Solver Performance using Multigrid vs CG et al

Mark Nudelman

Anish Yennapusa

Shen Gao

A Linear Algebra Problem

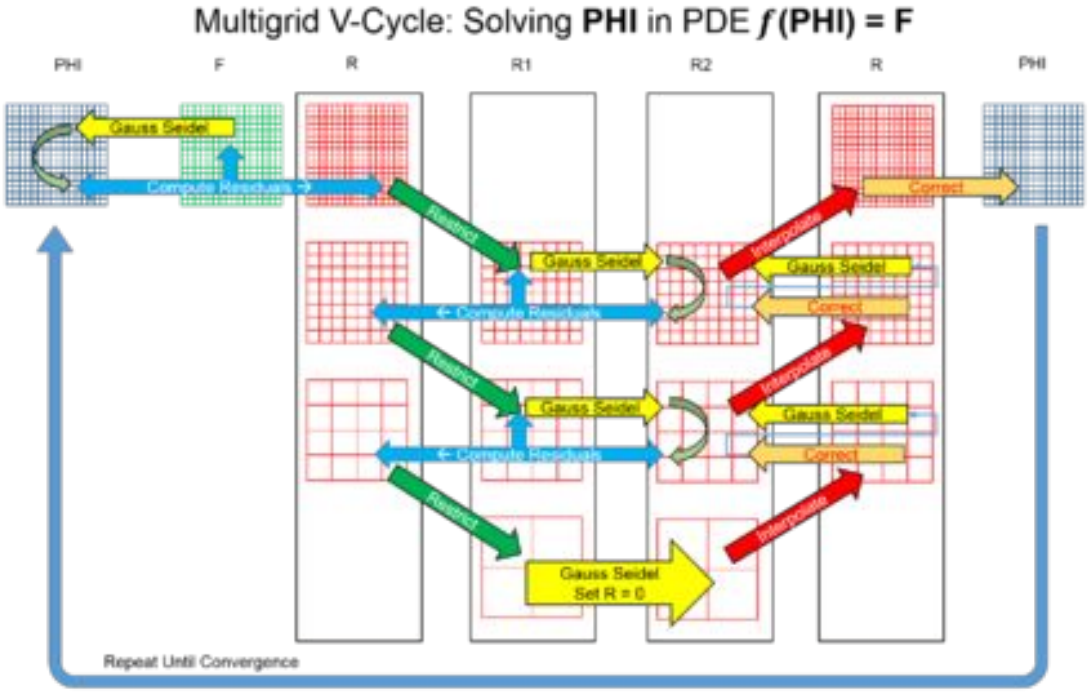
- Ax = b
- Sparse matrix
- Very small eigenvalues (~0.0001)
  - Jacobi, Gauss-Seidel take too long
  - Conjugate gradient also takes too long

- Multigrid is a solution

What is a Multigrid?

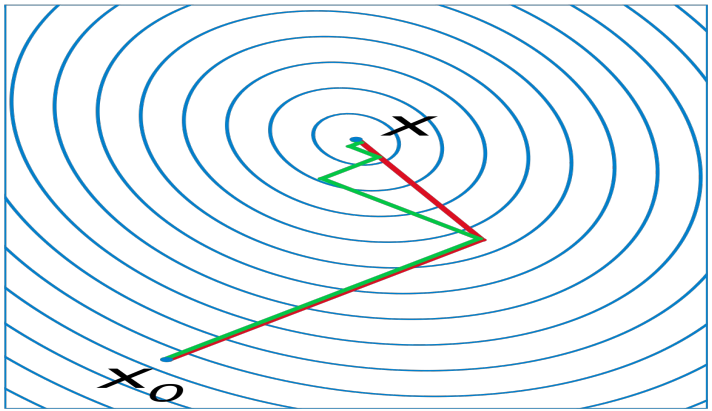
- It is a recursive divide and conquer method used in many fast algorithms for computer science tasks and numerical methods for high performance computing.

