Computational Linear Algebra Project
Computation of the von Neumann entropy by comparing Rational Lanczos to the Stochastic Trace estimator and Probing Method

Kleon Karapas

May 2024

Abstract

In this project, rational Krylov subspaces have been studied as well as their implementation in methods to compute the trace of the von Neumann entropy, $S(A) = \text{Tr}(-A\log(A))$. It has been verified that rational Krylov subspace iterations coincide with polynomial iterations when the poles are infinite. The stochastic trace estimator was then implemented, showing its potential for achieving fast results, but with a particularly high amount of samples needed to obtain high accuracies. Finally, the probing method was studied, coupled with two stopping criterions for the Lanczos method as well as two methods to compute the optimal value of d.

1

1.1 Definition of a rational Krylov subspace $\mathcal{Q}_m(A, b)$

Let's suppose the existence of a polynomial q_{m-1} of degree $m-1$, $q_{m-1} \in \mathcal{P}_{m-1}$, where we will assume in the following that q_{m-1} is factored as:

$$q_{m-1}(z) = \prod_{j=1}^{m-1} (1 - \frac{z}{\xi_j}), \tag{1}$$

where the poles $\xi_1, \xi_2, ...$ are numbers in the extended complex plane $\mathbb{C} := \mathbb{C} \cup \{\infty\}$ different from all eigenvalues $\lambda \in \Lambda(A)$ and 0, where $\Lambda(A)$ denotes the spectrum of A. The rational Krylov space of order m associated with (A, b) is defined as:

$$\mathcal{Q}_m(A, b) := q_{m-1}(A)^{-1}\text{span}\{b, Ab, ..., A^{m-1}b\}.$$

1.2 Proof of Lemma 3.1 [2]

Lemma 3.1 (Exactness) *Let $V_m \in \mathbb{C}^{N\times m}$ be an orthonormal basis of $\mathcal{Q}_m(A, b)$, and let $A_m = V_m^* A V_m$. Then for any rational function $r_m \in \mathbb{P}_m/q_{m-1}$ we have*

$$(V_m V_m^*) r_m(A) b = V_m r_m(A_m) V_m^* b,$$

provided that $r_m(A_m)$ is defined. In particular, if $r_m \in \mathbb{P}_{m-1}/q_{m-1}$, then

$$r_m(A)b = V_m r_m(A_m) V_m^* b,$$

i.e., the rational Arnoldi approximation for $r_m(A)b$ is exact.

Proof. Define $q = q_{m-1}(A)^{-1}b$. We first show by induction that

$$(V_m V_m^*) A^j q = V_m A_m^j V_m^* q \quad \text{for all } j \in \{0, 1, ..., m\}. \tag{2}$$

The A^j terms stand for the terms belonging to the polynomial in the numerator of the rational function r_m. The assertion (2) is obviously true for $j = 0$. Assume that it is true for some $j < m$. Then by the definition of a rational Krylov space we have:

- $(V_m^* V_m)A^j = A^j$ (as $V_m = [v_1, ..., v_m] \in \mathbb{C}^{N\times m}$ is an orthonormal basis of $\mathcal{Q}_m(A, b)$)
- We know that the orthogonal projector of $\mathbb{C}^m$ onto the rational Krylov space $\mathcal{Q}_m(A, b)$ is: $V_m(V_m^* V_m)^{-1}V_m^* A^j q = V_m V_m^* A^j q = A^j q$. This last equality is due to the fact that $A^j q$ already belongs to the rational Krylov space, meaning that its projection on this space is itself. We also know that A^j and q commute which completes the justification for the last equality.

