A VARIABLE DIMENSION SKETCHING STRATEGY FOR NONLINEAR LEAST-SQUARES

STEFANIA BELLAVIA*, GRETA MALASPINA*, BENEDETTA MORINI*

Abstract. We present a stochastic inexact Gauss-Newton method for the solution of nonlinear least-squares. To reduce the computational cost with respect to the classical method, at each iteration the proposed algorithm approximately minimizes the local model on a random subspace. The dimension of the subspace varies along the iterations, and two strategies are considered for its update: the first is based solely on the Armijo condition, the latter is based on information from the true Gauss-Newton model. Under suitable assumptions on the objective function and the random subspace, we prove a probabilistic bound on the number of iterations needed to drive the norm of the gradient below any given threshold. Moreover, we provide a theoretical analysis of the local behavior of the method. The numerical experiments demonstrate the effectiveness of the proposed method.

1. Introduction. In this paper, we consider second-order methods for solving the nonlinear least-squares problem

$$\min_{x\in\mathbb{R}^n} f(x) = \frac{1}{2}\|F(x)\|_2^2, \tag{1.1}$$

where the residual function $F\colon \mathbb{R}^n \to \mathbb{R}^m$ is continuously differentiable, and the objective function $f\colon \mathbb{R}^n \to \mathbb{R}$ has a large number n of variables. Our focus is on using second-order models of reduced dimension which still incorporate some form of curvature information. To this end, we propose a sketching strategy of variable dimension.

In recent years, randomized linear algebra [24, 25, 31, 33] has emerged as a powerful tool for solving optimization problems with high computational and memory demands. Randomized sampling and randomized embeddings are the core of a variety of optimization methods with stochastic models that are suitable for solving many applications, including machine learning; see e.g., [2–4, 6, 9–17, 27, 28, 30, 32]. Referring to our problem (1.1), a large reduction of either the variable dimension n or the dimension m of the observations can be achieved via randomized linear algebra. The application of a random embedding, referred to as sketching, can be used to restrict the computation of the trial step to a random subspace of $\mathbb{R}^n$ of dimension considerably smaller than n. As a result, the per-iteration cost is reduced and savings occur in terms of both cost and memory. We refer to [15, §1] for a detailed overview of the existing literature of optimization methods based on sketching.

We turned our attention to sketching methods motivated by the works [13, 14, 30] where a general random subspace framework for unconstrained nonconvex optimization was proposed and then specialized to trust-region and quadratic regularization methods applied to problem (1.1). Under the assumption that the random subspace is an embedding of the gradient of f at the

*Dipartimento di Ingegneria Industriale, Università degli Studi di Firenze, Viale G.B. Morgagni 40, 50134 Firenze, Italia. Members of the INdAM Research Group GNCS. Emails: stefania.bellavia@unifi.it, greta.malaspina@unifi.it, benedetta.morini@unifi.it

†The research that led to the present paper was partially supported by INDAM-GNCS through Progetti di Ricerca 2023 and by PNRR - Missione 4 Istruzione e Ricerca - Componente C2 Investimento 1.1, Fondo per il Programma Nazionale di Ricerca e Progetti di Rilevante Interesse Nazionale (PRIN) funded by the European Commission under the NextGeneration EU programme, project "Advanced optimization METhods for automated central veIn Sign detection in multiple sclerosis from magneTic resonAnce imaging (AMETISTA)", code: P2022J9SNP, MUR D.D. financing decree n. 1379 of 1st September 2023 (CUP E53D23017980001), project "Numerical Optimization with Adaptive Accuracy and Applications to Machine Learning", code: 2022N3ZNAX MUR D.D. financing decree n. 973 of 30th June 2023 (CUP B53D23012670006), and by Partenariato esteso FAIR "Future Artificial Intelligence Research" SPOKE 1 Human-Centered AI. Obiettivo 4, Project "Mathematical and Physical approaches to innovative Machine Learning technologies (MaPLe)", Codice Identificativo EP_FAIR_002, CUP B93C23001750006.

current iteration, such methods drive the gradient of f below a specified threshold ϵ, with high probability, in a number of iterations of order $\mathcal{O}(\epsilon^{-2})$. While sketching preserves the order of worst-case complexity, the numerical experience reported in [13] is not conclusive on the benefit of using sketching.

The need for further investigation on the performance of randomized subspace methods led us to develop a variable dimension sketching strategy for Levenberg-Marquardt type models. Our main effort is to analyze if the trial step captures second-order information though it belongs to the random subspace generated by sketching. To this end we need to assume that the sketching matrices embed the transpose of the Jacobian of F at the iterates with some probability; consequently, we focus on cases where the Jacobian is low-rank, e.g., strongly underdetermined problems. Our algorithm employs the Armijo condition to test the acceptance of the trial step and to adapt the size of the random subspace; it belongs to the class of step search methods since the step direction can change during the back-tracking procedure [22] and retains worst-case iteration complexity of order $\mathcal{O}(\epsilon^{-2})$. The choice of the size of the random subspace is made in view of two conditions; the first condition is the acceptance or rejection of the iterate based on the Armijo condition, the latter condition checks if the trial step captures second-order information. The numerical experience shows that our strategy improves the performance of the algorithm with respect to the basic application of sketching. We remark that we are aware of only one recent work concerning variable size strategies. In [15], a cubic regularization method for unconstrained optimization problems is proposed along with variable size sketching matrices; the approach adopted in [15] is very different from ours as the procedure for choosing the size of the random matrices is based on the rank of the sketched Hessian matrix. This work is organized as follows. In Section 2 we present our algorithm and in Section 3 we study its theoretical properties. In Section 4 we discuss the effect of sketching on the approximate minimization of the full Levenberg-Marquardt model and introduce a practical strategy for choosing the random embedding size adaptively. In Section 5 we analyze the local convergence behavior of a variant of our algorithm to further support our proposal for the adaptive choice of the embedding size. Finally, in Section 6 we present numerical results that illustrate the benefit of using our variable dimension sketching strategy. The appendix summarizes some matrix distributions from the literature that are of interest for our purposes and contains some proofs.

2. A step search algorithm with random reduced models. We introduce our Algorithm 2.1 employing reduced models generated by randomized embedding and a step search strategy. Given the iterate $x_k \in \mathbb{R}^n$, the reduced model is built relying on the Gauss-Newton model

$$m_k(s) = \frac{1}{2}\|J(x_k)s + F(x_k)\|_2^2, \tag{2.1}$$

where $s \in \mathbb{R}^n$. Suppose that a sketching matrix, i.e., a random matrix $M_k \in \mathbb{R}^{\ell_k \times n}$, is sampled from a matrix distribution $\mathcal{M}_k$ and let $s_k = M_k^T \widehat{s}_k$, $\widehat{s}_k \in \mathbb{R}^{\ell_k}$. Then we can form the randomly generated model

$$\widehat{m}_k(\widehat{s}) = \frac{1}{2}\|J(x_k)M_k^T\widehat{s} + F(x_k)\|_2^2, \tag{2.2}$$

whose dimension is reduced whenever $\ell_k < n$. In what follows we assume that $\ell_k \in [\ell_{\min}, \ell_{\max}]$, $\ell_{\max} \leq n$ and we make use of the Levenberg-Marquardt model of the form

$$\begin{aligned} \min_{\widehat{s}\in\mathbb{R}^{\ell_k}} \widehat{m}_k^R(\widehat{s}) &= \frac{1}{2}\|J(x_k)M_k^T\widehat{s} + F(x_k)\|_2^2 + \frac{1}{2}\mu_k\|\widehat{s}\|_2^2 \\ &= \frac{1}{2}\left\| \begin{pmatrix} J(x_k)M_k^T \\ \sqrt{\mu_k} I_{\ell_k} \end{pmatrix} \widehat{s} + \begin{pmatrix} F(x_k) \\ 0 \end{pmatrix} \right\|_2^2, \end{aligned} \tag{2.3}$$

with $0 < \mu_{\min} \leq \mu_k \leq \mu_{\max}$. Note that $\widehat{m}_k^R$ is a model for (1.1) in the subspace generated by M_k^T and is strictly convex due to the regularization term. We observe that

$$\begin{aligned} \nabla \widehat{m}_k(\widehat{s}) &= M_k J(x_k)^T J(x_k) M_k^T \widehat{s}_k + M_k J(x_k)^T F(x_k) \\ \nabla \widehat{m}_k^R(\widehat{s}) &= \left(M_k J(x_k)^T J(x_k) M_k^T + \mu_k I_{\ell_k}\right) \widehat{s} + M_k J(x_k)^T F(x_k) \end{aligned}$$

and the optimality condition $\nabla \widehat{m}_k^R(\widehat{s}) = 0$ for (2.3) amounts to solving the linear system of dimension $\ell_k \times \ell_k$

$$\left(M_k J(x_k)^T J(x_k) M_k^T + \mu_k I_{\ell_k}\right) \widehat{s} = -M_k J(x_k)^T F(x_k). \tag{2.4}$$

which can be solved approximately finding a step $\widehat{s}_k$ s.t.

$$r_k = \begin{pmatrix} J(x_k) M_k^T \\ \sqrt{\mu_k} I_{\ell_k} \end{pmatrix} \widehat{s}_k + \begin{pmatrix} F(x_k) \\ 0 \end{pmatrix}, \tag{2.5}$$

$$(M_k J(x_k)^T J(x_k) M_k^T + \mu_k I_{\ell_k}) \widehat{s}_k = -M_k J(x_k)^T F(x_k) + \rho_k, \tag{2.6}$$

$$\|\rho_k\|_2 = \|(M_k J(x_k)^T \ \sqrt{\mu_k} I_{\ell_k}) r_k\|_2 \leq \eta_k \|M_k J(x_k)^T F(x_k)\|_2, \tag{2.7}$$

for some $0 \leq \eta_k \leq \eta_{\max} < 1$. Choosing $\eta_k = 0$ corresponds to finding the exact minimizer of $\widehat{m}_k^R$, otherwise the step is an approximate minimizer computed using an iterative solver, and in this work we adopt LSMR [20]. Once $\widehat{s}_k$ is available, the trial step $s_k = M_k^T \widehat{s}_k$ in the full space is recovered. The procedure above describes Step 1 of Algorithm 2.1.

Algorithm 2.1. General scheme: k-th iteration

Given $c, \gamma, \widehat{\gamma} \in (0,1)$, $t_{\max}, \eta_{\max}, \mu_{\min}, \mu_{\max} > 0$, $\ell_{\min}, \ell_{\max} \in \mathbb{N}$, $\ell_{\min} < \ell_{\max} \leq n$.
Given $x_k \in \mathbb{R}^n$, $t_k \in (0, t_{\max}]$, $\eta_k \in [0, \eta_{\max}]$, $\mu_k \in [\mu_{\min}, \mu_{\max}]$, $\ell_k \in \mathbb{N}$, $\ell_k \in [\ell_{\min}, \ell_{\max}]$.

Step 1. Draw a random matrix $M_k \in \mathbb{R}^{\ell_k \times n}$ from a matrix distribution $\mathcal{M}_k$.
Form a random model $\widehat{m}_k^R(\widehat{s})$ of the form (2.3).
Compute the inexact step $\widehat{s}_k$ in (2.5)−(2.7). Let $s_k = M_k^T \widehat{s}_k$.

Step 2. If $x_k + t_k s_k$ satisfies

$$f(x_k + t_k s_k) < f(x_k) + c t_k s_k^T \nabla f(x_k), \tag{2.8}$$

Then (successful iteration)
set $x_{k+1} = x_k + t_k s_k$, $t_{k+1} = \min\{t_{\max}, \gamma^{-1} t_k\}$, $\ell_{k+1} = \max\{\ell_{\min}, \widehat{\gamma} \ell_k\}$.
Else (unsuccessful iteration)
set $x_{k+1} = x_k$, $t_{k+1} = \gamma t_k$, $\ell_{k+1} = \min\{\ell_{\max}, \widehat{\gamma}^{-1} \ell_k\}$.

Step 3. Choose $\eta_{k+1} \in [0, \eta_{\max}]$, $\mu_{k+1} \in [\mu_{\min}, \mu_{\max}]$. Set $k = k + 1$.

Successively, in Step 2 of the algorithm, the step search is performed using the Armijo condition (2.8) with $c \in (0,1)$ being the small Armijo constant. The test is made on the trial iterate $x_k + t_k s_k$ where t_k is a positive steplength set at the previous iteration $k-1$. If the test (2.8) is satisfied, the iteration is successful, i.e., the trial iterate is accepted, the steplength t_{k+1} is enlarged for the next iteration and the sketching size ℓ_{k+1} is reduced for the next iteration taking into account that the current reduced model produced an accepted step. If the test (2.8) fails, the iteration is unsuccessful, i.e., the trial step is discarded, the steplength t_{k+1} is reduced for the next iteration and the sketching size ℓ_{k+1} is enlarged. According to a step search strategy, the step direction changes during the backtracking procedure. Finally, in Step 3 the forcing term η_{k+1} and the regularization parameter μ_{k+1} are defined for the next iteration.

The use of the Levenberg-Marquardt model instead of the Gauss-Newton model is motivated by the fact that, differently from the Gauss-Newton model, the step s_k is a descent direction for f as long as the sketched gradient $M_k\nabla f(x_k) = M_k J(x_k)^T F(x_k)$ is nonzero.

LEMMA 2.1. *Let s_k be as in Algorithm 2.1. It holds $s_k^T\nabla f(x_k) \leq -\mu_k\|\widehat{s}_k\|_2^2$, and $s_k^T\nabla f(x_k) < 0$ if $M_k\nabla f(x_k) \neq 0$.*

Proof. Let us first consider the case where the system (2.4) is solved inexactly. Namely, $\widehat{s}_k$ is computed applying LSMR method and satisfies (2.5). Let us define $G_k = \begin{pmatrix} J(x_k)M_k^T \\ \sqrt{\mu_k}I_{\ell_k} \end{pmatrix}$ and $\overline{F}_k = \begin{pmatrix} F(x_k) \\ 0 \end{pmatrix}$. Starting from the null initial guess $\widehat{s}_k^{(0)} = 0$, LSMR generates a sequence of iterates $\{\widehat{s}_k^{(j)}\}$, $j \geq 0$, such that

$$\|G_k\widehat{s}_k^{(j)} + \overline{F}_k\|_2^2 = \min_{\widehat{s}\in K_k^{(j)}} \|G_k\widehat{s} + \overline{F}_k\|_2^2, \tag{2.9}$$

with

$$K_k^{(j)} = \text{span}\left\{G_k^T\overline{F}_k, (G_k^TG_k)G_k^T\overline{F}_k, \ldots, (G_k^TG_k)^{j-1}G_k^T\overline{F}_k\right\}.$$

Then, the residual vector r_k in (2.5) corresponding to the inexact step $\widehat{s}_k = \widehat{s}_k^{(m)}$, for some $m \geq 0$, is orthogonal to any vector in $G_kK_k^{(m)}$, i.e., $\widehat{s}_k^TG_k^Tr_k = \widehat{s}_k^T\rho_k = 0$ with ρ_k defined in (2.7). Consequently, (2.6) yields

$$\begin{aligned} s_k^T\nabla f(x_k) &= \widehat{s}_k^TM_kJ(x_k)^TF(x_k) \\ &= \widehat{s}_k^T(-(M_kJ(x_k)^TJ(x_k)M_k^T + \mu_kI_{\ell_k})\widehat{s}_k + \rho_k) \\ &= -s_k^TJ(x_k)^TJ(x_k)s_k - \mu_k\|\widehat{s}_k\|_2^2 \leq -\mu_k\|\widehat{s}_k\|_2^2. \end{aligned} \tag{2.10}$$

since $J(x_k)^TJ(x_k)$ is positive semidefinite.

In case $\widehat{s}_k$ solves the system (2.4) exactly, the claim follows as above by (2.10) letting $\rho_k = 0$.

Finally, since by (2.6)-(2.7), it holds $\widehat{s}_k \neq 0$ if and only if $M_k\nabla f(x_k) \neq 0$, the proof is completed. ∎

Regarding the case where the sketched gradient $M_k\nabla f(x_k)$ is null, we observe that the vector s_k is null and the iteration is unsuccessful.

