Randomized Algorithms for Solving Linear Systems with Low-rank Structure

Michał Dereziński, Daniel LeJeune, Christopher Musco, Deanna Needell, Elizaveta Rebrova, and Jiaming Yang

Abstract

We consider the task of solving a large system of linear equations $Ax = b$, where for simplicity, we will assume that A is real, square, and full-rank. Iterative algorithms, such as LSQR, Conjugate Gradient and other Krylov subspace methods, are a powerful tool for solving such linear systems. Yet, the convergence properties of these methods are highly dependent on the singular value structure of the matrix A, and characterizing these properties effectively requires going beyond the usual notion of condition number. In this talk, we will consider this problem in the context of linear systems whose singular values exhibit a low-rank structure, in the sense that A can be decomposed into a low-rank ill-conditioned matrix (the "signal") and a full-rank well-conditioned matrix (the "noise"). Such linear systems are motivated by a range of problem settings, including in statistics, machine learning, and optimization, where the "signal" is often low-rank due to inherent structure of the data, while the "noise" may be coming from measurement error, data transformations, or an explicit regularizer imposed by the user. We will show how randomized sketching techniques, including our recent works on randomized preconditioning [DMY25] and stochastic solvers [DR24, DY24, DLNR24], can be used to exploit this low-rank structure in order to accelerate linear system solving in ways that go beyond what is possible with Krylov subspace methods.

Linear systems with low-rank structure. Consider the following linear system task:

$$\text{Solve} \quad Ax = b, \quad \text{given} \quad A \in \mathbb{R}^{n\times n} \quad \text{and} \quad b \in \mathbb{R}^n,$$

where A is a full-rank matrix with singular values $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_n$. For a given low-rank parameter $k \in \{1, ..., n\}$, we will allow the top-k part of the singular values to be very ill-conditioned, but assume that the tail is moderately well-conditioned, as measured by $\kappa_k = \sigma_{k+1}/\sigma_n$. For example, if the matrix A is explicitly regularized, e.g., $A = B + \lambda I_n$ as in ridge regression [AM15] or cubic-regularized Newton's method [NP06], then k may correspond to the number of singular values above the λ threshold. Similar regularization effect occurs when A is distorted by isotropic noise, $A = B + \delta N$, e.g., where N is Subgaussian [Joh01], or it is the error from stochastic rounding [DBM$^+$24]. Also, A may exhibit a power law singular value distribution ($\sigma_i \propto i^{-\beta}$), e.g., due to a data transformation with the Matérn kernel function [RW06]. Here, different values of k capture different signal-to-noise trade-offs. Our goal is to describe the convergence and computational cost of iterative algorithms for solving $Ax = b$ in terms of the parameters n, k, and κ_k. One can also consider the sparsity of A, but for simplicity, we will focus on the dense setting.

Effectiveness and limitations of Krylov subspace methods. A careful analysis of Krylov subspace methods such as LSQR and CG for solving linear systems with low-rank structure [AL86] shows that they need k iterations to capture the top-k singular vectors, and then $O(\kappa_k \log(1/\epsilon))$ iterations to converge at a rate that depends only on κ_k (with κ_k replaced by $\sqrt{\kappa_k}$ when A is positive definite). Thus, for a dense A, before reaching a fast convergence rate of $O(n^2\kappa_k \log 1/\epsilon)$ operations, Krylov methods require an initial $O(n^2k)$ cost (corresponding to roughly k matrix-vector products) to capture the low-rank structure of A, which is expensive for large k. This n^2k bottleneck can be established as a lower bound not just for Krylov methods but for any algorithms that access A only through matrix-vector products [DLNR24].

Given the above problem formulation and discussion, the central question of this talk is:

Can the n^2k bottleneck in solving linear systems with low-rank structure be overcome, when given direct access to A and allowing randomization?

Randomized preconditioning via sparse sketching. Randomized sketching offers a powerful set of tools for accelerating linear solvers. While these approaches have traditionally focused on very tall least squares problems [AMT10], linear systems with low-rank structure offer another setting where sketching can be beneficial. Such an algorithm starts by applying a random matrix $S \in \mathbb{R}^{s\times n}$ (e.g., Gaussian) to the matrix A, producing a smaller sketch $\tilde{A} = SA \in \mathbb{R}^{s\times n}$, where $s \ll n$ is the sketch size. This sketch can now be used to construct an approximate low-rank decomposition of A, e.g., by orthonormalizing the columns of $\tilde{A}^\top$ to obtain an $n \times s$ matrix Q and projecting A onto the subspace defined by those columns, $\hat{A} = AQQ^\top \approx A$ [HMT11]. The intuition here is that $\hat{A}$ approximates A reasonably well in the top-k singular directions as long as the sketch size s is sufficiently larger than k, and this approximation can be further boosted via subspace iteration.

