

# FAST LINEAR SYSTEM SOLVE VIA SUBSPACE ITERATION

IBROHIM NOSIROV\*, CHRISTOPHER MUSCO†, AND JONATHAN WEARE‡

**Abstract.** We propose a novel iterative method for the solution of linear systems that accelerates the convergences of Richardson iteration by simultaneously building an eigenvector basis. This method is particularly suited for matrices with substantial eigenvalue gaps, exemplified by the PageRank problem. In other scenarios, our scheme performs comparably to Richardson iteration. Our analysis currently assumes all system matrix eigenvalues to have positive real parts.

**1. Introduction.** Randomized algorithms are an exciting new class of fast numerical methods that offer accurate solutions to many problems in numerical linear algebra (NLA) while also scaling well to large datasets. Many randomized algorithms aim to ‘sketch’, or quickly project, a high-dimensional problem into a lower-dimensional basis where a smaller problem can be solved instead [1][5]. The choice of basis and the speed of its construction play a key role in determining the effectiveness of a sketched matrix computation [4]. Moreover, one may be particularly interested in rigorously justifying the acquired speed-up; this analysis is also dependent on how the basis is constructed [3].

We propose a scheme to iteratively build a basis of eigenvectors to solve a linear system. More specifically, our scheme amounts to performing subspace iteration [2], an extension of power method, to achieve a faster algorithm. This scheme also yields computational speedups for certain applications, like the PageRank problem; we present these results in section 3.

**2. Random Starting Subspace Iteration.** Here we present an algorithm to solve a linear system

$$(2.1) \quad \mathbf{Ax} = \mathbf{b},$$

where  $A \in \mathbb{R}^{n \times n}$  and  $\mathbf{b}$  are known and the goal is to find  $\mathbf{x}$  to satisfy this equality. We assume that all eigenvalues of  $A$  have positive real part and we choose  $\varepsilon > 0$  so that the spectral radius of  $I - \varepsilon A$  is less than 1. If  $A$  has eigenvalues with negative real part we can instead consider the equation  $A^\top \mathbf{Ax} = A^\top \mathbf{b}$ . Our scheme, which we call ‘Random Starting Subspace Iteration (SI)’, is detailed in Algorithm 2.1.

Forming the matrix

$$(2.2) \quad Y = \left( \varepsilon \sum_{j=0}^{q-1} (I - \varepsilon A)^j \mathbf{b} \quad (I - \varepsilon A)^q \Pi \right),$$

amounts to repeatedly multiplying an augmented matrix,

$$(2.3) \quad \bar{A} = \begin{pmatrix} 1 & \mathbf{0} \\ \varepsilon \mathbf{b} & I - \varepsilon A \end{pmatrix}.$$

by another augmented matrix of the form  $\bar{\Pi} = \begin{pmatrix} 1 & \mathbf{0} \\ \mathbf{0} & \Pi \end{pmatrix}$  and we get,

---

\*Department of Applied Mathematics & Statistics, Colorado School of Mines.

†Tandon School of Engineering, New York University.

‡Courant Institute of Mathematical Sciences, New York University.

---

**Algorithm 2.1** Random Starting Algorithm

---

**Input:** Matrix  $A \in \mathbb{R}^{n \times n}$ , right-hand side  $\mathbf{b} \in \mathbb{R}^n$ , spectral radius normalizing constant  $\varepsilon$ , number of iterations  $q$ , size of the subspace  $k$ .

**Output:** Approximate solution  $\hat{\mathbf{x}}$  to the linear system 2.1.

- 1: **function** RANDOM STARTING SI
  - 2:   Choose a random sketch matrix  $\Pi \in \mathbb{R}^{n \times (k-1)}$ .
  - 3:   Form the matrix  $Y \in \mathbb{R}^{n \times k}$ ,  $Y = \begin{pmatrix} \varepsilon \sum_{j=0}^{q-1} (I - \varepsilon A)^j \mathbf{b} & (I - \varepsilon A)^q \Pi \end{pmatrix}$ .
  - 4:   Perform a QR decomposition,  $Y = QR$ .
  - 5:   Solve the  $k \times k$  linear system  $Q^\top A Q \mathbf{c} = Q^\top \mathbf{b}$  for  $\mathbf{c} \in \mathbb{R}^{k \times k}$ .
  - 6:   Construct the approximate solution  $Q \mathbf{c} = \hat{\mathbf{x}}$ .
  - 7: **end function**
- 

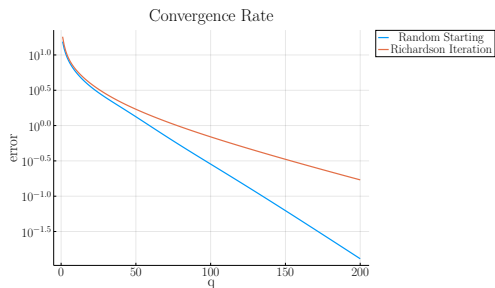
$$\bar{A}^q \bar{\Pi} = \begin{pmatrix} 1 & \mathbf{0} \\ \varepsilon \sum_{j=0}^{q-1} (I - \varepsilon A)^j \mathbf{b} & (I - \varepsilon A)^q \Pi \end{pmatrix}.$$

This construction, particularly the motivation for  $I - \varepsilon A$ , is non-obvious and we dedicate the Appendix (section 5) to its justification.