The trial step $\widehat{s}_k \in \mathbb{R}^{\ell_k}$ in (2.6) gives rise to two relative residuals with respect to the minimization of $\widehat{m}_k^R$ and $\widehat{m}_k$ defined as

$$\eta_k^* \stackrel{\text{def}}{=} \frac{\|\nabla\widehat{m}_k^R(\widehat{s}_k)\|_2}{\|M_k\nabla f(x_k)\|_2}, \qquad \nu_k^* \stackrel{\text{def}}{=} \frac{\|\nabla\widehat{m}_k(\widehat{s}_k)\|_2}{\|M_k\nabla f(x_k)\|_2}. \tag{2.11}$$

Such scalars characterize the approximate solution of the linear systems $\nabla\widehat{m}_k^R(\widehat{s}) = 0$ and $\nabla\widehat{m}_k(s) = 0$ respectively. The next lemma provides a relation between η_k^* and ν_k^* and an upper bound on the norm of $\widehat{s}_k$

LEMMA 2.2. *Let us assume that $\widehat{s}_k \in \mathbb{R}^{\ell_k}$ is such that*

$$(M_kJ(x_k)^TJ(x_k)M_k^T + \mu_kI)\widehat{s}_k = -M_k\nabla f(x_k) + \rho_k. \tag{2.12}$$

and $\|\rho_k\|_2 = \eta_k^*\|M_k\nabla f(x_k)\|_2$. Then, if $\eta_k^* = 0$

$$\frac{\mu_k}{\lambda_k^1 + \mu_k} \leq \nu_k^* \leq \frac{\mu_k}{\lambda_k^{r_k} + \mu_k}, \tag{2.13}$$

where λ_k^1 *and* $\lambda_k^{r_k}$ *are the largest and the smallest nonzero eigenvalue of* $M_k J(x_k)^T J(x_k) M_k^T$ *respectively, otherwise*

$$\nu_k^* \leq \frac{\mu_k}{\lambda_k^{r_k} + \mu_k} + \frac{\lambda_k^1}{\lambda_k^1 + \mu_k} \eta_k^*. \tag{2.14}$$

Further,

$$\|\widehat{s}_k\|_2 \leq \left(\frac{1}{\lambda_k^{r_k} + \mu_k} + \frac{\eta_k}{\mu_k} \right) \|M_k \nabla f(x_k)\|_2. \tag{2.15}$$

Proof. If $\eta_k^* = 0$ then $\widehat{s}_k = -\big(M_k J(x_k)^T J(x_k) M_k^T + \mu_k I\big)^{-1} M_k \nabla f(x_k)$ and

$$\nabla \widehat{m}_k(\widehat{s}_k) = \Big(- M_k J(x_k)^T J(x_k) M_k^T \big(M_k J(x_k)^T J(x_k) M_k^T + \mu_k I\big)^{-1} + I \Big) M_k \nabla f(x_k). \tag{2.16}$$

Let $B_k \stackrel{\text{def}}{=} M_k J(x_k)^T J(x_k) M_k^T = Q_k \Lambda_k Q_k^T$ be the eigendecomposition where

$$\Lambda_k = \operatorname{diag}(\lambda_k^1, \ldots, \lambda_k^{r_k}, 0, \ldots, 0) \in \mathbb{R}^{\ell_k \times \ell_k}, \;\; Q = \big(q_k^i \big| \ldots \big| q_k^{\ell_k}\big) \in \mathbb{R}^{\ell_k \times \ell_k},$$

r_k is the rank of the matrix, $\lambda_k^1 \geq \lambda_k^2 \geq \cdots, \lambda_k^{r_k} > 0$. Note that $\operatorname{span}(q_k^1, \ldots, q_k^{r_k}) = \operatorname{range}(B_k)$, $\operatorname{span}(q_k^{r_k+1}, \ldots q_k^{\ell_k}) = \ker(B_k)$. Then,

$$\nabla \widehat{m}_k(\widehat{s}_k) = Q_k \Big(- \Lambda_k (\Lambda_k + \mu_k I)^{-1} + I \Big) Q_k^T M_k \nabla f(x_k).$$

Since $M_k \nabla f(x_k) \in \operatorname{range}(B_k)$, then $(q_k^i)^T M_k \nabla f(x_k) = 0$ for $i = r_k + 1, \ldots \ell_k$, and

$$\begin{aligned} \nabla \widehat{m}_k(\widehat{s}_k) &= Q_k \Big(- \Lambda_k (\Lambda_k + \mu_k I)^{-1} + I \Big) \begin{pmatrix} (q_k^1)^T M_k \nabla f(x_k) \\ \vdots \\ (q_k^{r_k})^T M_k \nabla f(x_k) \\ 0 \\ \vdots \\ 0 \end{pmatrix} \\ &= -\sum_{i=1}^{r_k} \frac{\mu_k}{\lambda_k^i + \mu_k} \Big((q_k^i)^T M_k \nabla f(x_k) \Big) q_k^i. \end{aligned}$$

The vectors q_k^i are orthonormal, hence it follows

$$\begin{aligned} \|\nabla \widehat{m}_k(\widehat{s}_k)\|_2^2 &= \left\| \sum_{i=1}^{r_k} \frac{\mu_k}{\lambda_k^i + \mu_k} \Big((q_k^i)^T M_k \nabla f(x_k) \Big) q_k^i \right\|_2^2 \\ &= \sum_{i=1}^{r_k} \Big(\frac{\mu_k}{\lambda_k^i + \mu_k} \Big)^2 \Big((q_k^i)^T M_k \nabla f(x_k) \Big)^2 \|q_k^i\|_2^2 \\ &\leq \Big(\frac{\mu_k}{\lambda_k^{r_k} + \mu_k} \Big)^2 \sum_{i=1}^{r_k} \Big((q_k^i)^T M_k \nabla f(x_k) \Big)^2 \\ &= \Big(\frac{\mu_k}{\lambda_k^{r_k} + \mu_k} \Big)^2 \|Q^T M_k \nabla f(x_k)\|_2^2 \\ &= \Big(\frac{\mu_k}{\lambda_k^{r_k} + \mu_k} \Big)^2 \|M_k \nabla f(x_k)\|_2^2 \end{aligned}$$

which gives the upper bound in (2.13). Analogously, the lower bound in (2.13) follows.

In the general case $\eta_k^* \geq 0$, the step $\widehat{s}_k$ takes the form

$$\widehat{s}_k = (M_k J(x_k)^T J(x_k) M_k^T + \mu_k I)^{-1}(-M_k J(x_k)^T F(x_k) + \rho_k), \tag{2.17}$$

and taking into account (2.7), the equality $\|B_k(B_k + \mu_k I)^{-1}\| = \frac{\lambda_k^1}{\lambda_k^1 + \mu_k}$ and proceeding as for deriving the upper bound, we get (2.14).

Inequality (2.15) is obtained by (2.17) repeating the reasoning above. □

3. Theoretical analysis. Algorithm 2.1 generates a random sequence $\{x_k\}$ since at each iteration the model $\widehat{m}_k^R$ is random. Letting X_k be the random variable such that x_k is its realization and $\tau > 0$, the hitting time is defined as

$$N_\tau = \inf\{k : \|\nabla f(X_k)\|_2 \leq \tau\}. \tag{3.1}$$

Following [30], convergence to a τ-approximate first-order stationary point occurs if the algorithm is run for $k \geq N_\tau$ iterations; otherwise the algorithm has not converged.

In this section, exploiting the analysis in [13] and [30], we derive a probabilistic bound on the total number of iteration N_τ required to reach a τ-approximate first-order stationary point. We perform our analysis making the following assumption on the problem.

ASSUMPTION 3.1. *The objective function $f : \mathbb{R}^n \to \mathbb{R}$ in problem (1.1) is continuously differentiable and bounded below by f_*. The gradient of f is Lipschitz continuous, that is, there exist a positive scalar L such that for any $x, y \in \mathbb{R}^n$*

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2.$$

The first requirement for our analysis is to establish some properties of the iterate x_{k+1} when the iteration k is true, namely the random matrix M_k satisfies the following conditions, see [30].

DEFINITION 3.2. *Given the iteration independent constants $\varepsilon \in (0, 1)$, $M_{\max} > 0$, and matrix $M_k \in \mathbb{R}^{\ell_k \times n}$ drawn in Step 1 of the Algorithm 2.1, iteration k is true if*

$$\|M_k \nabla f(x_k)\|_2^2 \geq (1 - \varepsilon)\|\nabla f(x_k)\|_2^2 \tag{3.2}$$

$$\|M_k\|_2 \leq M_{\max}. \tag{3.3}$$

Thus, iteration k is true when M_k provides a one-sided (from below) embedding with distortion ε and when its norm is uniformly bounded from above. A relevant property of a true iteration is that it is successful if the steplength t_k is sufficiently small, independently of k.

LEMMA 3.3. *Let x_k be the iterate in Algorithm 2.1 and a true iteration be as in Definition 3.2. Suppose that Assumption 3.1 holds and iteration k is true. Let $t_{\text{low}} = \dfrac{2(1-c)\mu_{\min}}{LM_{\max}^2}$. If $t_k < t_{\text{low}}$, then iteration k is successful.*

Proof. Using the mean value theorem and Assumptions 3.1, we obtain

$$\begin{aligned}
f(x_k + t_k s_k) &= f(x_k) + \int_0^1 (\nabla f(x_k + w t_k s_k))^T (t_k s_k) dw \\
&= f(x_k) + t_k \nabla f(x_k)^T s_k + \int_0^1 t_k (\nabla f(x_k + w t_k s_k) - \nabla f(x_k))^T s_k dw \\
&\leq f(x_k) + t_k \nabla f(x_k)^T s_k + \frac{L}{2} t_k^2 \|s_k\|_2^2 \\
&\leq f(x_k) + t_k \nabla f(x_k)^T s_k + \frac{L}{2} t_k^2 M_{\max}^2 \|\widehat{s}_k\|_2^2.
\end{aligned}$$

Thus, the Armijo condition (2.8) holds if

$$t_k \nabla f(x_k)^T s_k + \frac{LM_{\max}^2}{2} t_k^2 \|\widehat{s}_k\|_2^2 < c t_k s_k^T \nabla f(x_k),$$

which is equivalent to $t_k < (c-1)\frac{2}{LM_{\max}^2}\frac{s_k^T \nabla f(x_k)}{\|\widehat{s}_k\|^2}$. Lemma 2.1 implies that

$$\frac{2}{LM_{\max}^2}(c-1)\frac{s_k^T \nabla f(x_k)}{\|\widehat{s}_k\|^2} > t_{\text{low}} \stackrel{\text{def}}{=} \frac{2(1-c)\mu_{\min}}{LM_{\max}^2},$$

which concludes the proof. □

Using again the concept of true iteration we can characterize the quantity $f(x_k) - f(x_{k+1})$ for all k. We need the following assumption on the generated sequence.

ASSUMPTION 3.4. *At any true iteration, it holds*

$$\sigma_1(M_k J(x_k)^T J(x_k) M_k^T) \leq \sigma_\dagger,$$

where $\sigma_1(\cdot)$ denotes the maximum singular value of a matrix and $\sigma_\dagger > 0$ is independent of k.

LEMMA 3.5. *Let $\{x_k\}$ be generated by Algorithm 2.1. Let true iterations be defined in Definition 3.2. Suppose that Assumption 3.4 holds.*

(i) If iteration k is true and successful with $k < N_\tau$, then

$$f(x_k) - f(x_{k+1}) \geq h(\tau, t_k),$$

where $h : \mathbb{R}^2 \to \mathbb{R}$ is a nonnegative function, non decreasing in its arguments $\tau > 0$, $t_k > 0$.

(ii) $f(x_k) - f(x_{k+1}) \geq 0$ for all $k \geq 0$.

Proof. (i) First we prove that

$$\|\nabla f(x)\|_2 \leq \frac{\sigma_\dagger + \mu_{\max}}{(1-\eta_{\max})\sqrt{(1-\varepsilon)}} \|\widehat{s}_k\|_2. \tag{3.4}$$

In fact, by (2.5)–(2.7), we have

$$\begin{aligned} \|M_k \nabla f(x_k)\|_2 &\leq \|(M_k J(x_k)^T J(x_k) M_k^T + \mu_k I_{\ell_k})\widehat{s}_k\|_2 + \|\rho_k\|_2 \\ &\leq (\sigma_\dagger + \mu_{\max})\|\widehat{s}_k\|_2 + \eta_{\max}\|M_k \nabla f(x_k)\|_2. \end{aligned}$$

and consequently

$$\|M_k \nabla f(x_k)\|_2 \leq \frac{\sigma_\dagger + \mu_{\max}}{1-\eta_{\max}} \|\widehat{s}_k\|_2.$$

Hence, using (3.2) we obtain (3.4). Now, if the iteration is true and successful, by the Armijo condition (2.8), Lemma 2.1, and the inequality (3.4) we have

$$\begin{aligned} f(x_{k+1}) &\leq f(x_k) - c t_k \mu_{\min} \|\widehat{s}_k\|_2^2 \\ &\leq f(x_k) - c t_k \frac{\mu_{\min}(1-\eta_{\max})^2(1-\varepsilon)}{(\sigma_\dagger + \mu_{\max})^2} \|\nabla f(x)\|_2^2 \\ &\leq f(x_k) - c t_k \frac{\mu_{\min}(1-\eta_{\max})^2(1-\varepsilon)}{(\sigma_\dagger + \mu_{\max})^2} \tau^2. \end{aligned}$$

Hence the claim follows with

$$h(\tau, t) \stackrel{\text{def}}{=} ct\frac{\mu_{\min}(1-\eta_{\max})^2(1-\varepsilon)}{(\sigma_\dagger + \mu_{\max})^2}\tau^2. \tag{3.5}$$

(ii) Lemma 2.1) and the acceptance rule of the step s_k in Algorithm 2.1 imply $f(x_k) - f(x_{k+1}) \geq 0$ for all $k \geq 0$. □

Random matrix distributions guarantee true iterations as in Definition 3.2 in probability. Following [30], we suppose that the iterations are true at least with a fixed probability as specified below. In what follows, L_k is a random variable and ℓ_k denotes its realization. We make the following assumption.

ASSUMPTION 3.6. *There exists $\delta_M \in (0,1)$ such that*

$$\mathbb{P}\left(T_k \mid X_k = x_k, L_k = \ell_k\right) \geq 1 - \delta_M, \qquad k = 0, 1, \ldots,$$

where T_k is the event T_k=$\{$iteration k is true$\}$. Moreover, T_k is conditionally independent on $T_0, T_1, \ldots, T_{k-1}$ given $X_k = x_k$ and $L_k = \ell_k$.

Conditions for ensuring the request of true iterations in probability are provided in the following Lemma.

LEMMA 3.7. *Suppose that there exist $\varepsilon \in (0,1)$, $\delta_M^{(1)} \in (0,1)$ such that for a(ny) fixed $y \in \{\nabla f(x) : x \in \mathbb{R}^n\}$, $M_k \in \mathbb{R}^{\ell_k \times n}$ drawn from a random matrix distribution $\mathcal{M}_k$ satisfies*

$$\mathbb{P}\left(\|M_k y\|_2^2 \geq (1-\varepsilon)\|y\|_2^2 \,|\, L_k = \ell_k\right) \geq 1 - \delta_M^{(1)}. \tag{3.6}$$

Further, suppose that there exists $\delta_M^{(2)} \in [0,1)$ such that M_k satisfies

$$\mathbb{P}\left(\|M_k\|_2 \leq M_{\max} \,|\, L_k = \ell_k\right) \geq 1 - \delta_M^{(2)}, \tag{3.7}$$

where $M_{\max}$ is an iteration independent constant and that $\delta_M^{(1)} + \delta_M^{(2)} < 1$.
For true iterations as in Definition 3.2, Assumption 3.6 is satisfied with $\delta_M = \delta_M^{(1)} + \delta_M^{(2)}$.