Therefore:

$$\begin{aligned}(V_m V_m^*)A^{j+1}q = (V_m V_m^*)AA^j q = (V_m V_m^*)A \underbrace{V_m V_m^* A^j q}_{=A^j q} &= (V_m V_m^*)AV_m V_m^* A^j \quad \underbrace{V_m V_m^* q}_{=V_m V_m^* A^0 q=q} \\ = V_m \underbrace{(V_m^* A V_m)}_{A_m} \underbrace{(V_m^* A^j V_m)}_{A_m^j} V_m^* q &= V_m A_m A_m^j V_m^* q = V_m A_m^{j+1} V_m^* q,\end{aligned}$$

which establishes (2). But now we have on both sides of the equality $q_{m-1}(A)^{-1}$ and the goal would be to have on the right-hand side $q_{m-1}(A_m)^{-1}$ so that we can have $r_m(A_m) = \frac{A_m^j}{q_{m-1}(A_m)}$. $q_{m-1}(A_m)^{-1}$ can be obtained in the following way:

$$\begin{aligned}b = q_{m-1}(A)q &= (a_0 + a_1 A + a_2 A^2 + \ldots + a_{m-1}A^{m-1})q \\ &= a_0 V_m V_m^* q + a_1 V_m V_m^* Aq + a_2 V_m V_m^* A^2 q + \ldots + a_{m-1} V_m V_m^* A^{m-1} q \quad \text{(rational Krylov space)} \\ &= a_0 V_m A_m^0 V_m^* q + a_1 V_m A_m^1 V_m^* q + a_2 V_m A_m^2 V_m^* q + \ldots + a_{m-1} V_m A_m^{m-1} V_m^* q \quad \text{(using (2))} \\ &= V_m (a_0 A_m^0 + a_1 A_m^1 + a_2 A_m^2 + \ldots + a_{m-1} A_m^{m-1}) V_m^* q \quad \text{(by linearity)} \\ &= V_m q_{m-1}(A_m) V_m^* q\end{aligned}$$

Therefore, we obtain:

$$b = q_{m-1}(A)q = V_m q_{m-1}(A_m) V_m^* q, \tag{3}$$

or equivalently, $V_m^* q = q_{m-1}(A_m)^{-1} V_m^* b$.

Hence, the proof can now be completed by writing $r_m(x) = p(x)/q_{m-1}(x)$, with $p(x) = \sum_{j=0}^m \alpha_j x^j$:

$$\begin{aligned}(V_m V_m^*) r_m(A) b &= \sum_{i=0}^m \alpha_j (V_m V_m^*) A^j q \\ &= \sum_{i=0}^m \alpha_j V_m A_m^j V_m^* q \qquad \text{Using (2)} \\ &= \sum_{i=0}^m \alpha_j V_m A_m^j q_{m-1}(A_m)^{-1} V_m^* b \qquad \text{Using (3)} \\ &= V_m p(A_m) q(A_m)^{-1} V_m^* b = V_m r_m(A_m) V_m^* b\end{aligned}$$

This completes the proof.

1.3 Proof of Lemma 4.3 of [1]

Lemma 4.3 *Let $A \in \mathbb{C}^{N\times N}$ be a Hermitian matrix. Let $p_{2m-1} \in \mathbb{P}_{2m-1}$ be a polynomial of degree at most $2m-1$, for an $m \in \mathbb{N}$. We define a rational function $r(z) = p_{2m-1}(z) q_{m-1}(z)^{-2}$ with q defined in Eq.1. Then:*

$$b^* r(A) b = b^* V_m r(A_m) V_m^* b \tag{4}$$

With A_m, V_m defined in Sect.1.2.

Proof It is sufficient to prove this for $p_{2m-1}(z) = z^k$ for $k = 0, \ldots, 2m-1$. We first assume k is odd, $k = 2j+1$, $j \leq m$ and $s(A) = q_{m-1}(A)^{-1} A^j$. We also know that A and $q_{m-1}(A)^{-1}$ as in Sect.1.2. We obtain:

$$\begin{aligned}
b^T r(A)b &= b^T A^{2j+1} q_{m-1}(A)^{-2} b \\
&= b^T q_{m-1}(A)^{-1} A^j A q_{m-1}(A)^{-1} A^j b \\
&= b^T s(A) A s(A) b \\
&= (s(A)^T b)^T A (s(A) b)
\end{aligned}$$

