Avoiding Communication in Two-Sided Krylov Subspace Methods

Nick Knight and Erin Carson

BERKELEY PARLAB

Research supported by Microsoft (Award #024263) and Intel (Award #024894) funding and by matching funding by U.C. Discovery (Award #DIG07-10227). Additional support comes from Par Lab affiliates National Instruments, NEC, Nokia, NVIDIA, Samsung, and Sun Microsystems.

Motivation

- Krylov Subspace Methods are commonly used for solving linear system
- Standard implementations are communication-bound due to required SpMV and orthogonalization in every iteration
- Solution: rearrange algorithms to perform s iterations at a time without communicating (s-step methods)
- SpMV in each iteration is replaced with a call to the Matrix Powers Kernel, which performs s SpMVs while reading the matrix only once
  - Used to generate s basis vectors for the Krylov Subspace

$$\mathcal{K}_k(A, v) = span\{v, Av, \ldots, A^{k-1}v\}$$

Previous Work

- Communication-Avoiding Kernels
  - Matrix Powers Kernel (one matrix, one input vector)
  - Tall-Skinny QR
- One-sided Krylov Subspace Methods
  - Conjugate Gradient [Hoemmen, 2010]
  - GMRES [MHDY09]
  - Lanczos [Hoemmen, 2010]
- Two-sided Krylov Subspace Methods
  - BiCG
    - Problem: BiCG is unstable in practice

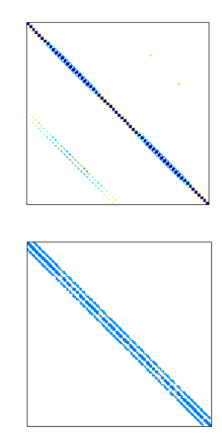
New CA-KSMs

- Preliminary Work
  - 2-Term recurrence version of BiCG
  - Conjugate Gradient Squared (CGS)
- Main result: CA-BiCGStab:

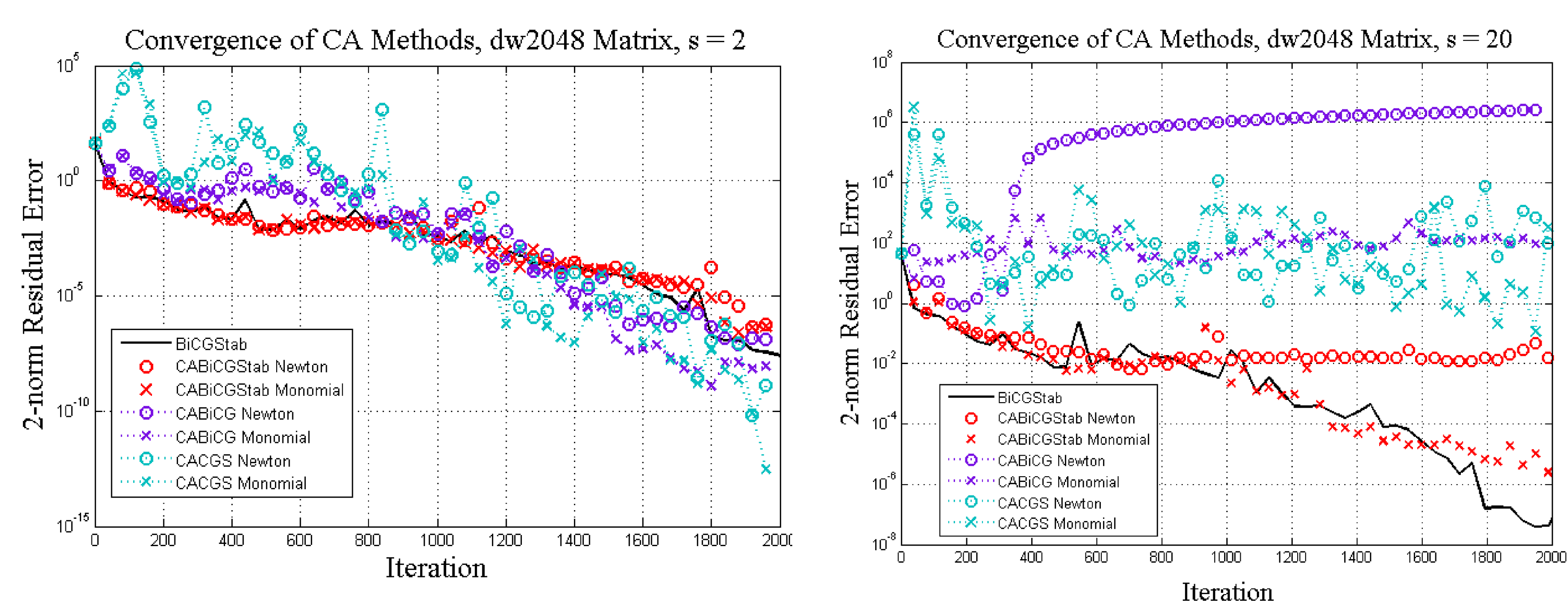
Computation and Storage Costs	
Matrix powers	3 x A
Storage	3Ns + 5N + O(s²)
Dense Work	O(Ns² + s³)