Proof. The proof closely follows that of Lemma 4.4.2 in [30]. □

In order to derive the result on the hitting time, we need the following technical result that represents a generalization to the case of variable sketching size of Lemma A.2 in [13]. Given k, let $\mathcal{T}_k$ and $\mathcal{M}_k$ be the random variables corresponding to the realizations t_k, M_k, respectively and let $\mathcal{F}_{k-1}$ denote the σ-algebra generated by X_0, $\mathcal{T}_0$, L_0, $\mathcal{M}_0$ …, X_{k-1}, $\mathcal{T}_{k-1}$, L_{k-1}, $\mathcal{M}_{k-1}$, X_k, $\mathcal{T}_k$, L_k.

LEMMA 3.8. *Let true iterations be defined in Definition 3.2. Suppose that Assumption 3.6 holds with $\delta_M \in (0,1)$.*
i) For any $\lambda > 0$ and $N \in \mathbb{N}$, we have

$$\mathbb{E}\left[e^{-\lambda \sum_{k=0}^{N-1} T_k}\right] \leq \left[e^{(e^{-\lambda}-1)(1-\delta_M)}\right]^N. \tag{3.8}$$

ii) If Algorithm 2.1 runs for N iterations, then, for any given $\delta_1 \in (0,1)$,

$$\mathbb{P}\left(N_S \leq (1-\delta_M)(1-\delta_1)N\right) \leq e^{-\frac{\delta_1^2}{2}(1-\delta_M)N}, \tag{3.9}$$

where $N_S = \sum_{k=0}^{N-1} T_k$ is the number of successful iterations.

Proof. See the Appendix A.2. ☐

LEMMA 3.9. *Given $\tau > 0$, suppose that $k < N_\tau$ and that Assumption 3.4 holds. Let t_{low} be defined as in Lemma 3.3. Then there exist $\psi_t = \min\left\{1, \left\lceil \log_\gamma\left(\frac{t_{\text{low}}}{t_0}\right)\right\rceil\right\} \in \mathbb{N}^+$ such that $t_{\min} = t_0\gamma^{\psi_t}$ satisfies $t_{\min} \leq \min\{t_{\text{low}}, \gamma t_0\}$*

Proof. [13, Lemma 2.1] ☐

We can now state the result on the iteration complexity.

THEOREM 3.10. *Suppose that Assumptions 3.1, 3.4 and 3.6 hold with $\delta_M \in (0, \frac{1}{4})$. Let N_τ be defined in* (3.1)*, h be given in (3.5) and ψ_t be given in Lemma 3.9. Assume that Algorithm 2.1 runs for N iterations. Then, for any $\delta_1 \in (0,1)$ such that $(1-\delta_M)(1-\delta_1) - \frac{3}{4} > 0$, if*

$$N \geq \left[(1-\delta_M)(1-\delta_1) - \frac{3}{4}\right]^{-1}\left[\frac{f(x_0) - f_*}{h(\tau, t_0\gamma^{1+\psi_t})} + \frac{\psi_t}{2}\right], \tag{3.10}$$

we have

$$\mathbb{P}\left(N \geq N_\tau\right) \geq 1 - e^{-\frac{\delta_1^2}{2}(1-\delta_M)N}.$$

Proof. Applying Theorem [13, Theorem 2.1] combined with the results in Lemma 3.5, Lemma 3.8 and Lemma 3.9 gives the desired result. ☐

4. Choosing the size ℓ_k. In this section we analyze the step s_k used in the step search strategy, summarize results from the literature on the enforcement of true iterations in probability and introduce a modification to Step 2 of Algorithm 2.1 that monitors the approximate minimization of the deterministic model m_k.

Algorithm 2.1 can be implemented using random matrix distributions that generate true iterations in probability according to Definition 3.2. Random ensembles $\mathcal{M}_k$ which satisfy Lemma 3.7 are: scaled Gaussian matrices, s-hashing matrices, stable 1-hashing matrices, scaled sampling matrices [25, 30, 33]. For the sake of completeness, in the Appendix A.1 we report the definition of such distributions and a table summarizing the relations between the values ε, $M_{\max}$, ℓ_k, $\delta_M^{(1)}$, $\delta_M^{(2)}$. In principle, due to sketched models, a single iteration of Algorithm 2.1 is computationally convenient with respect to the deterministic Levenberg-Marquardt algorithm. But the overall performance of the Algorithm 2.1 may be worse than that of the deterministic algorithm if the step s_k does not incorporate second-order information from the Gauss-Newton model m_k.

Minimizing the reduced model $\widehat{m}_k$ in (2.2) is equivalent to minimizing m_k in the subspace generated by the columns of M_k^T. In general, no hint can be given on $s_k = M_k^T\widehat{s}_k$ as an approximate minimizer of m_k and on the magnitude of the scalar

$$\theta_k^* \stackrel{\text{def}}{=} \frac{\|\nabla m_k(s_k)\|_2}{\|\nabla f(x_k)\|_2}, \tag{4.1}$$

which can be interpreted as a measure of the accuracy of s_k with respect to the optimality condition $\nabla m_k(s) = 0$. However, noticing that $\nabla f(x_k) = J(x_k)^T F(x_k)$, this limitation can be overcome using a ε-subspace embedding condition for $J(x_k)^T$ and reformulating the definition of true iteration as follows.

DEFINITION 4.1. *Given the iteration independent constants $\varepsilon \in (0,1)$, $M_{\max} > 0$, and a matrix $M_k \in \mathbb{R}^{\ell_k \times n}$ drawn in Step 1 of the Algorithm 2.1, iteration k is true if*

$$(1+\varepsilon)\|J(x_k)^T z\|_2^2 \geq \|M_k J(x_k)^T z\|_2^2 \geq (1-\varepsilon)\|J(x_k)^T z\|_2^2 \text{ for every } z \in \mathbb{R}^m, \tag{4.2}$$

$$\|M_k\|_2 \leq M_{\max}. \tag{4.3}$$

Now, recalling the definition of η_k^* and ν_k^* in (2.11) we characterize θ_k^* with respect to ν_k.

LEMMA 4.2. *Let $\widehat{s}_k \in \mathbb{R}^{\ell_k}$ be as in (2.5)–(2.7), $s_k = M_k^T \widehat{s}_k \in \mathbb{R}^n$, ν_k^* as in (2.11) and θ_k^* as in (4.1). Then, if iteration k is true as defined in Definition 4.1 it holds*

$$\Big(\frac{1-\varepsilon}{1+\varepsilon}\Big)^{1/2} \nu_k^* \leq \theta_k^* \leq \Big(\frac{1+\varepsilon}{1-\varepsilon}\Big)^{1/2} \nu_k^*.$$

Proof. Since $\nabla m_k(s_k)$ and $\nabla f(x_k)$ belong to $\mathrm{span}(J(x_k)^T)$, and $\nabla \widehat{m}_k(\widehat{s}_k) = M_k \nabla m_k(s_k)$, the inequality 4.2 implies

$$\nu_k^* = \frac{\|M_k \nabla m_k(s_k)\|_2}{\|M_k \nabla f(x_k)\|_2} \geq \frac{(1-\varepsilon)^{1/2}}{(1+\varepsilon)^{1/2}} \frac{\|\nabla m_k(s_k)\|_2}{\|\nabla f(x_k)\|_2} = \Big(\frac{1-\varepsilon}{1+\varepsilon}\Big)^{1/2} \theta_k^*,$$

and

$$\nu_k^* = \frac{\|M_k \nabla m_k(s_k)\|_2}{\|M_k \nabla f(x_k)\|_2} \leq \frac{(1+\varepsilon)^{1/2}}{(1-\varepsilon)^{1/2}} \frac{\|\nabla m_k(s_k)\|_2}{\|\nabla f(x_k)\|_2} = \Big(\frac{1+\varepsilon}{1-\varepsilon}\Big)^{1/2} \theta_k^*.$$

□

The property of ε-subspace embedding also yields results on the relation between the rank and the singular values of $M_k J(x_k)^T$ and $J(x_k)^T$.

THEOREM 4.3. *Given $\varepsilon \in (0,1)$ and $J(x_k)^T \in \mathbb{R}^{n \times m}$, suppose that $M_k \in \mathbb{R}^{\ell_k \times n}$ satisfies (4.2). Then*

i) $\mathrm{rank}(M(x_k)J(x_k)^T) = \mathrm{rank}(J(x_k)^T)$ and $\ker(M_k J(x_k)^T) = \ker(J(x_k)^T)$;

ii) letting $r_k = \mathrm{rank}(J(x_k)^T)$, and $\sigma_1(\cdot) \geq \cdots \geq \sigma_{r_k}(\cdot)$ be the nonsingular values of some given matrix of rank r_k, it holds

$$\sigma_1(M_k J(x_k)^T) \leq (1+\varepsilon)^{1/2} \sigma_1(J(x_k)^T), \tag{4.4}$$

and

$$\sigma_{r_k}(M_k J(x_k)^T) \geq (1-\varepsilon)^{1/2} \sigma_{r_k}(J(x_k)^T). \tag{4.5}$$

Proof. To ease the notation, we drop the iteration index k and we write M, J and ℓ in place of M_k, $J(x_k)$ and ℓ_k respectively.

The equality $\mathrm{rank}(J^T) = \mathrm{rank}(MJ^T)$ is proved in [30, Lemma 2.2.1]. As for the null space of J^T and MJ^T, trivially $\ker(J^T) \subseteq \ker(MJ^T)$ holds. Let us assume by contradiction that the inclusion is strict. Then, there exists $\bar{z} \in \ker(MJ^T)$ such that $\bar{z} \notin \ker(J^T)$, i.e., $\|MJ^T\bar{z}\|_2 = 0$ and $\|J^T\bar{z}\|_2 > 0$, which contradicts the embedding property (4.2).

We now prove the second part of the statement. Let $J^T = U\Sigma V^T$ and $MJ^T = P\widehat{\Sigma}Q^T$ be the singular value decompositions of J^T and MJ^T, respectively, with

$$\Sigma = \begin{pmatrix} \mathrm{diag}(\sigma_1, \ldots, \sigma_r, 0, \ldots, 0) \\ \\ O_{(n-m)\times m} \end{pmatrix} \in \mathbb{R}^{n \times m}, \ \widehat{\Sigma} = \begin{pmatrix} \mathrm{diag}(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_r, 0, \ldots, 0) \\ \\ O_{(\ell-m)\times m} \end{pmatrix} \in \mathbb{R}^{\ell \times m}$$

where $\sigma_1, \ldots, \sigma_r$ and $\widehat{\sigma}_1, \ldots, \widehat{\sigma}_r$ are the nonzero singular values of J^T and MJ^T, respectively. Moreover let us denote $V = (v_1, \ldots, v_m) \in \mathbb{R}^{m \times m}$, $Q = (q_1, \ldots, q_m) \in \mathbb{R}^{m \times m}$ where $\mathrm{range}(J) = \mathrm{span}(v_1, \ldots, v_r)$, $\mathrm{range}(JM^T) = \mathrm{span}(q_1, \ldots, q_r)$ and

$$\mathrm{span}(v_{r+1}, \ldots, v_m) = \ker(J^T) = \ker(MJ^T) = \mathrm{span}(q_{r+1}, \ldots, q_m).$$

Note that

$$\begin{aligned}\widehat{\sigma}_1^2 &= \|MJ^Tq_1\|_2^2 \le (1+\varepsilon)\|J^Tq_1\|_2^2 = (1+\varepsilon)\|U\Sigma V^Tq_1\|_2^2,\\ \widehat{\sigma}_r^2 &= \|MJ^Tq_r\|_2^2 \ge (1-\varepsilon)\|J^Tq_r\|_2^2 = (1-\varepsilon)\|U\Sigma V^Tq_r\|_2^2.\end{aligned} \tag{4.6}$$

Since Q is orthogonal, it holds

$$q_r \perp \operatorname{span}(q_{r+1},\dots,q_m) = \ker(MJ^T) = \ker(J^T) = \operatorname{span}(v_{r+1},\dots,v_m),$$

and therefore $q_1^Tv_i = q_r^Tv_i = 0$ for all $i = r+1,\dots,m$. Thus,

$$\begin{aligned}\widehat{\sigma}_1^2 &\le (1+\varepsilon)\sum_{i=1}^{r}\sigma_i^2(q_1^Tv_i)^2 \le (1+\varepsilon)\sigma_1^2\sum_{i=1}^{r}(q_1^Tv_i)^2\\ &= (1+\varepsilon)\sigma_1^2\sum_{i=1}^{n}(q_1^Tv_i)^2 = (1+\varepsilon)\sigma_1^2\|V^Tq_1\|_2^2 = (1+\varepsilon)\sigma_1^2,\end{aligned} \tag{4.7}$$

and

$$\begin{aligned}\widehat{\sigma}_r^2 &\ge (1-\varepsilon)\sum_{i=1}^{r}\sigma_i^2(q_r^Tv_i)^2 \ge (1-\varepsilon)\sigma_r^2\sum_{i=1}^{r}(q_r^Tv_i)^2\\ &= (1-\varepsilon)\sigma_r^2\sum_{i=1}^{n}(q_r^Tv_i)^2 = (1-\varepsilon)\sigma_r^2\|V^Tq_r\|_2^2 = (1-\varepsilon)\sigma_r^2,\end{aligned} \tag{4.8}$$

which concludes the proof. □

The previous lemma implies that the subspace embedding property cannot hold if $\ell_k < \operatorname{rank}(J(x_k)^T)$. Further, it characterizes λ_k^1 and $\lambda_k^{r_k}$ in Lemma 2.2 since $\lambda_k^1 = \sigma_1^2(M_kJ(x_k)^T)$ and $\lambda_k^{r_k} = \sigma_{r_k}^2(M_kJ(x_k)^T)$ and thus the condition number $\kappa_2(M_kJ(x_k)^TJ(x_k)M_k^T)$ in 2-norm of $M_kJ(x_k)^TJ(x_k)M_k^T$ is bounded above by $(1+\varepsilon)/(1-\varepsilon)\kappa_2(J(x_k)^TJ(x_k))$.

In order to take advantage of random models of reduced dimension, ℓ_k should be significantly smaller than n. Scaled Gaussian matrices of dimension $\ell_k \times n$ satisfy (4.2) with probability at least $1-\delta$ when $\ell_k = \mathcal{O}(\varepsilon^{-2}(r_k + \log(1/\delta)))$ with r_k being the rank of $J(x_k)$, but such matrices are dense and their use is not computationally convenient [33]. On the other hand, under suitable conditions, the distribution of s-hashing matrices may provide a subspace embedding and computational savings. Let us first introduce the notion of coherence $\mu(J(x_k)^T)$ of $J(x_k)^T$.

DEFINITION 4.4. *[24] Given a matrix $J(x_k)^T \in \mathbb{R}^{n\times m}$ with rank r_k, let $J(x_k)^T = U_k\Sigma_kV_k^T$ be the economic SVD decomposition where $U_k \in \mathbb{R}^{n\times r_k}$ has orthonormal columns, $\Sigma_k \in \mathbb{R}^{r_k\times r_k}$ has strictly positive diagonal entries, $V_k \in \mathbb{R}^{n\times r_k}$ has orthonormal columns. The coherence $\mu(J(x_k)^T)$ of $J(x_k)^T$ is defined as*

$$\mu(J(x_k)^T) = \max_{i=1,\dots,n}\|(U_k)_i\|_2,$$

where $(U_k)_i$ denotes the i-th row of U_k.

It holds $\sqrt{r_k/n} \le \mu(J(x_k)^T) \le 1$, see [30, Lemma 2.2.3].

From the literature we know that s-hashing matrices satisfy (4.2) in probability under different assumptions on the coherence of the matrix $J(x_k)^T$ and the size ℓ_k. In [30, Theorem 2.3.1] the author proves that if $\mu(J(x_k)^T) = \mathcal{O}(r_k^{-1})$ then 1-hashing matrices satisfy (4.2) with $\ell_k = \mathcal{O}(r_k)$. Further, results on larger values of ℓ_k state that if $\mu(J(x_k)^T) = O(\log^{-3/2}(r_k))$

then $\ell_k = O(r_k \log^2(r_k))$ while if no restrictions are put on the coherence, then $\ell_k = \mathcal{O}(r_k^2)$ is both necessary and sufficient to enforce a subspace embedding for 1-hashing matrices (see e.g., [30, Section 2.3]). Finally, if s-hashing matrices are used with $\ell_k = \mathcal{O}(r_k)$, then the request on $\mu(J(x_k)^T)$ is relaxed by $\sqrt{s}$, see [30, Theorem 2.4.1].