- The main idea of multigrid is to accelerate the convergence of a basic iterative method (known as relaxation) by a *global* correction of the fine grid solution approximation from time to time, accomplished by solving a coarse problem.


Multigrid V-Cycle: Solving PHI in PDE ƒ(PHI) = F
PHI
F
R
R1
R2
R
PHI
Gauss Seidel
Compute Residuals
Restrict
Gauss Seidel
Compute Residuals
Restrict
Gauss Seidel
Compute Residuals
Restrict
Gauss Seidel
Set R = 0
Interpolate
Gauss Seidel
Correct
Interpolate
Gauss Seidel
Correct
Interpolate
Correct
Repeat Until Convergence

What is Conjugate Gradient?

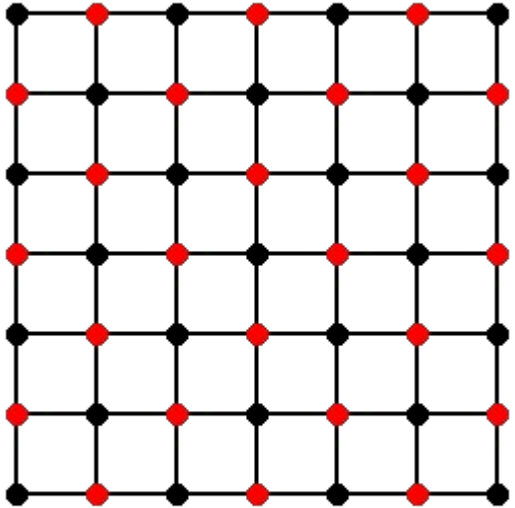
- The conjugate gradient method is an algorithm for the numerical solution of particular systems of linear equations, namely those whose matrix is positive-definite.
- The conjugate gradient method is often implemented as an iterative algorithm, applicable to sparse systems that are too large to be handled by a direct implementation.



What is Red Black

- An algorithm that reduces data dependence to only even and odd neighbors.
- This method is just as fast as Gauss-Seidel performing 2x faster than Jacobi
- However, like Jacobi it allows for parallel computation
    - Over all even sites
    - Over all odd sites

