Hierarchically Deflated Conjugate Gradient

Peter Boyle, University of Edinburgh

July 29, 2013

Eigenvector Deflation

Krylov solvers convergence controlled by the condition number

$$\kappa \sim \frac{\lambda_{max}}{\lambda_{min}}$$

- Lattice chiral fermions possess an exact index theorem
- Index theorem $\Rightarrow \exists$ near zero modes separated from zero only by quark mass
- Recent algorithmic progress eliminates low mode subspace from Krylov inversion

EigCG:

- Determine $N_{vec} \sim O(V)$ eigenvectors ϕ_i up to some physical λ
- $48^3 \Rightarrow 600$ vectors, $64^3 \Rightarrow 1500$ vectors
- Significant setup cost & storage cost $\propto V^2$
- Eliminates N_{vec} dimensional subspace $S = \mathrm{sp}\{\phi_i\}$ from problem

$$M = \begin{pmatrix} M_{\bar{s}\bar{s}} & \epsilon \\ \epsilon^\dagger & M_{ss} \end{pmatrix} \quad ; \quad M_{ss}^{-1} = \frac{1}{\lambda_i} |i\rangle\langle i|$$

Where $\epsilon = M_{\bar{s}s}$ is proportional to the error in the eigenvectors
Guess $\phi = \mathrm{diag}\{0\} \oplus \mathrm{diag}\{\frac{1}{\lambda_i}\}\eta$

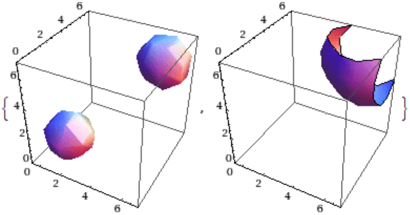
Why can we do better

Luscher's observation: local coherence of low modes

low virtuality solutions of gauge covariant Dirac equation locally similar

Consider N well separated instantons

- N-zero modes look like admixtures of single instanton eigenmodes
- Divide *one* mode into chunks centred on each each instanton
- All N-zero modes described by the span of these chunks



Luscher's inexact deflation

Avoid critical slowing down in Krylov solution of

$$M\psi = \eta$$

- Accelerate sparse matrix inversion by treating a vector subspace $S = \mathrm{span}\{\phi_\mathrm{k}\}$ exactly
- If the lowest lying eigenmodes are well contained in S the "rest" of the problem avoids critical slowing down

Setup:

- Must generate subspace vectors ϕ_k that are "rich" in low modes
- Subdividing these vectors into blocks b

$$\phi_k^b(x) = \begin{cases} \phi_k(x) & ; \quad x \in b \\ 0 & ; \quad x \notin b \end{cases}$$

yields a much larger subspace
$48^3 \times 96$ lattice with 4^4 blocks $\Rightarrow$ $12^3 \times 24$ coarse grid $\Rightarrow$ $O(10^4)$ bigger deflation space.

> Similar idea previously used in αSA adaptive multigrid (Brezina et al 2004)
>
> - *covariant derivative* $\leftrightarrow$ algebraically smooth.
> - blocks $\leftrightarrow$ aggregates.
>
> $\alpha SA \longrightarrow$ US multigrid collaboration & Wuppertal
> Attempt using $D^\dagger D$ for DWF arXiv:1205.2933 (Cohen, Brower, Clark, Osborn)

Inexact deflation framework

Introduce subspace projectors

$$P_S = \sum_{k,b} |\phi_k^b\rangle\langle\phi_k^b| \qquad ; \qquad P_{\bar{S}} = 1 - P_S \tag{1}$$

Compute M_{ss} as

$$M = \begin{pmatrix} M_{\bar{S}\bar{S}} & M_{S\bar{S}} \\ M_{\bar{S}S} & M_{SS} \end{pmatrix} = \begin{pmatrix} P_{\bar{S}} M P_{\bar{S}} & P_S M P_{\bar{S}} \\ P_{\bar{S}} M P_S & P_S M P_S \end{pmatrix}$$

