Cucheb: A GPU implementation of the Filtered Lanczos procedure

University of Minnesota
Supercomputing Institute

Jared L. Aurentz, Vassilis Kalantzis, and Yousef Saad

ICMAT Instituto de Ciencias Matemáticas

The filtered Lanczos procedure

Many applications require the computation of all eigenvalues and associated eigenvectors lying inside a real interval $[\alpha, \beta]$ of a large and sparse symmetric matrix $A \in \mathbb{R}^{n \times n}$.

The Lanczos method is an efficient approach when $[\alpha, \beta]$ lies on the periphery of the spectrum, and engages A only through Matrix- Vector products. Lanczos is based on a three-term recurrence:

$$Aq_i = \beta_{i-1}q_{i-1} + \alpha_i q_i + \beta_i q_{i+1}, \; (q_0 = 0, \; \beta_1 = 0).$$

In theory, $\{q_1, \ldots, q_{i+1}\}$ form an orthonormal basis. In practice, orthonormality must be explicitly enforced.

The eigenvalues of A are approximated by those of

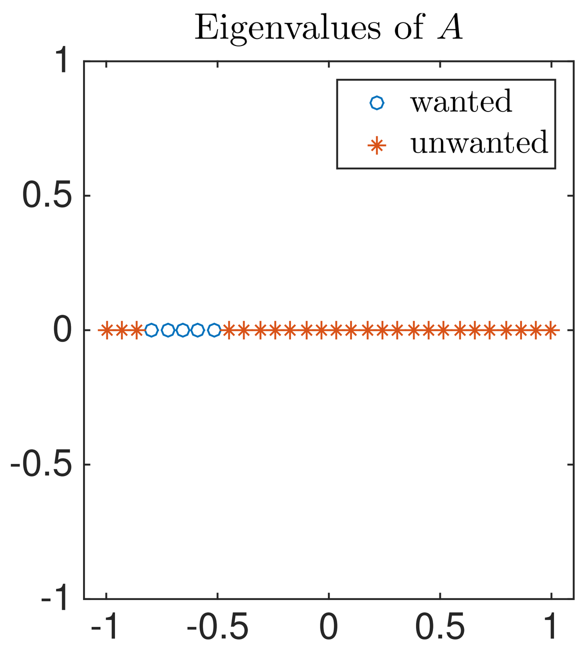
$$T_i = \begin{pmatrix} \alpha_1 & \beta_1 & & & \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \beta_2 & \ddots & \ddots & \\ & & \ddots & \alpha_{i-1} & \beta_{i-1} \\ & & & \beta_{i-1} & \alpha_i \end{pmatrix},$$

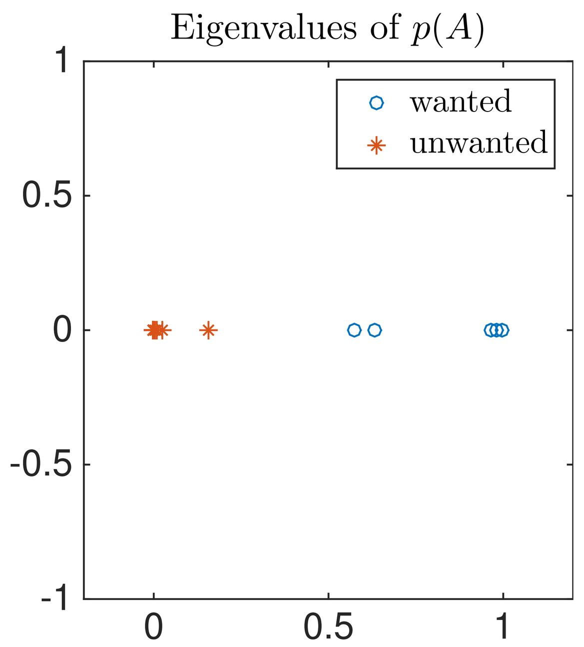
where the peripheral eigenvalues of A converge first and the convergence rate is affected by the relative separation.

