Deflation as a Method of Variance Reduction for Estimating the Trace of a Matrix Inverse

Andreas Stathopoulos
Parts of the work with A. S. Gambhir, J. Laeuchli, and K. Orginos

Computer Science Department and Physics Department
College of William and Mary

Acknowledgment: NSF, DOE, Jefferson Lab

The problem

Given a large, $N \times N$ matrix A and a function f

find trace of $f(A)$: $t(f(A))$

Common functions:

$$\begin{aligned} f(A) &= A^{-1} \\ f(A) &= \log(A) \\ f(A) &= R_i^T A^{-1} R_j \end{aligned}$$

Applications: Data mining, QMC, Uncertainty Quantification, ...

Our focus LQCD: $f(A) = A^{-1}$ or $f(A) = \Gamma A^{-1}$

Standard underlying method

Monte Carlo (Hutchinson 1989)

If x is a vector of random Z_2 variables

$$x_i = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}$$

then

$$\boxed{E(x^T A^{-1} x) = t(A^{-1})}$$

Monte Carlo Trace
for i=1:n
  x = randZ2(N,1)
  sum = sum + $x^T A^{-1} x$

trace = sum/n

Standard underlying method

Monte Carlo (Hutchinson 1989)

If x is a vector of random Z_2 variables

$$x_i = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}$$

then

$$\boxed{E(x^T A^{-1} x) = t(A^{-1})}$$

Monte Carlo Trace
for i=1:n
  x = randZ2(N,1)
  sum = sum + $x^T A^{-1} x$

trace = sum/n

2 problems
Large number of samples

How to compute $x^T A^{-1} x$

Standard underlying method

Monte Carlo (Hutchinson 1989)

If x is a vector of random Z_2 variables

$$x_i = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}$$

then

$$\boxed{E(x^T A^{-1} x) = t(A^{-1})}$$

Monte Carlo Trace
for i=1:n
  x = randZ2(N,1)
  sum = sum + $x^T A^{-1} x$

trace = sum/n

Solve $Ay = x$ vs quadrature $x^T A^{-1} x$
  Golub'69, Bai'95, Meurant'06,'09, Strakos'11
$O(100 - 1000s)$ statistically independent RHS
  Recycling (de Sturler), Deflation (Morgan, AS'07)

Standard underlying method

Monte Carlo (Hutchinson 1989)

If x is a vector of random Z_2 variables

$$x_i = \begin{cases} 1 & \text{with probability } 1/2 \\ -1 & \text{with probability } 1/2 \end{cases}$$

then

$$\boxed{E(x^T A^{-1} x) = t(A^{-1})}$$

Monte Carlo Trace
for i=1:n
  x = randZ2(N,1) What x to use to reduce variance?
  sum = sum + $x^T A^{-1} x$ Equivalently, transform A to have less variance

trace = sum/n

Variance of the Hutchinson estimator

The "squared error" of the statistical estimator $t(A^{-1})$ is its variance

$$\boxed{\mathrm{Var}(t(A^{-1})) = \frac{2}{n}\|\tilde{A}^{-1}\|_F^2 = \frac{2}{n}\left(\|A^{-1}\|_F^2 - \sum_{i=1}^{L} |A_{i,i}^{-1}|^2\right)}$$

where $\tilde{A}^{-1} = A^{-1} - \mathrm{diag}(\mathrm{diag}(A^{-1}))$

Variance of the Hutchinson estimator

The "squared error" of the statistical estimator $t(A^{-1})$ is its variance

$$\mathrm{Var}(t(A^{-1})) = \frac{2}{n}\|\tilde{A}^{-1}\|_F^2 = \frac{2}{n}\left(\|A^{-1}\|_F^2 - \sum_{i=1}^{L} |A_{i,i}^{-1}|^2\right)$$

where $\tilde{A}^{-1} = A^{-1} - \mathrm{diag}(\mathrm{diag}(A^{-1}))$

Thus, the goal of variance reduction:

remove weight from the off-diagonals elements of A^{-1}

Variance of the Hutchinson estimator

The "squared error" of the statistical estimator $t(A^{-1})$ is its variance

$$\boxed{\mathrm{Var}(t(A^{-1})) = \frac{2}{n}\|\tilde{A}^{-1}\|_F^2 = \frac{2}{n}\left(\|A^{-1}\|_F^2 - \sum_{i=1}^{L} |A_{i,i}^{-1}|^2\right)}$$

where $\tilde{A}^{-1} = A^{-1} - \mathrm{diag}(\mathrm{diag}(A^{-1}))$

Thus, the goal of variance reduction:

$$\boxed{\text{remove weight from the off-diagonals elements of } A^{-1}}$$

- Choose vectors that remove particular patterns of A^{-1} (Hierarchical Probing)
- Approximate $M \approx A^{-1}$, $t(A^{-1}) = t(M) + t(A^{-1} - M)$
  hope that $t(A^{-1} - M)$ has smaller variance (SVD deflation)

Dilution and Probing

Dilution: removes variance from a pre-determined pattern of A [Peardon 2005]

Spin/Color, Time, or Even/Odd dilution most common

Probing: a general algebraic technique to compute sparse matrix approximations

originally for computing Jacobians and banded matrix approximations [Coleman & Moré, 82][Chan & Mathew, 92]

Dilution uses the same technique as probing

Dilution and Probing

Dilution: removes variance from a pre-determined pattern of A [Peardon 2005]

Spin/Color, Time, or Even/Odd dilution most common

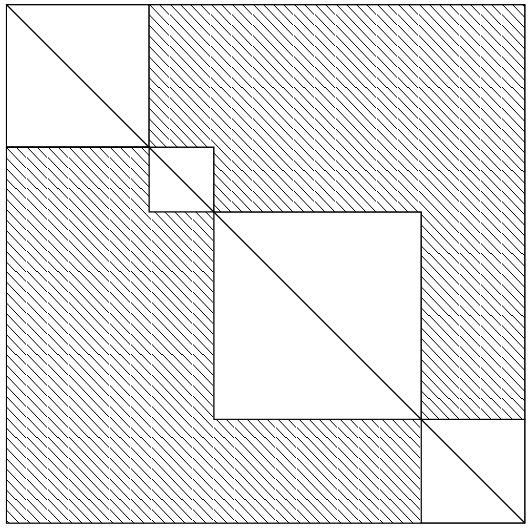
Probing: a general algebraic technique to compute sparse matrix approximations

originally for computing Jacobians and banded matrix approximations [Coleman & Moré, 82][Chan & Mathew, 92]

