

## SOME REMARKS ON GMRES FOR TRANSPORT THEORY

**Bruce W. Patton**

Propulsion Research Center  
Marshall Space Flight Center  
Huntsville, Alabama 35812  
bruce.w.patton@nasa.gov

**James Paul Holloway**

Department of Nuclear Engineering and Radiological Sciences  
University of Michigan  
2355 Bonisteel  
Ann Arbor, MI 48109-2104  
hagar@umich.edu

### ABSTRACT

We review some work on the application of GMRES to the solution of the discrete ordinates transport equation in one-dimension. We note that GMRES can be applied directly to the angular flux vector, or it can be applied to only a vector of flux moments as needed to compute the scattering operator of the transport equation. In the former case we illustrate both the delights and defects of ILU right-preconditioners for problems with anisotropic scatter and for problems with upscatter. When working with flux moments we note that GMRES can be used as an accelerator for any existing transport code whose solver is based on a stationary fixed-point iteration, including transport sweeps and DSA transport sweeps. We also provide some numerical illustrations of this idea. We finally show how space can be traded for speed by taking multiple transport sweeps per GMRES iteration.

*Key Words:* transport equation, GMRES, Krylov subspace

### 1. INTRODUCTION

The neutral particle transport equation, in either discrete or continuous form, can be written abstractly as

$$T\psi = CM\psi + S. \quad (1)$$

Here  $T$  represents the streaming and particle interaction processes,  $T\psi = \hat{\Omega} \cdot \vec{\nabla} \psi + \Sigma_t \psi$ . The operator  $M$  computes some number of spherical-harmonic moments of the flux  $\psi$  as needed for the order of scattering being modeled by the inscatter operator  $C$ , which could include up- and down-scatter in energy. The system is driven by an external source  $S$  and by some incoming flux boundary condition specified on a non-reentrant surface surrounding the region of interest. For the case of isotropic scattering we would have  $M\psi = \int_{4\pi} \psi d\hat{\Omega}$ , which is just the scalar flux, but the restriction to isotropic scattering thereby suggested is not necessary for the developments that follow, and the operator  $M$  could in fact compute several higher moments. In general, the Greek letter phi will represent all of the flux moments necessary for a given computation,  $\phi = M\psi$ . Further, since we are ultimately talking about numerical methods, the operators  $T$ ,  $C$  and  $M$  could represent some set of consistent numerical approximations. But none of these details is crucial to the discussions that follow.

Traditional approaches to the iterative solution of Eq. 1 are often in the form of a fixed-point iteration on the flux moments

$$\phi^{(i)} = L\phi^{(i-1)} + \phi_0. \quad (2)$$

In standard transport sweeps we have  $L = MT^{-1}C$  and  $\phi_0 = MT^{-1}S$  is the uncollided flux,  $\phi_u$ , while in DSA accelerated sweeps we have  $\phi_0 = [I + MD^{-1}C]\phi_u$  and  $L = M[T^{-1} + D^{-1}(CMT^{-1} - I)]C$ , where  $D$  represents a diffusion (or more generally a  $P_n$ ) correction that is applied after each sweep [1].

GMRES [2] (Generalized Minimal RESidual) is an algorithm for the solution of general linear systems  $Ax = b$ . The method begins with a guess  $x^{(0)}$ , from which the initial residual  $r = b - Ax^{(0)}$  is constructed; the method then approximately solves the problem for the correction  $Az = r$ , so that  $x = x^{(0)} + z$ . This is done by minimizing the residual  $\|r - Az\|$  over the Krylov subspace

$$K_n(A, r) = \text{span}\{r, Ar, A^2r, \dots, A^{n-1}r\}. \quad (3)$$

The approximate solution is thus sought in the affine space  $x^{(0)} + K_n(A, r)$ , which is iteratively allowed to grow as needed. Practical accomplishment of the residual minimization is greatly simplified by the properties of the Krylov subspace; these important details will not be reviewed here, but can be found in Saad and Schultz.[2]

It should be noted that GMRES requires the storage of  $n$  vectors, each the size of the solution vector  $x$ , to hold a basis for  $K_n(A, r)$ . Pure GMRES proceeds, iteratively increasing the dimension of the Krylov subspace at each step, until  $\|r - Az\|$  is smaller than some specified tolerance. In contrast, the algorithm called GMRES( $n$ ) limits the size of the Krylov subspace used to  $n$  dimensions and restarts the whole algorithm if this limit is reached. In practical problems GMRES( $n$ ) has a very important advantage over GMRES because it has a fixed and known storage requirement, but it can converge more slowly than pure GMRES. In either case, note that the algorithm is a non-stationary iteration, and cannot be written in the form of Eq. 2 with fixed operator  $L$ .