What if $[\alpha, \beta]$ lies in the interior of the spectrum and/or includes a large number of eigenvalues $\rightarrow$ Lanczos will perform a large number of steps, increasing memory usage and orthogonalization costs.

The filtered Lanczos procedure applies Lanczos on a carefully chosen polynomial transformation $\rho(.)$ of A (see [2] for details). The goals of $\rho(.)$ are:

1. Eigenvalues of A located inside $[\alpha, \beta]$ are mapped to the top eigenvalues of $\rho(A)$.
2. Construction of $\rho(.)$ requires minimal knowledge of $\Lambda(A)$.
3. Multiplying $\rho(A)$ by a vector is practical.





Chebyshev polynomial filtering

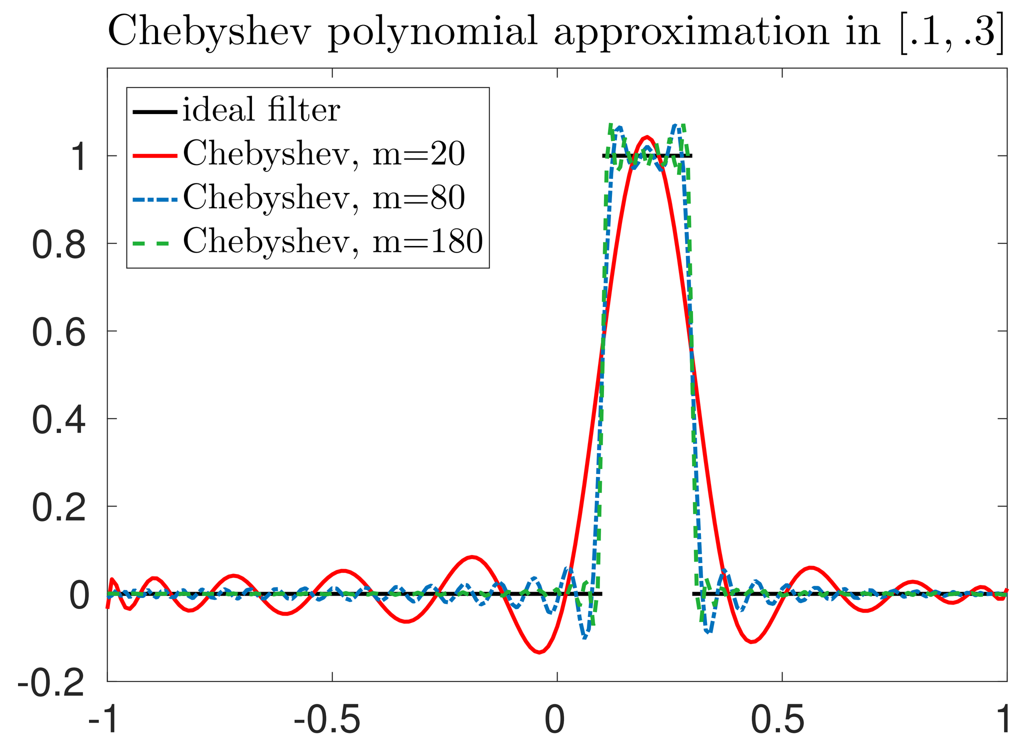
A simple and efficient approach for constructing $\rho(.)$ is to fix a degree m and approximate the step function $I_{[\alpha,\beta]}$ by

