SOR as a Preconditioner

M. A. DeLong and J. M. Ortega

November 7, 1994

1 Introduction

It is well-known (see, e.g. [2] and [4]) that the use of red/black or multicolor orderings to parallelize SSOR or ILU preconditioning may seriously degrade the rate of convergence of the conjugate gradient method, as compared with the natural ordering.

The SOR iteration itself, however, does not suffer this degradation. Indeed, if the coefficient matrix is consistently ordered with property A, the asymptotic rates of convergence of the natural and red/black orderings are identical (Young[9]); moreover, in practice one quite often sees faster convergence in the red/black ordering than in the natural ordering. This suggests the possible use of SOR as a parallel preconditioner. It cannot be a preconditioner for the conjugate gradient method on symmetric positive definite systems since the corresponding preconditioned matrix is not symmetric. But this restriction does not apply to nonsymmetric systems and conjugate-gradient type methods such as GMRES ([6]). In fact, Saad[5] showed promising results using several steps of Gauss-Seidel iteration as a preconditioner in conjunction with the GMRES iteration, and the present paper complements his results. Shadid and Tuminaro[7] have also reported experiments using Gauss-Seidel as a preconditioner. However, they used only one Gauss-Seidel iteration and, as our experiments show, this is usually not competitive.

2 Experimental Results on a Model Problem

We first present some experimental results on a two-dimensional convection-diffusion equation (see, e.g., [3])

$$-(u_{xx} + u_{yy}) + \sigma(u_x + u_y) = f(x, y) \tag{2.1}$$

on the unit square with Dirichlet boundary conditions. For simplicity we take σ to be constant. The equation (2.1) is discretized by standard five-point finite differences using centered differences for the first derivative terms. The right hand side f of (2.1) is chosen so that the exact solution of the discrete system is known.

If there are $n = N^2$ interior points, this results in an $n \times n$ linear system

$$A\mathbf{u} = \mathbf{b} \tag{2.2}$$

in which a typical equation is of the form

$$4u_i + \beta u_{i+1} + \gamma u_{i-1} + \beta u_{i+N} + \gamma u_{i-N} = h^2 f_i \tag{2.3}$$

where

$$\beta = -1 + \frac{\sigma h}{2}, \quad \gamma = -1 - \frac{\sigma h}{2} \tag{2.4}$$

Thus, in the natural ordering of the grid points used in (2.3), A has the same five-diagonal structure as the coefficient matrix of the corresponding Poisson problem. In the red/black ordering the coefficient matrix has the form

$$\begin{bmatrix} D_R & E \\ F & D_B \end{bmatrix}$$

where D_R and D_B are diagonal. The Gauss-Seidel iteration is then

$$\mathbf{u}_R^{k+1} = D_R^{-1}(\mathbf{b}_R - E\mathbf{u}_B^k), \quad \mathbf{u}_B^{k+1} = D_B^{-1}(\mathbf{b}_B - F\mathbf{u}_R^{k+1})$$

which can be implemented very effectively in parallel.

We first give some results for a serial code on an IBM RS/6000, Model 250, for the algorithm GS(k)−GMRES(m) in which we use m GMRES vectors before restarting and k Gauss-Seidel iterations as the preconditioner. GMRES was implemented with unmodified Gram-Schmidt orthogonalization.

	51^2 Equations				151^2 Equations			
	Natural		Red/Black		Natural		Red/Black	
Algorithm	iters	time	iters	time	iters	time	iters	time
GMRES(5)	508	10.9	508	11.2	4720	893	4720	909
ILU(0)-GMRES(5)	75	2.3	142	4.4	359	96	1108	290
GS(1)-GMRES(5)	236	6.9	139	4.3	2069	539	1180	318
GS(5)-GMRES(5)	46	2.9	44	2.8	296	167	242	139
GMRES(10)	281	7.4	281	7.4	2455	555	2455	562
ILU(0)-GMRES(10)	62	2.1	108	3.9	276	84	662	204
GS(1)-GMRES(10)	180	6.0	110	3.8	919	267	659	199
GS(5)-GMRES(10)	44	2.8	44	3.0	157	89	140	83
GS(10)-GMRES(20)	20	2.1	22	2.5	70	67	73	72
GS(20)-GMRES(20)	14	2.4	15	2.7	55	89	58	97

Table 1: *Natural vs. Red/Black Orderings, Equation (2.1)*

Table 1 shows CPU times in seconds and iterations for various values of k and m for two problem sizes: $51^2 = 2601$ equations, and $151^2 = 22,801$ equations. The convergence criterion is $||\mathbf{r}||_2 < 10^{-6}||A||_\infty$ and the initial approximation is zero. (We also used an initial approximation that was a random vector with elements between 0 and 1, uniformly distributed. The results are similar and not shown.)