We now use Lemma 3.1 of [2] for $r_m(A) = s(A)$ (second equality is used since $j \le m$):

$$\begin{aligned}
b^T r(A)b &= (V_m s(A_m)^T V_m^T b)^T A (V_m s(A_m) V_m^T b) \\
&= (b^T V_m s(A_m) \underbrace{V_m^T) A (V_m}_{A_m} s(A_m) V_m^T b) \\
&= b^T V_m s(A_m) A_m s(A_m) V_m^T b \\
&= b^T V_m r(A_m) V_m^T b
\end{aligned}$$

This proves the lemma for k odd.

For k even $\Leftrightarrow k = 2j$, we have the following (similar to odd case):

$$\begin{aligned}
b^T r(A)b &= b^T s(A)^2 b \\
&= (s(A)^T b)^T (s(A) b) \\
&= b^T V_m s(A_m) \underbrace{V_m^T V_m}_{I_m} s(A_m) V_m^T b \\
&= b^T V_m s(A_m)^2 V_m^T b \qquad = b^T V_m r(A_m) V_m^T b
\end{aligned}$$

This completes the proof.

2 Proof of Proposition 4.4 [1]

Proposition 4.4 *Let $A \in \mathbb{C}^{N \times N}$ be a Hermitian matrix with spectrum $\Lambda(A) \subset [\lambda_{min}, \lambda_{max}] := \Sigma$. Let $\psi = b^* f(A) b$ and $\psi_m = b^* V_m f(A_m) V_m^* b$. Then:*

$$|\psi - \psi_m| \le 2||b||_2^2 \min_{p \in \mathbb{P}_{2m-1}} ||f - p q_{m-1}^{-2}||_\Sigma \tag{5}$$

With the second norm on the right hand side term being the maximum norm on the compact set Σ.

Proof

$$\begin{aligned}
|\psi - \psi_m| &= |b^* f(A) b - b^* V_m f(A_m) V_m^* b| \\
&= |b^* f(A) b - b^* V_m f(A_m) V_m^* b - b^* r(A) b + b^* r(A) b| \\
&= |b^* f(A) b - b^* V_m f(A_m) V_m^* b - b^* r(A) b + b^* V_m r(A_m) V_m^* b| \qquad \text{Lemma 4.3} \\
&\le ||b||_2^2 \cdot ||f(A) - r(A) + V_m [r(A_m) - f(A_m)] V_m^*||_2 \\
&\le ||b||_2^2 \cdot (||f(A) - r(A)||_2 + ||r(A_m) - f(A_m)||_2) \qquad ||V_m||_2 = 1
\end{aligned} \tag{6}$$

The first inequality can be shown using the matrix 2-norm and then applying the Cauchy-Schwarz inequality:

$$||A||_2 = \sup_{x \in \mathbb{C}^N} ||Ax||_2 / ||x||_2 \Rightarrow ||Ax||_2 \le ||A||_2 ||x||_2 \forall x \in \mathbb{C}^N \tag{7}$$

$$|x^* A x| = | \langle x, Ax \rangle \le ||x||_2 ||Ax||_2 \le ||x||_2^2 ||A||_2 \qquad \forall x \in \mathbb{C}^n, \forall A \in \mathbb{C}^{N \times N} \tag{8}$$

Which justifies the first inequality in Eq.6. The second inequality is obtained using the triangle inequality:

$$||(f(A) - r(A)) + (r(A_m) - f(A_m))||_2 \le ||f(A) - r(A)||_2 + ||r(A_m) - f(A_m)||_2 \tag{9}$$

The following theorem [3] states that, for any function g that is analytical in the interior of $W(A) := \{v^* Av \mid v \in \mathbb{C}^N, ||v|| = 1\}$ and continuous up to the boundary of $W(A)$, the following inequality holds for any dimension N:

$$||g(A)|| \leq C \sup_{z \in W(A)} |g(z)| \tag{10}$$

With $C \leq 1 + \sqrt{2}$ [4] (In paper [3] it is 11.08 but this is a larger value so we ommit it).