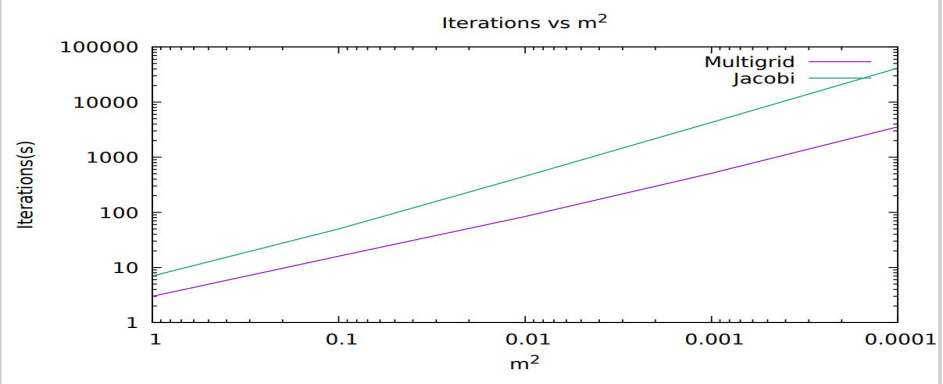
Red–Black Ordering of Grid Points

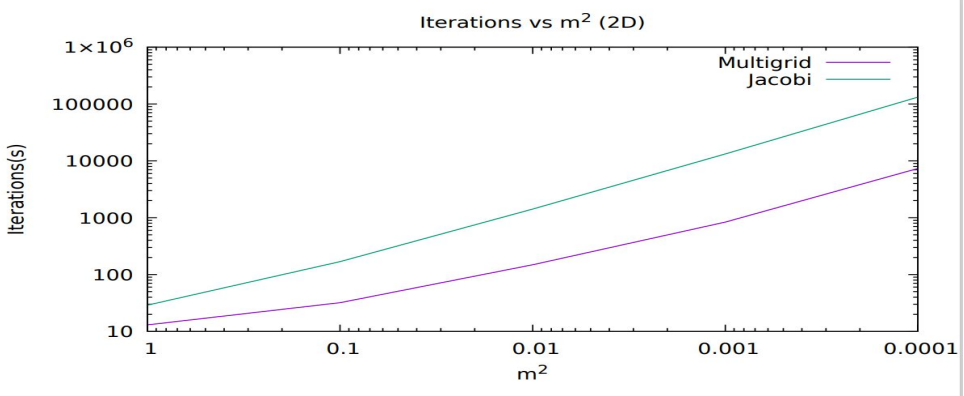


Black points have only Red neighbors

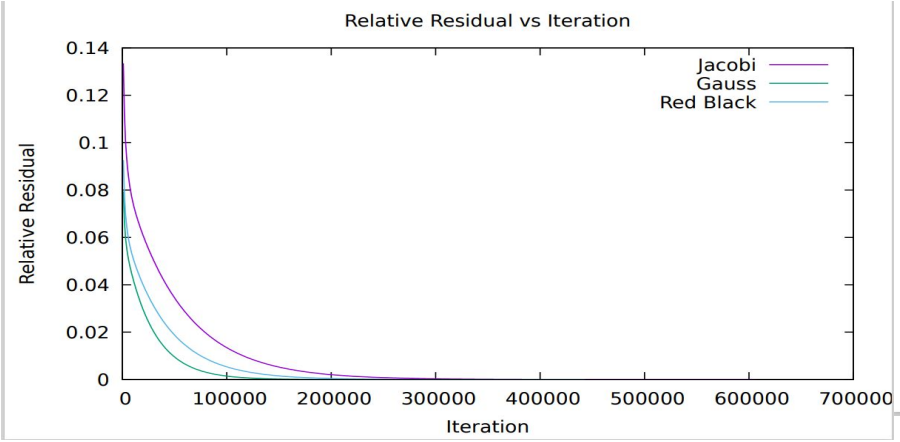
Red points have only Black neighbors

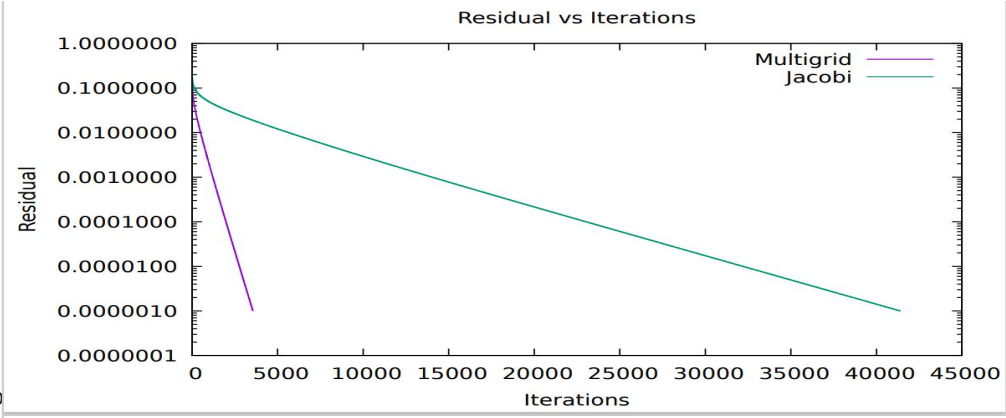
Results


Iterations vs m²
Multigrid
Jacobi
Iterations(s)
100000
10000
1000
100
10
1
1
0.1
0.01
0.001
0.0001
m²