For comparison, we also show results for no preconditioning and for ILU(0) as the preconditioner. As expected, the red/black ordering has little effect on the GMRES iteration but leads to a large degradation in the rate of convergence when ILU(0) is the preconditioner. With Gauss-Seidel as the preconditioner, there is no degradation between the natural and red/black orderings; in fact, the red/black ordering is usually better, sometimes markedly so. Finally, Table 1 shows that improvements can be expected for larger numbers of Gauss-Seidel iterations and GMRES vectors. This is consistent with the observations of Saad[5] for the Gauss-Seidel iterations. For these particular problems, the lowest times are obtained for GS(10)-GMRES(20).

The results of Table 1 are for $\sigma = 5$. From (2.4) the off-diagonal terms of the matrix are about $-1 \pm .05$ for the 51^2 problem and $-1 \pm .017$ for the 151^2 problem. Thus, the matrices may be viewed as small perturbations of the Poisson matrix. We will return in Section 4 to the effect of larger σ.

3 The Effect of ω

Saad[5] reports results only for the Gauss-Seidel iteration and in [1] there is a footnote to the effect that SOR is never used as a preconditioner because no polynomial acceleration is possible if the optimal ω is used. However, Figure 1 shows the effect of using SOR as a preconditioner for various values of ω. For this particular problem, roughly a factor of two improvement is obtained using the optimal ω rather than $\omega = 1$. Similar results are obtained for the 151^2 problem: 140 iterations for $\omega = 1$ and 61 iterations for the optimal $\omega \doteq 1.8$.

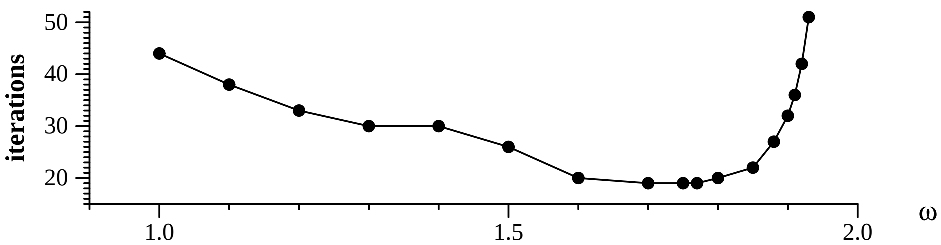

Figure 1: *Iterations for 2601 Equations, SOR(5)-GMRES(10), Red/Black Ordering*

The curve in Figure 1 is similar to, but not nearly as pronounced as, the corresponding convergence curve for SOR as a stand-alone iteration. In Figure 2, we compare the time to convergence curve for SOR with that for SOR(10)-GMRES(20) for a region near the optimal ω's. (SOR and SOR-GMRES each may have its own optimum.) This figure, which is for the 151^2 problem, illustrates two things. First, near the optimum ω's the times for SOR and SOR-GMRES are almost identical, presumably corroborating the previous statement that polynomial acceleration of SOR with the optimal ω is not effective. But for values of ω smaller than optimal, SOR-GMRES is considerably better than SOR. Moreover, the SOR-GMRES

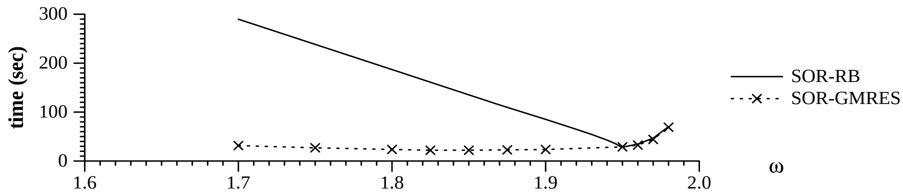

Figure 2: *Time for SOR and SOR(10)-GMRES(20). 22801 Equations*

convergence curve is relatively flat for $\omega < \omega_{opt}$, indicating that a rather crude approximation to ω_{opt} may be almost as good as the optimal ω.