If A is Hermitian, and g is restricted to a compact set $\Sigma \supseteq \Lambda(A)$, then $C = 1$ and the supremum can be changed to the maximum norm . In our case:

- $\Sigma = [\lambda_{min}, \lambda_{max}] \supseteq \Lambda(A)$
- $g = f - r$

Therefore

$$|\psi - \psi_m| \leq 2||b||_2^2 \cdot ||f - r||_\Sigma \tag{11}$$

Where $||f - r||_\Sigma$ is the maximum norm on Σ, and this inequality hold for the function $r \in q_{m-1}^{-2} \cdot \mathbb{P}_{2m-1}$ that minimizes $||f - r||_\Sigma$:

$$|\psi - \psi_m| \leq 2||b||_2^2 \inf_{p \in \mathbb{P}_{2m-1}} ||f - pq_{m-1}^{-2}||_\Sigma \tag{12}$$

We were not able to figure out why the infimum can be cahnged into a minimum.

3 Rational Lanczos algorithm

The rational Lanczos algorithm that we implemented in the general case where the poles are finite is the following :

Input: Function f, matrix A, starting vector v, pole sequence $\{\xi_1, \xi_2, \dots\}$, number of iterations m.
Output: Rational Lanczos approximation f_m, rational Krylov basis $V = v_1, v_2, \dots, v_m$

```
For $j = 1, 2, \dots, m$
    Set $\tilde{A} = I - A/\xi_j$, $v_1 = v/|v|$
    Compute $w_{j+1} = \tilde{A}Av_j$.
    Compute $\alpha = v_j^* w_{j+1}$
    Set $v = w - v_j w_{j+1}$
    If $j > 1$
        Set $v = v - \beta v_{j-1}$
    Compute $\beta = ||v||_2$
    Set $v_{j+1} = v/\beta$
    Set $h_{j,j} = \alpha$
    Set $h_{j+1,j} = h_{j,j+1} = \beta$
Define $K = I_m + H(1:m, 1:m) \cdot \text{diag}(\xi_1^{-1}, ..., \xi_m^{-1})$
Compute $f_m = ||v||_2^2 e_1^T H(1:m, 1:m) K^{-1} e_1$
```

Table 1: Rational Lanczos Algorithm

In our case, we take the poles to be infinite, meaning that this method is equivalent to the classical polynomial Lanczos method (as the Minnesota matrix is symmetric we use the Lanczos method). According to [2], we have the following rational Arnoldi decomposition:

$$AV_{m+1}\underline{K_m} = V_{m+1}\underline{H_m} \tag{13}$$

where,

$$\underline{H_m} := \begin{bmatrix} H_m \\ h_{m+1,m}\mathbf{e}_m^T \end{bmatrix} \quad \text{and} \quad \underline{K_m} := \begin{bmatrix} I_m + H_m D_m \\ h_{m+1,m}\xi_m^{-1}\mathbf{e}_m^T \end{bmatrix},$$

with $D_m = diag(\xi_1^{-1}, \xi_2^{-1}, \dots, \xi_m^{-1})$ and $\mathbf{e}_m$ denotes the m-th unit vector in $\mathbb{R}^m$. If all poles ξ_j are infinite, then (13) reduces to the standard *(polynomial) Arnoldi decomposition* $AV_m = V_{m+1}\underline{H_m}$.