We can now draw conclusions on the size of ℓ_k based on the discussion above. The overall efficiency of Algorithm 2.1 depends on two aspects: the use of embedding with small size ℓ_k and the rate of convergence since the reduction in the cost of minimizing the model $\widehat{m}_k^R$ may be offset by a large number of iterations performed. At this regard, we observe what follows.

- Embedding with small size ℓ_k can be obtained if the rank of the Jacobian matrix is sufficiently small; this fact occurs if the problem (1.1) is strongly underdetermined, i.e., $\text{rank}(J(x)) \leq m \ll n$, or more generally low rank, i.e., $\text{rank}(J(x)) \ll \max\{m, n\}$.
- Though s_k is a descent direction, see Lemma 2.1, it may be a poor descent direction for f at x_k. The rate of convergence depends on whether s_k retains second-order information from the Gauss-Newton model m_k, that ultimately can be measured by the magnitude of θ_k^*.
- If iteration k is true in the sense of Definition 4.1, inequality (2.14) and Lemma 4.2 yield

$$\theta_k^* \leq \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{1/2} \left(\frac{\mu_k}{\lambda_k^{r_k} + \mu_k} + \frac{\lambda_k^1}{\lambda_k^1 + \mu_k}\eta_k^*\right). \tag{4.9}$$

  Then, in case of true iterations we have $\theta_k^* = O\left(\mu_k/\lambda_k^{r_k} + \eta_k^*\right)$ and s_k is an Inexact Gauss-Newton direction corresponding to a forcing term of the order of $\frac{\mu_k}{\lambda_k^{r_k}} + \eta_k^*$.

Algorithm 4.1. Revised step 2 of Algorithm 2.1. Choosing ℓ_{k+1}

Given c, γ, $\widehat{\gamma} \in (0,1)$, $\theta > 0$, $t_{\max}$, $\ell_{\min}, \ell_{\max} \in \mathbb{N}$, $\ell_{\min} < \ell_{\max} \leq n$.
Given x_k, $s_k \in \mathbb{R}^n$, $t_k \in (0, t_{\max}]$, $\ell_k \in \mathbb{N}$.
If $x_k + t_k s_k$ satisfies

$$f(x_k + t_k s_k) < f(x_k) + c t_k s_k^T \nabla f(x_k),$$

Then (successful iteration)
  Set $x_{k+1} = x_k + t_k s_k$, $t_{k+1} = \min\{t_{\max}, \gamma^{-1} t_k\}$.
  Compute θ_k^* in (4.1). If

$$\theta_k^* \leq \theta \tag{4.10}$$

    set $\ell_{k+1} = \max\{\ell_{\min}, \widehat{\gamma}\ell_k\}$
  Else
    set $\ell_{k+1} = \min\{\ell_{\max}, \widehat{\gamma}^{-1}\ell_k\}$.
Else (unsuccessful iteration)
  set $x_{k+1} = x_k$, $t_{k+1} = \gamma t_k$, $\ell_{k+1} = \min\{\ell_{\max}, \widehat{\gamma}^{-1}\ell_k\}$.

In order to adaptively choose ℓ_k and at the same time to monitor the size of θ_k^*, that in case of false iterations can be large, we propose a modification in Step 2 of Algorithm 2.1 as described in Algorithm 4.1. We introduce a prefixed positive threshold θ and test the magnitude of the value θ_k^* defined in (4.1) with respect to θ. Then, we reduce the sketching size only in case of successful iterations such that $\theta_k^* \leq \theta$. Trivially, setting $\theta = \infty$ inhibits the control (4.10). At the extra cost of evaluating the full gradient $\nabla f(x_k)$, Algorithm 4.1 is a practical procedure for the adaptive choice ℓ_k that still allows reductions in the size of the sketching matrices but

exploits more information on the current model m_k and step s_k with respect to Algorithm 2.1. We denote as SLM (Sketched Levenberg-Marquardt) the combination of Algorithms 2.1 and 4.1.

5. Local Analysis of the Sketched Levenberg-Marquardt Algorithm. In this section we focus on the local convergence of a variant of the SLM Algorithm and show that, despite the use of sketching matrices, in case of true iterations we retain the local error decrease of the deterministic inexact Levenberg-Marquardt approach. The procedure is denoted as SLM-local and presented in Algorithm 5.1. We remark that the steplength t_k is maintained fixed throughout the iterations, i.e., $t_k = 1$, $\forall k$.

Algorithm 5.1. Algorithm SLM-local variant: k-th iteration

Given $\widehat{\gamma} \in (0,1), \theta > 0, \eta_{\max}, \mu_{\max} > 0$, $\ell_{\min}, \ell_{\max} \in \mathbb{N}$, $\ell_{\min} < \ell_{\max} \leq n$.
Given $x_k \in \mathbb{R}^n$, $\eta_k \in [0, \eta_{\max}]$, $\mu_k \in (0, \mu_{\max}]$, $\ell_k \in \mathbb{N}$, $\ell_k \in [\ell_{\min}, \ell_{\max}]$.
Step 1. Draw a random matrix from a matrix distribution $M_k \in \mathcal{M}_k$.
Form a random model $\widehat{m}_k^R(\widehat{s})$ of the form (2.3).
Compute the inexact step $\widehat{s}_k$ in (2.5)−(2.7). Let $s_k = M_k^T \widehat{s}_k$.
Step 2. Set $x_{k+1} = x_k + s_k$.
Compute θ_k^* in (4.1). If (4.10) is satisfied then
set $\ell_{k+1} = \max\{\ell_{\min}, \widehat{\gamma}\ell_k\}$
Else
set $\ell_{k+1} = \min\{\ell_{\max}, \widehat{\gamma}^{-1}\ell_k\}$.
Step 3. Choose $\eta_{k+1} \in [0, \eta_{\max}]$, $\mu_{k+1} \in (0, \mu_{\max}]$. Set $k = k + 1$.

Let Ω^* denote the set of all stationary points of f, x^* a point in Ω^*, and given $\zeta \in (0,1)$, let B_ζ be the closed ball of center x^* and radius ζ. Moreover, given any $x \in \mathbb{R}^n$, let $\mathrm{dist}(x, \Omega^*)$ denote the distance between x and Ω^*, i.e.

$$\mathrm{dist}(x, \Omega^*) = \min\{\|x - z\|_2 \mid z \in \Omega^*\}.$$

The convergence analysis follows the path of [1, 19] where exact and inexact deterministic Levenberg-Marquardt methods are studied and it is carried out under the following assumptions.

ASSUMPTION 5.1. *There exists $L_0 > 0$ such that for every $x, y \in B_\zeta$*

$$\|J(x) - J(y)\|_2 \leq L_0 \|x - y\|_2.$$

ASSUMPTION 5.2. *For every $x \in B_\zeta$ we have $\mathrm{rank}(J(x)^T J(x)) = \mathrm{rank}(J(x^*)^T J(x^*)) = r$, for some positive r and there exists a positive $\lambda_{\min}$ such that for every $x \in B_\zeta$*

$$\min\{\lambda > 0 | \lambda \in \mathrm{eig}(J(x)^T J(x))\} \geq \lambda_{\min},$$

where $\mathrm{eig}(J(x)^T J(x))$ denotes the spectrum of $J(x)^T J(x)$.

ASSUMPTION 5.3. *There exists $\omega > 0$ such that for every $x \in B_\zeta$*

$$\omega \, \mathrm{dist}(x, \Omega^*) \leq \|\nabla f(x)\|_2.$$

ASSUMPTION 5.4. *There exists $\sigma > 0$ and $\beta \in [0, 1]$ such that for every $x \in B_\zeta$ and every $z \in B_\zeta \cap \Omega^*$*

$$\|J(x)^T F(z)\|_2 \leq \sigma \|x - z\|_2^{1+\beta}. \tag{5.1}$$

Assumption 5.3 is an error-bound condition with $\|\nabla f(x)\|_2$ as the residual function [26]. This condition is weaker than the full-rank condition, see [1, 5, 19, 26]. We note that Assumption 5.4 is satisfied in case of zero residual problems for any $\sigma \geq 0$. In case of nonzero residual problems it is needed to handle the error due to the employment of $J(x)^T J(x)$ in place of the true Hessian of f.

The following Lemma collects a set of inequalities that follow directly from the Lipschitz continuity of ∇f and J. Note that since F is continuously differentiable, there exists $J_{\max}$ strictly positive such $\|J(x)\|_2 \leq J_{\max}$ for every $x \in B_\zeta$.

LEMMA 5.5. *Suppose that Assumptions 3.1 and 5.1 hold. Then for every $x, y \in B_\zeta$ and every $z \in B_\zeta \cap \Omega^*$,*

1. $\|F(y) - F(x) - J(x)(y-x)\|_2 \leq L_1 \|y-x\|_2^2$, *with* $L_1 = L_0/2$;
2. $\|\nabla f(y) - \nabla f(x) - J(x)^T J(x)(y-x)\|_2 \leq L_2 \|y-x\|_2^2 + \|(J(y)-J(x))^T F(y)\|_2$, *with* $L_2 = L_1 J_{\max}$;
3. $\|(J(y)-J(x))^T F(y)\|_2 \leq L_0 J_{\max}(\|x-z\|_2 \|y-z\|_2 + \|y-z\|_2^2) + \|J(x)^T F(z)\|_2 + \|J(y)^T F(z)\|_2$;
4. $\|J(x)^T F(x)\|_2 \leq L \operatorname{dist}(x, \Omega^*)$.

Proof. See [1, pp. 1102, 1103]. □

We now make the following assumption on the probability of having an iteration true in the sense of Definition 4.1. The σ-algebra $\mathcal{F}_{k-1}$ introduced in Section 3 is invoked below.

ASSUMPTION 5.6. *Let $\widehat{T}_k$ be the random variable such that*

$$\widehat{T}_k = \begin{cases} 1 & \text{if (4.2)-(4.3) hold at iteration } k \\ 0 & \text{otherwise.} \end{cases}. \tag{5.2}$$

There exists $\pi_M \in (0,1)$ such that for every $k \in \mathbb{N}$ we have

$$\mathbb{P}\left(\widehat{T}_k = 1 \mid \mathcal{F}_{k-1}\right) \geq 1 - \pi_M.$$

Moreover, $\mathbb{P}\left(\widehat{T}_0 = 1\right) \geq 1 - \pi_M$, and $\widehat{T}_k$ is conditionally independent on $\widehat{T}_{k-1}, \ldots, \widehat{T}_0$ given $\mathcal{F}_{k-1}$.

The following lemma proves that, for true iterations and suitable choices of μ_k and η_k, the step s_k is bounded by a multiple of the distance of the current iterate x_k from the set Ω^*. While in Algorithms 2.1 and 5.1 the values assigned to the regularization parameter μ_k and the forcing term η_k were not specified, here we enforce conditions on the choice of μ_k and η_k in order to recover fast local decrease at true iterations.

LEMMA 5.7. *Let Assumptions 3.1, 5.1, 5.2 hold and suppose that there exists a positive constant $\bar{c}$ such that $\eta_k/\mu_k = \bar{c}$ for every k. Then, there exists c_1 such that, if $\widehat{T}_k = 1$ and $x_k \in B_\zeta$ then*

$$\|s_k\|_2 \leq c_1 \operatorname{dist}(x_k, \Omega^*).$$

Proof. See the Appendix A.3. □

LEMMA 5.8. *Under the same assumptions as Lemma 5.7 and Assumption 5.4, if $x_k, x_{k+1} \in B_\zeta$ and $\widehat{T}_k = 1$ then there exist $L_3, L_4 > 0$ such that*

$$\omega \operatorname{dist}(x_{k+1}, \Omega^*) \leq L_3 \operatorname{dist}(x_k, \Omega^*)^{1+\beta} + L_4 \operatorname{dist}(x_k, \Omega^*)^2 + \theta_k^* L \operatorname{dist}(x_k, \Omega^*),$$

with θ_k^* as in (4.1).

Proof. See the Appendix A.3. □

To proceed in our analysis we let $\bar{\eta}$, $\bar{\mu}$ be positive scalars, and

$$\bar{\theta} = \frac{(1+\varepsilon)^{1/2}}{(1-\varepsilon)^{1/2}} \left(\frac{\bar{\mu}}{\lambda_{\min}} + \bar{\eta} \right), \tag{5.3}$$

with ε in (4.2), and $\lambda_{\min}$ as in Assumption 5.2. Moreover, for some $\xi \in (0,1)$ let

$$\varsigma = \begin{cases} \min\left\{\zeta, \frac{\xi\omega - L_3}{L_4 + \bar{\theta}L^2}\right\} & \text{if } \beta = 0, \\ \min\left\{\zeta, \left(\frac{\xi\omega}{L_3 + L_4 + \bar{\theta}L^{1+\beta}}\right)^{1/\beta}\right\} & \text{if } \beta \in (0,1]. \end{cases} \tag{5.4}$$

and distinguish the cases $\beta = 0$ and $\beta \in (0,1]$ in Assumption 5.4 and in the following additional assumption.

ASSUMPTION 5.9. *Let ω be the scalar in Assumption 5.3, σ be the scalar in Assumption 5.4, c_1 be the scalar in Lemma 5.7, L_3, L_4 the scalars in Lemma 5.8, $\bar{\theta}$ the scalar in (5.3).*
If $\beta = 0$, suppose that $\sigma < \xi\omega/(2+c_1)$, $\eta_k = \bar{\eta}\|J(x_k)^T F(x_k)\|$, $\mu_k = \bar{\mu}\|J(x_k)^T F(x_k)\|$.
If $\beta \in (0,1]$, suppose that $\eta_k = \bar{\eta}\|J(x_k)^T F(x_k)\|^\beta$, $\mu_k = \bar{\mu}\|J(x_k)^T F(x_k)\|^\beta$.

We remark that η_k/μ_k is constant as supposed in Lemma 5.7. We also note that in case $\beta = 0$, the scalar σ is supposed to be sufficiently small. This is in line with the convergence analysis of Gauss-Newton methods for nonzero residual problems, see [18]. By definition of L_3 (see the proof of lemma 5.8), the additional condition on σ implies $\xi\omega - L_3 > 0$, and consequently ς in (5.4) is strictly positive.

We now prove that, assuming that both x_k and x_{k+1} belong to B_ς, in the true iterations the distance $\operatorname{dist}(x_{k+1}, \Omega^*)$ decreases with respect to $\operatorname{dist}(x_k, \Omega^*)$. Note that, since $\eta_k^* \le \eta_k$ by (2.11), then θ_k^* in (4.9) satisfies

$$\theta_k^* \le \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{1/2} \left(\frac{\mu_k}{\lambda_k^{r_k} + \mu_k} + \frac{\lambda_k^1}{\lambda_k^1 + \mu_k}\eta_k \right). \tag{5.5}$$

LEMMA 5.10. *Let Assumptions 3.1,5.1, 5.2 and 5.3 hold. Given any $\xi \in (0,1)$, $\bar{\eta} \ge 0$ and $\bar{\mu} > 0$, suppose that Assumptions 5.4 and 5.9 hold with $\beta \in [0,1]$. If $\operatorname{dist}(x_k, \Omega^*) \le \varsigma$ with ς given in (5.4), $x_k, x_{k+1} \in B_\varsigma$ and $\widehat{T}_k = 1$, then*

$$\operatorname{dist}(x_{k+1}, \Omega^*) \le \xi \operatorname{dist}(x_k, \Omega^*).$$

Proof. We first consider the case $\beta = 0$. Inequality (5.5), Assumption 5.2, the choice of η_k and μ_k, Item 4 in Lemma 5.5 yield

$$\theta_k^* \le \bar{\theta} L \operatorname{dist}(x_k, \Omega^*), \tag{5.6}$$

with $\bar{\theta}$ given in (5.3). From Lemma 5.8, using $\operatorname{dist}(x_k, \Omega^*) \le \varsigma$, we have

$$\omega \operatorname{dist}(x_{k+1}, \Omega^*) \le L_3 \operatorname{dist}(x_k, \Omega^*) + L_4 \operatorname{dist}(x_k, \Omega^*)^2 + \bar{\theta}L^2 \operatorname{dist}(x_k, \Omega^*)^2. \tag{5.7}$$

This implies

$$\omega \operatorname{dist}(x_{k+1}, \Omega^*) \le (L_3 + (L_4 + \bar{\theta}L^2)\varsigma) \operatorname{dist}(x_k, \Omega^*),$$

and we get the thesis by (5.4).

