eta(\|T_i\|_F^2) \frac{1}{N} \sum_{i=1}^{N} \|\Psi(x_i, u) - y_i\|^2,$$

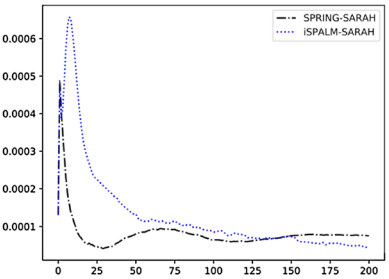
without changing the algorithm, where η is a smooth cutoff function of the interval $(-\infty, 4000d]$. Now, simple calculations yield that the function $\tilde{H}$ is globally Lipschitz continuous. Since it is also bounded from below by 0 we can conclude that our convergence results of iSPALM are applicable.



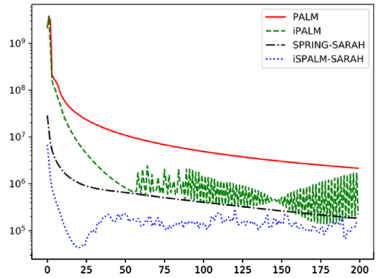
(a) Average loss versus epochs on the training set.



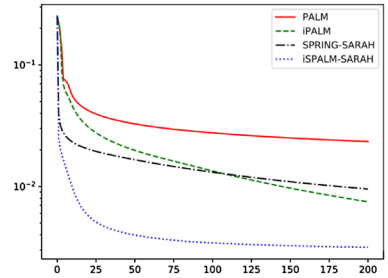
(b) Average loss versus execution time on the training set.



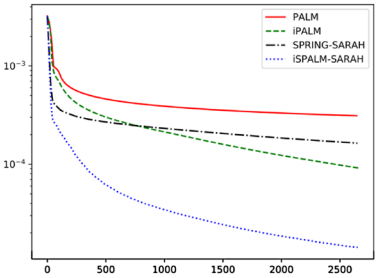
(c) Standard deviation of the loss versus execution time on the training set.



(d) Riemannian gradient of the loss versus epochs on the training set.



(e) Average loss versus epochs on the test set.



(f) Average loss versus execution time on the test set.

Fig. 2 Loss function versus number of epochs and versus execution time for training a PNN for MNIST classification

Remark 6.1 For the implementation, we need to calculate $\operatorname{prox}_{\tilde{f}}$, which is the orthogonal projection $P_{\mathcal{U}}$ onto $\mathcal{U}$. This includes the projection of the matrices $T_i, i = 1, 2, 3$ onto the Stiefel manifold. In [23,Sect. 7.3,7.4] it is shown, that the projection of a matrix A onto the Stiefel manifold is given by the U-factor of the polar decomposition $A = US \in \mathbb{R}^{d,n}$, where $U \in \operatorname{St}(d, n)$ and S is symmetric and positive definite. Note that U is only unique, if A is non-singular. Several possibilities for the computing U

are considered in [22,Chapter 8]. In particular, U is given by VW, where $A = V\Sigma W$ is the singular value decomposition of A. For our numerical experiments we use the iteration

$$Y_{k+1} = 2Y_k(I + Y_k^{\mathrm{T}} Y_k)^{-1}$$

with $Y_0 = A$, which converges for any non-singular A to U, see [22]. □

Now we run PALM, iPALM and SPRING, iSPRING for 200 epochs using a batch size of $b = 1500$. One epoch contains for SPRING and iSPALM 40 steps and for PALM and iPALM 1 step. As in the previous example we repeat the experiment 10 times with the same initialization and plot for the resulting loss functions mean and standard deviation to represent the randomness of the algorithms. Figure 2 shows the mean and standard deviation of the loss versus the number of epochs or the execution time as well as the squared norm of the Riemannian gradient for the iterates of iSPALM after each epoch. We observe that iSPALM performs much better than SPRING and that iPALM performs much better than PALM. Therefore this example demonstrates the importance of the inertial parameters in iPALM and iSPALM. Further, iSPALM and SPRING outperform their deterministic versions significantly. The resulting weights from iSPALM reach after 200 epochs an average accuracy of 0.985 on the test set.

7 Conclusions

We combined a stochastic variant of the PALM algorithm with the inertial PALM algorithm to a new algorithm, called iSPALM. We analyzed the convergence behavior of iSPALM and proved convergence results, if the gradient estimators are inertial variance-reduced. In particular, we showed that the expected distance of the subdifferential to zero converges to zero for the sequence of iterates generated by iSPALM. Additionally, the sequence of function values achieves linear convergence for functions satisfying a global error bound. We proved that a modified version of the negative log-likelihood function of Student-t MMs fulfills all necessary convergence assumption of PALM, iPALM. We demonstrated the performance of iSPALM for two quite different applications. In the numerical comparison, it turns out that iSPALM shows the best performance of all four algorithms. In particular, the example with the PNNs demonstrates the importance of combining inertial parameters and stochastic gradient estimators.

