The directions received by the minimization task were used to generate an orthonormal basis for the minimization subspace, and the product of the matrix A with these basis vectors (included in the counts above) was performed by the minimization task. Upon restart, the current residual was used as the right hand side.

Algorithm KMS was run in two modes: no communication from $Task_0$ to the other tasks, and communication every 20 iterations in order to have the splitting tasks work with the best solution vector found thus far. The maximal size for the minimization subspace in the KMS and GMRES algorithms was 20. The conjugate gradient algorithm, of course, was run without restarts.

Results of the experiments are given in Figures 1 and 2, in which the norms of the residuals for the various algorithms are plotted as a function of number of matrix-vector multiplications.

There are two curves for each KMS algorithm. The upper ones correspond to no communication from $Task_0$ to the other tasks and indicate that the algorithms stall and fail to converge. This is due to the generation of directions that are effectively random when Gram-Schmidt orthogonalization is applied to a long sequence of Krylov vectors. The same problem limits the

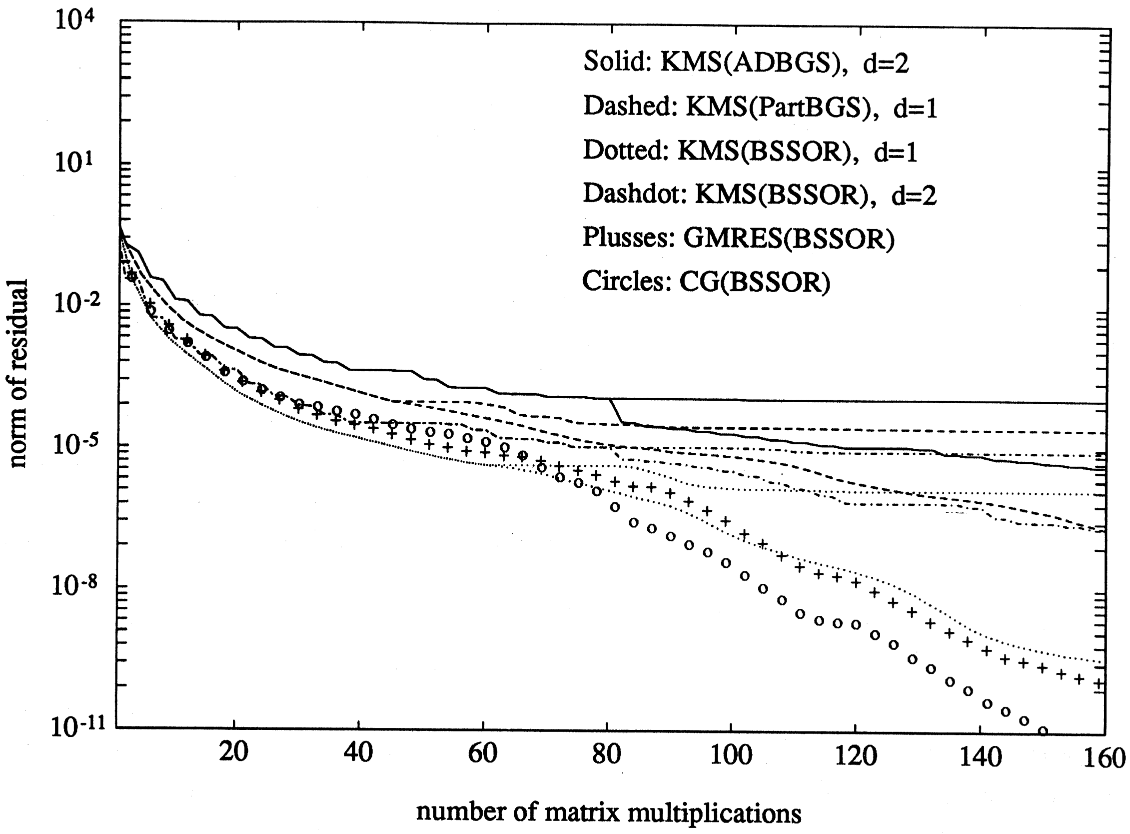

FIG. 1. 50 × 50 grid, Problem 1.