In case $\beta \in (0,1]$, inequality (5.5), Assumption 5.2, the form of η_k and μ_k, and Item 4 in Lemma 5.5 yield

$$\theta_k^* \leq \bar{\theta} L^{\beta} \operatorname{dist}(x_k, \Omega^*)^{\beta}. \tag{5.8}$$

Using again Lemma 5.8 we get

$$\omega \operatorname{dist}(x_{k+1}, \Omega^*) \leq L_3 \operatorname{dist}(x_k, \Omega^*)^{1+\beta} + L_4 \operatorname{dist}(x_k, \Omega^*)^2 + \bar{\theta} L^{1+\beta} \operatorname{dist}(x_k, \Omega^*)^{1+\beta}. \tag{5.9}$$

Since $\varsigma < 1$ by construction, it follows that

$$\omega \operatorname{dist}(x_{k+1}, \Omega^*) \leq (L_3 + L_4 + \bar{\theta} L^{1+\beta}) \varsigma^{\beta} \operatorname{dist}(x_k, \Omega^*)$$

and the thesis follows using (5.4). ☐

We now prove that if x_0 belongs to B_ς and $\bar{k}$ consecutive iterations are true, then all the iterates $\{x_k\}_{k=0}^{\bar{k}}$ belong to the ball B_ζ and $\operatorname{dist}(x_k, \Omega^*)$ is smaller than some specified positive scalar.

LEMMA 5.11. *Let Assumptions 3.1,5.1, 5.2 and 5.3 hold. Given any $\xi \in (0,1)$, $\bar{\eta} \geq 0$ and $\bar{\mu} > 0$, suppose that Assumptions 5.4 and 5.9 hold with $\beta \in [0,1]$. Let*

$$\bar{\varsigma} = \min \left\{ \varsigma, \frac{\zeta(1-\xi)}{1-\xi+c_1} \right\}. \tag{5.10}$$

If $x_0 \in B_{\bar{\varsigma}}$, and there exists some positive $\bar{k}$ such that $\widehat{T}_k = 1$ for every $k = 0, \dots \bar{k}$, then we have $\operatorname{dist}(x_k, \Omega^*) \leq \bar{\varsigma}$ and $x_{k+1} \in B_\zeta$ for every $k = 0, \dots \bar{k}$.

Proof. The proof is analogous to that of Lemma 4.2 in [1]. ☐

Lemma 5.10 and 5.11 above ensure that for all values of β the distance of x_k from the set of stationary points decreases at least linearly in case of true iterations. We now show that the decrease is superlinear and quadratic when $\beta \in (0,1)$ and $\beta = 1$, respectively.

LEMMA 5.12. *Let the assumptions of Lemma 5.11 hold. Let $\beta \in (0,1]$ and $\bar{\theta}$ given in (5.3). If $x_0 \in B_{\bar{\varsigma}}$ and there exists some positive $\bar{k}$ such that $\widehat{T}_k = 1$ for every $k = 0, \dots \bar{k}$, then we have*

$$\operatorname{dist}(x_{k+1}, \Omega^*) \leq \frac{L_3 + \bar{\theta} L^{1+\beta}}{\omega} \operatorname{dist}(x_k, \Omega^*)^{1+\beta} + \frac{L_4}{\omega} \operatorname{dist}(x_k, \Omega^*)^2, \tag{5.11}$$

for every $k = 0, \dots \bar{k}$.

Proof. Since $x_0 \in B_{\bar{\varsigma}}$, Lemma 5.11 ensures that Lemma 5.8 can be applied for all $k = 0, \dots, \bar{k}$. Then, (5.9) holds and gives the thesis. ☐

The following theorem shows that, in case $\beta = 0$ we retain the linear decrease of Levenberg-Marquardt approaches in $\bar{k}$ consecutive iterations with probability $(1 - \pi_M)^{\bar{k}}$. If Assumption 5.4 holds with $\beta = 1$ we get quadratic decrease with the same probability.

THEOREM 5.13. *Let Assumptions 3.1,5.1, 5.2, 5.3 and 5.6 hold. Given any $\xi \in (0,1)$, $\bar{\eta} \geq 0$ and $\bar{\mu} > 0$, suppose that Assumptions 5.4 and 5.9 hold with $\beta \in [0,1]$.*

If $x_0 \in B_{\bar{\varsigma}}$ with $\bar{\varsigma}$ given in (5.10), then for every $\bar{k} \in \mathbb{N}$ we have

$$\mathbb{P}\left(\operatorname{dist}(x_{\bar{k}}, \Omega^*) \leq \xi^{\bar{k}} \operatorname{dist}(x_0, \Omega^*) \right) \geq (1 - \pi_M)^{\bar{k}} \qquad \text{if } \beta = 0,$$

and

$$\mathbb{P}\left(\operatorname{dist}(x_{\bar{k}}, \Omega^*) \leq C^{\sum_{j=0}^{\bar{k}-1} (1+\beta)^j} \operatorname{dist}(x_0, \Omega^*)^{(1+\beta)^{\bar{k}}} \right) \geq (1 - \pi_M)^{\bar{k}} \qquad \text{if } \beta \in (0,1],$$

with $C = (L_3 + L_4 + \bar{\theta} L^{1+\beta})/\omega$.

Proof. First consider the case $\beta = 0$ and let A_j be the event $\operatorname{dist}(x_{j+1}, \Omega^*) \leq \xi \operatorname{dist}(x_j, \Omega^*)$. We prove by induction that

$$\mathbb{P}\left(\bigcap_{j=0}^{\bar{k}-1} A_j\right) \geq (1 - \pi_M)^{\bar{k}},$$

i.e., the probability of linear reduction of $\operatorname{dist}(x_{j+1}, \Omega^*)$ with respect to $\operatorname{dist}(x_j, \Omega^*)$ for $\bar{k}$ subsequent iteration, $j = 0, \ldots, \bar{k}$, is at least $(1 - \pi_M)^{\bar{k}}$. For $\bar{k} = 1$, by Lemma 5.10 we have that $\operatorname{dist}(x_1, \Omega^*) \leq \xi \operatorname{dist}(x_0, \Omega^*)$ if $\widehat{T}_0 = 1$. Therefore,

$$\mathbb{P}(A_0) \geq \mathbb{P}\left(\widehat{T}_0 = 1\right) \geq 1 - \pi_M.$$

Let us now assume that

$$\mathbb{P}\left(\bigcap_{j=0}^{\bar{k}-2} A_j\right) \geq (1 - \pi_M)^{\bar{k}-1}. \tag{5.12}$$

We have

$$\mathbb{P}\left(\bigcap_{j=0}^{\bar{k}-1} A_j\right) = \mathbb{P}\left(A_{\bar{k}-1} \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right) \mathbb{P}\left(\bigcap_{j=0}^{\bar{k}-2} A_j\right). \tag{5.13}$$

Let us now consider the first term on the right-hand side of (5.13). Lemma 5.10 and 5.11 ensure that

$$\mathbb{P}\left(A_{\bar{k}-1} \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right) \geq \mathbb{P}\left(\widehat{T}_{\bar{k}-1} = 1 \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right) = \mathbb{E}\left[\widehat{T}_{\bar{k}-1} \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right].$$

Then, by the definition of expected value and the law of total expectation we have

$$\begin{aligned}
\mathbb{P}\left(A_{\bar{k}-1} \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right) &\geq \mathbb{E}\left[\mathbb{E}\left[\widehat{T}_{\bar{k}-1} \mid \mathcal{F}_{\bar{k}-2}\right] \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right] \\
&= \mathbb{E}\left[\mathbb{P}\left(\widehat{T}_{\bar{k}-1} = 1 \mid \mathcal{F}_{\bar{k}-2}\right) \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right] \\
&\geq \mathbb{E}\left[(1 - \pi_M) \mid \bigcap_{j=0}^{\bar{k}-2} A_j\right] = 1 - \pi_M.
\end{aligned} \tag{5.14}$$

Using inequalities (5.12) and (5.14) into (5.13), we get

$$\mathbb{P}\left(\bigcap_{j=0}^{\bar{k}-1} A_j\right) \geq (1 - \pi_M)^{\bar{k}}.$$

Therefore

$$\mathbb{P}\left(\operatorname{dist}(x_{\bar{k}}, \Omega^*) \leq \xi^{\bar{k}} \operatorname{dist}(x_0, \Omega^*)\right) \geq \mathbb{P}\left(\bigcap_{j=0}^{\bar{k}-1} A_j\right)$$

which completes the proof.

If $\beta \in (0, 1]$, denoting with A_j the event that inequality (5.11) holds at iteration j, the proof follows the above arguments and invokes Lemma 5.11 and 5.12. ∎

6. Numerical results. In this section, we investigate the numerical performance of the SLM Algorithm. We also consider the deterministic version of Algorithm 2.1 where sketching is not applied, which reduces to the standard Linesearch Levenberg-Marquardt (LLM) procedure since the direction does not change in case of unsuccessful iterations.

Computing s_k amounts to using the QR decomposition if $\eta_k = 0$, and the LSMR algorithm [20] otherwise. In the case $\eta_k = 0$ we measure the computational cost for solving the linear system as $2mn^2 + n^2$ and $2m\ell_k^2 + \ell_k^2$ for LLM and SLM, respectively (see [7, Section 2.7.2] for the computational cost of QR with regularization). If LSMR is applied, the computational cost of the procedure is given by $2m\ell_k q_k$ for the sketched system and $2mnq_k$ for the unsketched system, letting q_k be the number of LSMR iterations performed.

As for the parameters, we set $c = 10^{-4}$, $\gamma = 0.5$, $\widehat{\gamma}^{-1} = 1.1$, $t_{\max} = 1$, $\ell_{\min} = n/10$, $\ell_{\max} = n$, $\mu_k = 10^{-4}$, $\eta_k = \eta$, $\eta \in [0, 1)$, $\forall k \geq 0$. The maximum number of LSMR iterations is set equal to $\min\{m, \ell_k\}$. The considered matrix distribution $\mathcal{M}_k$ consists of 1-hashing matrices of dimension $\ell_k \times n$. We denote SLM$\widehat{p}$ the procedure where the initial sketching size ℓ_0 is $\widehat{p}\%$ of dimension n and specify the couple (η, θ) used in practice. We terminate Algorithms LLM and SLM when either $\|\nabla f(x_k)\|_2 < 10^{-3}$ or 500 nonlinear iterations are performed.

6.1. Problems of varying size and rank. We consider a set of artificially generated low rank problems. Given an optimization problem

$$\min_{y \in \mathbb{R}^p} \|\Phi(y)\|_2^2, \quad \text{with } \Phi : \mathbb{R}^p \longrightarrow \mathbb{R}^m, \tag{6.1}$$

and a size $n > p$, we consider the following augmented problem:

$$\min_{x \in \mathbb{R}^n} f(x) = \|\Phi(Ax)\|_2^2, \tag{6.2}$$

where A is random matrix $p \times n$ with components uniformly distributed in $[0, 1]$ scaled so that $\|A\|_F = 1$. The problems considered are from the CUTEst [21] collection. The number of variables p and observations m of each problem in the collection is determined by a problem-specific parameter d. Table 6.1 displays the relation between m, p and d for the considered problems.

To illustrate the effect of using (4.10), we first consider OSCIGRNE problem with $m = 500$ and $p = 500$, and we define the objective function f as in (6.2) with $n = 1000$. We run LLM and SLM50 with $\eta_k = \eta = 0$, $\forall k \geq 0$, and initial guess $x_0 = (1, \ldots, 1)^T$. In Figure 6.1 we plot the norm of $\nabla f(x_k)$ versus the computational cost and the value ℓ_k versus the iterations. The computational cost is defined as follows. We assign cost m to each evaluation of the residual vector $F(x_k)$, and cost mn to the evaluation of the Jacobian $J(x_k)$. The computational cost for solving the regularized linear system is given by $2m\ell_k^2 + \ell_k^2$. In the computation of θ_k^*, the products $J(x_k)^T F(x_k)$ and $J(x_k)^T J(x_k) s_k$ cost $3mn$ overall. To summarize, the per-iteration cost is given by $2m\ell_k^2 + \ell_k^2 + 4mn + m$.

Figure 6.1 shows that SLM50 is convenient in terms of cost with respect to LLM when $\theta = 10^{-3}$ and $\theta = 10^{-1}$. On the contrary, using $\theta = \infty$ gives poor results. More insight into

Problem	p	m
ARTIF	$d+2$	d
BRATU2D	d^2	$(d-2)^2$
BROYDN3D	d	d
DRCAVTY1	$(d+4)^2$	d^2
FREURONE	d	$2(d-1)$
OSCIGRNE	d	d

Table 6.1: CUTEst problems

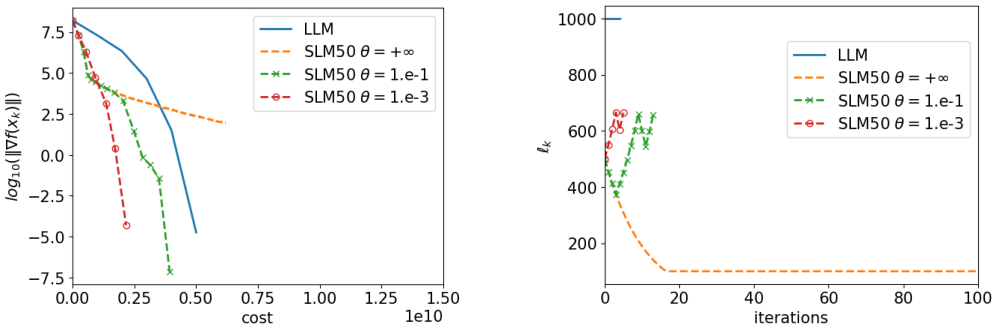


Figure 6.1: OSCIGRNE problem $m = 500$, $n = 1000$. History of LLM and SLM50 $\eta = 0$.

these results is provided by Table 6.2 and 6.3. For the case where the control (4.10) is inhibited, in Table 6.2 we report, for selected iterations k, the values of $f(x_k)$, $\nabla f(x_k)$, ℓ_k along with the relative residuals $\eta_k^*, \nu_k^*, \theta_k^*$, and the occurrence of successful (S), unsuccessful (U) iteration (Itn).

Table 6.3 reports the same data obtained with $\theta = 10^{-1}$ in (4.10).

Table 6.2 shows that the reported iterations are successful and consequently ℓ_k decreases steadily, as shown in Figure 6.1 (right). But the decrease of $f(x_k)$ and $\|\nabla f(x_k)\|_2$ is low starting from $k = 4$, and this behavior can be ascribed to the the quality of s_k with respect to the model m_k in (2.1). The values of η_k^* are nonzero but very small due to round-off precision, the values ν_k^* are small as expected, while the value of θ_k^* are close to one for $k \geq 4$. This occurrence affects the convergence history and SLM50 compares poorly with LLM. On the contrary, Table 6.3 shows that imposing the control (4.10) gives rise to an increase in the size ℓ_k at some iterations, even if the iteration is successful, and greatly improves the performance of SLM50.