Dilution uses the same technique as probing

E.g.,
the trace of an m-colorable sparse matrix is recovered exactly by m vectors

$$t(A) = t(h^T A h)$$



1	0	0	0
1	0	0	0
1	0	0	0
1	0	0	0
0	1	0	0
0	1	0	0
0	0	1	0
0	0	1	0
0	0	1	0
0	0	1	0
0	0	1	0
0	0	1	0
0	0	0	1
0	0	0	1
0	0	0	1

Problem 1: How do we determine the sparsity pattern for Probing/Dilution?
Answer: From graph theory: multi-coloring

Problem 2: A^{-1} is not sparse. How do we approximate it?
Answer: Green's function

A_{ij}^{-1} decay in magnitude with the distance between i and j in the graph of A

Problem 1: How do we determine the sparsity pattern for Probing/Dilution?
Answer: From graph theory: multi-coloring

Problem 2: A^{-1} is not sparse. How do we approximate it?
Answer: Green's function

> A^{-1}_{ij} decay in magnitude with the distance between i and j in the graph of A

Classic Probing for $t(A^{-1})$ [Bekas et al, 07][Tang & Saad,'10]
Color $A^k \equiv$ distance-k coloring of A. Captures largest elements of A^{-1}

Problem 1: How do we determine the sparsity pattern for Probing/Dilution?
Answer: From graph theory: multi-coloring

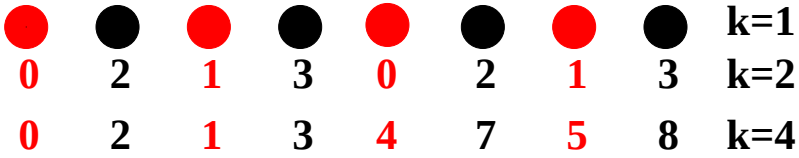
Problem 2: A^{-1} is not sparse. How do we approximate it?
Answer: Green's function

> A^{-1}_{ij} decay in magnitude with the distance between i and j in the graph of A

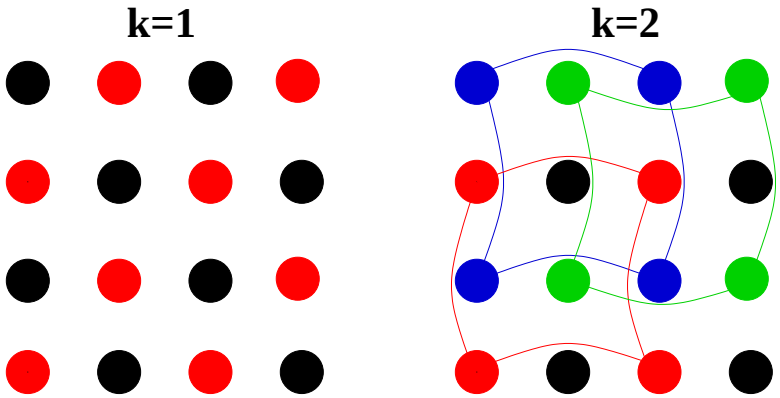
Classic Probing for $t(A^{-1})$ [Bekas et al, 07][Tang & Saad,'10]
Color $A^k \equiv$ distance-k coloring of A. Captures largest elements of A^{-1}

Limitations of CP:
(1) If $t(A^{-1})$ not accurate enough, discard work and repeat for larger k
(because the subspaces of the vectors h are not hierarchical)
(2) Coloring A^k very expensive for large k

Hierarchical Probing on lattices [JLAS,'13] Distances $k = 2^m, m = 1, 2, \ldots$



1D: HP(L lattice)
- Red-black color L
- HP(red nodes)
- HP(black nodes)



2D: HP(L lattice, $d = 2$)
- Red-black color L
- Split L to 2^d sublattices, L_i
- for $i = 1 : 2^d$
  - HP(L_i)

d-D: same algorithm

– Extremely efficient implementation
– Extensible to non-powers of 2
– Probing vectors H special permutations of Hadamard vectors
– $z \odot H(:, i)$ makes it statistical

SVD deflation for variance reduction

If U_1, Σ_1, V_1 are k known singular triplets of A, then

$$A = U_1 \Sigma_1 V_1^T + U_2 \Sigma_2 V_2^T = A_D + A_R$$

and

$$t(A^{-1}) = t(A_D^{-1}) + t(A_R^{-1})$$

SVD deflation for variance reduction

If U_1, Σ_1, V_1 are k known singular triplets of A, then

$$A = U_1 \Sigma_1 V_1^T + U_2 \Sigma_2 V_2^T = A_D + A_R$$

and

$$t(A^{-1}) = t(A_D^{-1}) + t(A_R^{-1})$$

$t(A_D^{-1})$ computed trivially

$t(A_R^{-1})$ using the Hutchinson estimator $t(A_R^{-1})$

SVD deflation for variance reduction

If U_1, Σ_1, V_1 are k known singular triplets of A, then

$$A = U_1 \Sigma_1 V_1^T + U_2 \Sigma_2 V_2^T = A_D + A_R$$

and

$$t(A^{-1}) = t(A_D^{-1}) + t(A_R^{-1})$$

$t(A_D^{-1})$ computed trivially

$t(A_R^{-1})$ using the Hutchinson estimator $t(A_R^{-1})$

Singular vectors vs Eigenvectors

- Easier to model theoretically
- $\Gamma A = (\Gamma U)\Sigma V^T$ is also SVD decomposition, if Γ unitary
- Works better in practice

SVD deflation for variance reduction

Question: does the variance reduce?

$$\mathrm{Var}(t(A_R^{-1})) = \|\tilde{A}_R^{-1}\|_F^2 < \|\tilde{A}^{-1}\|_F^2 = \mathrm{Var}(t(A^{-1}))$$

WLOG consider seeking $t(A)$ instead of $t(A^{-1})$

Variance analysis

SVD based deflation implies:

$$\|A\|_F^2 = \|A_D\|_F^2 \; + \; \|A_R\|_F^2$$

Let $D = \mathrm{diag}(A)$, $D_D = \mathrm{diag}(A_D)$ and $D_R = \mathrm{diag}(A_R)$

Theorem:

$$\boxed{\|\tilde{A}\|_F^2 = \|\tilde{A}_D\|_F^2 + \|\tilde{A}_R\|_F^2 - 2\mathrm{Real}(D_D^T D_R)}$$

Possible that variance increases!

