PROCEEDINGS OF SCIENCE

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

Andreas Frommer[a] **and Mostafa Nasr Khalil**[a,*]

[a] *Department of Mathematics, Bergische Universität Wuppertal*

E-mail: mostafa.khalil@uni-wuppertal.de

Hutchinson's method estimates the trace of a matrix function $f(D)$ stochastically using samples $\tau^H f(D)\tau$, where the components of the random vectors τ obey an isotropic probability distribution. Estimating the trace of the inverse of a discretized Dirac operator or variants thereof have become a major challenge in lattice QCD simulations, as they represent the disconnected contribution to certain observables. The Hutchinson Monte Carlo sampling, however, suffers from the fact that its accuracy depends quadratically on the sample size, making higher precision estimation very expensive. Meyer, Musco, Musco and Woodruff recently proposed an enhancement of Hutchinson's method, termed Hutch++, in which the sample space is enriched by several vectors of the form $f(D)\zeta$, ζ a random vector as in Hutchinson's method. Theoretical analyses show that under certain circumstances the number of these added sample vectors can be chosen in a way to reduce the dependence of the variance of the resulting estimator from the number N of samples from $O(1/N)$ to $O(1/N^2)$.

In this study we combine Hutch++ with our recently suggested multigrid multilevel Monte Carlo approach. We present results for the Schwinger discretization of the 2-dimensional Dirac operator, revealing that the two approaches contribute additively to variance reduction.

The 39th International Symposium on Lattice Field Theory,
8th-13th August, 2022,
Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn, Germany

[*] Speaker

PoS(LATTICE2022)017

1. Introduction

In this study, we consider the task of estimating the trace of the inverse of a large sparse matrix $D \in C^{n\times n}$, $\mathrm{tr}(D^{-1}) = \sum_i^n (D^{-1})_{ii}$. While this task arises in a variety of different fields, we focus on applications in Lattice QCD, where the disconnected fermion loop contribution to an observable is obtained from the trace of the inverse of the discretized Dirac operator, possibly after multiplication with certain γ-matrices; see [1]. The disconnected fermion loop contributions become increasingly important, as they cannot be neglected anymore given the accuracy of current state-of-the-art lattice simulations. Due to its sheer size, the $n \times n$ matrix D^{-1} cannot be computed directly, and the only way to access information on the entries of D^{-1} is through matrix-vector multiplications $D^{-1}\zeta$, i.e. via the solution of linear systems with matrix D. This is where stochastic estimation techniques come into play, starting with Hutchinson's method [2]. Its key component is the use of random vectors $\zeta \in \mathbb{C}^n$, whose components ζ_i obey an isotropic distribution, i.e.

$$\mathbb{E}[|\zeta_i|^2] = 1, \ \ \mathbb{E}[\zeta_i \zeta_j] = 0 \text{ for } i, j = 1, \dots, n, i \neq j. \tag{1}$$

Typically, one takes the components to be identically independent distribution (i.i.d.) complex numbers z with $\mathbb{E}[z] = 0$ and $\mathbb{E}[|z^2|] = 1$. A prominent example is the Rademacher vectors, where z is uniform in $\{-1, 1\}$. Averaging $\zeta^H D^{-1}\zeta$ over s independent random vectors ζ gives an unbiased estimator for the trace. Algorithm 1 shows how to proceed if a given relative target accuracy ϵ (actually: a confidence level of 68% corresponding to the 1σ confidence interval if we rely on the law of large numbers) is to be achieved.

Algorithm 1 plain Hutchinson

```
Input: $D \in \mathbb{C}^{n\times n}$ nonsingular, $\epsilon$ relative accuracy
Output: Approximation $\tau$ for $\mathrm{tr}(D^{-1})$
for $s = 1, 2, \dots$ do
    generate next random vector $\zeta_s$                ▷ $\zeta_s$ i.i.d. satisfying (1)
    $\tau_s \leftarrow \zeta_s^H D^{-1} \zeta_s$          ▷ solve linear system
    $\tau = \frac{1}{s}\sum_{i=1}^{s} \tau_i$             ▷ sample mean
    $V = \frac{1}{s-1}\sum_{i=1}^{s} |\tau_i - \tau|^2$   ▷ sample variance
    if $V/s \leq (\tau\epsilon)^2$ then
        stop
```

More precise theoretical results are known for special classes of matrices as exemplified by the following theorem from [3].