- Can represent matrix M exactly on this subspace by computing its matrix elements, known as the *little Dirac operator* [1]

$$A_{jk}^{ab} = \langle\phi_j^a|M|\phi_k^b\rangle$$

$$(M_{SS}) = A_{ij}^{ab}|\phi_i^a\rangle\langle\phi_j^b|$$

and the subspace inverse can be solved by Krylov methods and is:

$$Q = \begin{pmatrix} 0 & 0 \\ 0 & M_{SS}^{-1} \end{pmatrix}$$

$$M_{SS}^{-1} = (A^{-1})_{ij}^{ab}|\phi_i^a\rangle\langle\phi_j^b|$$

A inherits a sparse structure from M - well separated blocks do *not* connect through M

[1] Coarse grid matrix in MG

Subspace Schur decomposition

We can Schur decompose any matrix

$$M = UDL = \begin{bmatrix} M_{\bar{s}\bar{s}} & M_{\bar{s}s} \\ M_{s\bar{s}} & M_{ss} \end{bmatrix}$$

$$= \begin{bmatrix} 1 & M_{\bar{s}s} M_{ss}^{-1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & M_{ss} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ M_{ss}^{-1} M_{s\bar{s}} & 1 \end{bmatrix}$$

Note that

$$P_L M = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix}$$

yields the Schur complement $S = M_{\bar{s}\bar{s}} - M_{\bar{s}s} M_{ss}^{-1} M_{s\bar{s}}$
L and U related to Luscher's projectors P_L and P_R[2]

$$P_L = P_{\bar{S}} U^{-1} = \begin{pmatrix} 1 & -M_{\bar{S}S} M_{SS}^{-1} \\ 0 & 0 \end{pmatrix}$$

$$P_R = L^{-1} P_{\bar{S}} = \begin{pmatrix} 1 & 0 \\ -M_{SS}^{-1} M_{S\bar{S}} & 0 \end{pmatrix}$$

Also, $QM = 1 - P_R$

[2] Galerkin oblique projectors in MG

Luscher's algorithm

Multiply $M\psi = \eta$ by $1 - P_L$ and P_L yielding $(1 - P_R)\psi$ and $P_R\psi$:

$$(1 - P_R)\psi = M_{ss}^{-1}\eta_s$$
$$(P_L M)\chi = P_L\eta$$
$$\psi = P_R\chi + M_{ss}^{-1}\eta_s$$

- Each step of an outer Krylov solver involves an *inner* Krylov solution of the little Dirac op
  - This enters the matrix $P_L M$ being inverted and errors propagate into solution
  - Luscher tightens the precision during convergence; uses history forgetting *flexible* GCR
- Suppress little Dirac operator with Schwarz alternating procedure (SAP) preconditioner

$$(P_L M)M_{SAP}\phi = P_L\eta \quad ; \quad \psi = M_{SAP}\phi$$

- Non-hermitian system possible as evalues of D_W live in right half of complex plane:
- Little Dirac operator for D_W is *nearest neighbour*
  - Red black preconditioning of Little dirac op possible
  - Schwarz alternating procedure possible as D_W does not connect red to red.

Generalisation to 5d Chiral fermions

Krylov solution of Hermitian system necessary (CG-NR, MCR-NR)
Aim to speed up the red-black preconditioned system as this starts better conditioned

$$\mathcal{H} = \left(M_{oo} - M_{oe}M_{ee}^{-1}M_{eo}\right)^\dagger \left(M_{oo} - M_{oe}M_{ee}^{-1}M_{eo}\right) = M_{\rm prec}^\dagger M_{\rm prec}$$

Matrix being deflated is is next-to-next-to-next-to-nearest-neighbour!

Tasks!

- Must find *further suppression of little Dirac operator* overhead as LDop more costly
- Must find a replacement for the Schwarz preconditioner
- Must find appropriate solver: $(P_L M)M_{SAP}$ nonhermitian matrix so unsuitable for CG
- Must ensure system is tolerant to ill convergence of inner Krylov solver(s).

Little Dirac Operator

4 hop operator is painful as it connects 3280 neighbours!