Variance analysis

SVD based deflation implies:

$$\|A\|_F^2 = \|A_D\|_F^2 \; + \; \|A_R\|_F^2$$

Let $D = \mathrm{diag}(A)$, $D_D = \mathrm{diag}(A_D)$ and $D_R = \mathrm{diag}(A_R)$

Theorem:

$$\|\tilde{A}\|_F^2 = \|\tilde{A}_D\|_F^2 + \|\tilde{A}_R\|_F^2 - 2\mathrm{Real}(D_D^T D_R)$$

Possible that variance increases!

Example:

```
>> [U,~] = qr([ -1  1  1          A = U * [ 2   0   0
                1  1  1                     0  1.5  0
                1  1 -1 ]);                 0   0   1 ] * U';
>> Ar = U(:,2:3)*diag([1.5, 1])*U(:,2:3)';
Var(t(A))/Var(t(Ar))
    0.4074
```

Variance analysis with singular values and vectors

Theorem:
Assume we deflate the largest k singular triplets U_1, Σ_1, V_1
Let $\Delta = (U \odot \bar{V})^H (U \odot \bar{V})$ which implies

$$\Delta_{ml} = \sum_{i=1}^{N} \bar{u}_{im} v_{im} u_{il} \bar{v}_{il}, \qquad m, l = 1, \ldots, N$$

Then,

$$\frac{1}{2}\mathrm{Var}(t(A_R)) = \sum_{m=k+1}^{N} \sigma_m^2 - \sum_{m=k+1}^{N} \sum_{l=k+1}^{N} \sigma_m \sigma_l \Delta_{ml}$$

$$\frac{1}{2}(\mathrm{Var}(t(A)) - \mathrm{Var}(t(A_R))) = \sum_{m=1}^{k} \sigma_m^2 (1 - \Delta_{mm}) - \sum_{m=1}^{k} \sum_{l=m+1}^{N} \sigma_m \sigma_l (\Delta_{ml} + \Delta_{lm}).$$

Variance analysis with singular values and vectors

Problems, if Δ_{ml} has highly localized density

Best case scenario:
If $\sigma_1 > 2\sigma_2 > 4\sigma_3 > \ldots > 2^k\sigma_{k+1}$ variance reduces with any singular vectors

Difficult to characterize Δ_{ml}

Δ Hermitian positive definite

For Hermitian matrices $\Delta_{ml} = \sum_{i=1}^{N} |u_{im}|^2 |u_{il}|^2 \Rightarrow \Delta$ doubly stochastic

How can we factor Δ_{ml} out? Average case

The average case for Δ

Assumption:

U, V standard random unitary matrices (Haar distribution)

Assumption justified in Lattice QCD [Shuryak, Verbaarschot, Carlsson]

Jiang showed that $\sqrt{N}U_{ml} \sim \mathcal{N}(0,1)$

but only for submatrices up to $O(\sqrt{N} \times \sqrt{N})$

Our formula involves more than $O(N)$ elements of U and V

Instead we used random matrix theory to obtain $E(\Delta_{ml})$ and $\mathrm{Var}(\Delta_{ml})$ directly

Bounding the Δ_{ml}

Lemma: For non-Hermitian matrices,

$$\begin{aligned} E(\Delta_{ml}) &= 0, \\ \mathrm{Var}(\Delta_{ml}) &= 1/(N(N+1)^2), \\ E(\Delta_{mm}) &= 1/N, \\ \mathrm{Var}(\Delta_{mm}) &= (N-1)/(N^2(N+1)^2). \end{aligned}$$

$$|\Delta_{ml}| = O(N^{-1.5}),\ \Delta_{mm} = O(N^{-1})$$

Lemma: For Hermitian matrices,

$$\begin{aligned} E(\Delta_{ml}) &= 1/(N+1), \\ \mathrm{Var}(\Delta_{ml}) &= 2(N-1)/((N+1)^2(N+2)(N+3)), \\ E(\Delta_{mm}) &= 2/(N+1), \\ \mathrm{Var}(\Delta_{mm}) &= 4(N-1)/((N+1)^2(N+2)(N+3)). \end{aligned}$$

$$\Delta_{ml} = O(N^{-1}) = \Delta_{mm}$$

Expected variance of the deflated estimator

Theorem: Define the mean and the variance of the $N-k$ singular values of A_R

$$\mu_k = 1/(N-k) \sum_{m=k+1}^{N} \sigma_m$$

$$V_k = 1/(N-k) \sum_{m=k+1}^{N} (\sigma_m - \mu_k)^2$$

Then, for non-Hermitian matrices it holds

$$\frac{1}{2}E(\mathrm{Var}(t(A_R))) = (N-k)(1-\frac{1}{N})(V_k+\mu_k^2)$$

and for Hermitian matrices,

$$\frac{1}{2}E(\mathrm{Var}(t(A_R))) = (N-k)\left(V_k\frac{N}{N+1}+\mu_k^2\frac{k}{N+1}\right).$$

A model only on σ_i to predict whether deflation will reduce variance and how much

A remarkable result

For non-Hermitian matrices and for any $1 \leq k \leq N$, variance does not increase!