Further experiments were carried out solving strongly underdetermined problems where $m = 100$ and the artificial size n is set to 1000. The initial guess was fixed as $x_0 = (1, \ldots, 1)^T$ and SLMp algorithm was run 11 times for each tested value p. The LLM method is deterministic, and therefore it is not necessary to run it repeatedly. We present results obtained with the constant forcing terms $\eta = 10^{-3}$, $\theta \in \{\infty, 10^{-1}\}$, and plot the median one in terms of overall computational cost. The computational cost is measured as described above taking into account that each iteration of LSMR method has cost $2m\ell_k$. Hence, the per-iteration cost is given by $2m\ell_k q_k + 4mn + m$, where q_k is the number of LSMR iterations performed at iteration k. In Figure 6.2 the norm of the gradient is plotted against the cost, in Figure 6.3 the subspace dimension ℓ_k is displayed versus the iterations.

The sketched algorithms with $\theta = 10^{-1}$ perform well compared to LLM algorithm on all

k	$f(x_k)$	$\nabla f(x_k)$	ℓ_k	η_k^*	ν_k^*	θ_k^*	Itn
0	3.50e+8	1.65e+8	500	2.41e−16	1.83e−11	1.93e−3	S
1	1.73e+7	2.06e+7	454	2.61e−16	8.18e−11	4.54e−3	S
2	4.73e+5	1.94e+6	412	5.13e−16	4.17e−10	2.65e−2	S
3	1.10e+5	7.37e+4	374	1.03e−14	5.72e−9	5.46e−1	S
4	6.97e+4	4.01e+4	340	8.10e−15	8.17e−9	7.42e−1	S
5	4.70e+4	2.98e+4	309	6.56e−15	5.52e−9	8.82e−1	S
6	3.69e+4	2.63e+4	280	6.61e−15	5.34e−9	9.02e−1	S
7	2.99e+4	2.37e+4	254	4.93e−15	4.37e−9	9.53e−1	S
8	2.38e+4	2.26e+4	230	3.96e−15	2.60e−9	8.29e−1	S
9	2.02e+4	1.87e+4	209	2.86e−15	1.88e−9	9.16e−1	S
10	1.84e+4	1.87e+4	189	3.54e−15	1.83e−9	9.11e−1	S
100	1.82e+3	3.93e+3	100	1.48e−15	6.03e−10	9.74e−1	S
200	2.40e+2	1.36e+3	100	1.30e−15	5.23e−10	9.79e−1	S
300	3.83e+1	5.10e+2	100	1.10e−15	5.32e−10	1.01e+0	S
400	7.35e+0	2.30e+2	100	1.27e−15	6.44e−10	1.02e+0	S

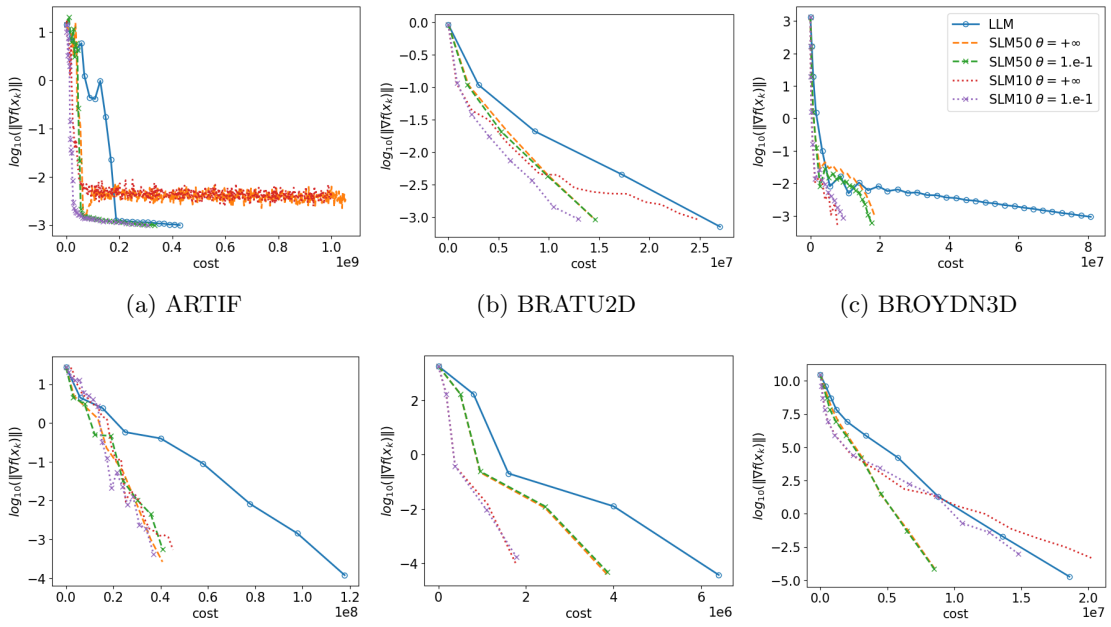
Table 6.2: OSCIGRNE problem $m = 500$, $n = 1000$. History of SLM50 along the iterations, $\eta = 0$, $\theta = +\infty$. Itn: successful iteration (S), unsuccessful iteration (U).

k	$f(x_k)$	$\nabla f(x_k)$	ℓ_k	η_k^*	ν_k^*	θ_k^*	Itn
0	3.50e+8	1.64e+8	500	4.20e−16	2.35e−11	1.54e−3	S
1	1.72e+7	2.07e+7	454	2.43e−16	6.57e−11	4.38e−3	S
2	4.53e+5	1.94e+6	412	5.90e−16	4.19e−10	2.46e−2	S
3	9.44e+4	7.05e+4	374	6.07e−15	5.26e−9	4.98e−1	S
4	6.06e+4	3.50e+4	411	1.53e−14	1.09e−8	8.43e−1	S
5	3.81e+4	2.96e+4	452	3.88e−14	2.11e−8	5.91e−1	S
6	1.76e+4	1.75e+4	497	2.82e−14	2.89e−8	7.11e−1	S
7	8.04e+3	1.25e+4	546	4.35e−14	3.33e−8	5.23e−1	S
8	2.08e+3	6.54e+3	600	2.09e−13	1.48e−7	3.07e−1	S
9	2.24e+2	2.02e+3	660	9.74e−14	7.76e−8	1.42e−6	S
10	1.64e−4	2.84e+1	600	6.05e−15	5.69e−9	9.43e−3	S
11	3.51e−6	2.67e−1	545	2.82e−14	3.05e−8	5.08e−1	S
12	1.14e−6	1.36e−1	599	8.75e−14	8.06e−8	3.73e−1	S
13	1.29e−7	5.06e−2	658	9.08e−14	8.26e−8	1.67e−6	S
14	5.0e−19	8.67e−8					

Table 6.3: OSCIGRNE problem $m = 500$, $n = 1000$. History of LLM and SLM50 along the iterations, $\eta = 0$, $\theta = 10^{-1}$. Itn: successful iteration (S), unsuccessful iteration (U).

the considered problems. In the solution of DRCAVTY1 problem, SLM10 and SLM50 perform significantly better than LLM for all θ. In the solution of BROYDN3D and FREURONE problems, SLM10 algorithm appears to be the most effective and the value θ used does not affect the performance significantly. For problem OSCIGRNE SLM50 is significantly cheaper than LLM and is not affected by the choice of θ, while SLM10 is comparable to LLM for $\theta = 10^{-1}$ and more costly for $\theta = +\infty$. In the solution of problem BRATU2D, SLM50 with both choices of θ and SLM10 with $\theta = 10^{-1}$ perform similarly and significantly better than LLM, while the method

that employs $\ell_0 = 10$ and $\theta = +\infty$ is comparable to LLM. ARTIF is the only considered problem where employing the sketching does not seem to yield significant gains in terms of computational cost. If (4.10) is inhibited, runs fails while using $\theta = 10^{-1}$ yields to $\ell_k = n$ in the last iterations, as shown in Figure 6.3. For the other problems, Figure 6.3 shows that the sketching size ℓ_k is truly adaptive and changes along the iterations. It's behavior heavily depends on the parameters θ, ℓ_0 employed and on the specific problem. We note that some curves of SLM$\widehat{p}$ are overlapped for a given $\widehat{p}$ and varying θ.

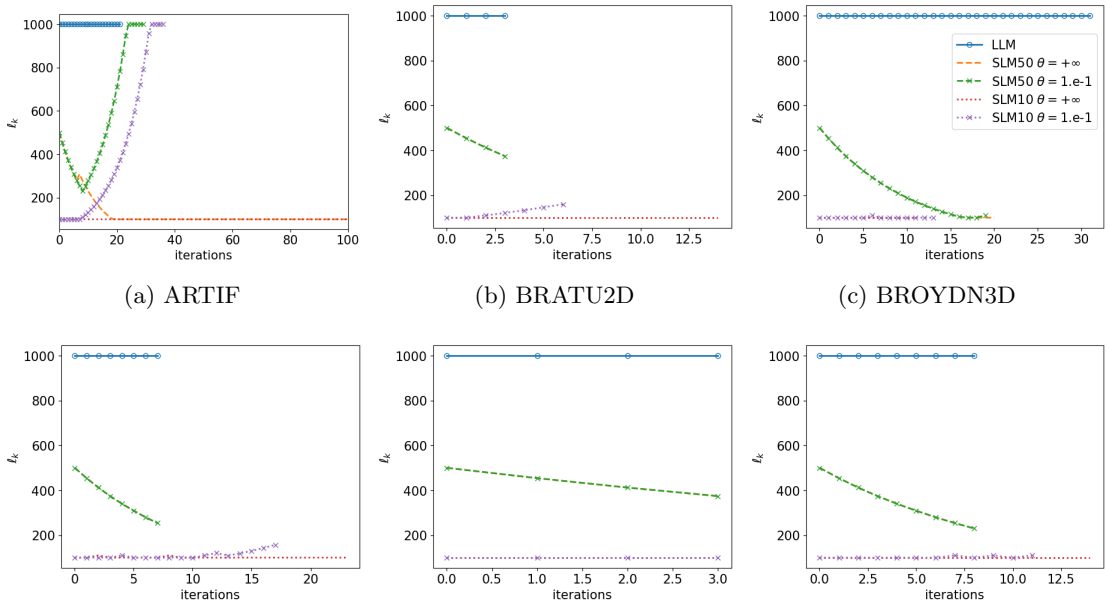


(a) ARTIF (b) BRATU2D (c) BROYDN3D

(d) DRCAVTY1 (e) FREURONE (f) OSCIGRNE

Figure 6.2: CUTEst problems solved with LLM, SLM10 and SLM50, $\eta = 10^{-3}, \theta \in \{+\infty, 10^{-1}\}$. Norm of $\nabla f(x_k)$ vs computational cost.

We conclude the current set of experiments showing that the adaptive choice of the sketching size positively affects the performance of the algorithm SLM. Hence, we repeat the tests conducted above using constant sketching dimension, i.e., $\ell_k = \ell_0$, $\forall k$. All other parameters are set as in the previous tests. We solved BROYDN3D, DRCAVTY1 and OSCIGRNE, $m = 100$, $n = 1000$, with $\ell_0 \in \{750, 500, 100\}$, corresponding to $75\%, 50\%$ and 10% of the problem dimension, respectively. We denote SLM$\widehat{p}$_fixed the corresponding algorithm. The results are reported in Figure 6.4 and we are interested in comparing such results with those in Figure 6.2 obtained with the adaptive strategy. We already noticed in Figure 6.3 that, on the considered problems, for SLM10 with $\theta = +\infty$, the size ℓ_k remains constant and always equal to 100; therefore the behavior of SLM10 and SLM10_fixed are analogous. Regarding the fixed values $\ell_k = 500$ and $\ell_k = 750$, $\forall k$, the performance of SLM75_fixed is significantly worse than that of SLM50_fixed and SLM50, and in two problems out of three, the overall cost of SLM75_fixed is comparable to that of LLM. Moreover, for the three considered problems, the final computational cost of the SLM50_fixed is significantly higher than the cost of SLM50. Overall the results of Figures 6.2-6.4 suggest that SLM with constant sketching size can work well in practice, but the most

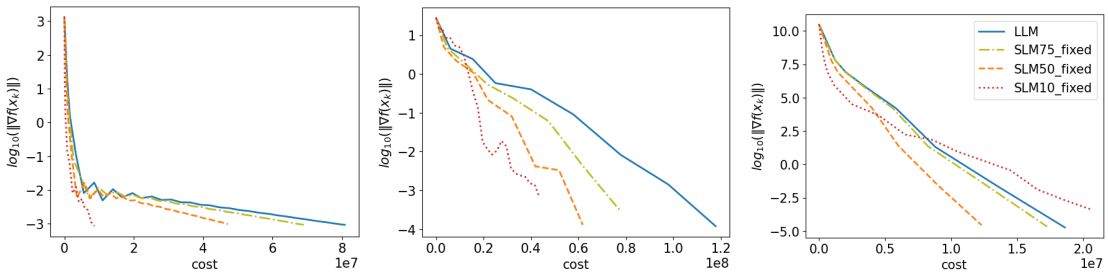


(a) ARTIF (b) BRATU2D (c) BROYDN3D

(d) DRCAVTY1 (e) FREURONE (f) OSCIGRNE

Figure 6.3: CUTEst problems solved with LLM, SLM10 and SLM50, $\eta = 10^{-3}, \theta \in \{+\infty, 10^{-1}\}$. Sketching size ℓ_k vs iterations.

effective sketching size seems to depend heavily on the considered problem. Hence, employing an adaptive strategy for the choosing ℓ_k seems to improve the robustness and the performance of the SLM strategy.



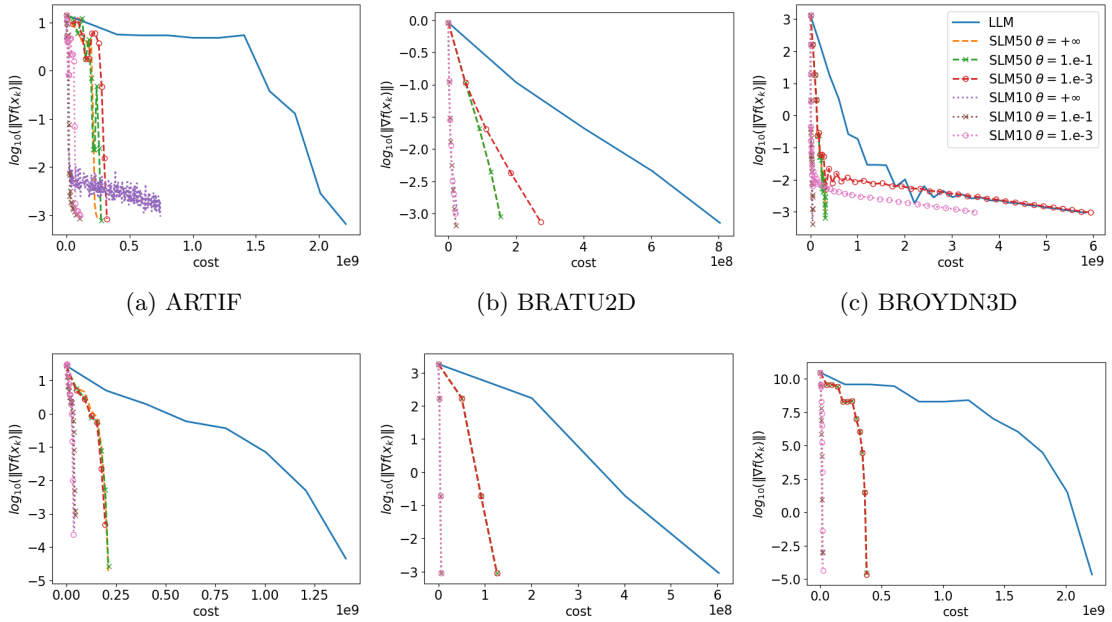
(a) BROYDN3D (b) DRCAVTY1 (c) OSCIGRNE

Figure 6.4: CUTEst problems solved with $\eta = 10^{-3}$ and constant $\ell_k = \ell_0$, $\forall k$. Results for LLM, SLM75_fixed, SLM50_fixed and SLM10_fixed. Norm of $\nabla f(x_k)$ vs computational cost.