4 The Effect of σ

σ	ω = 1	1.3	1.4	1.5	1.6	1.7	1.75	1.8	1.9
-41	92	75	71	DNC	DNC	DNC	DNC	DNC	DNC
-31	90	77	71	60	DNC	DNC	DNC	DNC	DNC
-21	82	79	59	61	56	DNC	DNC	DNC	DNC
-11	79	58	46	49	50	42	40	83	DNC
-5	57	39	39	39	37	35	32	32	88
0	42	40	35	31	29	30	30	29	31
5	44	31	26	26	22	20	19	20	28
11	42	22	20	18	15	12	12	14	31
21	35	15	11	9	7	9	12	18	DNC
31	26	10	8	5	7	15	20	DNC	DNC
41	19	7	4	6	10	149	DNC	DNC	DNC

Table 2: *Iterations for SOR(5)-GMRES(10), 2601 Equations, Natural Ordering*

In the previous sections, we have used only $\sigma = 5$, giving an almost symmetric system. Tables 2 and 3 show iteration counts as a function of both ω and σ in the natural and red/black orderings. DNC indicates no convergence due to stagnations. In the natural ordering (Table 2), as σ increases, the iterations usually decrease whereas as σ becomes more negative, the iterations tend to increase. This is explained by the fact that as σ increases, the system is becoming more strongly lower triangular, which is beneficial for Gauss-Seidel. But as σ becomes more negative, the system is becoming more upper triangular. This is, however, not the case for the red/black ordering (Table 3) and the iterations tend to increase as $|\sigma|$ increases.

For small values of $|\sigma|$ the iterations decrease as a function of ω until the optimum ω is passed. As $|\sigma|$ increases, the optimum ω decreases. Indeed, the whole

σ	ω 1	1.3	1.4	1.5	1.6	1.7	1.75	1.8	1.9
-41	68	30	19	40	DNC	DNC	DNC	DNC	DNC
-31	59	31	30	18	52	DNC	DNC	DNC	DNC
-21	61	38	30	21	18	40	104	DNC	DNC
-11	54	36	30	22	19	18	19	26	DNC
-5	44	30	30	26	20	19	19	20	32
0	40	21	19	12	11	12	13	16	28
5	44	30	30	26	20	19	19	20	32
11	54	36	30	22	19	18	19	26	DNC
21	61	38	30	21	18	40	104	DNC	DNC
31	59	31	30	18	52	DNC	DNC	DNC	DNC
41	68	30	19	40	DNC	DNC	DNC	DNC	DNC

Table 3: *Iterations for SOR(5)-GMRES(10), 2601 Equations, Red-Black Ordering*

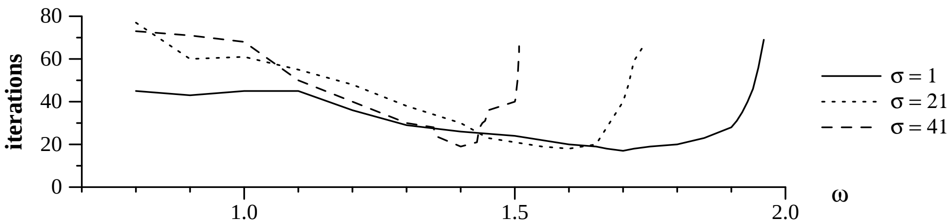

Figure 3: ω *Effects, 2601 Equations, Red/Black Ordering*

convergence curve shifts to the left as illustrated in Figure 3. The typical very rapid increase in the number of iterations as ω moves to the right of the optimum ω occurs for smaller ω as $|\sigma|$ increases and leads to no convergence for sufficiently large ω.