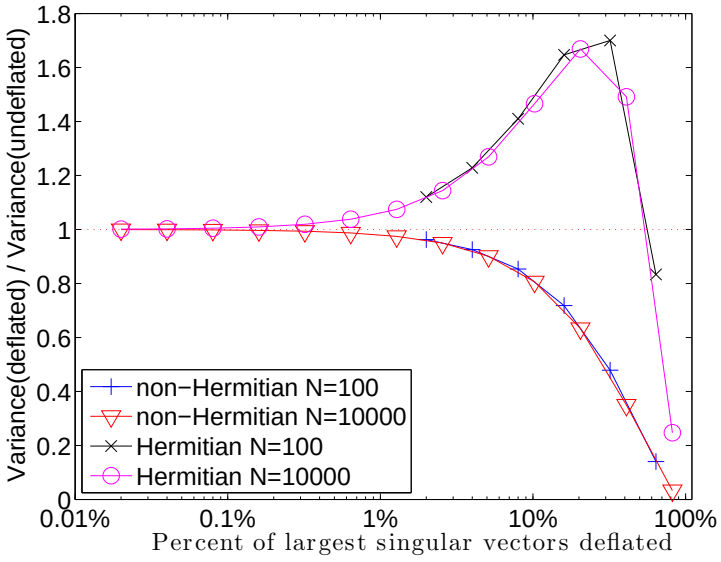
$$E(\mathrm{Var}(t(A_R))) \leq E(\mathrm{Var}(t(A)))$$

For Hermitian matrices, the expected deflated variance reduces only if

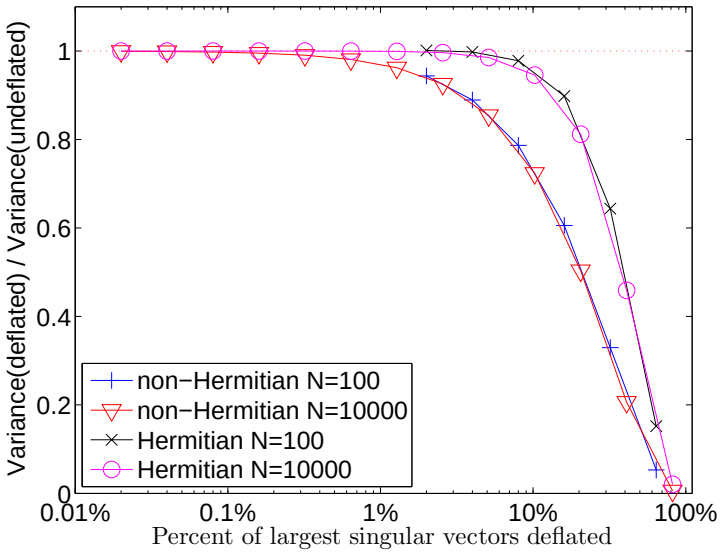
$$\mu_0^2 - \frac{(N-k)^2}{N^2}\mu_k^2 < \frac{1}{N}\sum_{i=1}^{k}\sigma_i^2$$

We are not aware of another property where non-Hermitian matrices may outperform Hermitian ones

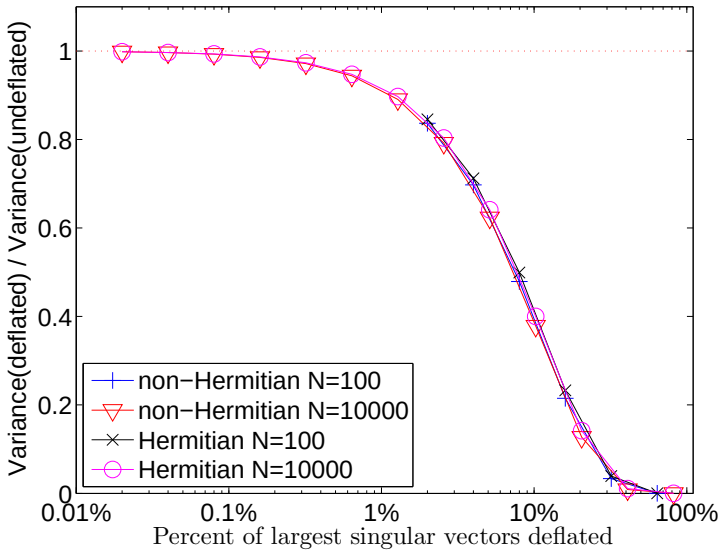
Deflation benefits wrt singular spectra decay