Theorem 1. *Assume that the matrix A is symmetric and positive semidefinite. Let $\mathrm{tr}^H(A)$ denote the Hutchinson estimator with s samples which are Rademacher vectors. Let $\epsilon, \delta \in (0, 1)$. Then, if*

$$s \geq \frac{6}{\epsilon^2} \log \frac{2}{\delta} \tag{2}$$

one has

$$\mathbb{P}\left(\left| \mathrm{tr}_s^H(A) - \mathrm{tr}(A) \right| \leq \epsilon \mathrm{tr}(A) \right) \geq 1 - \delta. \tag{3}$$

The above theorem is a quantitative illustration of the crucial draw-back of Monte Carlo trace estimation: The accuracy increases only with the square root of the number of samples, which makes high accuracy samples practically infeasible unless modifications are found which reduce the variance substantially. In section 2 we will discuss the most common methods for variance reduction of the Hutchinson estimator based on projections. The recent Hutch++ algorithm presented in [4] fits into this category with a special choice for the projection subspace. We do not consider probing methods, which can be used additionally for variance reduction.

In section 3 we then first briefly recall the multilevel Monte Carlo approach relying on a multigrid hierarchy for the matrix D, and then present a new approach which combines multigrid multilevel Monte Carlo with the Hutch++ idea. Numerical results for the Schwinger model will be reported in section 4.

2. Variance Reduction via Projection

For Rademacher vectors, the variance of the Hutchinson estimator for $\text{tr}(D^{-1})$ is given by $\frac{1}{2}\|\text{offdiag}(D^{-1} + D^{-T})\|_F^2$, for Z_4-vectors it is $\|\text{offdiag}(D^{-1})\|_F^2$; see [5], e.g., and the heuristics underlying variance reduction techniques typically rely on just reducing $\|D^{-1}\|_F^2$.

2.1 Deflation

Deflation aims to “remove” a part from the operator which contributes most to the Frobenius norm. Using an oblique or orthogonal projector Π on a yet to be determined k-dimensional subspace one splits $D^{-1} = (I - \Pi)D^{-1} + \Pi D^{-1}$. Usually, $\text{tr}(\Pi D^{-1})$ can be reduced to the trace of a $k \times k$ matrix which can be evaluated directly, and the Hutchinson estimator is used on $(I - \Pi)D^{-1}$. A summary of different choices for the deflating subspace can be found in [6].

Often, the deflating subspace is built from (approxmations to) small eigenmodes of D, i.e. large eigenmodes of D^{-1}. Deflation will thus become increasingly inefficient if the number of large eigenvectors increases with the dimension of D^{-1} (“volume dependence”). Actually, as is argued in [6], it can be advantageous to base deflation on singular triplets rather than eigenmodes. This is because the Frobenius norm is the 2-norm of the vector of singular values, so deflating the k largest singular triplets via a projection on the space spanned by the corresponding k (right) singular vectors of D sets the k largest singular values of D^{-1} to 0 in $(I - \Pi)D^{-1}$.

2.2 Exact Deflation

With (u_i, v_i, σ_i) denoting the singular triplets of D, $Dv_i = \sigma_i u_i$, and the singular values σ_i ordered increasingly, exact deflation uses the orthogonal projector $\Pi = V_k(U_k^H D V_k)^{-1} U_k^H D = V_k V_k^H$, where $U_k = [u_1|\ldots|u_k]$, $V_k = [v_1|\ldots|v_k]$. Then the trace of D^{-1} can be split as

$$\text{tr}(D^{-1}) = \text{tr}((I - \Pi)D^{-1}) + \text{tr}(\Pi D^{-1}). \tag{4}$$

The first term in eq. (4) can be expected to have reduced variance and can be estimated stochastically via Alg. 1 with less samples. The second term is available directly since $\text{tr}(\Pi D^{-1}) = \text{tr}(V_k^H D^{-1} V_k) = \sum_{i=1}^{k} \frac{1}{\sigma_i} u_i^H v_i$. If instead U_k and V_k contain the left and right eigenvectors belonging to the smallest eigenvalues λ_i of D, then the oblique projector $\Pi = V_k(U_k^H D V_k)^{-1} U_k^H D = V_k U_k^H$

achieves $\mathrm{tr}(\Pi D^{-1}) = \mathrm{tr}(U_k^H D^{-1} V_k) = \sum_{i=1}^{k} \frac{1}{\lambda_i}$. If D is Hermitian and positive definite, the two deflation approaches coincide, since then left and right eigenvectors as well as left and right singular vectors all coincide, and the singular values are the eigenvalues.