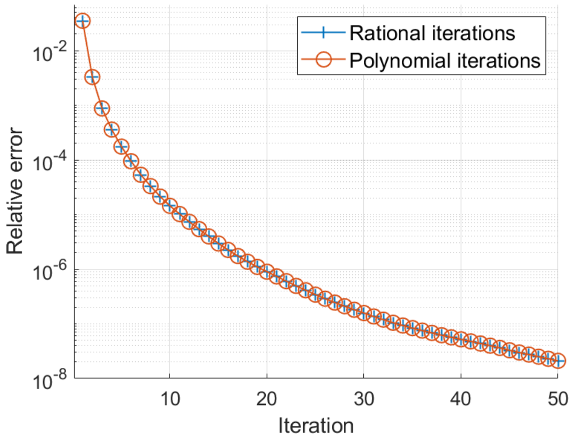

Figure 1: Comparison between rational Lanczos iterations with poles at infinity and polynomial Lanczos iterations.

The rational and polynomial Lanczos algorithms are implemented using the `minnesota` matrix, with its largest connected component extracted, then we take its sub-graph with nodes sorted in ascending order. The A which we will use is the graph Laplacian of this sub-graph divided by its trace.

Figure 1 shows the outcomes of testing the rational Lanczos algorithm with infinite poles, in comparison to polynomial Lanczos iterations and by using a random vector for the quadratic form. Both methods yield identical results(the difference between the vzlues being $\sim 10^{-14}$), which is expected since rational Krylov iterations with infinite poles are essentially polynomial Krylov iterations. These curves closely resemble the results shown in Figure 3 of [1]: the initial iterations converge quickly, but the rate of convergence slows down asymptotically.

4 Stochastic trace estimator

The function $f(A)$ of a symmetric matrix $A \in \mathbb{R}^{n\times n}$ can be defined via a spectral decomposition $A = U\Lambda U^T$ where U is orthogonal and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ is the diagonal matrix containing the eigenvalues. Then,

$$f(A) := Uf(\Lambda)U^T, \quad f(\Lambda) := \text{diag}(f(\lambda_1), \dots, f(\lambda_n)),$$

provided that f is defined on the eigenvalues of A. A matrix function can be computed via diagonalization, using polynomial or rational approximants as we have done previously with Krylov subspace methods. In the following sections we present two algorithms that estimate the trace of $f(A)$ and do not require the explicit computation of the diagonal entries of $f(A)$. The first is the stochastic trace estimator that we will present in this section and the other is the probing method (see Section 5).

We consider the problem of computing $\text{tr}(B)$, where $B \in \mathbb{R}^{n\times n}$ that can be easily accessed via matrix-vector products Bx and quadratic forms $x^T Bx$, for $x \in \mathbb{R}^n$, if its computationally expensive to diagonalize it. The case of $B = f(A)$ can be seen as an instance of this problem, since $f(A)$ is expensive to compute, but $f(A)x$ and $x^T f(A)x$ can be efficiently approximated using Krylov methods, as we recall in Section 3. The quadratic forms are computed as it follows. $V_m = [\mathbf{v}_1 \dots \mathbf{v}_m]$ denotes the matrix whose orthonormal columns span the Krylov subspace $\mathcal{Q}_m(A, \mathbf{b})$, and by $A_m = V_m^T AV_m$ the projection of A onto this subspace. We can then project the problem on $\mathcal{Q}_m(A, \mathbf{b})$ and approximate the *quadratic form* $\psi = \mathbf{b}^T f(A)\mathbf{b}$ in the following way,

$$\psi \approx \psi_m = \mathbf{b}^T V_m f(A_m) V_m^T \mathbf{b}$$

If the basis V_m is constructed incrementally using the rational Arnoldi algorithm in section 3, we have $\mathbf{v}_1 = \mathbf{b}/\|\mathbf{b}\|_2$ and therefore

$$\psi_m = \|\mathbf{b}\|_2^2 e_1^T f(A_m) e_1$$

which is the relation that we use to compute the forms in Matlab.