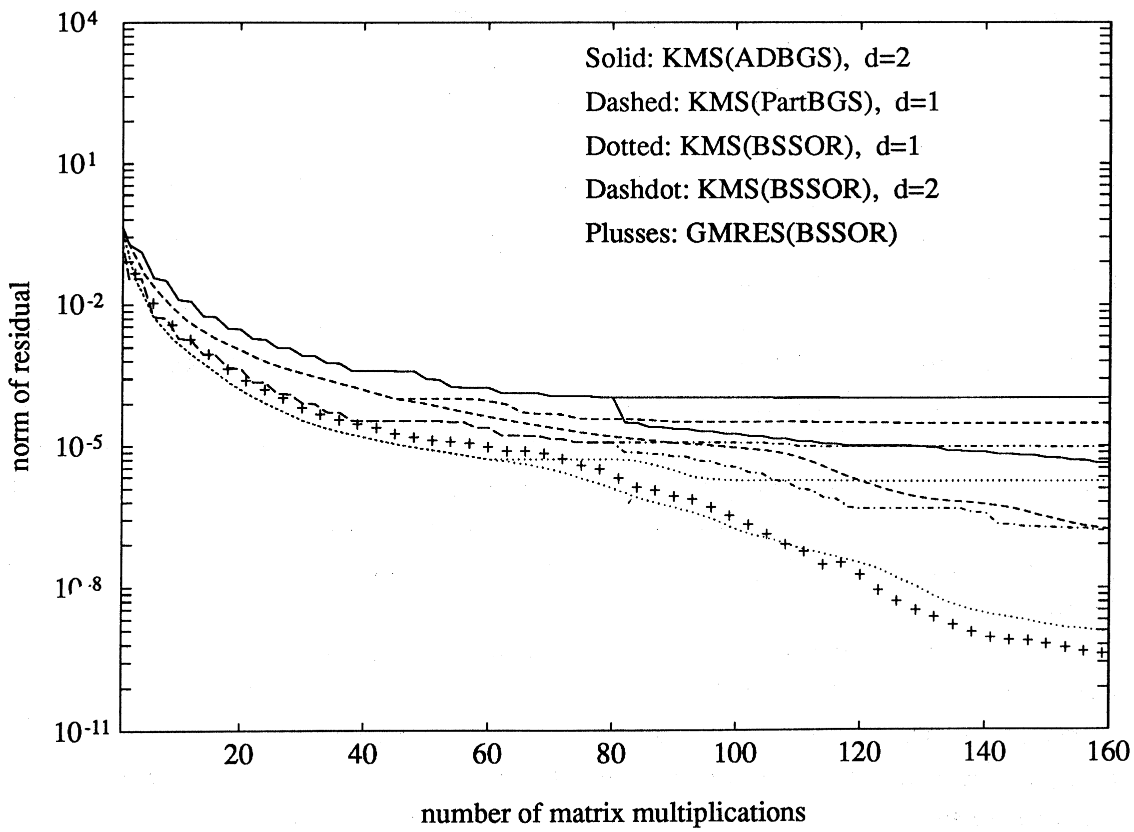

FIG. 2. 50×50 grid, Problem 2.

number of steps that can be performed in the s-step conjugate gradient algorithm of Chronopoulos and Gear, and can be somewhat alleviated by using Householder orthogonalization [20]. With feedback every 20 iterations, all KMS algorithms converged.

These experiments lead us to believe that Algorithm KMS is an effective alternative to standard implementations of Krylov subspace methods even on sequential computers, and the KMS formulation enables us to take advantage of multisplitting preconditioners. On parallel computers, the Algorithm KMS has the further advantage of not requiring frequent communication from $Task_0$ to the multisplitting tasks, and in balancing the number of matrix-vector multiplications among the tasks.

6. CONCLUSIONS

We have presented a nontraditional implementation of Krylov subspace methods that does not use an orthogonal basis to compute the subspace.

If only the vectors $\Delta \hat{x}$ are used for minimization, then this algorithm is equivalent to preconditioned conjugate gradients or GMRES; if additional vectors are used, then the minimization is performed over a larger subspace that includes the standard Krylov subspace. This algorithm has more flexibility than standard implementations and many advantages for parallel computation:

1. The number of synchronization points between the multisplitting tasks and the minimization task is greatly reduced.
2. Additional vectors can be added to the subspace if the Krylov generators are not working fast enough to keep the minimization task busy.
3. Vectors can be dropped if they do not provide a sufficient decrease.
4. We have the option of creating several vectors from any particular basis vector by partitioning it into subvectors and creating a vector from each of these padded with zeros. This might improve the convergence if the preconditioner is locally good but normalization between pieces of the vector is not so good.
5. The minimization task can reinitialize the direction generators at any time by sending the updated x vector. If the minimization has been performed using only the $\Delta \hat{x}$ vectors, then this has no effect on the computation, but if other directions have been added, then convergence can be accelerated without significant synchronization penalty.
6. There are natural extensions of these ideas to nonlinear problems [9].