2.3 Inexact Deflation

Exact deflation requires the precise computation of singular triplets or eigenpairs, which can be quite costly. We can instead work with approximations and still build the projection Π the same way as in exact deflation. Now, $\mathrm{tr}(\Pi D^{-1})$ is not directly available from approximate singular triplets or eigenvalues and the projector $V_k(U_k^H D V_k)^{-1} U_k^H D$ differs from the projector $V_k V_k^H$, e.g. Using the former gives $\mathrm{tr}(\Pi D^{-1}) = \mathrm{tr}(V_k(U_k^H D V_k)^{-1} U_k^H)$, which requires the inversion of a small $k \times k$ matrix and k multiplications with D. Using the latter gives $\mathrm{tr}(\Pi D^{-1}) = \mathrm{tr}(V_k^H D^{-1} V_k)$ which requires k system solves with the large matrix D.

If we have a sparse representation for U_k and V_k, we can efficiently use very large values for k in inexact deflation. This is the case with multigrid prolongation and restriction operators; see [5, 7, 8] and section 3.

2.4 Hutch++

Hutch++ [4] is an inexact deflation method, where the deflating subspace is obtained from D^{-1}-images of random vectors: We precompute $y_i := D^{-1} s_i$ for d i.i.d. isotropic random vectors s_i. This is one step of a block power method to approximate the largest eigenpairs of D^{-1}. We build an orthogonal projector Π on the space spanned by the y_i as $\Pi = QQ^H$ with the columns of $Q \in \mathbb{C}^{n \times d}$ representing an orthonormal basis for that space spanned, typically obtained through a QR-factorization of $Y = [y_1 | \cdots | y_d]$. The range of the vectors y_i contains, with increasing probability as d increases, good approximations to eigenvectors belonging to large eigenvalues of D^{-1}. As before, we decompose the matrix as

$$D^{-1} = (I - Q)D^{-1} + QD^{-1}. \tag{5}$$

As usual, the trace of the first summand in eq. (5) is estimated stochastically and should have a reduced variance. For the trace of the second term, we use $\mathrm{tr}(QD^{-1}) = \mathrm{tr}(V^H D^{-1} V)$, which requires another d system solves with D. Under the assumption that a system solve with D has cost $O(n^2)$, an asymptotic analysis in [4] shows that for a given budget of N system solves, the optimal choice for d is $d = N/3$. The recent paper [9] develops an adaptive technique to choose d optimally for a given target accuracy. All these results rely on the matrix being Hermitian (and positive definite).

3. Multilevel Monte Carlo (MLMC)

MLMC [10, 11] is a generalization of standard Monte Carlo. The idea is to represent a random variable X as a sum

$$X = \sum_{\ell=1}^{L} X_\ell \tag{6}$$

using additional random variables X_ℓ such that the variance of the X_ℓ is small when it is costly to evaluate and possibly large when it is cheap to evaluate. The different random variables can now be estimated stochastically and independently to obtain an estimator for X.

The variance ρ^2 for the resulting estimator for $\mathbb{E}[X]$ is the sum of the variances of the estimators for $\mathbb{E}[X_\ell]$. In the *uniform* approach one chooses the number N_ℓ of samples at each 'level' ℓ such that $\mathbb{V}[X_\ell]/N_\ell = \rho^2/L$. If one knows the cost C_ℓ for an evaluation of X_ℓ, the problem of minimizing the total cost under the constraint to obtain a variance of ρ^2 is solved for the *optimal* values [11]

$$N_\ell = \frac{1}{\rho^2}\sqrt{\mathbb{V}[X_\ell]/C_\ell}\sum_{j=1}^{L-1}\sqrt{\mathbb{V}[X_j]C_j}. \tag{7}$$