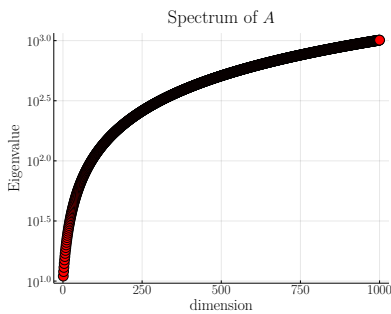
**3. Results.** We first present results numerically demonstrating that the convergence rate of our method depends on gaps between eigenvalues in the spectrum of  $A$ . Figure 3.1 presents both adversarial and amicable cases to show that a gap in the spectrum leads to faster convergence. We then describe the PageRank problem, an application where our random starting scheme performs especially well.

**3.1. Application: PageRank problem.** The PageRank problem, with applications in web search and other fields like social networks and protein network analysis, seeks to rank nodes in a graph based on their relative importance. The crux of the problem lies in determining the stationary distribution of a related Markov chain. A prevalent method to solve this involves an iterative algorithm that successively multiplies a vector by the matrix  $A = I - \omega P$ , where  $\omega$  is a specified probability (typically set to 0.85, representing the likelihood of a user continuing their random walk), and  $P$  is the stochastic matrix embodying the structure of the network. The algorithm continues until the ranks converge within a designated tolerance, thus providing a solution to the PageRank problem. Figure 3.2 visually encapsulates the faster convergence rate that our Random Starting SI method exhibits when solving the PageRank problem. The chart illustrates how the presence of a significant spectral gap in the system matrix  $A = I - \omega P$  influences the speed of convergence. This validates our mathematical assumptions about the relationship between the eigenvalue gap and convergence speed.

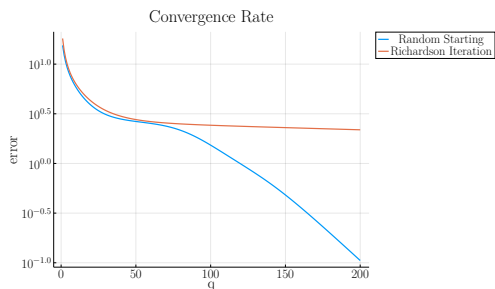
**4. Discussion.** Our proposed Random Starting SI method leverages iteratively constructed eigenvector bases to solve linear systems efficiently. Our results reveal that the eigenvalue gap in the system’s spectrum has a significant impact on the convergence speed. The algorithm performs well when a gap exists, as seen in use-cases like the PageRank problem, but comparably to traditional gradient descent in cases where the spectrum lacks significant gaps. Our current analysis, as presented in the Appendix, assumes all eigenvalues of the system matrix to have positive real parts and generalizing this analysis further is one future direction for our work.



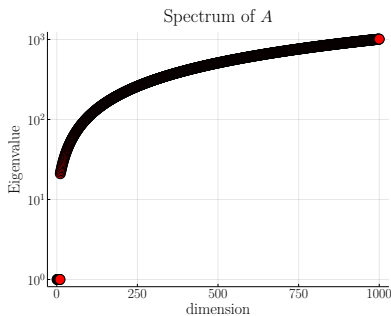
(a) Illustration of convergence rate, measured in  $q$  number of matrix-vector multiplications by the matrix  $\bar{A}$ . In this case, the spectrum of  $A$  inside the augmented matrix is graphed in 3.1b. The slopes of convergence for Richardson iteration and our random starting scheme are very similar.



(b) Spectrum of a matrix where eigenvalues are uniformly arranged. The gap between any two eigenvalues is the same throughout the spectrum.



(c) Illustration of convergence rate for a matrix whose spectrum is depicted in 3.1d. The random starting scheme converges at a much faster rate than Richardson iteration, as indicated by the significant difference in slopes.