$\sigma_{N-i+1} = \sqrt{i}$

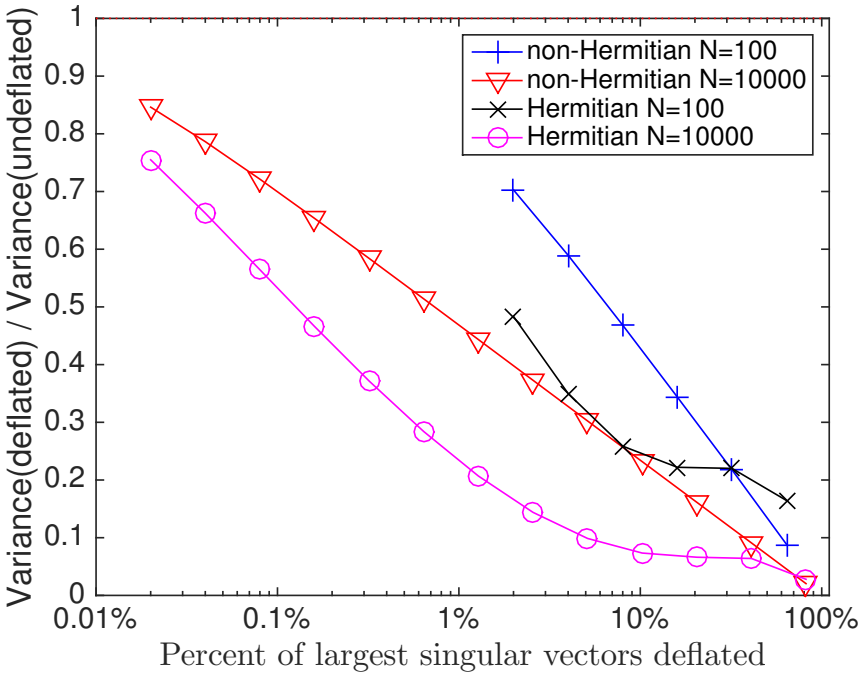


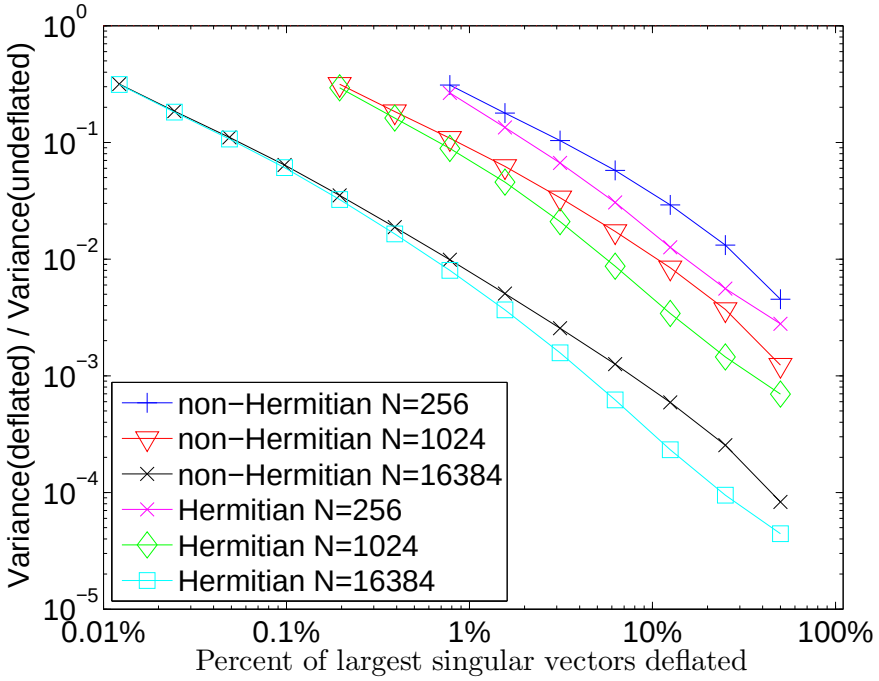
$\sigma_{N-i+1} = i$



$\sigma_{N-i+1} = i^3$

Deflation benefits wrt singular spectra decay



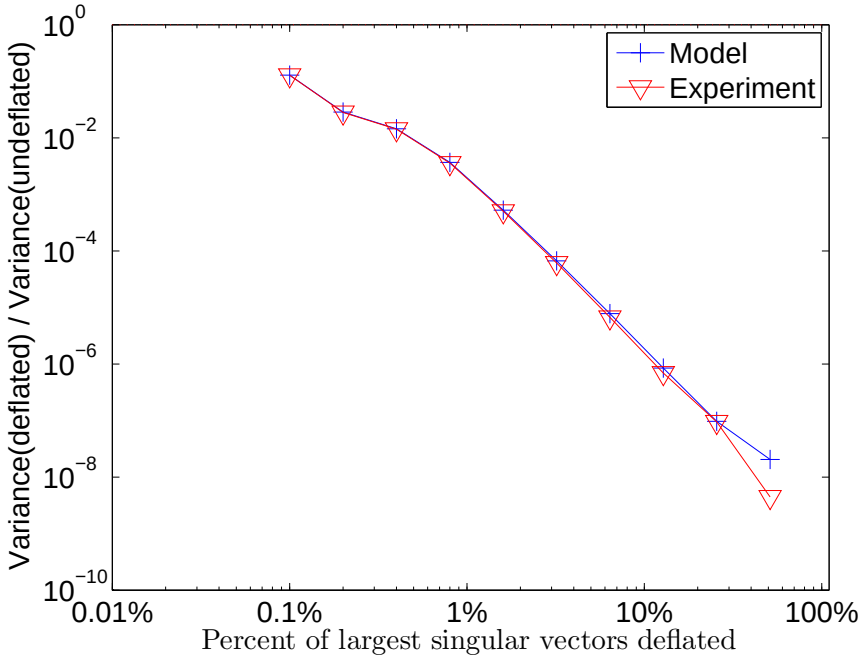


$\sigma_i = 1/\sqrt{i}$

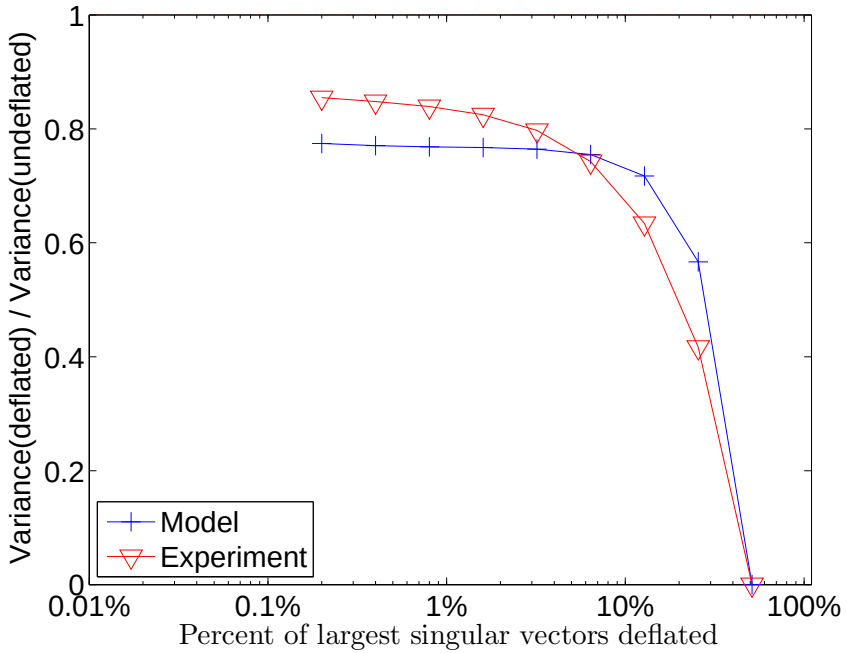
σ_i inverse of 2D Laplacian

Works better for inverses

Robust for matrices with non random singular vectors



