A Sparse Iterative Method (SIM) for Method of Moments Calculations

A.P.C. Fourie, D.C. Nitch and A.R.Clark
Department of Electrical Engineering
University of the Witwatersrand, Johannesburg
2050 South Africa
a.clark@ee.wits.ac.za

Abstract— **The Sparse Iterative Method (SIM) provides a faster solution to Method of Moments (MoM) matrix equations than does LU-decomposition with forward and back substitution. The SIM produces a solution with computational time proportional to N^2, as opposed to the N^3 time dependence associated with LU-decomposition. The SIM is implemented in an object oriented MoM program which is functionally equivalent to NEC2. In three examples, the SIM is shown to produce results as accurate as LU-decomposition. For dipoles, a flat wire grid and a generic three dimensional missile shape, the speed increase ranged from 3–30 times the speed of LU-decomposition; greater speed increases can be expected with electrically larger problems. The SIM requires no problem formulation changes, such as segment renumbering, and despite the fact that it is demonstrated for a wire MoM based on NEC2, it is general enough to be incorporated to any MoM formulation.**

I. Introduction

We have investigated a stationary[1] iterative method of moments (MoM) solution technique for implementation in an object oriented method of moments program [1]. When a method of moments problem has N unknowns, and N is large, it is computationally time-consuming; most time is spent either filling the matrix and/or in the LU-decomposition of the matrix. Typical method of moment programs execute in a time, t, that is given by

$$t \propto AN^2 + BN^3 + \text{other smaller terms} \qquad (1)$$

where A is the time taken to calculate a single matrix element and B is the time taken to modify a single matrix element during LU-decomposition on a specific machine.

The B coefficient in equation (1) is much smaller than the A coefficient. The A coefficient is larger because computation of an element of the interaction matrix usually involves both numerical integration and other operations. The B coefficient is smaller; modifying a matrix element during LU-decomposition involves only a few floating point operations.

Problems involving a small number of unknowns have solution times proportional to N^2. As the problem size increases, the time associated with the BN^3 term increases more rapidly than for the AN^2 term; this latter term ultimately governs the solution time for large MoM problems.

[1] A *stationary* method is one where the *rule* used to determine the next guess does not change from iteration to iteration.

Iterative methods concentrate on reducing the BN^3 component of the solution. This paper describes a method derived from the physics underlying the MoM. This Sparse Iterative Method (SIM) replaces LU-decomposition for MoM problems and results in solutions in a time proportional to N^2, which is considerably faster than LU-decomposition. Provided a suitable number of iterations are performed, the SIM results are as accurate as those obtained by LU-decomposition. In principle, since only a small fraction of the normal interaction matrix needs to be stored, it is possible to reduce computer storage requirements using the SIM. However, a trade off in terms of computer time versus storage exists; this trade-off is discussed later.

The development and evaluation of the new method was implemented on a redesigned and rewritten version of the well known NEC2 [2] program. Nitch [1] redesigned the FORTRAN version of NEC2 using the Object Oriented Programming (OOP) paradigm and implemented it using the C++ computer language. The C++ program is functionally equivalent to NEC2, but the improved software implementation enables modifications to be easily effected. The iterative solution technique could thus be incorporated easily into NEC, allowing the SIM to be evaluated for large realistic problems. In the past, new methods have been evaluated based on simple examples and implementations, due to the programming effort required to do the evaluation for realistic cases. The use of simplified codes can result in misleading conclusions, due to the fact that the methods are only evaluated for simple cases. We believe that testing methods using realistic problems is crucial to an adequate evaluation; the implementation of the SIM in the C++ NEC made such an evaluation possible.

II. Background to Method of Moments

The detailed derivation of MoM is available from many sources, and only the features of importance to the development of the SIM will be discussed here. The method of moments for perfectly conducting wire segments, as implemented in NEC2, will be considered. In the case of point matching (equivalent to using impulse or delta weighting functions), the MoM involves finding a current distribution across all segments such that the tangential E-field at the centre of each segment is zero. Zero tangential E-Fields satisfy the boundary conditions at the interface of a perfect conductor. In order to find the currents on a structure,

the current on each segment, $I_a(n)$, is assumed to have a generic shape defined by the basis function, J_n, i.e.,

$$\mathbf{I}_a = \sum_{n=1}^{N} I_n J_n \tag{2}$$

where I_n are the unknown coefficients associated with the current on each segment (when subdomain basis functions are used). The value of these unknown coefficients are solved using MoM.

In the case of NEC2, the basis functions consist of the superposition of a sine, cosine and a constant term. The relative relationship between the three terms is obtained from junction boundary conditions, leaving unknown scaling factors, I_n, to be determined for each segment; MoM solves for these scaling factors.

The matrix equation formulated in a MoM solution to an electromagnetic problem can be mathematically stated as

$$\mathbf{ZI} = \mathbf{V} \tag{3}$$

where $\mathbf{Z}$ is an $N \times N$ interaction matrix, $\mathbf{I}$ is the unknown current coefficient vector and $\mathbf{V}$ is the specified excitation vector. The matrix equation is normally solved using LU-decomposition followed by forward and back substitution. An element of the interaction matrix, Z_{ij}, represents the tangential electric field, E^t_{ij}, induced on segment i due to radiation from the generic current density J_j on segment j.