The variance of the estimator for X_ℓ with N_ℓ samples is then

$$\mathbb{V}[X_\ell]/N_\ell = \rho^2\sqrt{\mathbb{V}[X_\ell]C_\ell}\ \Bigg/ \sum_{j=1}^{L-1}\sqrt{\mathbb{V}[X_j]C_j}\ . \tag{8}$$

3.1 Multigrid Multilevel Monte Carlo (MG-MLMC) for the trace

In [5] we proposed a multilevel Monte Carlo method based on a multigrid hierarchy to reduce the variance. One splits the original matrix $D_1^{-1} = D^{-1}$ into a telescopic sum as:

$$\begin{aligned} D_1^{-1} &= (D_1^{-1} - P_1D_2^{-1}R_1) + (P_1D_2^{-1}R_1 - P_1P_2D_3^{-1}R_2R_1)\ldots + P_1\cdots P_{L-1}D_L^{-1}R_{L-1}\cdots R_1 \\ &= \sum_{\ell=1}^{L-1}\left(\hat{P}_\ell D_\ell^{-1}\hat{R}_\ell - \hat{P}_{\ell+1}D_{\ell+1}^{-1}\hat{R}_{\ell+1}\right) + \hat{P}_L D_L^{-1}\hat{R}_L, \end{aligned} \tag{9}$$

where $\hat{P}_\ell = P_1\cdots P_{\ell-1}$, $\hat{R}_\ell = R_{\ell-1}\cdots\hat{R}_1$.

Here, the P_ℓ and R_ℓ are the prolongation and restriction operators between consecutive levels of the multigrid hierarchy, respectively, $D_{\ell+1} = R_\ell D_\ell P_\ell$ are the (Galerkin) coarse grid operators, and $\hat{P}_\ell$ and $\hat{R}_\ell$ are the accumulated prolongations and restrictions which transport between level 1 and ℓ. Note that with the projector $\Pi_1 = P_1D_2^{-1}R_1D_1$ we have $\Pi_1D_1^{-1} = P_1D_2^{-1}R_1$ and similarly for the coarser levels, thus establishing the connection with inexact deflation discussed in section 2. In multigrid, the prolongations P_ℓ are precisely constructed in a manner that they contain good approximations to the small eigenmodes or singular triplets of D_ℓ.

The decomposition eq. (9) gives a multilevel decomposition for the trace as

$$\operatorname{tr}\left(D_1^{-1}\right) = \sum_{\ell=1}^{L-1}\operatorname{tr}\left(\hat{P}_\ell D_\ell^{-1}\hat{R}_\ell - \hat{P}_{\ell+1}D_{\ell+1}^{-1}\hat{R}_{\ell+1}\right) + \operatorname{tr}\left(\hat{P}_L D_L^{-1}\hat{R}_L\right) \tag{10}$$

to be used in a MLMC method. We expect the variance for each level difference $\hat{P}_\ell D_\ell^{-1}\hat{R}_\ell - \hat{P}_{\ell+1}D_{\ell+1}^{-1}\hat{R}_{\ell+1}$ to be small, since the prolongations $P_{\ell+1}$ are built to approximate small eigenpairs or singular triples of D_ℓ. The sizes of the matrices to invert on each level difference decrease significantly with the level, thus making a stochastic sample increasingly less costly.

On the coarsest level L, depending on the size of the matrix D_L, we might be able to compute the trace directly as $\sum_{i=1}^{N_L} e_i^T D_L^{-1}\hat{R}_L\hat{P}_L e_i$. If we do it stochastically, we have to invert a matrix whose dimension is very small compared to that of D.

In the successful multigrid approaches for the Wilson-Dirac matrix or its twisted mass variant, see [12–15], the restrictions and prolongations are aggregation based with $R_\ell = P_\ell^H$, and their columns are orthonormal, $P_\ell^H P_\ell = I$. This is why, using the cyclic property of the trace, eq. (9) gives

$$\mathrm{tr}(D_1^{-1}) = \sum_{\ell=1}^{L-1} \mathrm{tr}\left(D_\ell^{-1} - P_\ell D_{\ell+1}^{-1} P_\ell^H\right) + \mathrm{tr}\left(P_{L-1} D_L^{-1} P_{L-1}^H\right). \tag{11}$$

In contrast to eq. (10) this allows to work with random vectors of the smaller size n_ℓ instead of n on the various difference levels.