Recently there has been growing interest in applying this method to the transport equation (see, for example, [3–9, 11, 12].) One approach is to apply GMRES to a discretized version of the right-preconditioned integro-differential transport equation  $(T - CM)P^{-1}P\psi = S$ , with preconditioner  $P$ . The initial residual  $\delta S = S - (T - CM)\psi^{(0)}$  used by GMRES is then the particle source that would determine the correction between the initial guess  $\psi^{(0)}$  and the true flux. Without a preconditioner the Krylov subspace over which the residual is minimized would be  $K_n(T - CM, \delta S)$ , built from powers of the transport operator applied to this particle source. While perhaps algebraically acceptable, physically we would not expect that  $K_n$ , for small  $n$ , could accurately represent the flux. But if a preconditioner  $P$  represents an approximate transport process then the subspace  $K_n((T - CM)P^{-1}, \delta S)$  involves applying the transport operator to a flux,  $P^{-1}\delta S$ , and the angular flux  $\psi$  will then be found by minimizing the residual  $\|(T - CM)\psi - S\|$  over the affine space of angular fluxes  $\psi^0 + P^{-1}K_L((T - CM)P^{-1}, \delta S)$ . So GMRES applied directly to the transport equation without a preconditioner looks physically suspect, but with a good preconditioner it might attack otherwise quite difficult problems. We will illustrate the use of GMRES with ILU preconditioners to solve directly for the angular flux in Section 2.

Another approach to using GMRES for transport problems is to use it within a standard fixed point iteration, such as transport or DSA sweeps. GMRES can be applied to the fixed-point problem

$$\phi = L\phi + \phi_0 \quad (4)$$

whose solution is sought by the iteration of Eq. 2. This idea seems to originate with Faber & Manteuffel's [10] application of the conjugate gradient method to the problem, followed by the application of GMRES by Ashby, et al. [3], and later by Kelley and Xue [4]. However, its interesting interpretation as an *accelerator* for the fixed point iteration first appears in reference [11], where we show that applying GMRES( $n$ ) to  $(I - L)\phi = \phi_0$  is equivalent to finding a linear combination of the iterates  $\phi^{(i)}$ ,  $i = 0, 1, \dots, n - 1$ , from Eq. 2 that minimizes the residual  $\|(L - I)\phi + \phi_0\|$ . It is easy to show that if the fixed-point iteration converges ( $\|L\| < 1$ )

$$\|\phi^{(\infty)} - \phi^{(n)}\| \leq \frac{\|(L - I)\phi^{(n)} + \phi_0\|}{1 - \|L\|}, \quad (5)$$

so we see that GMRES( $n$ ), by minimizing the residual, is in a sense finding the best next iterate for the flux moments by using the information available in all of the iterates  $i = 0, 1, \dots, n - 1$ . This is in contrast to the iterate  $L\phi^{(n-1)}$  used by the standard iteration, which uses information only from the last iterate and which is not designed to minimize a residual. Since usually the fixed point iteration is run to an iterate  $n$  where  $\|\phi^{(n)} - \phi^{(n-1)}\| = \|(L - I)\phi^{(n-1)} + \phi_0\|$  is small, we can always do better by applying GMRES( $n$ ) instead, because GMRES( $n$ ) will actually minimize the quantity that we are trying to make small. This can be contrasted with the common game of acceleration schemes, which is to find a way to make  $\|L\|$  as small as possible; GMRES gives us another approach based on minimizing  $\|(L - I)\phi - \phi_0\|$  over a space of previous iterates.