In future work, it would be interesting to compare the performance of the iSPALM algorithm with more classical algorithms for estimating the parameters of Student-t MMs, in particular with the EM algorithm and some of its accelerations. For first experiments in this direction we refer to our work [17, 19]. Further, we intend to apply iSPALM to other practical problems, e.g. in more sophisticated examples of deep learning.

Acknowledgements The authors want to thank T. Pock (TU Graz) for fruitful discussions on iPALM.

Funding Open Access funding enabled and organized by Projekt DEAL. Funding by the German Research Foundation (DFG) within the project STE 571/16-1 is gratefully acknowledged.

A KL functions

Let us recall the notation of Kurdyka-Łojasiewicz functions. For $\eta \in (0, \infty]$, we denote by Φ_η the set of all concave continuous functions $\phi\colon [0, \eta) \to \mathbb{R}_{\geq 0}$ which fulfill the following properties:

(i) $\phi(0) = 0$.
(ii) ϕ is continuously differentiable on $(0, \eta)$.
(iii) For all $s \in (0, \eta)$ it holds $\phi'(s) > 0$.

Definition A.1 (*Kurdyka-Lojasiewicz property)* A proper, lower semicontinuous function $\sigma\colon \mathbb{R}^d \to (-\infty, +\infty]$ has the Kurdyka-Łojasieweicz (KL) property at $\bar{u} \in \operatorname{dom} \partial\sigma = \{u \in \mathbb{R}^d : \partial\sigma \neq \emptyset\}$ if there exist $\eta \in (0, \infty]$, a neighborhood U of $\bar{u}$ and a function $\phi \in \Phi_\eta$, such that for all

$$u \in U \cap \{v \in \mathbb{R}^d : \sigma(\bar{u}) < \sigma(v) < \sigma(\bar{u}) + \eta\},$$

it holds

$$\phi'(\sigma(u) - \sigma(\bar{u}))\operatorname{dist}(0, \partial\sigma(u)) \geq 1.$$

We say that σ is a KL function, if it satisfies the KL property in each point $u \in \operatorname{dom} \partial\sigma$.

B Properties of the objective function in MMs

We start with the KL property.

Lemma B.1 *The function $H : \mathbb{R}^K \times \mathbb{R}^K \times \mathbb{R}^{d\times K} \times \operatorname{Sym}(d)^K \to \mathbb{R}$ defined in (17) is a KL function. Moreover, it is bounded from below.*

Proof 1. Since the Gamma function is real analytic, we have that H is a combination of sums, products, quotients and concatenations of real analytic functions. Thus H is real analytic. This implies that it is a KL function, see [1,Example 1] and [28, 29].

2. First, we proof that $f(x|\nu, \mu, \Sigma)$ is bounded from above for $\nu > \epsilon$, $\mu \in \mathbb{R}^d$ and $\Sigma \succeq \epsilon I_d$. By definition of the Gamma function and since

$$\Gamma(\tfrac{\nu+d}{2})/\Gamma(\tfrac{\nu}{2})\nu^{\frac{d}{2}} \to 1 \quad \text{as} \quad \nu \to \infty \tag{18}$$

we have that (18) is bounded from below for $\nu \in [\epsilon, \infty)$. Further, we see by assumptions on ν and Σ that $|\Sigma|^{-\frac{1}{2}} \leq \epsilon^{-\frac{d}{2}}$ and $\left(1 + \frac{1}{\nu}(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)\right)^{-\frac{d+\nu}{2}} \leq 1$. Thus, $f(x|\nu, \mu, \Sigma)$ is the product of bounded functions and therefore itself bounded by some $C > 0$. This yields for $\tilde{\alpha} = \varphi_1(\alpha)$, $\tilde{\nu} = \varphi_2(\nu)$ and $\tilde{\Sigma} = \varphi_3(\Sigma)$ that

$$-\sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} \tilde{\alpha}_k f(x_i|\tilde{\nu}_k, \tilde{\mu}_k, \tilde{\Sigma}_k)\right) \leq -\sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} \tilde{\alpha}_k C\right) \leq -n \log C,$$

which finishes the proof.

□

Next we state the Lipschitz properties of H.