Stochastic trace estimators compute approximations of $\mathrm{tr}(f(A)$ by making use of the fact that, for any matrix $f(A)$ and any random vector $\mathbf{x}$ such that $\mathbb{E}[\mathbf{x}\mathbf{x}^T] = I$, we have $\mathbb{E}[\mathbf{x}^T f(A)\mathbf{x}] = \mathrm{tr}(f(A))$, where $\mathbb{E}$ denotes the expected value. Hutchinson's trace estimator [1] is a simple stochastic estimator that generates N vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ with i.i.d. random $\mathcal{N}(0,1)$ entries and approximates $\mathrm{tr}(f(A))$ with

$$\begin{aligned} \mathrm{tr}_N^{\mathrm{Hutch}}(f(A)) &= \frac{1}{N}\sum_{j=1}^{N} \mathbf{x}_j^T f(A)\mathbf{x}_j = \frac{1}{N}\mathrm{tr}(X^T f(A) X), \quad X = [\mathbf{x}_1 \ldots, \mathbf{x}_N] \\ &\approx \frac{1}{N}\sum_{j=1}^{N} \mathbf{x}_j^T \{V_m f(A_m) V_m^T\}_j \mathbf{x}_j = \frac{1}{N}\sum_{j=1}^{N} \mathbf{\Psi}_{\mathbf{m},j} = \frac{1}{N}\mathrm{tr}(\mathbf{\Psi}_{\mathbf{m}}) \end{aligned}$$

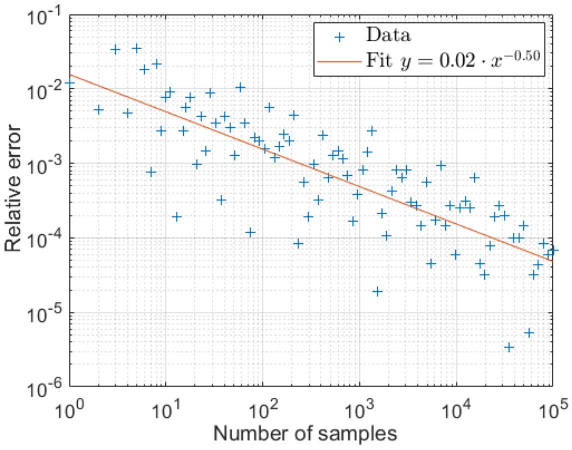

Figure 2: Hutchinson's trace estimator error as a function of the number of samples for the `minnesota` matrix.

The Lanczos method seen in class for polynomial Krylov subspaces is implemented, with $m = 20$ iterations and a Hessenberg matrix $H_m \in \mathbb{R}^{m\times m}$ is obtained with $H_m = A_m = V_m^* A V_m$. Therefore we compute $f(A_m) = f(H_m)$ which is computationally cheap as H_m is a way smaller matrix (20x20) compared to the original `minnesota` matrix of size(2642x2642), and can be done using the function `logm` in `MATLAB`.

The Hutchinson estimator is used with the sample number n going up to 10^5. The plot in Figure 2 illustrates the relationship between the number of samples (on the x-axis) and the relative error (on the y-axis) on a log-log scale. A power law fit, represented by the red line, is applied to the data, with the fit equation $y = 0.02 \cdot x^{-0.50}$ displayed on the plot. This fit indicates that the relative error decreases approximately in proportion to $n^{-1/2}$ as the number of samples increases. We did not find any reference to this power law in the relative bibliography suggesting that this might be random.

5 Probing method

The probing method is a technique used to approximate the trace of a matrix function $f(A)$ for a sparse matrix A. It involves representing the sparse matrix as a graph $\mathcal{G}(A)$, where nodes and edges correspond to the non-zero elements of A. The nodes are then partitioned into subsets $V_1, \ldots, V_s$ using distance-d coloring, ensuring that no two nodes within the same subset are close to each other based on the geodesic distance