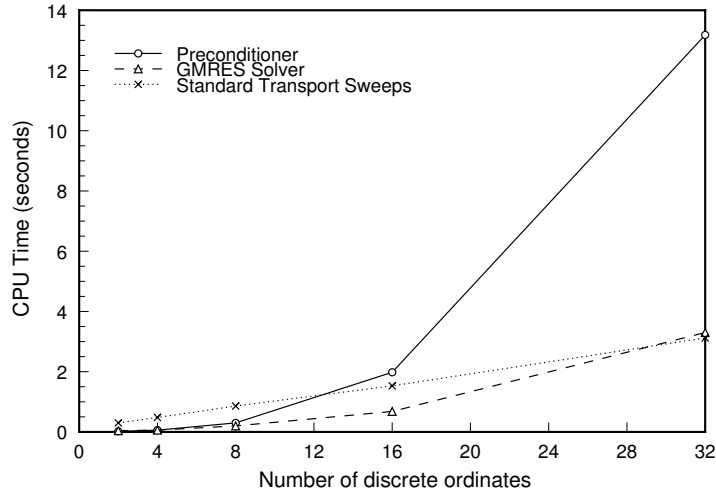
GMRES can therefore be expected to decrease the number of iterations required for *any* fixed point iteration for the flux moments. GMRES might even be able to accelerate DSA. GMRES thus has the potential to provide an accelerator for any existing transport solver. This capability can be retrofitted relatively easily into an existing code, because GMRES requires only a means to evaluate the operator  $(I - L)$  on a vector, and this capability is trivially provided by a code that already evaluates the action of  $L$  on any iterate. In Section 3 we will provide some illustrations of this approach.

## 2. ILU PRECONDITIONED GMRES APPLIED TO THE ANGULAR FLUX

The most direct application of GMRES to the transport equation comes from writing the discretized equation as  $(T - CM)\psi = S$ . After storing  $(T - CM)$  as a sparse matrix, we proceed to apply preconditioned GMRES to solve directly for the angular flux. In this section we report on some experiences with this approach using the "off-the-shelf" SPARSKIT package [13] to provide the preconditioners and the GMRES solver. We use the ILU( $\tau$ ) algorithm for preconditioning, with dropping parameter  $\tau = 10^{-4}$ . This algorithm performs an LU factorization of the system matrix, except as each element of the LU factorization is generated it is set to zero if it is smaller than  $\tau$  times the norm of the row from which it came in  $(T - CM)$ . This keeps the factorization incomplete and sparse.

The first set of problems solved considers a 5 mfp thick slab with an isotropic incoming flux applied at both boundaries; the discretization is diamond difference discrete ordinates. The transport sweeps were converged to a relative error of  $10^{-6}$  on the scalar flux and the net current, unless otherwise noted, and the GMRES residual was converged to  $10^{-6}$  in the  $l_2$  norm.

We have studied the speed of ILU preconditioned GMRES as a function of spatial mesh size, number of discrete ordinates, and scattering ratio [12]. The CPU time required to achieve a given convergence criterion increases roughly linearly with the number of mesh points. This is a direct consequence of the need to perform matrix-vector multiplies in the GMRES algorithm, and provides the same scaling as seen



**Figure 1.** Increase in CPU time with the number of discrete ordinates, with the ILU preconditioner and the GMRES solve separated, compared to standard sweeps. Note the rapid growth of the work required to construct the preconditioner.

in standard transport sweeps. But, as shown in Fig. 1, the CPU time increases nonlinearly in the number of discrete ordinates, and this growth in CPU time is dominated by the preconditioner construction; this growth is not surprising since the bandwidth of the system matrix is proportional to the number of discrete ordinates,  $M$ , so a matrix-vector multiply scales with  $M^2$ , and the ILU construction time will increase even more rapidly than this. On the other hand, the ILU preconditioned GMRES is largely insensitive to the scattering ratio  $c$ , as shown in Fig. 2. The computational time required for GMRES convergence is observed to be essentially independent of the scattering ratio  $c$ , unlike methods based on transport sweeps. This independence is a consequence of the ILU( $\tau$ ) preconditioner, which dynamically selects the important matrix elements for the various values of  $c$ .