3.2 Multigrid Multilevel Monte Carlo++ (MG-MLMC++)

The idea of MG-MLMC++ is to apply the Hutch++ estimator for each of the level differences in the multilevel decomposition eq. (10). We describe the method in Algorithm 2.

Algorithm 2 MLMC++, optimal accuracies, fixed numbers of deflation vectors

Input: $D \in \mathbb{C}^{n\times n}$ nonsingular, ϵ relative accuracy, L number of levels, $\hat R_\ell, \hat P_\ell$ restriction and prolongation operators between levels 1 and ℓ, $D_\ell \in \mathbb{C}^{n_\ell \times n_\ell}$ matrix on level ℓ, d_ℓ number of deflation vectors on level ℓ, $\ell = 1, \dots, L$,

Output: Approximation $\sum_{\ell=1}^{L-1}(\tau_\ell^{\mathrm{lr}} + \tau_\ell) + \tau_L$ for $\mathrm{tr}(D^{-1})$

```
 1: $\tau_L \leftarrow \sum_{i=1}^{N_L}(e_i^T \hat P_L) D_L^{-1}(\hat R_L e_i)$    ▹ coarsest level is computed directly
 2: for $\ell = 1, \dots, L-1$ do    ▹ obtain deflation vectors
 3:     generate $d_\ell$ i.i.d. random vectors $s_i, i = 1, \dots, d_\ell$,    ▹ with distribution satisfying (1)
 4:     collect them as columns in $S_\ell \in \mathbb{C}^{n\times d_\ell}$
 5:     $Y_\ell \leftarrow \left(\hat P_\ell D_\ell^{-1} \hat R_\ell - \hat P_{\ell+1} D_{\ell+1}^{-1} \hat R_{\ell+1}\right) S_\ell$,    ▹ $Y_\ell \in \mathbb{C}^{n\times d_\ell}$, solve linear system.
 6:     Compute QR-factoriz. $Y_\ell = Q_\ell K_\ell$    ▹ $Q_\ell = [q_1|\cdots|q_{d_\ell}] \in \mathbb{C}^{n\times d_\ell}$ has orthon. cols
 7:     $\tau_\ell^{\mathrm{lr}} \leftarrow \sum_{i=1}^{d_\ell} q_i^H \left(\hat P_\ell D_\ell^{-1} \hat R_\ell - \hat P_{\ell+1} D_{\ell+1}^{-1} \hat R_{\ell+1}\right) q_i$    ▹ *low rank part*, use mg to solve lin. sys.
 8: Set all levels $\ell$ to active    ▹ non active levels have reached required accuracy
 9: for $s = 1, 2, \dots$ until all levels $\ell$ not active do    ▹ *stochastic part*
10:     for $\ell = 1, \dots, L-1$ and $\ell$ is active do
11:         generate next random vector $\zeta_s$    ▹ $\zeta_s$ i.i.d. satisfying (1)
12:         $z_s = \zeta_s - Q_\ell(Q_\ell^H \zeta_s)$    ▹ projected vector
13:         $\tau_{s,\ell} \leftarrow z_s^H \left(\hat P_\ell D_\ell^{-1} \hat R_\ell \zeta_s - \hat P_{\ell+1} D_{\ell+1}^{-1} \hat R_{\ell+1} \zeta_s\right)$
14:         $C_{s,\ell} \leftarrow$ cost for lines 12 - 13
15:         $\tau_\ell = \frac{1}{s}\sum_{i=1}^{s} \tau_{i,\ell}$, $V_\ell = \frac{1}{s-1}\sum_{i=1}^{s} |\tau_{i,\ell} - \tau_\ell|^2$    ▹ sample mean and variance
16:         $C_\ell = \frac{1}{s}\sum_{i=1}^{s} C_{i,\ell}$    ▹ average cost per sample
17:     $\tau = \sum_{\ell=1}^{L}(\tau_\ell + \tau_\ell^{\mathrm{lr}})$
18:     for $\ell = 1, \dots, L-1$ and $\ell$ is active do    ▹ update target accuracies $\rho_\ell$
19:         $\rho_\ell \leftarrow \left(\sqrt{C_\ell V_\ell}\,/\,\sum_{j=1}^{L-1}\sqrt{C_j V_j}\right)^{1/2} \cdot (\epsilon\tau)$
20:         if $V_\ell/s \le \rho_\ell^2$ then
21:             set level $\ell$ to inactive
```