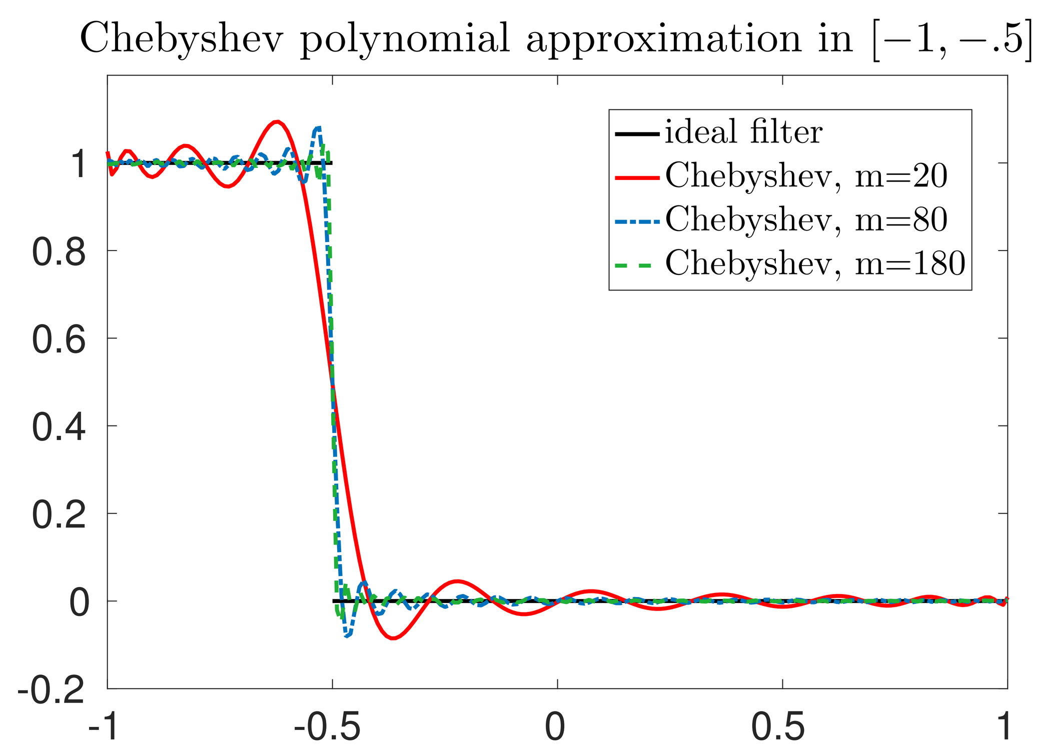
$$\rho_m(z) = \sum_{j=0}^{m} b_j T_j(z),$$

where T_j denotes the j'th degree Chebyshev polynomial of the first kind.

For a given α and β the $\{b_j\}$ are known analytically,

$$b_j = \begin{cases} (\arccos(\alpha) - \arccos(\beta))/\pi, & j = 0, \\ 2(\sin(j\arccos(\alpha)) - \sin(j\arccos(\beta)))/j\pi, & j > 0. \end{cases}$$





References

[1] Jared L. Aurentz, Vassilis Kalantzis, and Yousef Saad. *Cucheb: A GPU implementation of the filtered Lanczos procedure.* Submitted.

[2] Haw-Ren Fang, and Yousef Saad. *A Filtered Lanczos Procedure for Extreme and Interior Eigenvalue Problems.* SIAM J. Sci. Comput., **34**, A2220-A2246 (2012).

U.S. Department of Energy

GPU implementation

Image courtesy of *www.pny.com*

Cucheb is written in CUDA and as of now targets NVIDIA Graphic Processing Units:

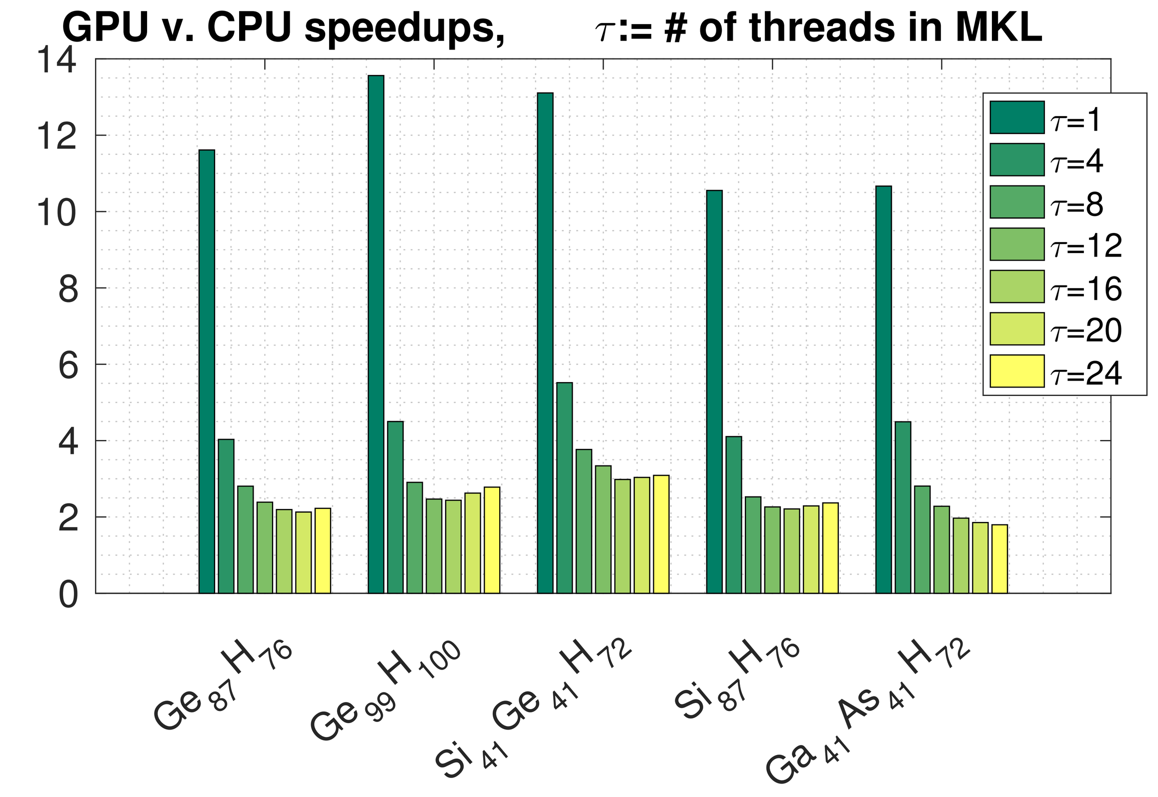
⇒ Cucheb implements a non-restarted, filtered block Lanczos procedure using full orthogonalization.

⇒ Matrices are loaded and handled using the CSR format. Sparse (dense) linear algebra is performed by cuSPARSE (cuBLAS).

⇒ The user need provide only the matrix and the interval of interest $[\alpha, \beta]$.

https://github.com/jaurentz/cucheb

Performance of Cucheb using a block variant of Lanczos



Matrix	n	nnz/n
Ge87H76	112,985	69.9
Ge99H100	112,985	74.8
Si41Ge41H72	185,639	80.9
Si87H76	240,369	44.4
Ga41As41H72	268,096	69.0

Test matrices: We tested Cucheb on a few Hamiltonians generated using the PARSEC package.

Hardware: K40m GPU with 11 GB of RAM and 2880 CUDA cores. The host CPU was a Haswell Xeon E5-2680 processor.

Matrix	interval	eigs	m	iters	MV	time
			50	210	31,500	31
			100	180	54,000	40
Ge87H76	$[-0.645, -0.0053]$	212	150	150	67,500	44
			50	210	31,500	32
			100	180	54,000	41
Ge99H100	$[-0.650, -0.0096]$	250	150	180	81,000	56
			50	210	31,500	56
			100	180	54,000	73
Si41Ge41H72	$[-0.640, -0.0028]$	218	150	180	81,000	99
			50	150	22,500	38
			100	90	27,000	35
Si87H76	$[-0.660, -0.3300]$	107	150	120	54,000	63
			200	180	144,000	225
			300	180	162,000	236
Ga41As41H72	$[-0.640, 0.0000]$	201	400	180	216,000	306