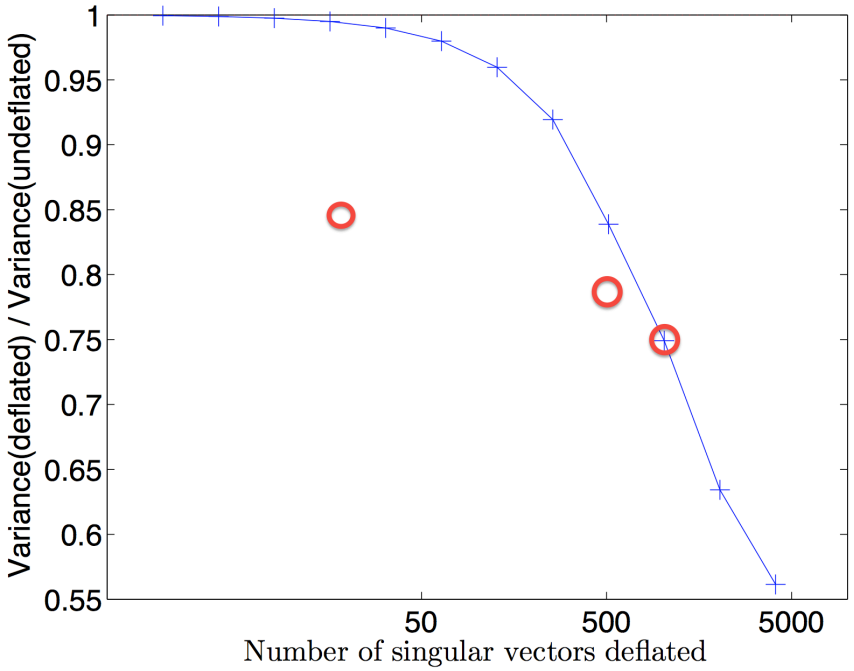
BWM2000



OLM1000

4D lattice: $32^3 \times 64$ with 12 unknowns per node (25 million matrix size)

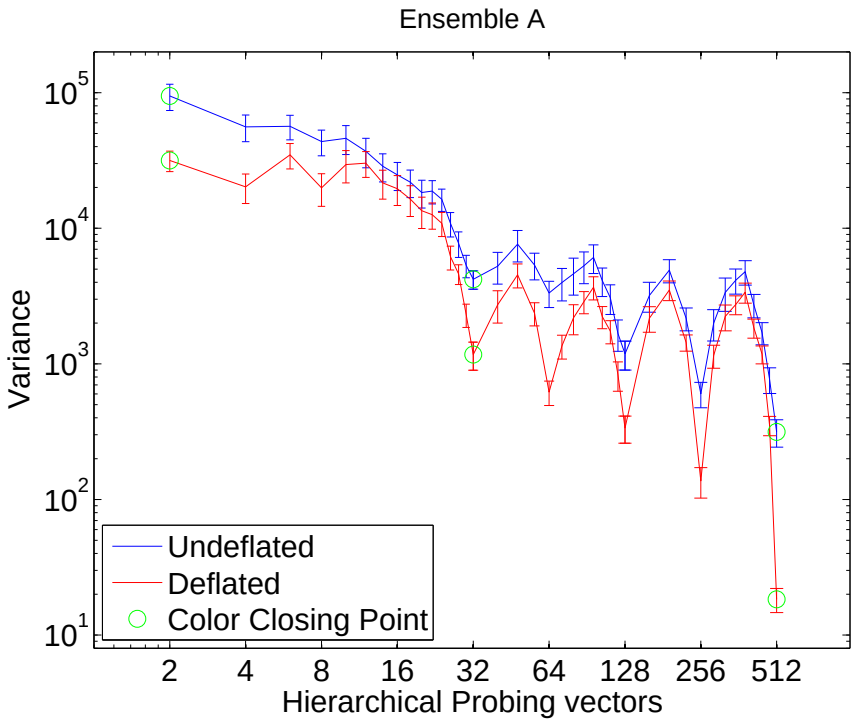
Variance improvement estimated through MC (red circles)

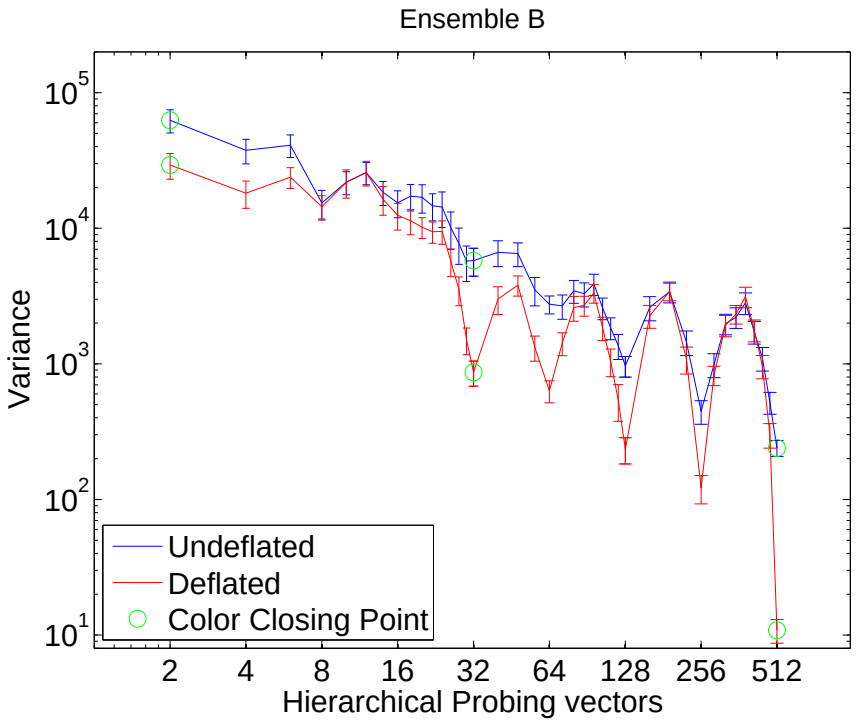


Model and experiment agree that 1000 singular vectors give 25% speedup

Hierarchical Probing gets 2 or 3-fold speedup on this ill conditioned problem!