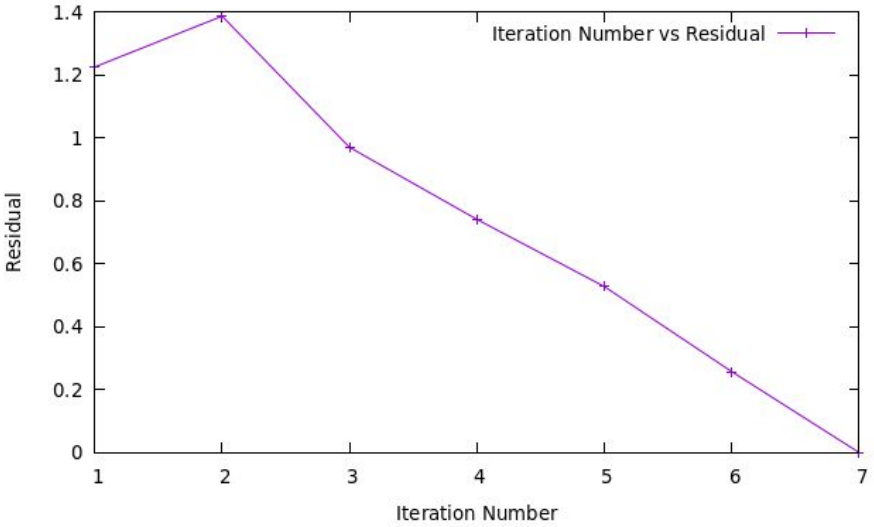

Iterations vs m² (2D)
Multigrid
Jacobi
Iterations(s)
1×10⁶
100000
10000
1000
100
10
1
0.1
0.01
0.001
0.0001
m²

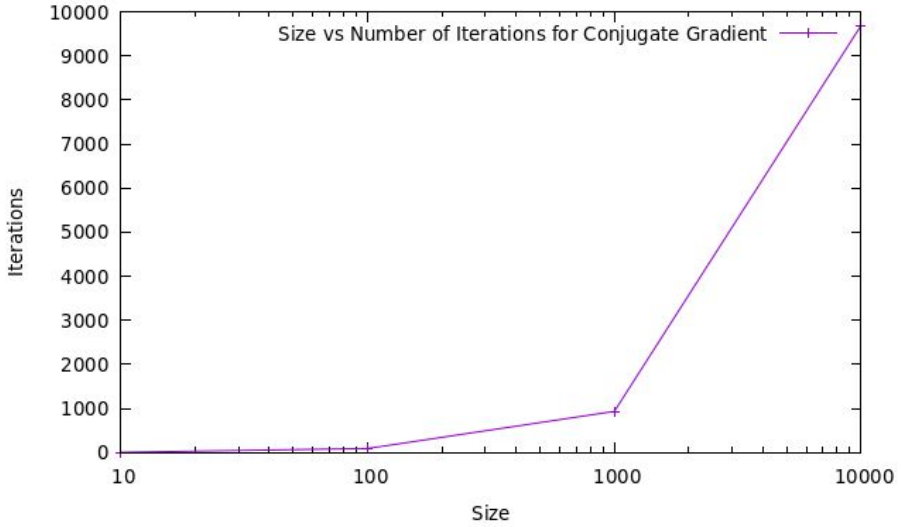
Results


Relative Residual vs Iteration
Relative Residual
Iteration
Jacobi
Gauss
Red Black
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
0
100000
200000
300000
400000
500000
600000
700000


Residual vs Iterations
Residual
Iterations
Multigrid
Jacobi
1.0000000
0.1000000
0.0100000
0.0010000
0.0001000
0.0000100
0.0000010
0.0000001
0
5000
10000
15000
20000
25000
30000
35000
40000
45000

Results


Iteration Number vs Residual
Residual
Iteration Number
1.4
1.2
1
0.8
0.6
0.4
0.2
0
1
2
3
4
5
6
7


Size vs Number of Iterations for Conjugate Gradient
Iterations
Size
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
0
10
100
1000
10000

Conclusion

- Multigrid solver converges in relatively fewer iterations as the size of the problem increases.
- It also performs less matrix vector computations as size increases.
- Multigrid solver converges in relatively less time as the size of the problem increases.

Future Work

- Compare performance on a larger scale.
- Parallelize with MPI instead of OpenAcc.
- Compare with preconditioned conjugate gradient method.
- Experiment with number of processors

Thank You