If implemented naïvely, sketching does not appear to overcome the $O(n^2k)$ computational barrier exhibited by Krylov methods, due to three bottlenecks: (1) applying the sketching matrix S, (2) projecting via the orthogonal matrix Q, and (3) performing subspace iteration. Each of these require at least k matrix-vector products to produce a decent preconditioner for a linear system with rank k structure. However, given direct access to A, the sketching cost (bottleneck 1) can be reduced by using fast sketching methods, e.g., by making S extremely sparse, which is known to retain similar guarantees as a Gaussian matrix. Moreover, recent works have shown that a careful construction of the preconditioner can avoid the full projection step (bottleneck 2): in the positive definite case, by relying on Nyström approximations [FTU23], and in the general case, by using an inner solver to construct the preconditioner implicitly [DMY25]. In the latter work, we showed that this approach can be used to solve a linear system in $\tilde{O}(n^2\kappa_k\sqrt{n/k}\log 1/\epsilon + k^3)$ operations (up to minor logarithmic factors), where the term $\sqrt{n/k}$ comes as a trade-off from omitting subspace iteration (bottleneck 3). When k is sufficiently large and κ_k small enough, this overcomes the n^2k barrier.

Stochastic solvers via Sketch-and-Project. Another class of methods that use randomized sketching and/or sub-sampling to go beyond matrix-vector products are stochastic iterative solvers such as randomized Kaczmarz and coordinate descent, among others. Viewed in the context of sketching, many of these methods can be unified under the framework of Sketch-and-Project [GR15]. Here, we consider a solver that updates an iterate x_t by repeatedly sketching the system $Ax = b$ and projecting x_t onto the solutions of the sketched system:

$$x_{t+1} = \arg\min_{x\in\mathbb{R}^n} \|x_t - x\| \quad \text{subject to} \quad SAx = Sb.$$

While stochastic solvers have traditionally been considered effective primarily in specialized settings where we may not be able to perform full matrix-vector products with A (e.g., due to memory or bandwidth constraints), we have shown in recent works that these methods can also be particularly effective for linear systems with low-rank structure. Here, the intuition is that the sketched system $SAx = Sb$ retains the information about the top-k singular directions of A, which gives the Sketch-and-Project solver a convergence rate akin to being preconditioned with a rank k approximation [DR24]. We have adapted this approach to a simple Randomized Block Kaczmarz method [DY24], as well as a variant with Nesterov's acceleration [DLNR24], showing that these algorithms can solve a linear system in $\tilde{O}((n^2 + nk^2)\kappa_k \log 1/\epsilon)$ operations. This recovers the fast Krylov convergence of $\tilde{O}(n^2\kappa_k \log 1/\epsilon)$ operations for up to $k = O(\sqrt{n})$, while entirely avoiding the n^2k bottleneck.

References

[AL86] Owe Axelsson and Gunhild Lindskog. On the rate of convergence of the preconditioned conjugate gradient method. *Numerische Mathematik*, 48:499–523, 1986.

[AM15] Ahmed El Alaoui and Michael W. Mahoney. Fast randomized kernel ridge regression with statistical guarantees. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 775–783, 2015.

[AMT10] Haim Avron, Petar Maymounkov, and Sivan Toledo. Blendenpik: Supercharging lapack's least-squares solver. *SIAM Journal on Scientific Computing*, 32(3):1217–1236, 2010.

[DBM+24] Gregory Dexter, Christos Boutsikas, Linkai Ma, Ilse CF Ipsen, and Petros Drineas. Stochastic rounding implicitly regularizes tall-and-thin matrices. *arXiv preprint arXiv:2403.12278*, 2024.

[DLNR24] Michał Dereziński, Daniel LeJeune, Deanna Needell, and Elizaveta Rebrova. Fine-grained analysis and faster algorithms for iteratively solving linear systems. *arXiv preprint arXiv:2405.05818*, 2024.

[DMY25] Michał Dereziński, Christopher Musco, and Jiaming Yang. Faster linear systems and matrix norm approximation via multi-level sketched preconditioning. *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2025.

[DR24] Michał Dereziński and Elizaveta Rebrova. Sharp analysis of sketch-and-project methods via a connection to randomized singular value decomposition. *SIAM Journal on Mathematics of Data Science*, 6(1):127–153, 2024.

[DY24] Michał Dereziński and Jiaming Yang. Solving dense linear systems faster than via preconditioning. In *56th Annual ACM Symposium on Theory of Computing*, 2024.

[FTU23] Zachary Frangella, Joel A Tropp, and Madeleine Udell. Randomized Nyström preconditioning. *SIAM Journal on Matrix Analysis and Applications*, 44(2):718–752, 2023.

[GR15] Robert M Gower and Peter Richtárik. Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1660–1690, 2015.

[HMT11] Nathan Halko, Per-Gunnar Martinsson, and Joel A Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[Joh01] Iain M Johnstone. On the distribution of the largest eigenvalue in principal components analysis. *The Annals of Statistics*, 29(2):295–327, 2001.

[NP06] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[RW06] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.