$d(i, j)$ in the graph. For each subset V_ℓ, a probing vector $\mathbf{v}_\ell$ is constructed by summing the canonical basis vectors corresponding to the nodes in V_ℓ. The trace of the matrix function is then approximated by summing the quadratic forms of the probing vectors and the matrix function, given by

$$T_d(f(A)) := \sum_{\ell=1}^{s} \mathbf{v}_\ell^T f(A)\mathbf{v}_\ell. \tag{14}$$

where the associated probing vectors are:

$$\mathbf{v}_\ell = \sum_{i\in V_\ell} \mathbf{e}_i \in \mathbb{R}^n, \quad \ell = 1, \dots, s \tag{15}$$

This method efficiently reduces the number of required quadratic forms by leveraging the sparsity and symmetry of the matrix. It is important to keep s as small as possible since, in view of (14), it is the number of quadratic forms needed to compute $T_d(f(A))$. The problem here is that finding the minimum number s of colors, or sets, in the partition needed to get a distance-d coloring for a fixed d is NP-complete for general graphs. A greedy and efficient algorithm to get a quasi-optimal distance-d coloring is given by Algorithm 1 [1].

What we will try to accomplish is to find the smallest possible ϵ such that:

$$|tr(f(A)) - T_d(f(A))| \le \epsilon \cdot tr(f(A)) \tag{16}$$

To do this we first try to find the minimum value for s which corresponds to the number of iterations m for Lanczos. We can bound the error by using the following lower and upper bounds based on the theorem 4.8 of Sect.4.3 of [1] :

$$\|\mathbf{b}\|_2^2 \min_{z\in[\lambda_{\min},\lambda_{\max}]} |g_m(z)| \le |\psi - \psi_m| \le \|\mathbf{b}\|_2^2 \max_{z\in[\lambda_{\min},\lambda_{\max}]} |g_m(z)|. \tag{17}$$

where A is symmetric with spectrum $\sigma(A) \subseteq [\lambda_{\min}, \lambda_{\max}]$, $\psi = \mathbf{v}^T f(A)\mathbf{v}$ the *exact quadratic form* and ψ_m the rational quadratic form as seen previously. The function g_m is taken as defined in equation 4.8 of [1], and is computed numerically by discretizing it on the interval $[\lambda_{\min}, \lambda_{\max}]$ (200 points). The upper bound is the one that is more interesting as it allows for a reliable stopping criterion for the Lanczos algorithm. Following the development in section 5.1 of [1] we obtain the following condition:

$$\left|\mathbf{v}_\ell^T f(A)\mathbf{v}_\ell - \hat{\psi}_\ell\right| \le \hat{\epsilon} \cdot \frac{|V_\ell|}{n} \quad \ell = 1, \dots, s \tag{18}$$

where $\{\mathbf{v}_\ell\}_{\ell=1}^s$ are the probing vectors used in the distance-d coloring,$\hat{\psi}_\ell$ denote the approximation of $\mathbf{v}_\ell^T f(A)\mathbf{v}_\ell$ obtained with a Krylov method and by recalling that $\|\mathbf{v}_\ell\|_2 = |V_\ell|^{1/2}$, where V_ℓ denotes the set of the partition associated to the ℓ-th color.$\hat{\epsilon}$ is used to represent the absolute error inequality $|tr(f(A)) - T_d(f(A))| \le \hat{\epsilon}$. Thus for every Lanczos iteration, the upper bound of (18) is compared to the upper bound of (17).

It is also crucial to determine what is the right value for d. We use two different methods:

- **Theoretical upper bound**: The first way is to use the theoretical upper bound from corollary 3.2 of [1], which states that:
  Let $A \in \mathbb{R}^{n\times n}$ be symmetric with $\sigma(A) \subseteq [a, b]$, $0 \le a < b$, and put $\gamma = a/b$. Then

$$|S(A) - T_d(-A\log(A))| \le nb(1-\sqrt{\gamma})\frac{1+\gamma+2d\sqrt{\gamma}}{2(d^2-1)}\left(\frac{1-\sqrt{\gamma}}{1+\sqrt{\gamma}}\right)^d \le \hat{\epsilon}, \tag{19}$$

  for all $d \ge 2$. The goal is to find the smallest d so that the previous relation is respected.