We repeated the tests in Figures 6.2, 6.3 using a direct method for the solution of the linear system, $\eta = 0$. The results are reported in Figures 6.5 and 6.6. Figure 6.5 shows that SLM10 and SLM50 procedures are effective in all runs except the case when $\theta = 10^{-3}$ is used for problem

BROYDN3D. In such costly run, Figure 6.6 displays that the condition (4.10) is not satisfied and the embedding size increases steadily.

Summarizing the results presented, SLM$\widehat{p}$ showed to be effective in terms of computational cost. For some problems, testing the condition (4.10) was crucial for improving the performance of SLMp algorithm; further using moderate values of θ, such as $\theta = 10^{-1}$ did not deteriorate the behavior of the sketched algorithm.



(a) ARTIF (b) BRATU2D (c) BROYDN3D

(d) DRCAVTY1 (e) FREURONE (f) OSCIGRNE

Figure 6.5: CUTEst problems solved with LLM, SLM10 and SLM50, $\eta = 0$, $\theta \in \{+\infty, 10^{-1}, 10^{-3}\}$. Norm of $\nabla f(x_k)$ vs computational cost.

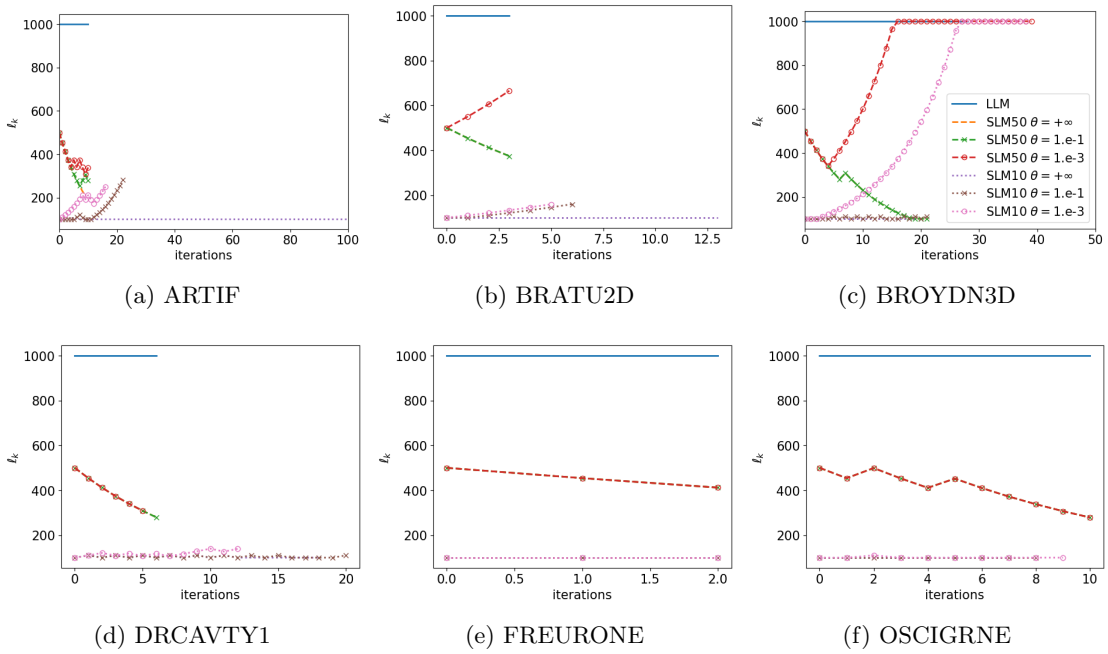
6.2. Binary Classification. We consider a binary classification problem with logistic model and least-squares loss of the form (1.1) where

$$F_i(x) = b_i - \frac{1}{1 + e^{-x^T a_i}}, \quad i = 1, \ldots, m, \tag{6.3}$$

and $a_i \in \mathbb{R}^n$, $b_i \in \{0, 1\}$ are the features vectors and the labels of the training set respectively.

The datasets used are GISETTE [23] and REJAFADA [29]. Regarding GISETTE, the problem dimension is $n = 5000$, $m = 6000$ samples were used as the training set and the validation set has size 1000. Regarding REJAFADA, the problem dimension is $n = 6824$. Out of the 1996 couples $\{(a_i, b_i)\}$, $m = 1597$ couples were used as training set to define problem (6.3) while the remaining 399 couples were used as validation set. The corresponding least-squares problem is in this case underdetermined. The accuracy in the classification problems is measured as the percentage of labels correctly predicted in the validation set.

We solved this problem with LLM and SLM Algorithms and null initial guess x_0. We present results obtained using SLM10 and SLM50 with constant forcing terms $\eta_k = \eta = 10^{-3}$, $\forall k \geq 0$,



(a) ARTIF (b) BRATU2D (c) BROYDN3D

(d) DRCAVTY1 (e) FREURONE (f) OSCIGRNE

Figure 6.6: CUTEst problems solved with LLM, SLM10 and SLM50, $\eta = 0$, $\theta \in \{+\infty, 10^{-1}, 10^{-3}\}$. Embedding size ℓ_k vs iterations.

and $\theta \in \{+\infty, 10^{-1}\}$. For every couple of parameters (η, θ), we run SLM 11 times. The results obtained are reported in Figures 6.7–6.9.

In Figure 6.7, for each run of LLM and SLM and each pair (η, θ) we plot the accuracy at termination versus the total computational cost. The computational cost is defined as follows. Each evaluation of the residual function $F(x_k)$ requires the computation of m scalar products of the form $a_i^T x$ and the overall cost is mn. Such scalar products can be stored and used to evaluate the Jacobian $J(x_k)$, whose cost can therefore be disregarded. The evaluation of the gradient $J(x_k)^T F(x_k)$ has cost mn and the evaluation of $J(x_k)^T J(x_k) s_k$ has cost $2mn$. Finally, each iteration of LSMR method requires two matrix vector products of sizes $m\ell_k$. To summarize, the per-iteration cost is given by $2m\ell_k q_k + 4mn$, where q_k is the number of LSMR iterations performed at outer-iteration k. For the LLM method, such cost is evaluated setting $\ell_k = n$, $\forall k \geq 0$. In figure 6.8 we plot the norm of the gradient $\nabla f(x_k)$ versus the computational cost, for the median run in terms of overall computational cost. Figure 6.9 shows how the sketching size ℓ_k evolves through the iterations. For each algorithm and pair (η, θ), we plot the results that correspond to the median run in terms of the final computational cost.

In Figure 6.7 we can notice that all runs achieve approximately the same accuracy on the validation set. However, the overall computational cost is significantly smaller for the SLM algorithms compared to the LLM algorithm, except for two runs of SLM10 applied to GISETTE problem, and the best results are obtained by the SLM10 algorithm. The savings obtained with the SLM Algorithm are also shown in Figure 6.8.

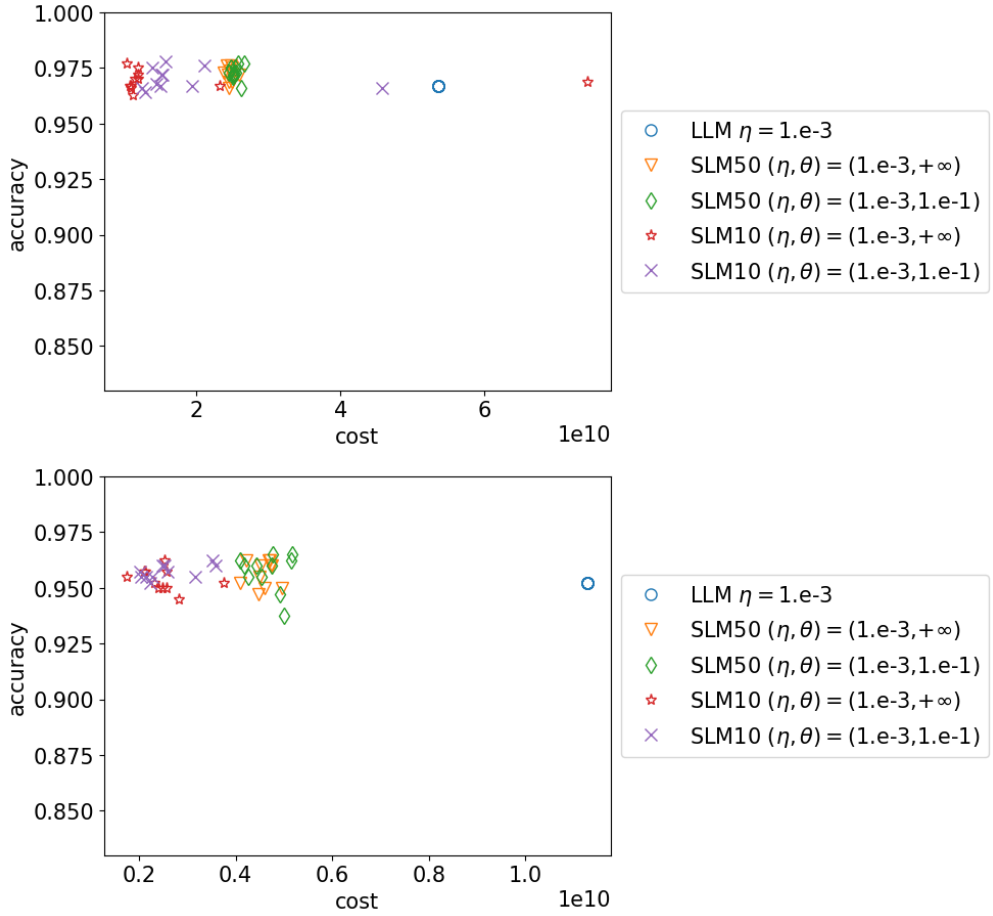


Figure 6.7: Accuracy at termination versus total computational cost. Upper: GISETTE dataset. Lower: REJAFADA dataset.

A. Appendix.

A.1. Matrix distributions. We summarize the definition of some matrix distributions of interest and specify their parameters with respect to properties (3.2) and (3.3). We denote $M_{i,j}$ the entries of a matrix M.

DEFINITION A.1. *$M \in \mathbb{R}^{\ell \times n}$ is a Scaled Gaussian matrix if its entries $M_{i,j}$ for $i = 1, \ldots, \ell$, $j = 1, \ldots, n$ are i.i.d. and distributed as $\mathcal{N}(0, \ell^{-1})$.*

DEFINITION A.2. *Given $\ell \leq n$, $s \leq \ell$, $M \in \mathbb{R}^{\ell \times n}$ is an s-hashing matrix if for every column index $j \in \{1, \ldots, n\}$ we sample without replacement $i_1, \ldots i_s$ uniformly at random and set $M_{i_p,j} = \pm 1/\sqrt{s}$, $p = 1, \ldots, s$.*

DEFINITION A.3. *Let $\ell \in \mathbb{N}^+$ with $\ell < n$. A stable 1-hashing matrix $M \in \mathbb{R}^{\ell \times n}$ has one non-zero per column, whose value is ± 1 with equal probability, with the row indices of the non-zeros given by the sequence $\mathcal{I}$ constructed as follows. Repeat the set $\{1, 2, \ldots, \ell\}$ for $\lceil n/\ell \rceil$ times*

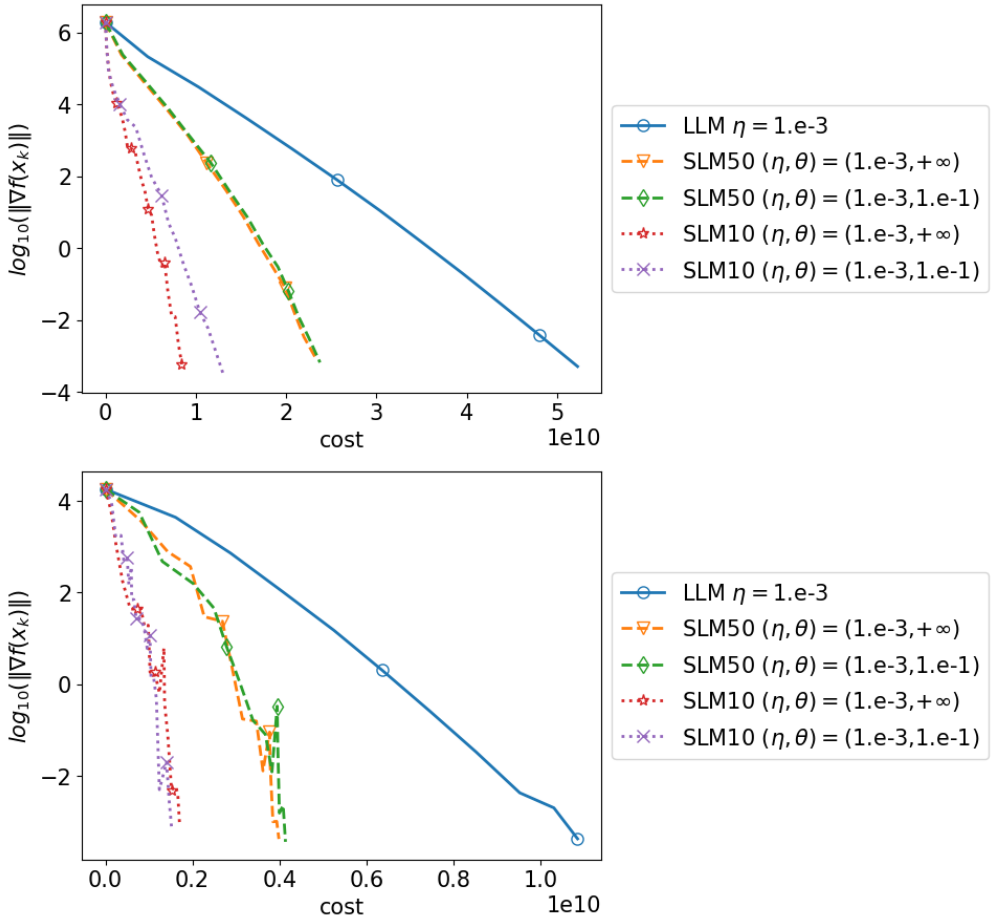


Figure 6.8: Norm of the gradient $\nabla f(x_k)$ versus computational cost. Upper: GISETTE dataset. Lower: REJAFADA dataset.

to obtain a set $\mathcal{D}$. Then randomly sample n elements from $\mathcal{D}$ without replacement to construct the sequence $\mathcal{I}$.

DEFINITION A.4. *$M \in \mathbb{R}^{\ell \times n}$ is a Scaled Sampling matrix if for every row index $i = 1, \ldots, \ell$ we sample $j \in \{1, \ldots, n\}$ uniformly at random and set $M_{i,j} = \sqrt{n/m}$.*

For the classes of matrices introduced above, Table A.1 from [30, Table 4.2] summarizes the value of ε, $M_{\max}$, $\delta_M^{(1)}$, $\delta_M^{(2)}$, ℓ for the fulfillment of (3.2) and (3.3). Notice that (3.3) holds with probability 1 for all considered matrices except for scaled Gaussian matrices, and that the value $M_{\max}$ decreases as ℓ increases for stable 1-hashing and scaled sampling matrices.

A.2. Proof from Section 3.