- Limit the stencil of the Little Dirac operator by requiring block $\geq 4^4$
- Mobius fermions M_{ee}^{-1} is non-local in s-direction $\Rightarrow$ blocks stretch full s-direction
- Sparse in 4d with next-to-next-to-next-to-nearest coupling
- Matrix *still* connects to 80 neighbours

$$\begin{gathered}(\pm\hat{x}),\ (\pm\hat{y}),\ (\pm\hat{z}),\ (\pm\hat{t})\\ (\pm\hat{x}\pm\hat{y}),\ (\pm\hat{x}\pm\hat{z}),\ (\pm\hat{x}\pm\hat{t}),\ (\pm\hat{y}\pm\hat{z}),\ (\pm\hat{y}\pm\hat{t}),\ (\pm\hat{z}\pm\hat{t})\\ (\pm\hat{x}\pm\hat{y}\pm\hat{z}),\ (\pm\hat{x}\pm\hat{y}\pm\hat{t}),\ (\pm\hat{x}\pm\hat{z}\pm\hat{t}),\ (\pm\hat{y}\pm\hat{z}\pm\hat{t})\\ (\pm\hat{x}\pm\hat{y}\pm\hat{z}\pm\hat{t})\end{gathered}$$

- Underlying cost at least ten times more than non-Hermitian system
- Reducing to 4d has saved L_s factor *but may require more vectors* to describe 5th dimension

Little Dirac Operator Implementation

- 10×10 matrix-vector complex multiply reasonably high cache reuse
  - Using IBM xlc vector intrinsics gives adequate performance
  - Single precision accelerated gives around 50 Gflop/s per node in L2 cache
  - (re)Discovered bug in L2 cache around 4 months after Argonne
- 80 small messages of order 1-5 KB
  - Programme BG/Q DMA engines directly to eliminate MPI overhead
  - Asynchronous send overhead under 10 microseconds with precomputed DMA descriptors.
  - 50x faster than MPI calls.

Infra-red shift preconditioner

Since we are deflating the low modes, seek approximate inverse preconditioner for Hermitian system that is accurate for high modes.

- Naive left-right preconditioner:

$$L^\dagger (P_L \mathcal{H}) L \phi = L^\dagger P_L \eta$$

$$L \sim (\mathcal{H})^{-\frac{1}{2}}$$

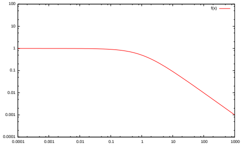
- Better to use preconditioned CG (p 278 Saad) with Hermitian preconditioner M_P

$$M_P = L^\dagger L \sim (\mathcal{H})^{-1}$$

- Use fixed number of shifted CG iterations as preconditioner (IR shifted preconditioner)
    - Krylov solver seeks optimal polynomial under some norm

$$M_{IRS} = \frac{1}{\mathcal{H} + \lambda}$$

- λ is an gauge covariant infra-red regulator that shifts the lowest modes
    - Plays similar role to the domain size in SAP
- Keeps the Krylov solver working hard on the high mode region
    - Does not have locality benefit of SAP[3]



[3]Comms in BG/Q tolerate this, but Additive Schwarz is worth investigating for future machines (suggested by Mike Clark)

Robustness

Two inner Krylov solvers

- Little Dirac operator inversion $Q \equiv M_{SS}^{-1}$
- IR shifted preconditioner inversion $M_{IRS} = \frac{1}{\mathcal{H}+\lambda}$

Curious robustness effects: during solution to 10^{-8} on a 16^3 lattice

M_{SS}^{-1} residual	M_{IRS} residual	Iteration count
10^{-11}	10^{-8}	36
10^{-8}	10^{-8}	Non converge [4]
10^{-11}	10^{-8}	36
10^{-11}	10^{-4}	36
10^{-11}	10^{-2}	36

Although *flexible* CG (Notay 1999) exists better to understand *why* the CG is tolerant to variability in M but not Q

[4] smallest residual is 10^{-7} then diverges. Here Luscher introduced flexible algorithms

Robustness

Consider preconditioned CG with $A = P_L \mathcal{H} = \begin{pmatrix} 1 & -M_{\bar{S}S} M_{SS}^{-1} \\ 0 & 0 \end{pmatrix} \mathcal{H}$