Schwinger model					
L		$\ell = 1$	$\ell = 2$	$\ell = 3$	$\ell = 4$
4	n_ℓ	$2 \cdot 128^2$	$4 \cdot 32^2$	$8 \cdot 8^2$	$8 \cdot 2^2$
	$\mathrm{nnz}(D_\ell)$	$2.94e5$	$1.64e5$	$2.46e4$	1024
mass	$m_1 = -0.1320$	$m_2 = -0.1325$	$m_3 = -0.1329$	$m_4 = -0.1332$	$m_5 = -0.1333$
defl. vects.	384	384	512	512	512

Table 1: Parameters used in the Schwinger model and number of deflated eigenvectors chosen in exactly deflated Hutchinson. $\mathrm{nnz}(D_\ell)$ denotes the number of non-zero elements in D_ℓ

Some of its more important features are:

- We assume that we have a cost model to measure the cost for a stochastic sample. We take averages of the cost for each stochastic sample to get increasingly accurate average costs C_ℓ.
- With this measured cost and the measured sample variance V_ℓ we determine the optimal target variance from eq. (8) for each level difference. This target variance is updated at each additional sample on that level difference.
- We describe the algorithm using the decomposition eq. (10) with the accumulated prolongations and restrictions. The adaptation to eq. (11), should it apply, is straightforward.
- The number of deflation vectors d_ℓ for each level difference must be chosen a priori.
- Lines 5 and 6 perform one step of the block power iteration, the crucial ingredient of the Hutch++ method. We can perform more than 1, k say, iterations of the block power method by repeating these lines with S_ℓ in the next sweep equal to Q_ℓ from the previous sweep.

4. Numerical Results

Numerical computations were performed using Python on a single core of a node with 44 cores Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20GHz. We demonstrate the benefits of MG-MLMC++ over exactly deflated Hutchinson and the benefits of MG-MLMC with the two types of accuracies by using the Schwinger discretization of the 2-dimensional Dirac operator [16] with the same configuration and parameters as in [5]. In particular, we use 5 different (negative) masses m to shift the mass-less Schwinger operator by the respective multiple of the identity, thus yielding operators with increasing condition number. The multigrid hierarchy was constructed with a bootstrap setup and aggregation based prolongations as in DDαAMG [14]. Properties of the matrices at the various levels are summarized in the top part of Table 1.

To assess the performance of the algorithms we use a simple cost model which counts the arithmetic operations in all occurring matrix-vector multiplications, i.e. in the projections, the restrictions and prolongations and in the smoothing iteration in the multigrid solver. This arithmetic cost is proportional to the number of non-zeros in the respective matrix, and as an indication, this number is reported for the operators at the different levels in Table 1.

We use a deflated Hutchinson method as our reference for comparison. We did not use non-deflated Hutchinson, because its performance is by two orders of magnitude worse than that of

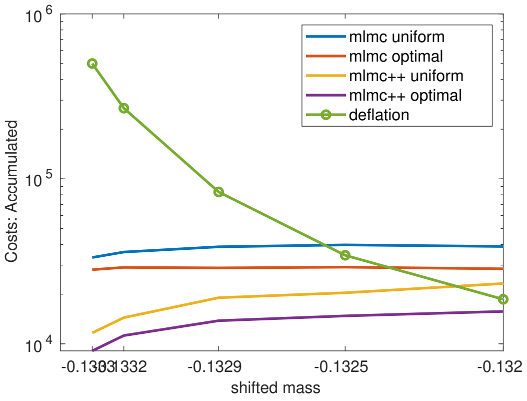

Figure 1: MG-MLMCM, MG-MLMC++ and deflated Hutchinson for the Schwinger matrix: total cost for different masses with uniform and the optimized target variances.

deflated Hutchinson. For deflation, we used the k smallest eigenmodes that we precomputed, and then optimized k so as to obtain the smallest overall cost, *excluding* the cost for the eigenvector computation. So the work for deflated Hutchinson is actually higher than what we report.