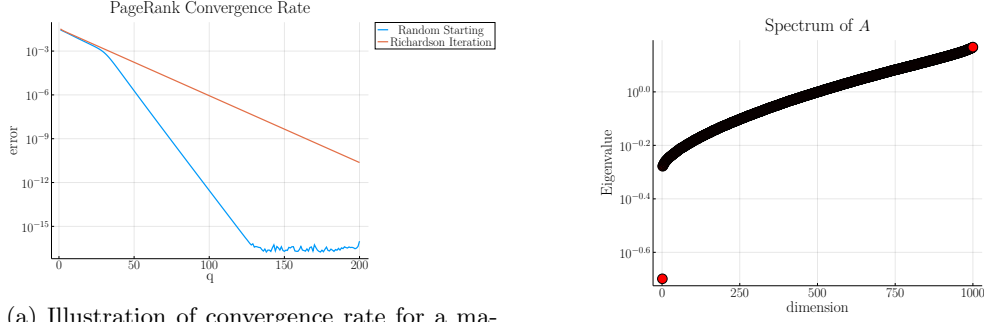
(d) Spectrum that is exactly the same as in 3.1b, but the smallest 10 eigenvalues are set to 1. This creates an order of magnitude gap in the spectrum.

Fig. 3.1: An illustration of the effect a gap in the spectrum of eigenvalues has on the convergence rate of our scheme as opposed to Richardson iteration. Figures 3.1a and 3.1b illustrate the convergence in the adversarial case for our random starting scheme. Figures 3.1c and 3.1d illustrate the effect of a gap in the spectrum has on the convergence; namely, this results in a significantly faster convergence rate. For both cases, the size of the subspace,  $k$ , is held constant at 20.

**5. Appendix.** This section is primarily dedicated to motivating the construction of the augmented matrix in 2.3. In section 5.1 we first show that performing repeated applications of  $\bar{A}$  to a non-zero vector is equivalent to performing Richardson iteration on the matrix  $A$ . In section 5.2, we opt for a fixed-point analysis in to demonstrate convergence in a simpler case with a starting vector as opposed to a subspace.

**5.1.  $\bar{A}(1 \ x_0^\top)$  amounts to power iteration on  $I - \epsilon A$ .**

*Proof.* Let  $\bar{A}$  be defined as in 2.3, let  $x_0$  be any non-zero vector, let  $q$  be the



(a) Illustration of convergence rate for a matrix of the form  $A = I - \omega P$  where  $\omega \in [0, 1]$  and  $P$  is a stochastic matrix. The random starting algorithm leverages the gap in the spectrum, as seen in figure 3.2b, to achieve a much faster convergence rate than Richardson iteration.

(b) Graph of the spectrum of the matrix described in the PageRank problem. We can see that there is a collection of small eigenvalues, a sizable gap, and a collection of large eigenvalues.

Fig. 3.2: Illustration of the correlation between the spectral gap in the system matrix and the convergence speed of the Random Starting SI method. The graph captures the comparative performance with a clear gap (example: PageRank problem) versus those lacking a significant gap. The  $x$ -axis represents the size of the spectral gap, while the  $y$ -axis measures the rate of convergence.

number of iterations (matrix-vector multiplications). We will show that performing power method with  $\bar{A}$  as the operator and with  $\begin{pmatrix} 1 \\ \mathbf{x}_0 \end{pmatrix}$  as the starting vector amounts to performing Richardson iteration to solve  $A\mathbf{x} = \mathbf{b}$ . First, consider

$$\begin{pmatrix} 1 \\ \mathbf{x}_{q+1} \end{pmatrix} = \bar{A}^{q+1} \begin{pmatrix} 1 \\ \mathbf{x}_0 \end{pmatrix} = \bar{A} \begin{pmatrix} 1 \\ \mathbf{x}_q \end{pmatrix} = \begin{pmatrix} 1 \\ \varepsilon \mathbf{b} + (I - \varepsilon A)\mathbf{x}_q \end{pmatrix}.$$

Since the first entry is always equal to 1, we now focus on the remaining entries in the vector and notice that

$$(5.1) \quad \mathbf{x}_{q+1} = \varepsilon \mathbf{b} + (I - \varepsilon A)\mathbf{x}_q.$$

Re-expressing this in terms of  $\mathbf{x}_0$ , we get

$$(5.2) \quad \mathbf{x}_{q+1} = \left( \varepsilon \sum_{j=0}^q (I - \varepsilon A)^j \mathbf{b} \right) \mathbf{x}_0. \quad \square$$

This is the definition of Richardson iteration.

## 5.2. Random Starting SI converges (one starting vector).

*Proof.* Assuming that  $A$  is symmetric, we will show that  $\bar{A}^q \begin{pmatrix} 1 \\ \mathbf{x}_0 \end{pmatrix}$  converges to  $\begin{pmatrix} 1 \\ \mathbf{x}^* \end{pmatrix}$  for sufficient  $q$  iterations. First, note that the first entry in both the starting vector and the desired vector is 1. Hence, only focus on the remaining entries.