Lemma B.2 *For $H : \mathbb{R}^K \times \mathbb{R}^K \times \mathbb{R}^{d\times K} \times \mathrm{Sym}(d)^K \to \mathbb{R}$ defined by (17) and all $\alpha \in \mathbb{R}^K$, $\nu \in \mathbb{R}^K$, $\mu \in \mathbb{R}^{d\times K}$ and $\Sigma \in \mathbb{R}^{d\times d\times K}$ we have that the gradients $\nabla_\alpha H(\cdot, \nu, \mu, \Sigma)$, $\nabla_\nu H(\alpha, \cdot, \mu, \Sigma)$, $\nabla_\mu H(\alpha, \nu, \cdot, \Sigma)$, and $\nabla_\Sigma H(\alpha, \nu, \mu, \cdot)$ are globally Lipschitz continuous.*

The proof follows by straightforward computation. The technical details, in particular the computation of the outer derivatives of the objective function in (17) can be found in [21].

References

1. Attouch, H., Bolte, J.: On the convergence of the proximal algorithm for nonsmooth functions involving analytic features. Mathematical Programming. A Publication of the Mathematical Programming Society **116**(1-2, Ser. B), 5–16 (2009)
2. Attouch, H., Bolte, J., Redont, P., Soubeyran, A.: Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the Kurdyka-łojasiewicz inequality. Math. Oper. Res. **35**(2), 438–457 (2010)
3. Banerjee, A., Maji, P.: Spatially constrained Student's t-distribution based mixture model for robust image segmentation. J. Math. Imaging Vis. **60**(3), 355–381 (2018)
4. Bolte, J., Sabach, S., Teboulle, M.: Proximal alternating linearized minimization for nonconvex and nonsmooth problems. Math. Program. **146**(1–2, Ser. A), 459–494 (2014)
5. Bottou, L.: In: Large-scale machine learning with stochastic gradient descent, In Proceedings of COMPSTAT'2010, volume 1, pp. 177–186. Springer, (2010)
6. Byrne, C.L.: The EM Algorithm: Theory. University of Massachusetts, Applications and Related Methods. Lecture Notes (2017)
7. Cappe, O., Moulines, E.: On-line expectation-maximization algorithm for latent data models. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **71**(3), 593–613 (2009)
8. Chambolle, A., Ehrhardt, M.-J., Richtárik, P., Schoenlieb, C.-B.: Stochastic primal-dual hybrid gradient algorithm with arbitrary sampling and imaging applications. SIAM J. Optim. (2018)