5 Other Test Problems

We next give experimental results on two problems used in [7]:

$$-u_{xx} + u_x + (1 + y^2)(-u_{yy} + u_y) = f(x, y) \tag{5.1}$$

and

$$-u_{xx} - u_{yy} + 100(x^2 u_x + y^2 u_y) - \beta u = f(x, y) \tag{5.2}$$

Both problems are on the unit square with Dirichlet boundary conditions and forcing function f chosen so that the exact solution of the differential equation

(5.1) or the discrete system for (5.2) is known. Both problems are discretized with centered differences and a uniform mesh.

Table 4 gives experimental results for (5.1) for various values of the number of SOR steps, the number of GMRES vectors and the relaxation parameter ω. For the smaller problem, $\omega = 1.7$ is a nominal guess for the optimum ω, and likewise for $\omega = 1.9$ for the larger problem. Again, for all results in this section, the initial approximation is zero, the convergence criterion is $\|r\|_2 < 10^{-6}\|A\|_\infty$, and the machine is the same IBM RS/6000, Model 250.

	51^2 Equations		151^2 Equations	
Algorithm	iters	time	iters	time
GMRES(10)	499	12.18	3347	738.98
GS(1)-GMRES(10)	199	6.28	1162	338.02
SOR(1.7,1)-GMRES(10)	152	4.90	1005	293.65
SOR(1.9,1)-GMRES(10)	137	4.47	872	254.62
GS(5)-GMRES(10)	50	3.25	270	152.38
SOR(1.7,5)-GMRES(10)	24	1.52	96	54.07
SOR(1.9,5)-GMRES(10)	28	1.82	65	36.80
GS(10)-GMRES(10)	36	3.57	137	124.10
SOR(1.7,10)-GMRES(10)	18	1.85	44	40.18
SOR(1.9,10)-GMRES(10)	18	1.83	32	29.52
GS(1)-GMRES(20)	112	4.65	649	242.47
SOR(1.7,1)-GMRES(20)	102	4.32	578	216.58
SOR(1.9,1)-GMRES(20)	169	6.95	586	218.85
GS(10)-GMRES(20)	27	2.98	90	86.37
SOR(1.7,10)-GMRES(20)	15	1.55	40	38.73
SOR(1.9,10)-GMRES(20)	18	1.93	25	24.30
GS(1)-GMRES(60)	74	5.68	260	181.78
SOR(1.7,1)-GMRES(60)	75	5.65	258	181.13
SOR(1.9,1)-GMRES(60)	109	8.62	274	191.42
GS(10)-GMRES(60)	26	3.05	65	83.28
SOR(1.7,10)-GMRES(60)	15	1.58	32	34.00
SOR(1.9,10)-GMRES(60)	18	1.98	24	23.98

Table 4: *GMRES for Equation (5.1) - Red/Black Ordering*

The results in Table 4 show the same general pattern as was seen for the model problem in Section 2: using only one step of the Gauss-Seidel preconditioner is always far worse than using multiple steps. Moreover, with only one step, the introduction of ω produces relatively little benefit, especially for larger numbers of GMRES vectors. However, when at least five SOR iterations are used, there is always a significant benefit from either of the two nominal values of ω. The optimal ω's are not known for the problem of Table 4. However, additional experiments for the 151^2 problem showed that for SOR(1)-GMRES(60), any ω in the interval (1.47, 1.60) produces the lowest iteration count: 252. For SOR(10)-GMRES(60), $\omega = 1.9$ was indeed an optimal ω.

The effect of the number of GMRES vectors is not so clear. For the larger problem, more vectors are beneficial when using only one GS iteration but for multiple SOR iterations, there is little benefit in using more than 20 GMRES vectors. This, of course, may be a consequence of this particular problem.

We cannot give a direct comparison of our results with [7] since they used larger problem sizes than we could. For example, for a 512×512 grid, they reported a time (on 256 processors of an NCUBE2) of 237 seconds for GMRES using 64 vectors and with one GS step as the preconditioner. Our results suggest that if they had used multiple GS/SOR steps and a nominal value of ω, the time might have been decreased by a factor of six or more. On this particular problem, a factor of six improvement would have made the SOR preconditioner comparable to their best least squares polynomial preconditioner, which in turn was far slower than a multigrid preconditioner.

However, on the other three (and more difficult) problems in [7] the multigrid and polynomial preconditioners did not show such superiority and, in fact, failed on some problems. We next consider (5.2), which was Problem 4 in [7]. Table 5 gives results for a nominal value of $\beta = 100$ and for 64 GMRES vectors.

