Block CG Solvers Documentation
github.com/lkeegan/blockCG

Liam Keegan
liam@keegan.ch

June 27, 2018

1 Introduction

Formulations of block CG solvers, which simultaneously invert a hermitian matrix acting on n_{RHS} vectors. Notation conventions used:

- A is the hermitian positive definite $L \times L$ matrix to be inverted
- x, r, p, etc. are L–component vectors
- X, R, P, etc. are $L \times n_{\text{RHS}}$ block–vectors
- α, β, ζ, etc. are $n_{\text{RHS}} \times n_{\text{RHS}}$ matrices

2 Solvers

2.1 CG

To establish notation, the standard CG solver for $Ax = b$ is given in Alg. 1.

Algorithm 1 CG: Solve $Ax = b$

$x, p, r, t \in \mathcal{C}^L$; $\alpha, \beta \in \mathcal{R}$
$x_0 = t_0 = 0$, $r_0 = p_0 = b$, $\alpha_0 = b^\dagger b$
for $k = 1, 2, \ldots$ until $|Ax_k - b| \,/\, |b| < \epsilon$ **do**
    $t_k \leftarrow Ap_k$
    $\beta_k \leftarrow (p_k^\dagger t_k)^{-1}(r_{k-1}^\dagger r_{k-1})$
    $r_k \leftarrow r_{k-1} - t_k\beta_k$
    $\alpha_k \leftarrow (r_{k-1}^\dagger r_{k-1})^{-1}(r_k^\dagger r_k)$
    $x_k \leftarrow x_{k-1} + p_k\beta_k$
    $p_k \leftarrow r_k + p_k\alpha_k$
end for

2.2 SCG

Since the Krylov basis is shift–invariant, the shifted equation $(A + \sigma)x = b$, where σ is a real and positive scalar, can be solved without additional applications of the matrix A, which allows the construction of a shifted CG solver [1], given in Alg. 2.

Algorithm 2 SCG: Solve $(A+\sigma_j)x^{\sigma_j} = b$, where $j = 0, \ldots, N_{\text{shifts}} - 1$

$x^{\sigma_j}, p^{\sigma_j}, r, t \in \mathcal{C}^L$; $\alpha^{\sigma_j}, \beta^{\sigma_j}, \zeta^{\sigma_j} \in \mathcal{R}$?
$x_0^{\sigma_j} = t_0 = 0$, $r_0 = p_0^{\sigma_j} = b$, $\alpha_0 = b^\dagger b$
for $k = 1, 2, \ldots$ until $|(A+\sigma_0)x_k^{\sigma_0} - b| \,/\, |b| < \epsilon$ **do**
  $t_k \leftarrow (A+\sigma_0)p_k^{\sigma_0}$
  $\beta_k^{\sigma_0} \leftarrow ((p_k^{\sigma_0})^\dagger t_k)^{-1}(r_{k-1}^\dagger r_{k-1})$
  $r_k \leftarrow r_{k-1} - t_k \beta_k^{\sigma_0}$
  $\alpha_k^{\sigma_0} \leftarrow (r_{k-1}^\dagger r_{k-1})^{-1}(r_k^\dagger r_k)$
  $x_k^{\sigma_0} \leftarrow x_{k-1}^{\sigma_0} + p_k^{\sigma_0}\beta_k^{\sigma_0}$
  $p_k^{\sigma_0} \leftarrow r_k + p_k^{\sigma_0}\alpha_k^{\sigma_0}$
  for $j = 1, \ldots, N_{\text{shifts}} - 1$ **do**
    $\beta_k^{\sigma_0} \leftarrow ((p_k^{\sigma_0})^\dagger t_k)^{-1}(r_{k-1}^\dagger r_{k-1})$
    $\beta_k^{\sigma_0} \leftarrow ((p_k^{\sigma_0})^\dagger t_k)^{-1}(r_{k-1}^\dagger r_{k-1})$
    $\beta_k^{\sigma_0} \leftarrow ((p_k^{\sigma_0})^\dagger t_k)^{-1}(r_{k-1}^\dagger r_{k-1})$
    $x_k^{\sigma_j} \leftarrow x_{k-1}^{\sigma_j} + p_k^{\sigma_j}\beta_k^{\sigma_j}$
    $p_k^{\sigma_j} \leftarrow r_k \zeta^{\sigma_j} + p_k^{\sigma_j}\alpha_k^{\sigma_j}$
  end for
end for

3 Block Solvers

3.1 BCG

The BlockCG (BCG) [2] algorithm is a straightforward extension of the CG algorithm, described in Alg. 3.

Algorithm 3 BCG: Solve $AX = B$

$X, P, R, T \in \mathcal{C}^{L \times n_{\text{RHS}}}$; $\alpha, \beta \in \mathcal{C}^{n_{\text{RHS}} \times n_{\text{RHS}}}$
$X_0 = T_0 = P_0 = 0$, $R_0 = B$, $\alpha_0 = B^\dagger B$
for $k = 1, 2, \ldots$ until $|AX_k - B| \,/\, |B| < \epsilon$ **do**
  $P_k \leftarrow R_{k-1} + P_{k-1}\alpha_{k-1}$
  $T_k \leftarrow AP_k$
  $\beta_k \leftarrow (P_k^\dagger T_k)^{-1}(R_{k-1}^\dagger R_{k-1})$
  $R_k \leftarrow R_{k-1} - T_k\beta_k$
  $X_k \leftarrow X_{k-1} + P_k\beta_k$
  $\alpha_k \leftarrow (R_{k-1}^\dagger R_{k-1})^{-1}(R_k^\dagger R_k)$
end for

3.2 BCGrQ

Algorithm 4 BCGrQ: Solve $AX = B$

$X, P, R, T \in \mathcal{C}^{L\times n_{\mathrm{RHS}}}$; $\alpha, \beta, \delta \in \mathcal{C}^{n_{\mathrm{RHS}}\times n_{\mathrm{RHS}}}$
$X_0 = T_0 = P_0 = 0$, $\{R_0, \delta_0\} = \mathrm{qr}(B), \alpha_0 = 1$
for $k = 1, 2, \ldots$ until $\max_i \sqrt{\delta_k(i,i)/\delta_0(i,i)} < \epsilon$ **do**
    $P_k \leftarrow R_{k-1} + P_{k-1}\alpha_{k-1}^\dagger$
    $T_k \leftarrow AP_k$
    $\beta_k \leftarrow (P_k^\dagger T_k)^{-1}$
    $\{R_k, \alpha_k\} \leftarrow \mathrm{qr}(R_{k-1} - T_k\beta_k)$
    $X_k \leftarrow X_{k-1} + P_k\beta_k\delta_{k-1}$
    $\delta_k \leftarrow \alpha_k\delta_{k-1}$
end for

References

[1] B. Jegerlehner, *Krylov space solvers for shifted linear systems*, hep-lat/9612014.

[2] D. P. O'Leary, *The block conjugate gradient algorithm and related methods*, *Linear Algebra and its Applications* **29** (1980) 293 – 322. Special Volume Dedicated to Alson S. Householder.