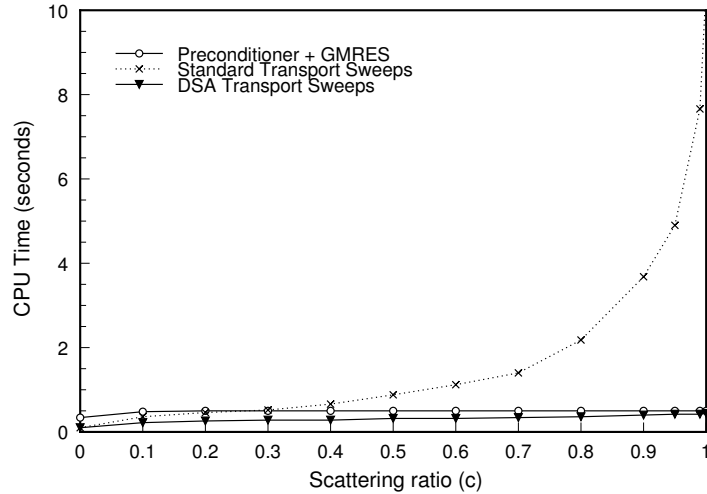
Note that for this particular problem with  $S_8$  quadrature, the combined preconditioner construction time and GMRES solver CPU times are competitive with traditional transport sweeps when the value of  $c$  becomes greater than about 0.25. ILU right preconditioned GMRES is roughly comparable in speed to DSA, but DSA is always somewhat faster. It is worth noting that if the CPU time required for the preconditioner construction phase is neglected (about 0.3 sec), then GMRES is competitive with DSA for any value of  $c > 0$ . This is significant, since the cost of the preconditioner can be amortized over several different values of the source vector,  $S$ , which contains only the external source data and the boundary values.

The next problem is designed to test the ability of GMRES to handle severe anisotropic scattering. The scattering cross section moments used are those derived by Morel [14] to model charged particle scattering. These cross sections moments are given by

$$\Sigma_{s,n} = \frac{1}{200}(M(M + 1) - n(n + 1)) \quad n = 0, 1, \dots, M. \tag{6}$$

The medium is purely scattering and held five mean free paths thick (independent of  $M$ ), with a mesh spacing of 0.1 mean free paths.\* There is once again an isotropic incoming flux on both boundaries. A

\*Note that as the value of  $M$  changes, the total cross section changes as well, and so to keep the slab five mean free paths thick



**Figure 2.** The effect of scattering ratio  $c$  on the run time of ILU preconditioned GMRES, compared to standard sweeps and DSA. GMRES is insensitive to the scattering ratio. This problem uses  $S_8$  quadrature.

**Table I.** Effect of anisotropic scatter on various transport solvers. The convergence criteria is generally  $10^{-6}$ , but on those runs marked with an asterisk was increased to  $10^{-5}$ .

M	ILU( $\tau$ )	GMRES		Sweeps		DSA	
		Iters	CPU	Iters	CPU	Iters	CPU
2	0.02	1	0.02	82	1.06	39	0.62
4	0.06	2	0.10	111	4.82	49	1.80
8	0.26	3	0.26	213*	46.92	41*	9.02
16	0.76	4	0.80	43*	61.32	21*	30.30

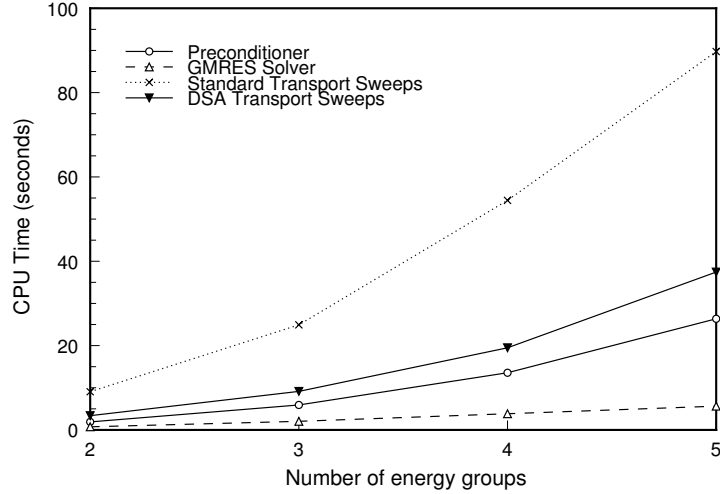
sweeps convergence criteria of  $10^{-6}$  was used on all the angular flux moments in standard sweeps and in DSA, except for the cases  $M = 8$  and  $M = 16$ , where the criteria was relaxed to  $10^{-5}$  in order to achieve convergence within a reasonable number of iterations. Table I compares GMRES with transport sweeps for different values of scattering order  $M$ .

GMRES with ILU preconditioning is seen to be very efficient at handling the severe anisotropy in this problem; it can be seen from Table I that GMRES is from one to two orders of magnitude faster than standard transport sweeps for this test problem.

Finally, we consider a multi-group problem with upscatter and downscatter. A non-symmetric iterative technique such as GMRES can be applied to the full multigroup equations, thereby eliminating the need for energy group iterations. The homogeneous five mfp thick slab problem was considered with isotropic

---

it is necessary to adjust the physical dimension of the slab.