1. $r_0 = b - Ax_0$
2. $z_0 = M_{IRS} r_0$; $p_0 = z_0$
3. for iteration k
4. $\alpha_k = (r_k, z_k)/(p_k, Ap_k)$
5. $x_{k+1} = x_k + \alpha_k p_k$
6. $r_{k+1} = r_k - \alpha_k A p_k$
7. $\mathbf{z_{k+1} = M_{IRS} r_{k+1}}$
8. $\beta_\mathbf{k} = (\mathbf{r_{k+1}}, \mathbf{z_{k+1}})/(\mathbf{r_k}, \mathbf{z_k})$
9. $\mathbf{p_{k+1}} = \mathbf{z_{k+1}} + \beta_\mathbf{k} \mathbf{p_k}$
10. end for

- Noise in the preconditioner M_{IRS} *only* enters the search direction α_k is based on matrix elements of $P_L \mathcal{H}$.
- Better to use the Little Dirac operator inverse as a preconditioner ...and not separate the solution into subspace and complement ...already discussed as advantage of MG in Boston papers

Combining preconditioners

- Have little Dirac operator Q and M_{IRS} representing approximate inverse
  - Q on subspace containing low mode
  - M_{IRS} on high mode space
  - splitting is necessarily *inexact*
- Options for combining these as a preconditioner
  - Additive

$$M_{IRS} + Q$$

  - Consider alternating error reduction steps

$$\begin{aligned}
x_{i+1} &= x_i + M_{IRS}[b - \mathcal{H}x_i] \\
x_{i+2} &= x_{i+1} + Q[b - \mathcal{H}x_{i+1}] \\
&= x_i + M_{IRS}[b - \mathcal{H}x_i] + Q[b - \mathcal{H}[x_i + M_{IRS}[b - \mathcal{H}x_i]]] \\
&= x_i + [(1 - Q\mathcal{H})M_{IRS} + Q](b - \mathcal{H}x_i) \\
&= x_i + [P_R M_{IRS} + Q](b - \mathcal{H}x_i)
\end{aligned}$$

  - Infer family of preconditioner

Sequence	Preconditioner	Name
additive	$M_{IRS} + Q$	AD
M_{IRS}, Q	$P_R M_{IRS} + Q$	A-DEF2
Q, M_{IRS}	$M_{IRS} P_L + Q$	A-DEF1
Q, M_{IRS}, Q	$P_R M_{IRS} P_L + Q$	Balancing Neumann Neumann (BNN)
Q, M_{IRS}, Q	$M_{IRS} P_L + P_R M_{IRS} + Q - M_{IRS} P_L \mathcal{H} M_{IRS}$	MG Hermitian V(1,1) cycle

Generalised framework for inexact deflation solvers

Extend framework of Tang, Nabben, Vuik, Erlangga (2009) to three levels

Take $Q = \begin{pmatrix} 0 & 0 \\ 0 & M_{SS}^{-1} \end{pmatrix}$ and $M_{IRS} = (\mathcal{H} + \lambda)^{-1}$

Method	$V_{\rm start}$	M_1	M_2	M_3	$V_{\rm end}$
PREC	x	M_{IRS}	$\mathbb{1}$	$\mathbb{1}$	x_{k+1}
AD	x	$M_{IRS} + Q$	$\mathbb{1}$	$\mathbb{1}$	x_{k+1}
DEF1	x	M_{IRS}	$\mathbb{1}$	P_L	$Qb + P_R x_{k+1}$
DEF2	$Qb + P_R x$	M_{IRS}	P_R	$\mathbb{1}$	x_{k+1}
A-DEF1	x	$M_{IRS} P_L + Q$	P_R	$\mathbb{1}$	x_{k+1}
A-DEF2	$Qb + P_R x$	$P_R M_{IRS} + Q$	$\mathbb{1}$	$\mathbb{1}$	x_{k+1}
BNN	x	$P_R M_{IRS} P_L + Q$	$\mathbb{1}$	$\mathbb{1}$	x_{k+1}