Fig. 1 reports the arithmetic cost in MFlops for five different methods: Deflated Hutchinson for reference, MG-MLMC with uniform target variances on the difference levels and its modification working with optimal target variances, and then the corresponding two versions for MG-MLMC++. Here, we determined the number k of steps of the block power iteration and the number d_ℓ of vectors to be used there by a parameter scan on each level. This scan is reported in Fig. 2. We find that $k = 2$ is a better choice than $k = 1$, and that increasing k further does not result in significant further gains. Also, $d_\ell \approx 50$ appears as a good choice on all level differences.

The plot in Fig. 1 shows that for all masses considered, the best MLMC method now outperforms deflated Hutchinson (with an optimal number of deflated vectors and without counting the work for computing those). It also shows that with optimal numbers of vectors in the block power iteration, the "++"-enhancement improves MLMC by a factor of 1.5 to 3, with a stronger improvement for the smaller values of m, i.e. the more ill-conditioned matrices. The influence of the strategy to determine the target variance ("uniform" or "optimized") is, on the other hand, not very significant.

As a supplementary information, Tab. 2 reports the number of stochastic samples that were carried out on the different level differences. These numbers directly illustrate the variance reductions achieved in the different approaches. Each stochastic sample involves the solution of two linear systems (with matrices D_ℓ and $D_{\ell+1}$). These are done via multigrid and are thus quite efficient. This is why the numbers of stochastic samples do not reflect the total arithmetic cost of the methods, in which, in particular, performing the projections has a high cost when the deflating subspace becomes larger. Interestingly, there is no visible dependence on the mass parameter for the MLMC approaches as was already observed in [5].

5. Conclusion

We have developed MG-MLMC++, a new trace estimator for the inverse which combines multigrid multilevel Monte Carlo with the recent Hutch++ approach. We have shown that the

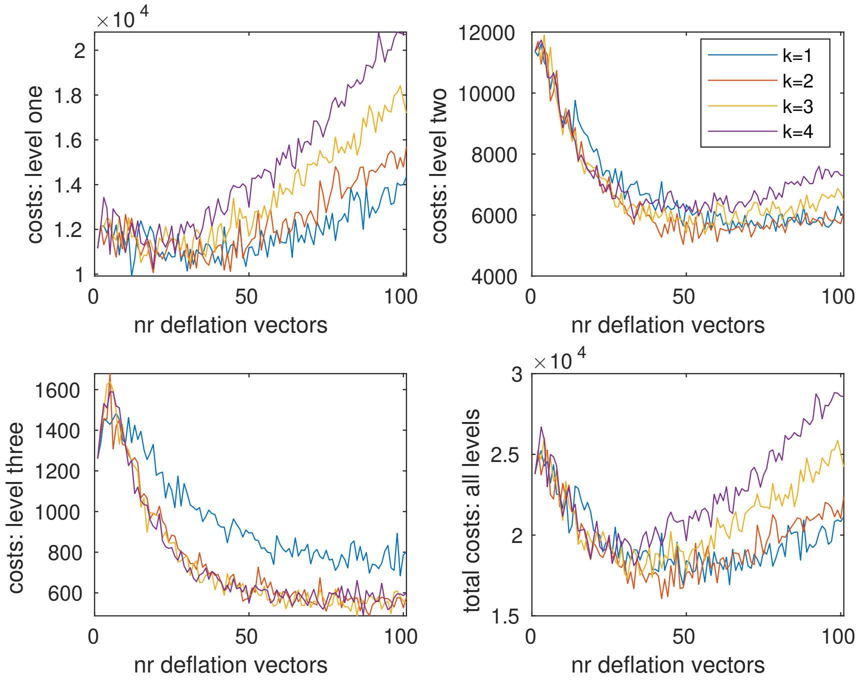

Figure 2: Work at each level difference as a function of the number of vectors in the block power iteration and total work when taking the same number on all levels. .