**Figure 3.** The run times of various solvers applied to a problem with both upscatter and downscatter, as a function of the number of energy groups. The preconditioner construction time is shown separately from the GMRES iteration time.

incoming fluxes all energy groups and with multigroup macroscopic cross sections defined as

$$\Sigma_t^g = 1.0 \quad (7)$$

$$\Sigma_{s,0}^{g \rightarrow g} = 0.5 \quad (8)$$

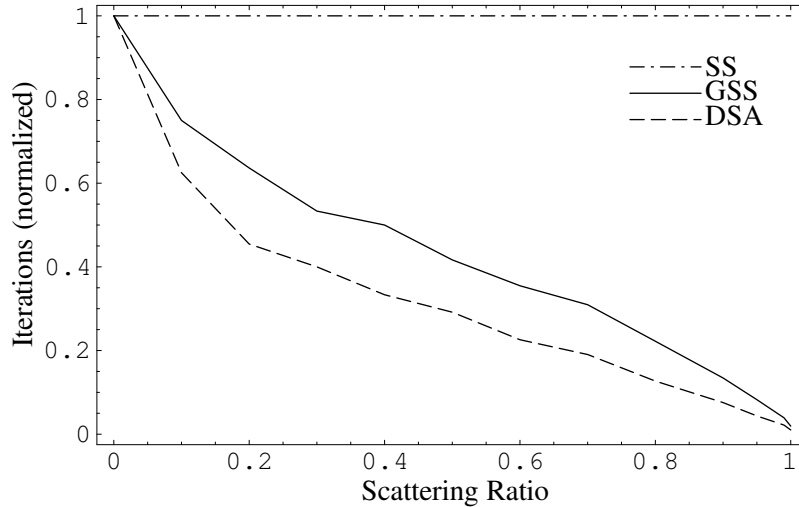
$$\Sigma_{s,0}^{g' \rightarrow g} = 0.2 \Sigma_{s,0}^{g \rightarrow g} \quad g' < g \quad (9)$$

$$\Sigma_{s,0}^{g' \rightarrow g} = 0.1 \Sigma_{s,0}^{g \rightarrow g} \quad g < g' . \quad (10)$$

The traditional approach to this problem for sweep-based methods is to perform additional cycles through those energy groups which have upscattering sources; for comparison, we have performed such upscatter iterations with a convergence criteria of  $10^{-6}$  in the scalar flux. Depending on the number of energy groups and the relative amount of upscattering between the energy groups, the computational effort for upscatter iterations can increase rapidly. In contrast, as applied here, GMRES solves for all of the unknown angular fluxes at each position, angular direction, and energy group simultaneously. It would be expected that an iterative method which can calculate the unknowns in such a global manner would be very efficient at solving upscatter problems.

Figure 3 shows a comparison of GMRES, unaccelerated transport sweeps, and DSA transport sweeps, for different numbers of energy groups for this multigroup problem with upscatter. Note the large increase in CPU time for standard transport sweeps when upscatter is present in the problem. The preconditioner CPU time increase as a function of the number of groups is again nonlinear, but the preconditioned GMRES algorithm is faster than DSA transport sweeps.

While the ILU preconditioned GMRES has proven rather effective in solving the slab geometry transport problems considered, it is worrying that as the computational work scales nonlinearly with the number of discrete ordinates. The worst aspects of this are due to the preconditioner, and at some point the preconditioner construction time by itself could exceed the CPU time for standard transport sweeps. This may limit the utility of the direct application of an ILU preconditioned GMRES to specific classes of



**Figure 4.** Number of iterations required by GMRES accelerated simple transport sweeps (GSS) and by DSA, both normalized to the number of iterations required for unaccelerated simple sweeps (SS), as a function of scattering ratio, for a slab 15 mean-free-paths thick. Quadrature order is  $S_8$ .

problems where the preconditioner cost is acceptable, or else can be amortized over many uses. Fortunately, GMRES can also be applied to accelerate traditional transport sweeps and perhaps even DSA transport sweeps, in which the physics of the problem provides much the same benefit as a preconditioner. We discuss this next.