9. Chambolle, A., Pock, T.: A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imaging Vis. **40**(1), 120–145 (2011)
10. Chen, J., Zhu, J., Teh, Y.W., Zhang, T.: Stochastic expectation maximization with variance reduction. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, Advances in Neural Information Processing Systems 31, pp. 7967–7977. Curran Associates, Inc., (2018)
11. Davis, D., Edmunds, B., Udell, M.: The sound of apalm clapping: Faster nonsmooth nonconvex optimization with stochastic asynchronous palm. In Advances in Neural Information Processing Systems, pp. 226–234 (2016)
12. Ding, M., Huang, T., Wang, S., Mei, J., Zhao, X.: Total variation with overlapping group sparsity for deblurring images under Cauchy noise. Appl. Math. Comput. **341**, 128–147 (2019)
13. Driggs, D., Tang, J., Liang, J., Davies, J., Schönlieb, C.-B.: SPRING: A fast stochastic proximal alternating method for non-smooth non-convex optimization. ArXiv preprint arXiv:2002.12266 (2020)
14. Gerogiannis, D., Nikou, C., Likas, A.: The mixtures of Student's t-distributions as a robust framework for rigid registration. Image Vis. Comput. **27**(9), 1285–1294 (2009)
15. Gitman, I., Lang, H., Zhang, P., Xiao, L.: Understanding the role of momentum in stochastic gradient methods. In Advances in Neural Information Processing Systems, pp. 9633–9643, (2019)
16. Griewank, A., Walther, A.: Evaluating derivatives: principles and techniques of algorithmic differentiation, volume 105. Siam (2008)
17. Hasannasab, M., Hertrich, J., Laus, F., Steidl, G.: Alternatives to the EM algorithm for ML estimation of location, scatter matrix, and degree of freedom of the student t distribution. Numerical Algorithms, pp. 1–42 (2020)
18. Hasannasab, M., Hertrich, J., Neumayer, S., Plonka, G., Setzer, S., Steidl, G.: Parseval proximal neural networks. J. Fourier Anal. Appl. **26**, 59 (2020)
19. Hertrich, J.: Superresolution via Student-t Mixture Models. Master Thesis, TU Kaiserslautern (2020)
20. Hertrich, J., Neumayer, S., Steidl, G.: Convolutional proximal neural networks and plug-and-play algorithms. arXiv preprint arXiv:2011.02281 (2020)
21. Hertrich, J., Steidl, G.: Inertial stochastic palm (iSPALM) and applications in machine learning. ArXiv Preprint arXiv:2005.02204v2, (2020)
22. Higham, N.J.: Functions of Matrices: Theory and Computation. SIAM, Philadelphia (2008)
23. Horn, R.A., Johnson, C.R.: Matrix Analysis. Oxford University Press (2013)
24. Johnson, R., Zhang, T.: Accelerating stochastic gradient descent using predictive variance reduction. Advances in Neural Information Processing Systems, pp. 315–323. (2013)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. ArXiv preprint arXiv:1412.6980 (2014)
26. Lange, K.L., Little, R.J., Taylor, J.M.: Robust statistical modeling using the t distribution. J. Am. Stat. Assoc. **84**(408), 881–896 (1989)
27. Laus, F., Steidl, G.: Multivariate myriad filters based on parameter estimation of Student-t distributions. SIAM J. Imag. Sci. **12**(4), 1864–1904 (2019)
28. Łojasiewicz, S.: Une propriété topologique des sous-ensembles analytiques réels. In *Les Équations aux Dérivées Partielles (Paris, 1962)*, pages 87–89. Éditions du Centre National de la Recherche Scientifique, Paris, (1963)
29. Łojasiewicz, S.: Sur la géométrie semi- et sous-analytique. Université de Grenoble. Annales de l'Institut Fourier **43**(5), 1575–1595 (1993)
30. McLachlan, G., Krishnan, T.: The EM Algorithm and Extensions. John Wiley and Sons Inc (1997)
31. Meng, X.-L., Van Dyk, D.: The EM algorithm—an old folk-song sung to a fast new tune. J. R. Stat. Soc. Ser. B (Statistical Methodology) **59**(3), 511–567 (1997)
32. Nesterov, Y.E.: A method for solving the convex programming problem with convergence rate $O(1/k^2)$. Dokl. Akad. Nauk SSSR **269**(3), 543–547 (1983)
33. Nguyen, L.M., Liu, J., Scheinberg, K., Takáč, M.: Sarah: A novel method for machine learning problems using stochastic recursive gradient. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 2613–2621, (2017)
34. Nguyen, T.M., Wu, Q.J.: Robust Student's-t mixture model with spatial constraints and its application in medical image segmentation. IEEE Trans. Med. Imaging **31**(1), 103–116 (2012)
35. Peel, D., McLachlan, G.J.: Robust mixture modelling using the t distribution. Stat. Comput. **10**(4), 339–348 (2000)
36. Pock, T., Sabach, S.: Inertial proximal alternating linearized minimization (iPALM) for nonconvex and nonsmooth problems. SIAM J. Imag. Sci. **9**(4), 1756–1787 (2016)

37. Polyak, B.T.: Some methods of speeding up the convergence of iteration methods. USSR Comput. Math. Math. Phys. **4**(5), 1–17 (1964)
38. Qian, N.: On the momentum term in gradient descent learning algorithms. Neural Netw. **12**(1), 145–151 (1999)
39. Reddi, S.J., Hefny, A., Sra, S., Póczos, B., Smola, A.: Stochastic variance reduction for nonconvex optimization. In Proc. 33rd International Conference on Machine Learning, (2016)
40. Rockafellar, R.T., Wets, R.J.: Variational Analysis. A Series of Comprehensive Studies in Mathematics, vol. 317. Springer, Berlin, Heidelberg (1998)
41. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**(6088), 533–536 (1986)
42. Sfikas, G., Nikou, C., Galatsanos, N.: Robust image segmentation with mixtures of Student's *t*-distributions. In 2007 IEEE International Conference on Image Processing, volume 1, pp. I – 273–I – 276, (2007)
43. Sutskever, I., Martens, J., Dahl, G., Hinton, G.: On the importance of initialization and momentum in deep learning. In International conference on machine learning, pp. 1139–1147, (2013)
44. Van Den Oord, A., Schrauwen, B.: The Student-*t* mixture as a natural image patch prior with application to image compression. J. Mach. Learn. Res. **15**(1), 2061–2086 (2014)
45. van Dyk, D.A.: Construction, implementation, and theory of algorithms based on data augmentation and model reduction. The University of Chicago (1995). (**PhD Thesis**)
46. Xu, Y., Yin, W.: Block stochastic gradient iteration for convex and nonconvex optimization. SIAM J. Optim. **25**(3), 1686–1716 (2015)
47. Yang, Z., Yang, Z., Gui, G.: A convex constraint variational method for restoring blurred images in the presence of alpha-stable noises. Sensors **18**(4), 1175 (2018)
48. Zhou, Z., Zheng, J., Dai, Y., Zhou, Z., Chen, S.: Robust non-rigid point set registration using Student's-*t* mixture model. PloS one **9**(3), e91381 (2014)