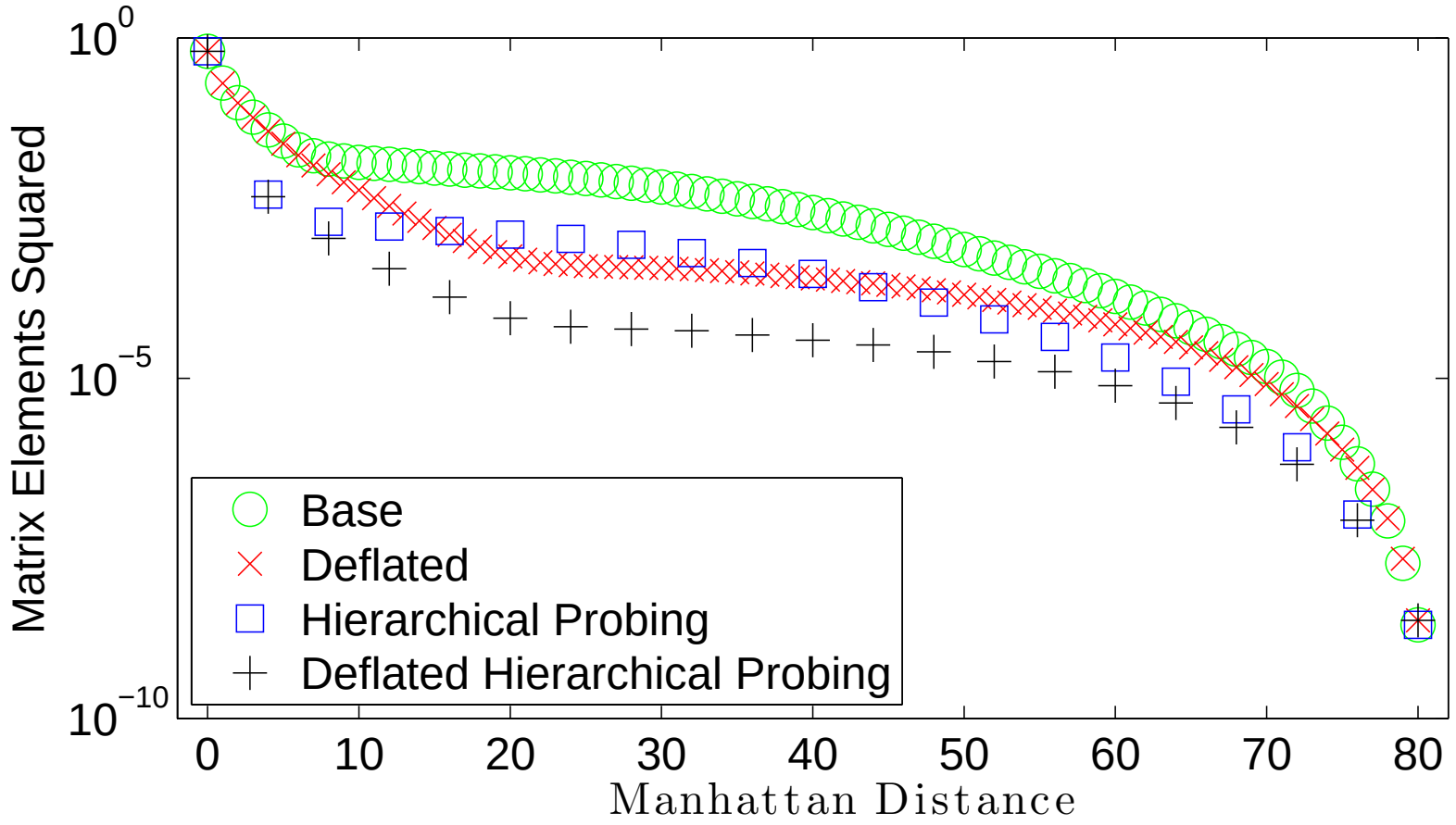
The synergy of HP with Deflation





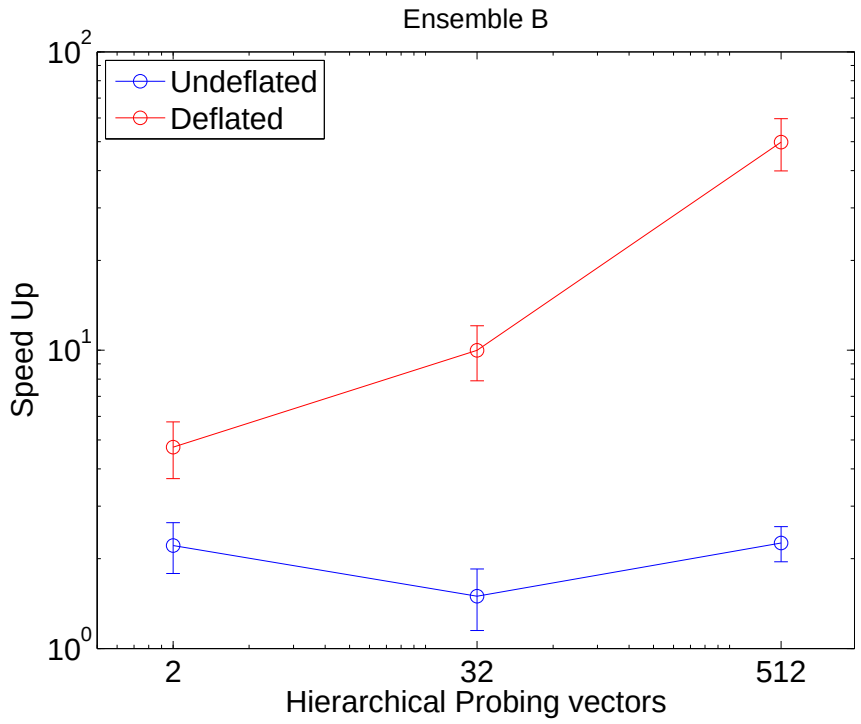
Speedup: 10-15 over HP!