- **Heuristic method** From section 5.1 of [1] we have the following relation:

$$|\operatorname{tr}(f(A)) - T_d(f(A))| \approx |T_{d+1}(f(A)) - T_d(f(A))| = \frac{C}{d^k}q^d, \quad d = 1, 2. \tag{20}$$

for k = 2 and some parameters $C > 0$ and $q \in (0, 1)$. However sometimes the behavior is best described by k=3 so we tested both values. We select d as:

$$d_\star = \min \left\{ d : \frac{C}{d^k} q^d \leq \hat{\epsilon} \right\}, \tag{21}$$

and if the previous system (20) is solved:

$$q = 2^k \frac{|T_3(f(A)) - T_2(f(A))|}{|T_2(f(A)) - T_1(f(A))|} \tag{22}$$

$$C = \frac{|T_2(f(A)) - T_1(f(A))|}{q} \tag{23}$$

Test matrix	n	Error	d	Colors	Time (s)
Yeast	2224	2.89e-04	3	222	27.441
		6.74e-05	28	2224	5.6962
Minnesota	2640	5.06e-04	5	24	12.7711
		7.96e-05	19	276	1.2316
ca-HepTh	8638	2.89e-04	3	252	68.116
		6.75e-05	30	8638	33.8286
bcsstk29	13830	1.96e-04	2	87	12.0268
		2.25-05	13	2424	8.7986

Test matrix	n	Error	Iterations	Time (s)
Yeast	2224	1.09e-06	18295	19.2082
		6.19e-06	283304	864.8032
Minnesota	2640	4.52e-06	4121	3.4625
		2.61e-06	89815	542.2226
ca-HepTh	8638	2.97e-06	49639	48.936
		2.43e-06	222980	1376.113
bcsstk29	13830	2.81e-06	4117	19.2311
		8.00e-06	34006	358.6217

Table 2: Reproduction of Table 2 and 3 of [1]. (Top): comparison of the heuristic estimate (top row) and the theoretical bound (bottom row) for choosing d, with $\epsilon = 10^{-3}$. (Bottom) Comparison of the geometric mean estimate (top row) and the upper bound (bottom row) for $\epsilon = 10^{-5}$.

In table 2 there are many similarities with table 2 from [1] for the values of the errors, d and colorings. The most flagrant discrepancy is in the run times which is most probably due to the computation of $\mathcal{T}_d(f(A))$ for $d = 1, 2, 3$, $k = 2, 3$, using the upper bound (17) equation, as there are far more iterations being done. In the case we use the theoretical bound (19) and the polynomial Lanczos is implemented we observe very similar results to [1] which is unexpected as it has a slower convergence rate than the rational method. The most probable explanation for this is the use of a faster processor (AMD Ryzen 9 5900X 12-Core Processor running at 3.70 GHz).

For table 3, we notice a discrepancy in the run times for the upper bound which is orders of magnitude slower. For the geometric mean estimates we have similar run times.What is quite surprising, is that the error obtained with the theoretical upper bound is higher than the error obtained with the geometric mean estimate.

References

[1] Benzi, M., Rinelli, M., & Simunec, I. (2023). Computation of the von Neumann entropy of large matrices via trace estimators and rational Krylov methods. Numerische Mathematik, 155(3), 377-414.

[2] Güttel, S. (2013). Rational Krylov approximation of matrix functions: Numerical methods and optimal pole selection. GAMM-Mitteilungen, 36(1), 8-31.

[3] Crouzeix, M. (2007). Numerical range and functional calculus in Hilbert space. Journal of Functional Analysis, 244(2), 668-690.

[4] Crouzeix, M., & Palencia, C. (2017). The Numerical Range is a $(1+\sqrt{2})$-Spectral Set. SIAM Journal on Matrix Analysis and Applications, 38(2), 649–655. doi:10.1137/17M1116672