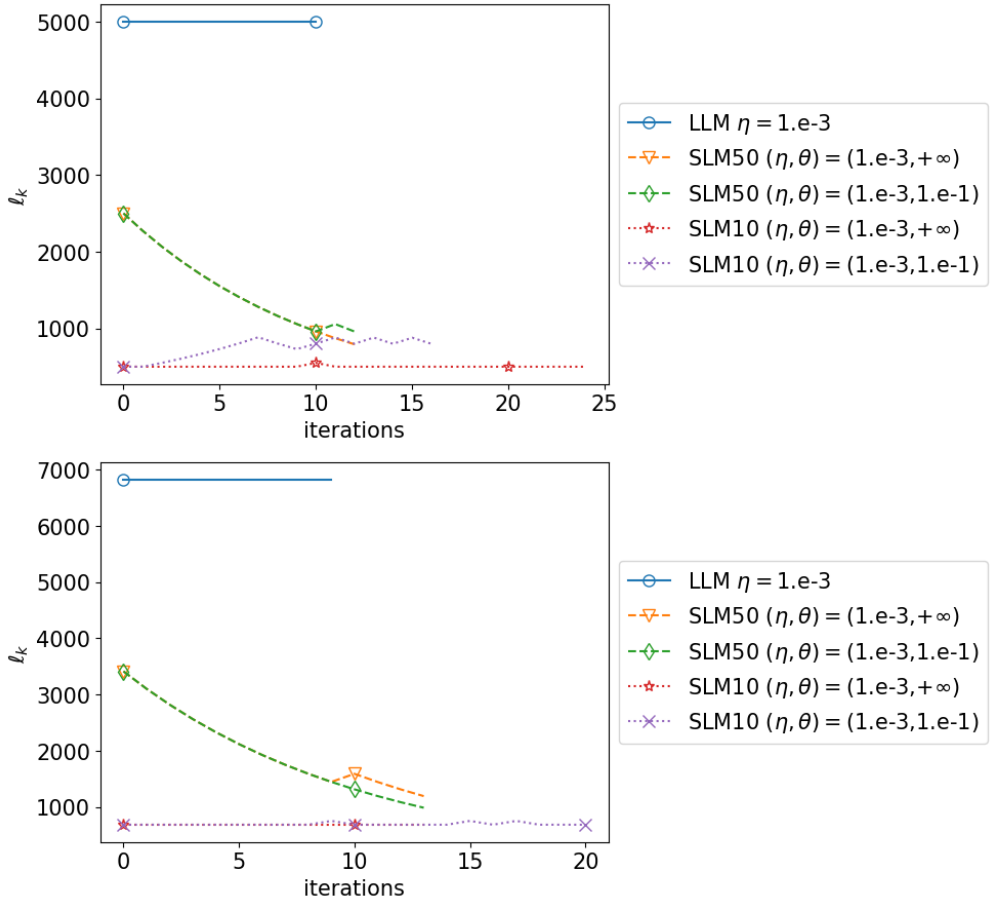


Figure 6.9: Sketching dimension ℓ_k at each iteration. Upper: GISETTE dataset. Lower: RE-JAFADA dataset.

Proof. [Proof of Lemma 3.8] *i*) We proceed by induction on N and consider $N = 1$ first. Since $= e^{-\lambda x}$ is convex, we have $e^{-\lambda T_0} \leq 1 + (e^{-\lambda} - 1)T_0$ and

$$\mathbb{E}\left[e^{-\lambda T_0}\right] \leq 1 + (e^{-\lambda} - 1)\mathbb{E}\left[T_0\right]. \tag{A.1}$$

Moreover, we have $\mathbb{E}\left[T_0\right] \geq \mathbb{P}(T_0 = 1) \geq 1 - \delta_M$, where the first inequality is due to $T_0 \geq 0$, and the second inequality to Assumption 3.6. Therefore, noting that $e^{-\lambda} - 1 < 0$, (A.1) gives

$$\mathbb{E}\left[e^{-\lambda T_0}\right] \leq 1 + (e^{-\lambda} - 1)(1 - \delta_M) \leq e^{(e^{-\lambda}-1)(1-\delta_M)}, \tag{A.2}$$

where the last inequality comes from $1 + y \leq e^y$ for $y \in \mathbb{R}$.

Now assume $\mathbb{E}\left[e^{-\lambda \sum_{k=0}^{N-2} T_k}\right] \leq \left[e^{(e^{-\lambda}-1)(1-\delta_M)}\right]^{N-1}$. Due to the Tower property, we have

$$\mathbb{E}\left[e^{-\lambda \sum_{k=0}^{N-1} T_k}\right] = \mathbb{E}\left[\mathbb{E}\left[e^{-\lambda \sum_{k=0}^{N-1} T_k} \,|\, \mathcal{F}_{N-2}\right]\right],$$

	ε	$\delta_M^{(1)}$	ℓ	$\delta_M^{(2)}$	$M_{\max}$
Scaled Gaussian	(0,1)	$e^{-\frac{\varepsilon^2\ell}{4}}$	$\frac{4}{\varepsilon^2}\log\left(\frac{1}{\delta_M^{(1)}}\right)$	(0,1)	$1+\sqrt{\frac{n}{\ell}}+\sqrt{\frac{2\log(1/\delta_M^{(2)})}{\ell}}$
s-hashing	(0,1)	$e^{-\frac{\varepsilon^2\ell}{C_1}}$	$\frac{C_1}{\varepsilon^2}\log\left(\frac{1}{\delta_M^{(1)}}\right)$	0	$\sqrt{\frac{n}{s}}$
Stable 1-hashing	(0,3/4)	$e^{-\frac{(\varepsilon-1/4)^2\ell}{C_1}}$	$\frac{C_3}{(\varepsilon-1/4)^2}\log\left(\frac{1}{\delta_M^{(1)}}\right)$	0	$\sqrt{\lceil\frac{n}{\ell}\rceil}$
Scaled Sampling	(0,1)	$e^{-\frac{\varepsilon^2\ell}{2n\nu^2}}$	$\frac{2n\nu^2}{\varepsilon^2}\log\left(\frac{1}{\delta_M^{(1)}}\right)$	0	$\sqrt{\frac{n}{\ell}}$

Table A.1: Values of $\varepsilon, M_{\max}, \delta_M^{(1)}$ and $\delta_M^{(2)}$ in (3.2) and (3.3) for different classes of matrices

and

$$\begin{aligned}\mathbb{E}\left[e^{-\lambda\sum_{k=0}^{N-1}T_k}|\mathcal{F}_{N-2}\right] &= \mathbb{E}\left[\prod_{k=0}^{N-1}e^{-\lambda T_k}\,|\,\mathcal{F}_{N-2}\right]\\ &= \prod_{k=0}^{N-2}e^{-\lambda T_k}\mathbb{E}\left[e^{-\lambda T_{N-1}}\,|\,\mathcal{F}_{N-2}\right] \qquad (A.3)\\ &\le e^{(e^{-\lambda}-1)(1-\delta_M)}\prod_{k=0}^{N-2}e^{-\lambda T_k}, \qquad (A.4)\end{aligned}$$

since T_{N-1} is conditionally independent of the past iterations $T_0,\dots,T_{N-1}$, and in the last inequality we used (A.2) and the arguments for the case $N>1$ (from Assumption 3.6, $\mathbb{E}\left[T_{N-1}\,|\,\mathcal{F}_{N-2}\right]=\mathbb{P}(T_{N-1}=1\,|\,\mathcal{F}_{N-2})\ge 1-\delta_M$). Hence, induction implies

$$\begin{aligned}\mathbb{E}\left[e^{-\lambda\sum_{k=0}^{N-1}T_k}\right] &\le e^{(e^{-\lambda}-1)(1-\delta_M)}\mathbb{E}\left[\prod_{k=0}^{N-2}e^{-\lambda T_k}\right]\\ &\le e^{(e^{-\lambda}-1)(1-\delta_M)}\left[e^{(e^{-\lambda}-1)(1-\delta_M)}\right]^{N-1},\end{aligned}$$

and the claim in Item i) follows.

ii) See [13, Proof of Lemma A.1]. □

A.3. Proofs from Section 5.

Proof. [Proof of Lemma 5.7] By (4.5) and Assumption 5.2 we have $\lambda_k^{r_k}\ge(1-\varepsilon)\lambda_r(J(x_k)^TJ(x_k))\ge(1-\varepsilon)\lambda_{\min}$. Using (2.15), the fact that iteration k is true, Item 4 in Lemma 5.5, and the assumption on η_k/μ_k we have

$$\begin{aligned}\|s_k\|_2 =&\|M_k^T\widehat{s}_k\|_2 \le M_{\max}\left(\frac{1}{(1-\varepsilon)\lambda_{\min}+\mu_k}+\frac{\eta_k}{\mu_k}\right)\|M_kJ(x_k)^TF(x_k)\|_2\\ &\le M_{\max}(1+\varepsilon)^{1/2}\left(\frac{1}{(1-\varepsilon)\lambda_{\min}+\mu_k}+\frac{\eta_k}{\mu_k}\right)\|J(x_k)^TF(x_k)\|_2 \qquad (A.5)\\ &\le M_{\max}(1+\varepsilon)^{1/2}\left(\frac{1}{(1-\varepsilon)\lambda_{\min}}+\bar{c}\right)L\,\mathrm{dist}(x_k,\Omega^*),\end{aligned}$$

therefore the thesis holds with $c_1=M_{\max}(1+\varepsilon)^{1/2}\left(\frac{1}{(1-\varepsilon)\lambda_{\min}}+\bar{c}\right)L$. □

Proof. [Proof of Lemma 5.8] By Assumption 5.3, the definition of θ_k^* in (4.1), and Item 4 in Lemma 5.5, we have

$$\begin{aligned}\omega \operatorname{dist}(x_{k+1}, \Omega^*) &\leq \|\nabla f(x_{k+1})\|_2 \leq \|\nabla f(x_{k+1}) - \nabla f(x_k) - J(x_k)^T J(x_k) s_k\|_2 + \\ &\quad \|\nabla f(x_k) + J(x_k)^T J(x_k) s_k\|_2 \\ &= \|\nabla f(x_{k+1}) - \nabla f(x_k) - J(x_k)^T J(x_k) s_k\|_2 + \theta_k^* \|J(x_k)^T F(x_k)\|_2 \\ &\leq \|\nabla f(x_{k+1}) - \nabla f(x_k) - J(x_k)^T J(x_k) s_k\|_2 + \theta_k^* L \operatorname{dist}(x_k, \Omega^*). \end{aligned} \tag{A.6}$$

Using Items 2 and 3 in Lemma 5.5, (5.1) and proceeding as in Lemma 4.1 in [1], we get the thesis, with $L_3 = \sigma(1 + (1 + c_1)^{1+\beta})$ and $L_4 = L_2 c_1^2 + L_0 J_{\max}(1 + c_1)(2 + c_1)$. □

REFERENCES

[1] Behling, R., Goncalves, D.S., Santos, S.A., *Local Convergence Analysis of the Levenberg-Marquardt Framework for Nonzero-Residue Nonlinear Least-Squares Problems Under an Error Bound Condition*, Journal of Optimization Theory and Applications, 183, pp. 1099–1122, 2019.

[2] S. Bellavia, G. Gurioli, B. Morini, P. L. Toint, *Trust-region algorithms: probabilistic complexity and intrinsic noise with applications to subsampling techniques*, EURO Journal on Computational Optimization, 10, pp. 1–37, 2022.

[3] S. Bellavia, G. Malaspina, B. Morini, *Inexact Gauss-Newton methods with matrix approximation by sampling for nonlinear least-squares and systems*, Mathematics of Computation, DOI: https://doi.org/10.1090/mcom/4073, 2025

[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, pp. 53-84 2023.

[5] S. Bellavia, B. Morini, *Strong local convergence properties of adaptive regularized methods for nonlinear least squares*, IMA Journal of Numerical Analysis, 35, pp. 947-968, 2015.

[6] A.S. Berahas, L. Cao, K. Scheinberg, *Global convergence rate analysis of a generic line search algorithm with noise*, SIAM Journal on Optimization, 31, pp. 1079–1603, 2021.

[7] A. Bjork, Numerical Methods for Least Squares Problems, Society for Industrial and Applied Mathematics, 1996

[8] I. Bongartz, A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *CUTE: Constrained and Unconstrained Testing Environment.* ACM Transactions on Mathematical Software, 21(1), pp. 123–160, 1995.

[9] J. Blanchet, C. Cartis, M. Menickelly, K. Scheinberg, *Convergence Rate Analysis of a Stochastic Trust Region Method via Submartingales*, INFORMS Journal on Optimization, 1, pp. 92-119, 2019.

[10] R.H. Byrd , G.M. Chin, W. Neveitt, J. Nocedal, *On the Use of Stochastic Hessian Information in Optimization Methods for Machine Learning*, SIAM Journal on Optimization, 21(3), pp. 977-995, 2011.

[11] R.H. Byrd, G.M. Chin, J. Nocedal, Y. Wu *Sample size selection in optimization methods for machine learning*, Mathematical Programming, 134(1), pp. 127-155, 2012.

[12] R.H. Byrd, S.L. Hansen, J. Nocedal, Y. Singer, *A Stochastic Quasi-Newton Method for Large-Scale Optimization*, SIAM Journal on Optimization, 26(2), pp. 1008–1021, 2016.

[13] C. Cartis, J. Fowkes, Z. Shao, *Randomised Subspace Methods for Non-Convex Optimization, with Applications to Nonlinear Least-Squares*, 2022, arXiv:2211.09873v1.

[14] C. Cartis, J. Fowkes, Z. Shao, *A Randomised Subspace Gauss-Newton Method for Nonlinear Least-Squares*, 2022, arXiv:2211.05727v1.

[15] C. Cartis, Z. Shao, E. Tansley, *Random Subspace Cubic-Regularization Methods with applications to Low-Rank Functions*, 2024, arXiv:2501.09734v1

[16] C. Cartis, K. Scheinberg, *Global convergence rate analysis of unconstrained optimization methods based on probabilistic model*, Mathematical Programming, 169, pp. 337–375, 2017.

[17] R. Chen, M. Menickelly, K. Scheinberg, *Stochastic optimization using a trust-region method and random models*, Mathematical Programming, 169, pp. 447-487, 2018.

[18] J.E. Dennis, R.B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice Hall, Englewood Cliffs, NJ, 1983.

[19] L. Fodor, D. Jakovetić, N. Krejić, G. Malaspina, *Parallel Inexact Levenberg-Marquardt Method for Nearly-Separable Nonlinear Least Squares*, arXiv preprint arXiv:2312.09064, 2023.

[20] D. C.-L. Fong, M. A. Saunders, *LSMR: An iterative algorithm for sparse least-squares problems*, SIAM J. Sci. Comput. 33:5, 2950-2971, 2011

[21] N. I. M. Gould, D. Orban, and Ph. L. Toint. *CUTEst: a constrained and unconstrained testing environment with safe threads for mathematical optimization.* Computational Optimization and Applications, 60(3):545-557, 2015.
[22] B. Jin, K. Scheinberg, M. Xie, *High Probability Complexity Bounds for Adaptive Step Search Based on Stochastic Oracles*, SIAM Journal on Optimization, 34(3), pp. 2169–3166, 2024.
[23] I. Guyon, S. Gunn, A. Ben-Hur, G. Dror, Gisette [Dataset], UCI Machine Learning Repository, 2004.
[24] M. W. Mahoney, *Randomized Algorithms for Matrices and Data*, Fund. Trends Mach. Learn., 3(2), pp. 123–224, 2011.
[25] P.G. Martinsson, J. A. Tropp, *Randomized numerical linear algebra: Foundations and algorithms*, Acta Numerica, 29, pp. 403–572, 2020.
[26] J.S. Pang, *Error bounds in mathematical programming Mathematical Programming*, 79, pp. 299-332, 1997.
[27] C. Paquette, K. Scheinberg, *A Stochastic Line Search Method with Expected Complexity Analysis*, SIAM Journal of Optimization, 30, pp. 349–376, 2020.
[28] M. Pilanci, M.J. Wainwright, *Newton sketch: A near linear-time optimization algorithm with linear-quadratic convergence*, SIAM Journal on Optimization 27(1), pp. 205–245, 2017.
[29] R. Pinheiro, S. M. L. de Lima, S. Murilo, E. Albuquerque, D. Souza, T. Monteiro, P. Lopes, R. Lima, J. Oliveira, S. Silva, REJAFADA [Dataset], UCI Machine Learning Repository, 2019.
[30] Z. Shao, *On Random Embeddings and Their Application to Optimization*, PhD thesis, University of Oxford, Oxford, UK, 2021.
[31] J.A. Tropp, *An Introduction to Matrix Concentration Inequalities*, Foundations and Trends in Machine Learning, 8, pp. 1–230, 2015.
[32] R. Yuan, A. Lazaric, R. M. Gower, *Sketched Newton–Raphson*, SIAM Journal on Optimization, 32, pp. 1499-2459, 2022.
[33] D. P. Woodruff, *Sketching as a tool for numerical linear algebra*, Foundations and Trends in Theoretical Computer Science, 10, No. 1-2, pp. 1–157, 2014