Algorithm	iters	time
GMRES(64)	DNC	DNC
GS(1)-GMRES(64)	2556	1961.7
SOR(1.7,1)-GMRES(64)	1779	1360.9
SOR(1.9,1)-GMRES(64)	DNC	DNC
GS(5)-GMRES(64)	118	117.0
SOR(1.7,5)-GMRES(64)	44	37.8
SOR(1.9,5)-GMRES(64)	100	93.6
GS(10)-GMRES(64)	56	72.3
SOR(1.7,10)-GMRES(64)	25	25.9
SOR(1.9,10)-GMRES(64)	42	48.9
GS(20)-GMRES(64)	39	70.5
SOR(1.7,20)-GMRES(64)	16	26.5
SOR(1.9,20)-GMRES(64)	22	37.1

Table 5: *Equation (5.2)* $\beta = 100$ - *Red/Black Ordering, 22801 Equations*

The results in Table 5 show some of the same patterns as in Table 4, but in a more pronounced way. First, with only one GS or SOR step, the times are more than a factor of 10 higher than with five or more GS/SOR steps. Second, with only one SOR step, the use of ω has relatively little benefit as opposed to when at least five SOR steps are taken, and $\omega = 1.9$ causes no convergence. Third, we see that whereas using 10 GS/SOR steps is rather significantly better than using 5, going to 20 steps produces little benefit. Additional experiments showed that for 5 and 10 SOR steps an optimal ω was indeed 1.7, and for 20 steps an optimal ω slightly larger than 1.7 required 15 iterations. We note that we also ran the problem of Table 5 using 32 GMRES vectors but the results were considerably worse.

6 BiCGSTAB

We next give some experimental results using BiCGSTAB[8] as the base iteration. Table 6 is for the model equation (2.1) and corresponds to Table 1. We see that, as with GMRES and as expected, the natural and red/black orderings are comparable. However, as opposed to GMRES, the Gauss-Seidel iteration has relatively little effect as a preconditioner, possibly because BiCGSTAB is already so effective on this problem relative to GMRES without a preconditioner. Moreover, the use of five GS iterations is superior for both problem sizes, and 20 GS iterations is worse than no preconditioning.

	51^2 Equations				151^2 Equations			
	Natural		Red-Black		Natural		Red-Black	
Algorithm	iters	time	iters	time	iters	time	iters	time
BiCGSTAB	86	3.10	88	3.15	225	64.23	249	73.20
GS(5)-BiCGSTAB	20	2.32	20	2.43	64	58.12	52	50.47
GS(10)-BiCGSTAB	14	2.63	14	2.75	39	60.28	40	64.65
GS(20)-BiCGSTAB	9	3.30	9	3.30	27	75.48	29	84.62

Table 6: *BiCGSTAB for Equation (2.1)*

Tables 7 and 8 give results for equations (5.1) and (5.2) respectively, corresponding to Tables 4 and 5 for GMRES. As in Table 6, we see that Gauss-Seidel has relatively little effect in most cases and using many Gauss-Seidel iterations can be detrimental.

For example, in Table 7, five Gauss-Seidel iterations produce some benefit but ten is worse than no preconditioning. Similarly, in Table 8, five Gauss-Seidel iterations produce a definite benefit whereas 10 and 20 iterations are worse than just one iteration. Both Tables 7 and 8 also show that the use of SOR with a nominal value of ω may produce far better results than just Gauss-Seidel although, as with GMRES, a value of ω that is too large may be detrimental and even lead to no convergence.

It is interesting that, although with no preconditioning BiCGSTAB performs very much better than GMRES, with the SOR preconditioner the best GMRES times are comparable with the best BiCGSTAB times on all problems, and rather considerably better on Equation (5.2).