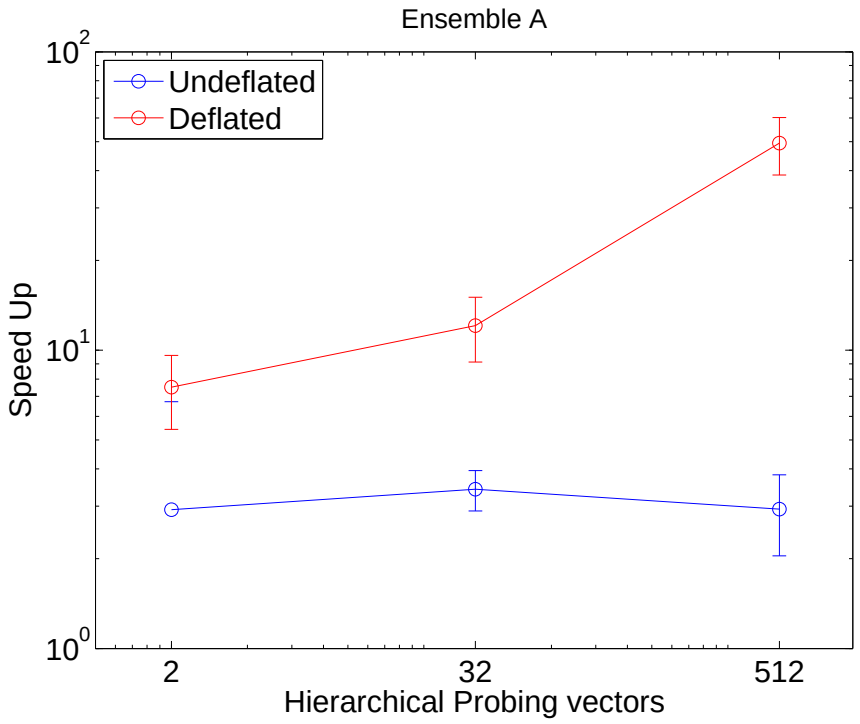
The synergy of HP with Deflation



HP acts at local distances, Deflation at long distances

The synergy of HP with Deflation

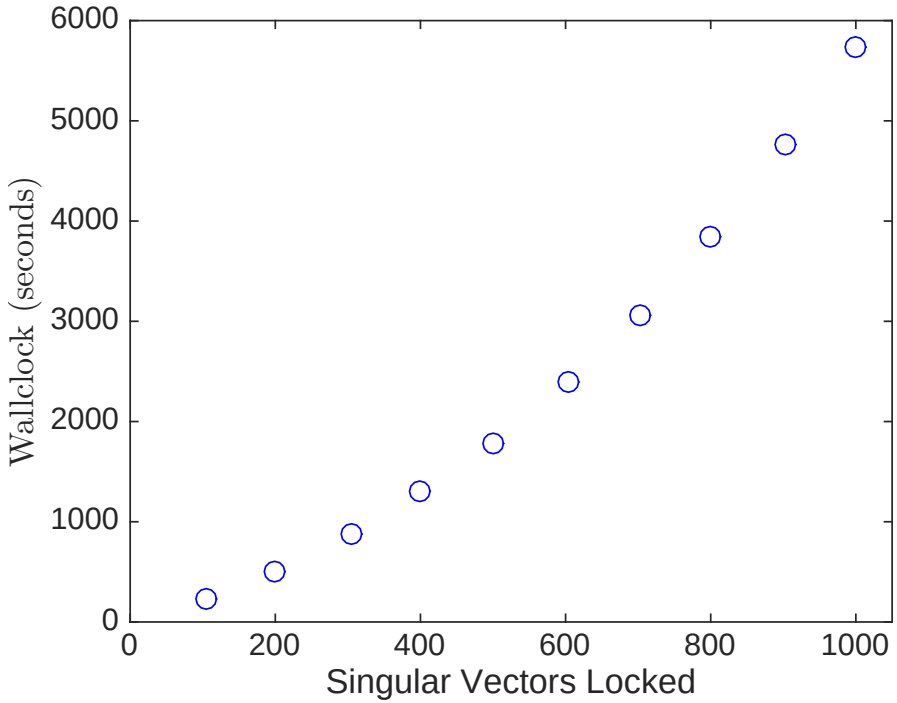


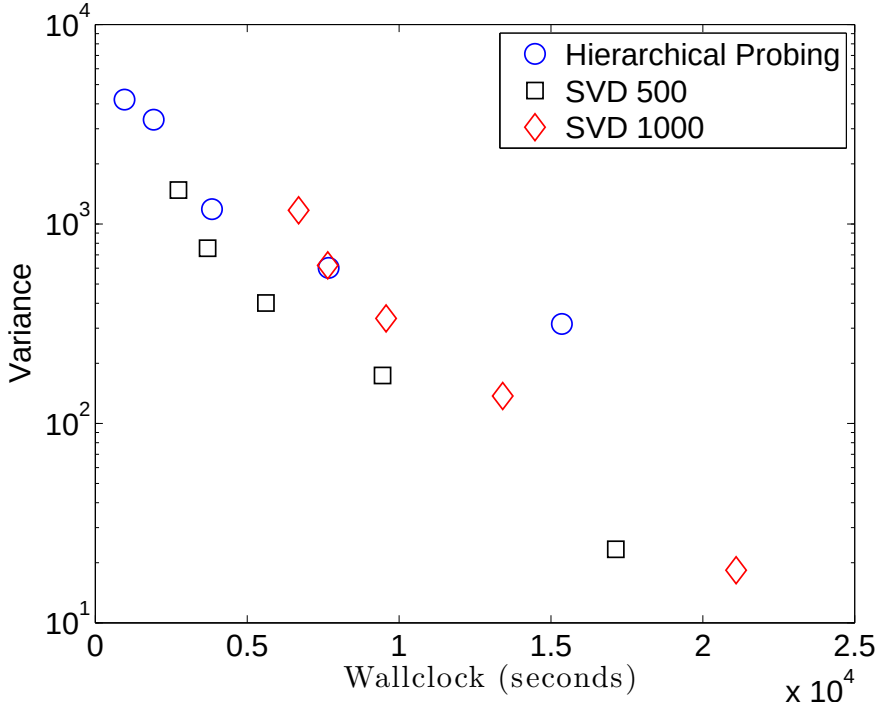


More than 50 speedup over MC!

Singular triplet computation and timings

PRIMME using block method and AMG preconditioner (∼30-fold speedup)





Overhead of PRIMME is less than 10% of a one-shot calculation

Singular triplets can be used for many $t(\Gamma A^{-1})$ and other computations

Scalability issues

Well-conditioned problems

deflation effect smaller but HP effect much larger

Ill-conditioned problems

HP effect reduces but deflation becomes important

Scalability issue:

The size of deflation subspace scales up with the volume

Using Multigrid for obtaining singular values scaled it back down

To reduce the cost of application of the subspace we are currently working on a multigrid representation of the singular vectors

Conclusions

- Analyzed effects of deflation on Hutchinson method
- Analyzed "expected" effects when U, V are standard random unitary matrices
- The model based on the distribution of singular values robust in general
- Applied results on a large scale disconnected diagrams calculation
  - Synergy between hierarchical probing and deflation
  - 50-fold speedup over MC
  - A challenging eigenvalue computation with PRIMME+AMG

For more details see arXiv 1603.05988