Clearly, further work remains to be done in developing effective multisplittings and in implementing the algorithm on parallel machines.

In the special case of $p = 1$ (i.e., a single splitting), the idea behind Algorithm KMS should be attributed to Gene Golub, who frequently asks the question, "But why do you need an orthogonal basis?"

Bob Plemmons made useful comments on a draft of the manuscript.

REFERENCES

1 Loyce Adams and J. Ortega, A multi-color SOR method for parallel computation, in *Proceedings of the 1982 International Conference on Parallel Processing*, 1982, pp. 53–56.

2 O. Axelsson and P. S. Vassilevski, A black box generalized conjugate gradient solver with inner iterations and variable-step preconditioning, *SIAM J. Matrix Anal. Appl.* 12:625–644 (1991).

3 Dimitri P. Bertsekas and John N. Tsitsiklis, *Parallel and Distributed Computation Numerical Methods*, Prentice-Hall, Englewood Cliffs, N.J., 1989.

4 T. F. Chan, Rank-revealing *QR* factorization, *Linear Algebra Appl.* 88/89:67–82 (1987).

5 A. T. Chronopoulos and C. W. Gear, On the efficient implementation of

preconditioned s-step conjugate gradient methods on multiprocessors with memory hierarchy, *Parallel Comput.* 11:37–53 (1989).

6 Andreas Frommer and Daniel B. Szyld, H-Splittings and Two-Stage Iterative Methods, *Numer. Math.* 63:345–356 (1992).

7 G. H. Golub and M. Overton, Convergence of a two-stage Richardson iterative procedure for solving systems of linear equations, in *Proceedings of the Dundee Biennial Conference on Numerical Analysis*, Univ. of Dundee, Scotland, June 1981, Springer-Verlag.

8 Gene H. Golub and Charles F. Van Loan, *Matrix Computations*, Johns Hopkins U.P., Baltimore, 1983.

9 Chiou-Ming Huang, Preconditioning Parallel Multisplittings for Solving Systems of Equations, Technical Report CS-TR-2948, Ph.D. Dissertation, Computer Science Dept., Univ. of Maryland, College Park, Aug. 1992.

10 Chiou-Ming Huang and Dianne P. O'Leary, Preconditioning parallel multisplittings for solving linear systems of equations, in *6th ACM International Conference on Supercomputing, Washington, DC*, July 1992.

11 M. Neumann and R. J. Plemmons, Convergence of parallel multisplitting iterative methods for M-matrices, *Linear Algebra Appl.* 88/89:559–573 (1987).

12 D. P. O'Leary, Ordering schemes for parallel processing of certain mesh problems, *SIAM J. Sci. Statist. Comput.* 5:620–632 (1984).

13 D. P. O'Leary and P. Whitman, Parallel QR factorization by Householder and modified Gram-Schmidt algorithms, *Parallel Comput.* 16:99–112 (1990).

14 Dianne P. O'Leary and R. E. White, Multi-splittings of matrices and parallel solution of linear systems, *SIAM J. Discrete Math.* 6:630–640 (1985).

15 J. M. Ortega, Efficient implementations of certain iterative methods, *SIAM J. Sci. Statist. Comput.* 9:882–891 (1988).

16 Theodore S. Papatheodorou and Yiannis G. Saridaki, Parallel algorithm and architectures for multisplitting iterative methods, *Parallel Comput.* 12:171–182 (1989).

17 R. Glowinski et al. (Eds.), *Proceedings of the First International Symposium on Domain Decomposition Methods for Partial Differential Equations, des Ponts et Chaussees, Paris, France, January 7–9, 1987*, SIAM, Philadelphia, 1988.

18 Youcef Saad and Martin H. Schultz, GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.* 7:856–869 (1986).

19 G. W. Stewart, An Updating Algorithm for Subspace Tracking, *IEEE Trans. on Signal Processing* 40:1535–1541 (1992).

20 Charles D. Swanson and Anthony T. Chronopoulos, Orthogonal s-Step Methods for Nonsymmetric Linear Systems of Equations, Technical Report UMSI91/341, Univ. of Minnesota Supercomputer Inst., Minneapolis, 1991.

21 R. E. White, Multisplitting with different weighting schemes, *SIAM J. Matrix Anal. Appl.* 10:481–493 (1989).

Received 10 June 1992; final manuscript accepted 2 September 1992