- DEF1/DEF2/ADEF1/ADEF2/BNN are equivalent
  - *identical iterates* with V_{start} up to Q, M_{IRS} error
  - Luscher's algorithm corresponds to DEF1
- Move little Dirac operator into the preconditioner with formally identical convergence to inexact deflation!
- A-DEF2 is most tolerant of preconditioner variability

Algorithm

1. x arbitrary
2. $x_0 = V_{\rm start}$
3. $r_0 = b - \mathcal{H} x_0$
4. $y_0 = M_1 r_0$; $p_0 = M_2 y_0$
5. for iteration k
6. $w_k = M_3 \mathcal{H} p_k$
7. $\alpha_k = (r_k, y_k)/(p_k, w_k)$
8. $x_{k+1} = x_k + \alpha_k p_k$
9. $r_{k+1} = r_k - \alpha_k w_k$
10. $\mathbf{y_k} = \mathbf{M_1 r_k}$
11. $\boldsymbol{\beta_k} = (\mathbf{r_{k+1}}, \mathbf{y_{k+1}})/(\mathbf{r_k}, \mathbf{y_k})$
12. $\mathbf{p_{k+1}} = \mathbf{M_2 y_{k+1}} + \boldsymbol{\beta_k} \mathbf{p_k}$
13. end for
14. $x = V_{\rm end}$

Remain in deflated Krylov picture but make it Heirarchical by deflating the deflation matrix Q

Why does CG work here?

- Hermiticity of M_1 clear for BNN but not A-DEF1/2
  Theorem: for $V_{\rm start} = Qb + P_R x$ A-DEF2 is identical to BNN.

- We have from $Q\mathcal{H} = (1 - P_R)$

$$Qr_0 = Q[\mathcal{H} V_{\rm start} - b] = (1 - P_R)[Q_b + P_R x] - Qb = P_R Q_b = 0$$

$$Q\mathcal{H} p_0 = (1 - P_R)[P_R M P_L + Q] r_0 = 0$$

- get induction steps:

$$Qr_{j+1} = Qr_j - \alpha_j Q\mathcal{H} p_j = 0$$

$$Q\mathcal{H} p_{j+1} = (1 - P_R)[P_R M P_L + Q] r_j + \beta_j Q\mathcal{H} p_j = 0$$

- Can also show $P_L r_0 = 0$ and $P_L \mathcal{H} p_0 = \mathcal{H} p_0$ so that

$$P_L \mathcal{H} p_{j+1} = \mathcal{H} P_R [P_R M P_L + Q] r_j + \beta_j p_j = \mathcal{H} p_{j+1}$$

  and

$$P_L r_{j+1} = P_L r_j - \alpha_j P_L \mathcal{H} p_j = r_j - \alpha_j \mathcal{H} p_j = r_{j+1}$$

BNN then retains $P_L r_j = r_j$ in exact arithmetic
$\Rightarrow$ BNN iteration ($P_R M P_L r_j$) and A-DEF2 iteration ($P_R M r_j$) equivalent up to convergence error

- DEF1(Luscher), DEF2, A-DEF1, A-DEF2, BNN are ALL equivalent up to convergence

BUT they *differ* hugely in sensitivity to convergence error in Q

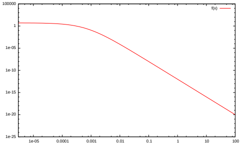
Hermiticity and improved subspace generation

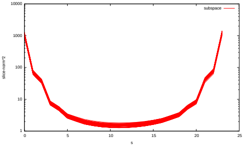
- Hermitian system gains the properties

$$P_L^\dagger = P_R \qquad (P_L M)^\dagger = P_L M$$

- Since we use $\mathcal{H} = M_{\rm prec}^\dagger M_{\rm prec}$ we have a Hermitian Positive (semi) Definite matrix. Generate subspace with rational multi-shift solver applied to Gaussian noise

$$R(\eta^{\rm Gaussian}) \propto \frac{1}{(\mathcal{H}+\lambda)(\mathcal{H}+\lambda+\epsilon)(\mathcal{H}+\lambda+2\epsilon)(\mathcal{H}+\lambda+3\epsilon)}$$





- Classic low pass filtering problem – use rational filter
  - Gain $1/x^4$ suppression in single pass *without* inverse iteration
  - $\epsilon \sim 10^{-3}$ adds IR safety to the inversion $O(1000)$ iterations per subspace vector
  - NB Also possible for $\gamma_5 D_W$
  - Subspace support only on walls possible. Potential to regain factor of L_s?