### 3. GMRES AS AN ACCELERATOR FOR TRANSPORT SWEEPS

In this section we apply GMRES to accelerate both standard transport sweeps and DSA transport sweeps in one-group slab geometry problems with isotropic scattering, and with isotropic incoming fluxes applied to both boundaries of the system. The basic discretization is again diamond-difference discrete ordinates. But the fundamental unknowns on which we iterate are now the flux moments  $\phi = M\psi$ , and we build upon traditional transport method iterations, which have the form of the fixed-point iteration of Eq. 2. We compare these iterations with the application of GMRES to the fixed point problem itself,  $(I - L)\phi = \phi_0$ , and note that this can be retrofitted into a code that already performs the fixed point iteration. Basically we simply have to apply  $L$  to a set of flux moments, as though we were performing a step of the fixed point iteration, and then compute  $\phi - L\phi$ . This is all GMRES needs in order to build the Krylov subspace basis and a representation of the operator in that basis.

In the tables and plots we refer to standard sweeps as SS, to DSA sweeps as DSA, and when accelerating either of these with GMRES we prepend the letter G. In all cases the convergence criterion is  $1 \times 10^{-6}$  on the  $l_2$  norm of the solution vector (scalar flux). Also in all cases the initial flux guess is 1, independent of space.

Figure 4 shows the ratio of the number of sweeps required by GSS to that required by standard transport sweeps (SS), as a function of the scattering ratio; the plots are for  $S_8$  computations in a 15 mfp thick slab. Also shown is the same ratio for DSA. We see that, as predicted, GSS requires fewer transport sweeps than simple sweeps alone. Indeed, its behavior with scattering ratio is very similar to that of DSA, although DSA in all cases requires fewer sweeps than GSS. It is surprising that GSS does so well for highly

scattering systems; for a 5 mean-free-path thick slab with  $c = 1$  (not shown, see [11]) GSS requires only 12 physical transport sweeps, while simple unaccelerated sweeps requires 176 physical sweeps. Thus, while about one particle in a million has collided 176 times inside the slab, by minimizing the residual GSS is able to construct a solution accurate to  $10^{-6}$  as a linear combination of only the first 12 simple transport sweeps. While not shown here, a comparison of CPU time shows that GMRES reduces the error in the scalar flux at about the same rate as DSA [11]. While these results do not show that GMRES is faster than DSA, it should be noted that GMRES' good properties come from its minimization of the residual, independent of the details of spatial discretization. We can therefore expect that it will show excellent performance even for discretizations in which consistent DSA is not useful; Ashby et al. [3] have shown this to be so in one-dimension.

GMRES can in principle be used to accelerate DSA, and Table II shows some results of this. These results are for a 15 mean-free-path thick slab using either  $S_8$ ,  $S_{16}$ , or  $S_{32}$  quadrature; the data format is  $x/y$  where  $x$  is the number of applications of  $T^{-1}$  (sweeps) and  $y$  is CPU time in milliseconds. Generally, the GMRES accelerated DSA (GDSA) requires the same or fewer transport sweeps than pure DSA. Occasionally it requires one extra sweep; this comes about because our implementation of the GMRES accelerated algorithms all compute and store the uncollided flux as part of their initialization, while the DSA algorithm does not need this extra step.

While GMRES generally reduces the number of sweeps required by DSA for larger values of the scattering ratio, it does not always reduce the CPU time required. GMRES has significant overhead (orthogonalization of a Krylov subspace basis, minimization of the residual, flux construction as a linear combination of basis vectors, etc.) Only if the sweeps are expensive, as in the  $S_{16}$  and  $S_{32}$  runs, does this extra overhead pay for itself by reducing the overall CPU time. In our computations the overhead from GMRES ranges from as much as 30% for  $S_8$  calculations down to about 15% for  $S_{32}$  calculations, while, for comparison, the overhead of DSA ranges from 10% down to about 1% for those same cases. Careful optimization of GMRES and improved efficiency of the algorithm are therefore important matters for practical implementation.

Table II also shows the scaling of GDSA with quadrature order. We see that it scales identically to DSA with the number of discrete ordinates. This is in contrast to the rapid growth of CPU time required for ILU preconditioned GMRES. The GMRES accelerated simple transport sweeps, GSS, CPU time also scales linearly with the number of discrete ordinate directions.