- Original method formulated by van der Vorst. 1992)
  - Variation of CGS, remedies irregular convergence patterns
- Polynomial defined recursively at each step acts as a smoother
  - smoothes against previous residual
- CA Formulation
  - 2-term recurrence, similar to CGS and BiCG

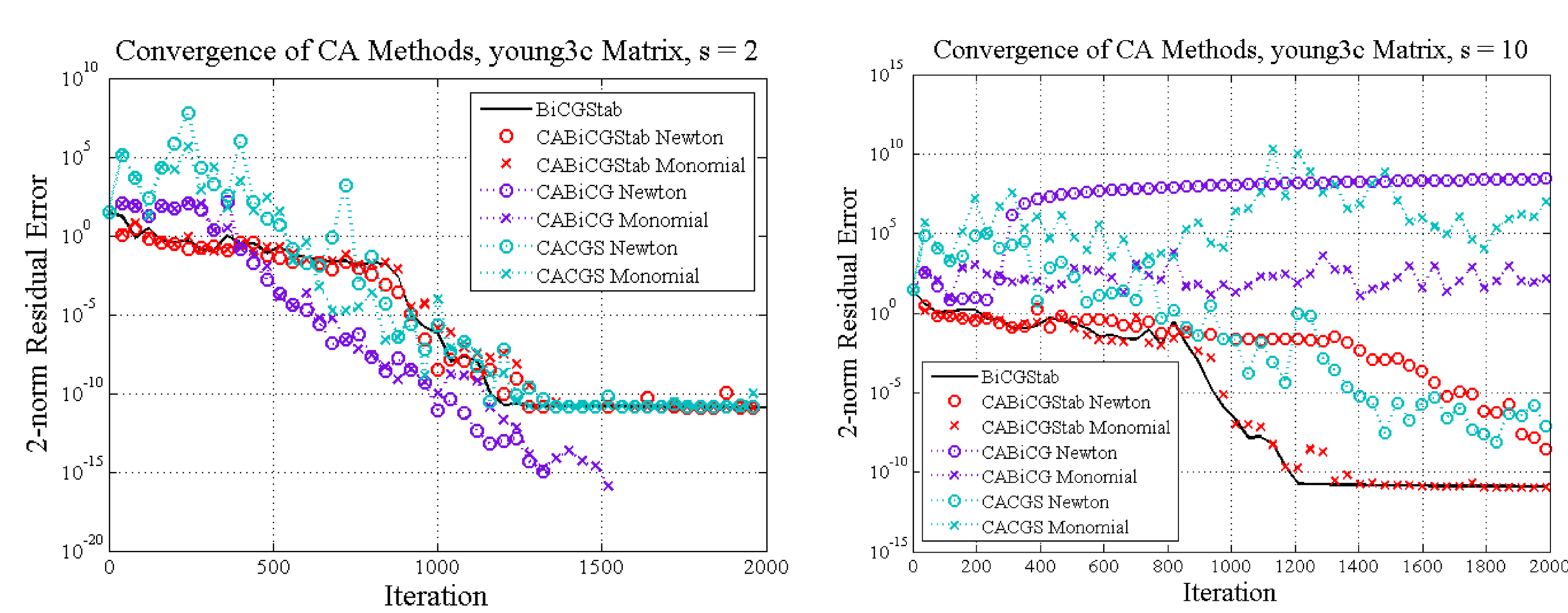
Convergence Results



Name	n	NNZ	Pattern Symmetry	Value Symmetry	Condition Number	Application
dw2048	2048	10114	No	No	5.3015e3	Electromagnetics Problem (H. Dong, 1993)
young3c	841	3988	No	No	1.1532e4	Acoustics Problem (D.Young, 1984)



- Figures: Convergence Results for dw2048 matrix for s=2 and s=20
- Shown for 2-term recurrence versions of BiCG, CGS, and BiCGStab. Black line indicates standard (Matlab) implementation.
- We see here that the BiCGStab method is indeed more stable for higher s values, especially using the monomial basis
  - Why? Too much roundoff error in Newton basis?