Subspace tricks

- Improved subspace generation

  1. Solve rational in single precision to loose tolerance (10^{-4}) *and with reduced* L_s
  2. Compute HDCG operator
  3. Refine subspace: loose (10^{-3}) shifted HDCG inverse fills into bulk
  4. Recompute HDCG operator

  - 2-4x reduction in subspace generation over double precision rational
  - Not all subspace vectors need be extensive in 5th dim
  - Removes L_s factor from the expensive low mode subspace
  - Gives same CG count as high precision rational filter

- Subspace reuse: recompute little Dop matrix elements with *no change to subspace*

  - Twisted boundary conditions
  - Moderate change in mass – not obvious for 5d chiral fermions but works!

Algorithm	Volume	mass	Twist	Solve time
CGNE	32^4	0.01	$\frac{\pi}{L}(0,0,0)$	30s
HDCG	32^4	0.01	$\frac{\pi}{L}(0,0,0)$	6.9s
HDCG	32^4	0.01	$\frac{\pi}{L}(0.2,0,0)$	6.9s
HDCG	32^4	0.01	$\frac{\pi}{L}(0.5,0.5,0.0)$	9.2s
HDCG	32^4	0.01	$\frac{\pi}{L}(0.5,0.5,0.5)$	9.8s
HDCG	32^4	0.1	$\frac{\pi}{L}(0,0,0)$	3.6s
HDCG	32^4	0.01	$\frac{\pi}{L}(0,0,0)$	6.9s
HDCG	32^4	0.005	$\frac{\pi}{L}(0,0,0)$	7.4s
HDCG	32^4	0.001	$\frac{\pi}{L}(0,0,0)$	7.8s

Hierarchical deflation

Deflate the deflation matrix !

- Block these vectors ϕ_k^b (e.g. $4^4 \times L_s$) and compute little Dirac operator
  Need only apply $N_{\mathrm{stencil}} = 80$ matrix multiplies per vector to compute little Dirac operator with a Fourier trick. Single precision suffices
  Can *detect* stencil from matrix application and generate optimal code for 1,2,4 hop operators
- Compute second level of deflation heirarchy using inverse iteration on Gaussian noise.
- Diagonalise this basis to make multiplication cheap
- Massively reduce convergence precision:
  - Use A-DEF2 to move the little Dirac operator into preconditioner
  - Can relax convergence precision to 10^{-2}
  - Eight order of magnitude gain, saving of O(10) in overhead
- Deflate the deflation matrix (Heirarchical).
  - Computing 128 low modes is cheap and saves another factor of 10.
  - Reduces O(2000) little Dirac operator iterations to O(20).

From 48^3 at physical quark masses

Precision	Heirarchical deflation	iterations
10^{-7}	N	4478
10^{-7}	Y	250
10^{-2}	Y	63

100 x reduction in little dirac operator overhead!

HDCG solver

Use outer CG A-DEF2 solver, DeflCG little dirac solver

Method	$V_{\rm start}$	M_1	M_2	M_3	$V_{\rm end}$
A-DEF2	$Qb + P_R x$	$P_R M_{IRS} + Q$	$\mathbb{1}$	$\mathbb{1}$	x_{k+1}
DeflCG	$Qb + P_R x$	$\mathbb{1}$	$\mathbb{1}$	$(1 - P_R)$	x_{k+1}

Where

$$Q = \begin{pmatrix} 0 & 0 \\ 0 & M_{SS}^{-1} \end{pmatrix} \quad ; \quad P_R = \begin{pmatrix} 1 & 0 \\ -M_{SS}^{-1} M_{S\bar{S}} & 0 \end{pmatrix}$$

$$\mathcal{H} = M_{\rm pc}^\dagger M_{\rm pc} \quad ; \quad M_{IRS} = [\mathcal{H} + \lambda_{\rm pc}]^{-1}$$

- Shifted matrix inversion M is solved with CG and fixed iteration count (N=8)
- M_{SS} inversion is itself deflated
- All operations in CG are perfromed in single precision except $\mathcal{H}$ multiply, x_j and r_j updates.