We recall equation 5.1 and consider that relation with respect to the solution vector  $\mathbf{x}^*$

$$\mathbf{x}_{q+1} - \mathbf{x}^* = \varepsilon \mathbf{b} + (I - \varepsilon A)\mathbf{x}_q - \mathbf{x}^*,$$

and we rearrange some terms to get

$$\mathbf{x}_{q+1} - \mathbf{x}^* = \varepsilon A\mathbf{x}^* + (I - \varepsilon A)\mathbf{x}_q - \mathbf{x}^*.$$

Finally, re-expressing our relation in terms of  $\mathbf{x}_0$ , we get

$$(5.3) \quad \mathbf{x}_{q+1} - \mathbf{x}^* = (I - \varepsilon A)^q(\mathbf{x}_0 - \mathbf{x}^*).$$

Notice that  $\mathbf{x}_0 - \mathbf{x}^*$  is a fixed-point where  $I - \varepsilon A$  is the operator. From this we can infer that the convergence, measured by the difference  $\mathbf{x}_{q+1} - \mathbf{x}^*$ , is dominated by  $(I - \varepsilon A)^q$ . A spectral decomposition of this matrix reveals

$$(5.4) \quad (I - \varepsilon A)^q = U^\top \begin{pmatrix} (1 - \varepsilon\lambda_n)^q & 0 & \cdots & 0 \\ 0 & (1 - \varepsilon\lambda_{n-1})^q & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & (1 - \varepsilon\lambda_1)^q \end{pmatrix} U,$$

where  $U$  is an orthonormal matrix and the eigenvalues are organized in descending order inside of a diagonal matrix.

Choosing  $\varepsilon < \frac{1}{\lambda_1}$ , we see that  $I - \varepsilon A$  tends to zero at a rate of  $\frac{1 - \varepsilon\lambda_{n-1}}{1 - \varepsilon\lambda_n} < \frac{\lambda_1 - \lambda_{n-1}}{\lambda_1 - \lambda_n}$ . We can rearrange some terms

$$\frac{\lambda_1 - \lambda_{n-1}}{\lambda_1 - \lambda_n} = 1 - \frac{\lambda_{n-1} - \lambda_n}{\lambda_1 - \lambda_n},$$

and we obtain the following rate

$$(5.5) \quad \left(1 - \frac{1}{\left(\frac{\lambda_1 - \lambda_n}{\lambda_{n-1} - \lambda_n}\right)}\right)^q \leq \exp\left(\frac{-q(\lambda_{n-1} - \lambda_n)}{\lambda_1 - \lambda_n}\right).$$

We can thus conclude that

$$\begin{aligned} \|\mathbf{x}_{q+1} - \mathbf{x}^*\| &= \|(I - \varepsilon A)^q(\mathbf{x}_0 - \mathbf{x}^*)\| \\ &\leq \|(I - \varepsilon A)^q\| \|\mathbf{x}_0 - \mathbf{x}^*\| \\ &\leq \exp\left(\frac{-q(\lambda_{n-1} - \lambda_n)}{\lambda_1 - \lambda_n}\right) \|\mathbf{x}_0 - \mathbf{x}^*\|. \quad \square \end{aligned}$$

#### REFERENCES

- [1] R. M. GOWER AND P. RICHTÁRIK, *Randomized iterative methods for linear systems*, SIAM Journal on Matrix Analysis and Applications, 36 (2015), pp. 1660–1690, <https://doi.org/10.1137/15M1025487>, <http://epubs.siam.org/doi/10.1137/15M1025487> (accessed 2023-06-06).

- [2] N. HALKO, P. G. MARTINSSON, AND J. A. TROPP, *Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions*, SIAM Review, 53 (2011), pp. 217–288, <https://doi.org/10.1137/090771806>, <https://doi.org/10.1137/090771806>, <https://arxiv.org/abs/https://doi.org/10.1137/090771806>.
- [3] C. MUSCO AND C. MUSCO, *Randomized block krylov methods for stronger and faster approximate singular value decomposition*, 2015, <https://arxiv.org/abs/1504.05477>.
- [4] Y. NAKATSUKASA AND J. A. TROPP, *Fast & accurate randomized algorithms for linear systems and eigenvalue problems*, 2022, <https://arxiv.org/abs/2111.00113>.
- [5] D. P. WOODRUFF ET AL., *Sketching as a tool for numerical linear algebra*, Foundations and Trends in Theoretical Computer Science, 10 (2014), pp. 1–157.