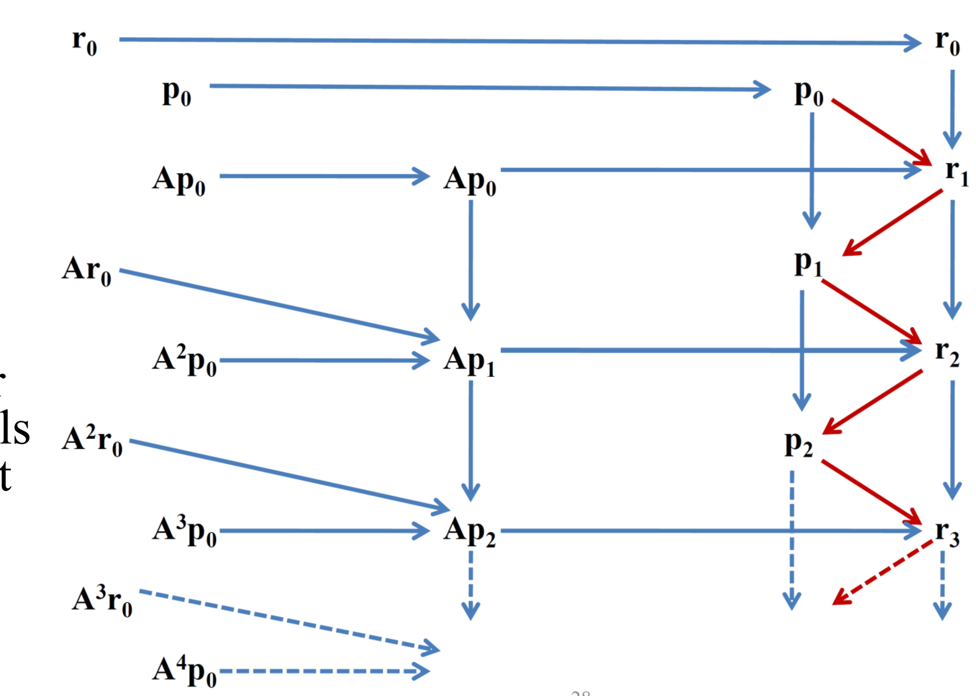


- Figures: Convergence Results for young3c matrix for s=2 and s=10
- Shown for 2-term recurrence versions of BiCG, CGS, and BiCGStab. Black line indicates standard (Matlab) implementation.
- We see here that the BiCGStab method is again more stable (monomial basis follows standard iterates up until convergence)
- The first plot indicates that although BiCGStab is, in general, more stable, the optimal method to use is problem dependent (BiCG does the best)

New Matrix Powers Kernels

- Our communication avoiding formulations of CGS and BiCGStab require more than one matrix powers evaluation
  - Same matrix, but different right hand sides (various state vectors stored for iterates)
  - IDEA: We can compute multiple RHSs at the same time!
    - SIMD parallelism
    - Still only requires reading A once

Figure (right): Dependencies in 2-Term Recurrence version of BiCG. Demonstrates need for two matrix powers calls (same matrix, different RHSs).



Preconditioning

- Naïve preconditioning approach: s SpMVs, s solves
- Problem: requires a different approach/implementation for each type of preconditioner!
- Current algorithms
  - Polynomial preconditioners (Saad, Toledo)
    - M is polynomial in A – incorporated into Newton basis
    - CA-Left-preconditioning (Hoemmen, 2010)
      - Preconditioners and matrices with low rank-off diagonal blocks, same sparsity structure
      - 1 + o(1) more messages than single SpMV, 1 preconditioner solve

Future Work

- Finish BiCGStab(l)
  - When s > 8, normal equations become ill-conditioned
  - Use rank-deficient least squares?
- Extension to other classes of preconditioners
- Chebyshev basis for matrix powers
  - Based on spectrum of A. Could provide more stability
- Parallel C Implementations for performance testing
- Tests with restarting and extended precision, varying s values
  - could help with stability and convergence