method type	samples nr. per mass					
	m1	m2	m3	m4	m5	level
deflated Hutchinson	529	1004	2318	7431	13845	
MG-MLMC, optimized target variances	325	321	315	313	306	$\ell = 1$
	854	873	837	833	791	$\ell = 2$
	4208	4218	4414	4287	4171	$\ell = 3$
MG-MLMC++, optimized target variances	181	158	143	108	177	$\ell = 1$
	221	221	200	148	111	$\ell = 2$
	278	272	243	162	173	$\ell = 3$

Table 2: Number of stochastic samples for different masses at each level ℓ for deflated Hutchinson, MG-MLMC and MG-MLMC++, both with optimized target variances

method outperforms other ones in trace computations for the Schwinger model. How to easily obtain a good choice for the number of vectors to use in the block power iteration is a subject of future research as is the application of our approach to the 4-dimensional (Wilson-) Dirac operator.

References

[1] J.C. Sexton and D. Weingarten, *Systematic expansion for full QCD based on the valence approximation*, *arXiv:hep-lat* (1994) [9411029].

[2] M.F. Hutchinson, *A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines*, *Comm. Statist. Simulation Comput.* **18** (1989) 1059.

[3] F. Roosta Khorasani and U. Ascher, *Improved bounds on sample size for implicit matrix trace estimators*, *Foundations of Computational Mathematics* **15** (2015) 1187.

[4] R.A. Meyer, C. Musco, C. Musco and D.P. Woodruff, *Hutch++: Optimal stochastic trace estimation*, *Proceedings of the SIAM Symposium on Simplicity in Algorithms (SOSA)* **2021** (2021) 142.

[5] A. Frommer, M.N. Khalil and G. Ramirez-Hidalgo, *A multilevel approach to variance reduction in the stochastic estimation of the trace of a matrix*, *SIAM J. on Sci. Comput.* **44** (2022) A2536.

[6] A.S. Gambhir, A. Stathopoulos and K. Orginos, *Deflation as a method of variance reduction for estimating the trace of a matrix inverse*, *SIAM J. on Sci. Comput.* **39** (2017) A532.

[7] S. Collins, G. Bali, A. Frommer, K. Kahl, I. Kanamori, B. Müller et al., *(Approximate) Low-Mode Averaging with a new Multigrid Eigensolver*, *PoS* **LATTICE 2015** (2016) 350.

[8] E. Romero, A. Stathopoulos and K. Orginos, *Multigrid deflation for lattice QCD*, *J. Comput. Phys.* **409** (2020) 109356.

[9] D. Persson, A. Cortinovis and D. Kressner, *Improved variants of the Hutch++ algorithm for trace estimation*, *SIAM J. Matrix Anal. Appl.* **43** (2022) 1162.

[10] M.B. Giles, *Multilevel Monte Carlo path simulation*, *Oper. Res.* **56** (2008) 607.

[11] M.B. Giles, *Multilevel Monte Carlo methods.*, *Acta Numer.* **24** (2015) 259.

[12] J. Brannick, R.C. Brower, M.A. Clark, J.C. Osborn and C. Rebbi, *Adaptive multigrid algorithm for lattice QCD*, *Phys. Rev. Lett.* **100:041601** (2007) .

[13] R. Babich, J. Brannick, R.C. Brower, M.A. Clark, T.A. Manteuffel, S.F. McCormick et al., *Adaptive multigrid algorithm for the lattice Wilson-Dirac operator*, *Phys. Rev. Lett. 105:201602* (2010) .

[14] A. Frommer, K. Kahl, S. Krieg, B. Leder and M. Rottmann, *Adaptive aggregation-based domain decomposition multigrid for the lattice Wilson–Dirac operator*, *SIAM J. Sci. Comp.* **36** (2014) A1581.

[15] C. Alexandrou, S. Bacchio, J. Finkenrath, A. Frommer, K. Kahl and M. Rottmann, *Adaptive aggregation-based domain decomposition multigrid for twisted mass fermions*, *Phys. Rev. D* **94** (2016) 114509.

[16] J. Schwinger, *Gauge invariance and mass II*, *Phys. Rev.* **128** (1962) 2425–2429.