Tunable parameters

Fine N_{vec}	40
Fine blocksize	$4^4 \times L_s$
Fine subspace filter	4th order rational $\lambda_S \sim 10^{-3}$
Fine subspace tolerance	10^{-6}
Coarse N_{vec}	128
Coarse blocksize	full volume
Coarse subspace filter	Inverse iteration (3)
Coarse subspace tolerance	10^{-7}
$\left[M_{\rm pc}^\dagger M_{\rm pc} + \lambda_{\rm pc}\right]^{-1}$	8 iterations (tol $\sim 10^{-1}$)
$\lambda_{\rm pc}$	1.0
M_{SS}^{-1}	tol 5×10^{-2}

1. x arbitrary
2. $x_0 = V_{\rm start}$
3. $r_0 = b - \mathcal{H} x_0$
4. $y_0 = M_1 r_0$; $p_0 = M_2 y_0$
5. for iteration k
6. $w_k = M_3 \mathcal{H} p_k$
7. $\alpha_k = (r_k, y_k)/(p_k, w_k)$
8. $x_{k+1} = x_k + \alpha_k p_k$
9. $r_{k+1} = r_k - \alpha_k w_k$
10. $\mathbf{y_k} = \mathbf{M_1 r_k}$
11. $\beta_\mathbf{k} = (\mathbf{r_{k+1}}, \mathbf{y_{k+1}})/(\mathbf{r_k}, \mathbf{y_k})$
12. $\mathbf{p_{k+1}} = \mathbf{M_2 y_{k+1}} + \beta_\mathbf{k} \mathbf{p_k}$
13. end for
14. $x = V_{\rm end}$

Performance

Both fine and coarse dirac operators give around 30-50Gflop/s per node on BG/Q.
On $48^3 \times 96 \times 24$, $M_\pi = 140\text{MeV}$, $a^{-1} = 1.73$ GeV on 1024 node rack

Algorithm	Tolerance	Cost	Matmuls
CGNE (double)	10^{-8}	1270s	16000
CGNE (mixed)			23000
EigCG (mixed)	10^{-8}	320s	11710
EigCG (mixed)	10^{-4}	55s	1400
EigCG (setup)		10h	
EigCG (vectors)		600 vectors	
HDCG (mixed)	10^{-8}	117s	2060
HDCG (mixed)	10^{-4}	9s	200
HDCG (setup)		40min	
HDCG (vectors)		44 vectors	

10^{-4} precision is used for the All-mode-averaging analysis

- Anticipate at least 5x speedup for RBC-UKQCD valence analysis over EigCG

Conclusions

Comparison	Gain
Exact Solve vs CGNE	11x
Exact Solve vs EigCG	2.7x
Inexact Solve vs EigCG	5x
Setup vs EigCG	10x
Footprint vs EigCG	15-40x

- Developed inexact deflation method to accelerating preconditioned normal equations
  Larger stencil required substantial algorithmic improvements
- Improved robustness with no formal change to inexact deflation:
  - Little Dirac operator in preconditioner: more robust solver (10x)
  - Heirarchical multi-level deflation (10x)
- Hermitian algorithm features
  - IR shifted preconditioner to replace SAP
  - Preconditioned CG tolerant to loose convergence of inner Krylov solver(s).
  - No flexible algorithm was required
- Approach based in Krylov space methods, bears similarities to multigrid
- Step towards alleviating L_s scaling of 5d Chiral Fermions (subspace generation)

To do:

- Investigate numerically efficiency of additive Schwarz preconditioning [5]
  Domain decomposed preconditioner should give 2x Gflop/s improvement
  Greater locality ⇒ candidate exascale algorithm

[5] suggested by Mike Clark