Algorithm	51^2 Equations iters	time	151^2 Equations iters	time
BiCGSTAB	92	3.22	238	67.45
GS(1)-BiCGSTAB	51	2.68	140	6023
SOR(1.7,1)-BiCGSTAB	49	2.55	169	71.72
SOR(1.9,1)-BiCGSTAB	73	3.65	149	63.33
GS(5)-BiCGSTAB	27	3.12	67	63.65
SOR(1.7,5)-BiCGSTAB	14	1.77	35	34.38
SOR(1.9,5)-BiCGSTAB	17	2.07	30	29.83
GS(10)-BiCGSTAB	20	3.85	50	79.77
SOR(1.7,10)-BiCGSTAB	11	2.23	25	41.22
SOR(1.9,10)-BiCGSTAB	12	2.33	16	27.50

Table 7: *BiCGSTAB for Equation (5.1), Red/Black Ordering*

7 Summary and Conclusions

We have shown by experimental results on some two-dimensional convection-diffusion type equations that the SOR iteration may have promise as a preconditioner for conjugate gradient type iterations such as GMRES. The rate of convergence is not degraded by the red/black ordering, which implies that efficient parallel implementation should be possible. We have shown for GMRES that using multiple GS/SOR steps is always far superior to using just one, and that at least a factor of two improvement over Gauss-Seidel as a preconditioner can be expected on these type problems by use of an approximation to the optimum ω. Moreover, the sensitivity to ω is very weak for $\omega < \omega_{opt}$, leading to the possibility of a suitably good ω even with a crude approximation. However, there is very strong sensitivity to ω if $\omega > \omega_{opt}$, and a slightly too large ω may cause no convergence. Thus, it is critical to always estimate ω on the low side.

For the BiCGSTAB iteration, the situation is not so clear. Multiple Gauss-Seidel steps have relatively little benefit but SOR with a nominal value of ω can reduce the time by a factor of two.

Future work will include parallel implementations, other test problems, and hopefully a better theoretical understanding of why and when SOR may be a good parallel preconditioner.

References

[1] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM,1993.

Algorithm	iters	time
BiCGSTAB	507	146.45
GS(1)-BiCGSTAB	223	94.23
SOR(1.1,1)-BiCGSTAB	249	104.57
SOR(1.7,1)-BiCGSTAB	374	155.72
SOR(1.9,1)-BiCGSTAB	DNC	DNC
GS(5)-BiCGSTAB	90	84.40
SOR(1.7,5)-BiCGSTAB	53	50.90
SOR(1.9,5)-BiCGSTAB	DNC	DNC
GS(10)-BiCGSTAB	66	104.05
SOR(1.7,10)-BiCGSTAB	26	42.85
SOR(1.9,10)-BiCGSTAB	DNC	DNC
GS(20)-BiCGSTAB	46	131.45
SOR(1.7,20)-BiCGSTAB	13	39.73
SOR(1.9,20)-BiCGSTAB	DNC	DNC

Table 8: *BiCGSTAB for Equation (5.2), $\beta = 100$, Red/Black Ordering 22801 Equations*

[2] I. Duff and G. Meurant, The Effect of Ordering on Preconditioned Conjugate Gradients, *BIT*, 29, pp.635-657, 1989.

[3] H. Elman and G. Golub, Iterative Methods for Cyclically Reduced Non-Self-Adjoint Linear Systems, *Mathematics of Computation*, 54(190), pp.671-700, 1990.

[4] J. Ortega, Orderings for Conjugate Gradient Preconditionings, *SIAM J. Optimization*, 1, pp.565-582, 1991.

[5] Y. Saad, Highly Parallel Preconditioners for General Sparse Matrices, in *Recent Advances in Iterative Methods*, (G. Golub, M. Lorkin, and A. Greenbaum, editors), Springer-Verlag, pp. 165-199, 1994.

[6] Y. Saad and M. Schultz, GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems, *SIAM J. Sci. Stat. Comput.,* 7, pp.856-869, 1986.

[7] J. Shadid and R. Tuminaro, A Comparison of Preconditioned Krylov Methods on a Large-Scale MIMD Machine, *SIAM J. Sci. Comp.,* 15, pp.440-459, 1994.

[8] H. van der Vorst, Bi-CGSTAB: a Fast and Smoothly Converging Variant of BI-CG for the Solution of Nonsymmetric Linear Systems, *SIAM J. Sci. Stat. Comput.,* 13, pp.631-644, 1992.

[9] D. Young, *Iterative Solution of Large Linear Systems,* Academic Press, 1971.