An important characteristic of DSA is, of course, that the number of iterations required for convergence is relatively insensitive to the scattering ratio. GSS shares this feature; the number of iterations required for GMRES accelerated sweeps does grow slowly with  $c$ , and this growth is slightly faster than that exhibited by DSA, but it is nothing like the exponential growth that unaccelerated sweeps would show. However, unlike DSA, GSS also shows a slow growth in the number of sweeps required as the optical thickness of the system is increased. Physically this growth is not surprising: GMRES acceleration of simple transport sweeps cannot be expected to magically provide all of the missing physics of highly collided particles far from their point of injection into the slab, and so in a thick slab it requires more physical transport sweeps in order to accumulate enough information on these highly collided particles to provide a good solution. DSA on the other hand is nearly optimal in describing particles far from the boundaries in a highly scattering medium, so the GMRES accelerated DSA also cannot be expected to take many fewer iterations than DSA for thick systems; this was seen in the only marginal acceleration of DSA displayed in Table II. Thus, while for thin slabs the GMRES accelerated simple transport sweeps are almost as fast as DSA, GSS slows down for thicker slabs. This may explain the results of other computations [5, 6] which seem to



**Table II.** GMRES acceleration of DSA (Iterations/msec)

$c$	$S_8$		$S_{16}$		$S_{32}$	
	DSA	GDSA	DSA	GDSA	DSA	GDSA
0.00	2/55	2/87	2/93	2/149	2/171	2/271
0.10	5/143	5/188	5/251	5/319	5/473	5/580
0.20	5/143	6/224	5/251	6/379	5/473	6/685
0.30	6/172	6/224	6/304	6/379	6/579	6/685
0.40	6/172	6/224	6/304	6/379	6/579	6/685
0.50	7/201	6/224	7/358	6/379	7/674	6/685
0.60	7/201	7/260	7/358	7/440	7/674	7/792
0.70	8/231	7/260	8/409	7/440	8/775	7/792
0.80	8/231	7/260	8/409	7/440	8/775	7/792
0.90	9/260	7/260	9/462	7/440	9/876	7/792
1.00	10/289	8/298	10/515	8/500	10/979	8/899

suggest that GMRES is not comparable with DSA; these other works focused on very thick systems.

While GMRES appears to offer a very effective accelerator for transport sweeps, it must be remembered that this comes at the cost of significant storage requirements. However, GMRES accelerated iterations provide a systematic way to trade storage for speed. If  $\phi^{(i)} = L\phi^{(i-1)} + \phi_0$  is a fixed point iteration for the flux moments and  $l$  is any integer, then  $\phi^{(i)} = L^l\phi^{(i-l)} + \sum_{j=0}^{l-1} L^j\phi_0$  is another fixed point iteration with the same solution. While this corresponds trivially to  $l$  iterations of the original algorithm, the acceleration of this algorithm using GMRES will depend on  $l$ . Applied to the simple transport sweeps algorithm,  $L = MT^{-1}C$  with  $\phi_0 = MT^{-1}S$ , such an algorithm corresponds to taking  $l$  simple transport sweeps within each iterate of GMRES, so that each basis vector of the Krylov subspace being built contains information about more highly collided fluxes. Since we are now giving GMRES better information about multiply scattered particles we can expect that for a highly scattering problem a smaller Krylov subspace will be required in order to achieve a good result. Thus, while the total number of transport sweeps might not be reduced (in fact, we shall see in the examples below that it is usually increased), we can expect a smaller storage requirement to hold the basis of the Krylov subspace (smaller  $n$ ). Thus, we may be able to trade storage for speed. We shall call this a GMRES accelerated  $l$ -step transport sweep, and denote it G2SS and G3SS for the case of 2 and 3 sweeps per Krylov subspace dimension, respectively.

Table III shows the CPU time and storage requirement for GMRES accelerated 2-step and 3-step transport sweeps (G2SS and G3SS). While we can reduce the storage requirements by performing these multiple sweeps, the total number of sweeps (applications of  $T^{-1}$ ), and hence the CPU time required, actually increases. Thus, we have an explicit ability to trade space for speed.

#### 4. CONCLUSIONS

In this brief paper we have reviewed some results on the application of GMRES to transport problems. GMRES can be applied to the direct solution of the transport equation for the angular flux, provided that a good preconditioner is used. A general purpose ILU preconditioner has proven effective for relatively small problems, including problems with upscatter and with highly anisotropic scatter. Especially in this latter problem the method performed extremely well. While some optimization may be possible, the

**Table III.** Storage required for GMRES accelerated sweep methods

$c$	GMRES		G2SS		G3SS	
	Vectors Stored	CPU (msec)	Vectors Stored	CPU (msec)	Vectors Stored	CPU (msec)
0.00	1	251	1	528	1	805
0.10	5	651	4	1101	3	1369
0.20	6	754	5	1294	4	1652
0.30	7	872	5	1294	4	1652
0.40	8	966	6	1488	5	1936
0.50	9	1074	7	1684	6	2232
0.60	10	1184	8	1880	6	2232
0.70	11	1295	9	2090	7	2509
0.80	13	1523	10	2277	8	2798
0.90	15	1756	11	2475	9	3087
1.00	18	2119	13	2879	11	3670

nonlinear scaling of ILU construction costs with the number of discrete ordinates makes its application as a general purpose transport solver questionable.

We have observed that classical iterations for transport solutions are in the form of fixed point iterations for flux moments [10]. The application of GMRES to such a fixed point problem can be interpreted as an accelerator or extrapolation method which constructs a new iterate as a linear combination of previous iterates [11]. It does so by minimizing the residual, which is an upper bound for the flux error itself, so GMRES applied to the fixed point problem can be expected to accelerate the fixed point iteration. This idea can be retrofitted into any existing transport code that is based on a stationary iteration. We have shown that this GMRES acceleration greatly accelerates standard transport sweeps, providing performance that is often comparable to DSA. DSA itself can be accelerated, but only if the cost of transport sweeps is high. The real utility of this GMRES acceleration of transport sweeps may be in problems for which DSA is unavailable, but that is a matter for others to explore.

The chief objection to GMRES as a transport solver is that it requires storage of multiple flux (or flux moment) vectors. We have shown that this space can be systematically traded for speed by taking multiple transport sweeps within each GMRES iteration. This allows additional flexibility in the application of GMRES to accelerate transport iterations.

## REFERENCES

- [1] R. E. Alcouffe, "Diffusion synthetic acceleration methods for the diamond-differenced discrete-ordinates equations," *Nuclear Science and Engineering*, **64**, pp. 344 (1977).
- [2] Y. Saad and M. H. Schultz, "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal of Scientific and Statistical Computing*, **7**, pp. 856 (1986).
- [3] S. F. Ashby, P. N. Brown, M. R. Dorr and A. C. Hindemarsch, "Preconditioned Iterative Methods for Discretized Transport Equations," *Proceedings of the International Topical Meeting on Advances in Mathematics, Computations, Reactor Physics*, Pittsburgh (1991).

- [4] C. T. Kelley and Z. Q. Xue, "GMRES and Integral Operators," *SIAM Journal of Scientific Computing*, **17**, pp. 217 (1996).
- [5] S. Oliveria, "Multigrid and Krylov Subspace Methods for the Transport Equation: Absorbion Case," *Proceedings of the 7<sup>th</sup> Copper Mountain Conference on Multigrid Methods* (1995).
- [6] S. Oliveria, "Krylov Subspace Methods for Transport Equations," *XENFIR III* (1995).
- [7] B. W. Patton and J. P. Holloway, "Application of Krylov Subspace Iterative Methods to the Slab Geometry Transport Equation," *Advancements and Applications in Radiation Protection and Shielding*, Cape Cod, April 21-25, **Vol 1**, pp. 384 (1996).
- [8] B. W. Patton, *Application of Krylov Subspace Iterative Techniques to the Numerical Solution of the Neutron Transport Equation*, Ph.D. Dissertation, University of Michigan (1996).
- [9] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, U.S.A (1997).
- [10] V. Faber and T. A. Manteuffel, "A Look at Transport Theory from the Point of View of Linear Algebra," in *Transport Theory, Invariant Imbedding and Integral Equations*, P. Nelson, ed, Marcel Dekker (1989).
- [11] B. Guthrie, J. P. Holloway and B. W. Patton, "GMRES as a Multi-Step Transport Sweep Accelerator," *Transport Theory and Statistical Physics*, **28**, pp. 83 (1999).
- [12] B. W. Patton and J. P. Holloway, "Application of preconditioned GMRES to the numerical solution of the neutron transport equation," *Annals of Nuclear Energy*, **29**, pp. 109 (2002).
- [13] Y. Saad, "Sparskit: a basic toolkit of sparse matrix computations," Version 2, Tech. Report, Research Institute for Advance Computer Science, NASA Ames Research Center, Moffet Field, CA (1994).
- [14] J. E. Morel, "Fokker-Planck calculations using standard discrete ordinates transport codes," *Nuclear Science and Engineering*, **71**, pp. 340 (1981).