A perhaps obvious, but important, property of the system of equations is that multiplication of the first row of $\mathbf{Z}$ with the current vector $\mathbf{I}$ gives E^t_1: the total tangential E-field on segment 1 due to radiation by the currents on all segments. If a segment is not excited, the tangential E-field should be zero. That is, for the general observation segment, i,

$$E^t_i = \sum_{j=1}^{N} E^t_{ij} \tag{4}$$

E^t_{ij} is the tangential E-field on segment i due to a current on source segment, j. The equation for obtaining this quantity as a function of current, geometry and segment dimensions usually has the form

$$E^t_{ij} = \int_{s_j} I_j J_j G_{ij} ds \tag{5}$$

where s_j denotes integration over the surface of segment j, and G_{ij} is a function of the geometric relationship between the two segments.

The only difference between E^t_{ij}, and the matrix element Z_{ij}, is that Z_{ij} is obtained using the *normalised* form of the current ($I_j = 1$) on the source segment while the *actual* current, $I_j J_j$, is used to obtain E^t_{ij}.

The solution obtained in this manner satisfies the initial premise; i.e. the tangential E-field at the center of each segment will be zero—excluding computer rounding errors and machine precision.

The components of the total simulation time mentioned in equation (1) result from the following operations:

- N^2 elements have to be calculated in order to initialise the Z-matrix. Each element is calculated by numerically solving an equation in the form of equation (5), with $I_n = 1$ for each element.
- The excitation vector, $\mathbf{V}$, is filled according to the excitation of each segment in the structure. Equation (3) is normally solved by factoring the Z-matrix using LU-decomposition and then solving for $\mathbf{I}$. LU-decomposition requires N^3 operations; for large problems, matrix factoring hence comprises the largest component of the execution time.

III. Previous Investigations and Attempts at Iterative Methods

Four previous publications describe iterative methods which relate to the SIM; these are: the banded Jacobi method [3], the block Seidel technique [4], the conjugate gradient method [5] [6] and the Impedance Matrix Localization (IML) method [7].

The banded Jacobi iterative scheme described by [3] was one of the first iterative methods to be implemented in a Method of Moments EM Code. This iterative technique is implemented in the GEMACS [8] program.

In GEMACS, a banded matrix is obtained by neglecting matrix elements some distance (or band) away from the main diagonal. A second matrix, consisting of those elements that were disregarded in the banded matrix, is also generated. A first estimate of the current is found by factoring and solving the banded system. The second matrix is then used to improve the current estimate iteratively until convergence is achieved. The method is similar to the standard Jacobi iterative scheme, except that a banded, rather than a diagonal matrix is employed. Mathematically the SIM is closely related to banded Jacobi iterative scheme, except that a sparse rather than banded matrix is employed—this difference has considerable advantages however.

The following formulation of the banded Jacobi method is given in terms of equation (3). The method is initialised by breaking the Z-matrix into two submatrices

$$\mathbf{Z} = \mathbf{B} + \mathbf{LU} \tag{6}$$

where $\mathbf{LU}$ is an upper and lower triangular matrix and $\mathbf{B}$ is a banded diagonal matrix with upper and lower bandwidths M (number of minor diagonals). GEMACS iteratively solves the equation

$$\mathbf{BI}^{k+1} = \mathbf{V} - (\mathbf{LU})\mathbf{I}^k \tag{7}$$

In order to achieve convergence while using small bands, the banded Jacobi method relies on renumbering segments to ensure that the majority of the largest elements of the Z-matrix lie about the main diagonal. However, with many structures, such an ordering is not always possible; when a multi-dimensional structure is electrically large, it is clearly impossible to ensure that segment numbers of adjacent

segments remain numerically close to each other—large cylinders and spherical shapes are particular examples. In such cases, the use of the banded Jacobi method invariably causes large matrix elements to lie far from the main diagonal, necessitating a wide band in the matrix for reasonable convergence rates.

A second iterative scheme was proposed by Baldwin, Boswell, Brewster and Allwright [4], who modified NEC2 to iteratively solve for a number of substructures. Their method is similar to the banded Jacobi method in that it consists of a **B**-matrix which contains all the self interaction submatrices for the substructures, and a **LU**-matrix, which contains the interaction submatrices between different substructures:

$$\mathbf{Z} = \begin{bmatrix} [\mathbf{Z}_1] & 0 & \cdots & 0 \\ 0 & [\mathbf{Z}_2] & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & [\mathbf{Z}_p] \end{bmatrix} + \begin{bmatrix} 0 & [\mathbf{Z}_{12}] & \cdots & [\mathbf{Z}_{1p}] \\ [\mathbf{Z}_{21}] & 0 & \cdots & [\mathbf{Z}_{2p}] \\ \vdots & \vdots & \vdots & \vdots \\ [\mathbf{Z}_{p1}] & [\mathbf{Z}_{p2}] & \cdots & 0 \end{bmatrix} \tag{8}$$

Each element in square brackets denotes a submatrix which contains either the self-interaction terms for a specific substructure, or the interaction between different substructures. This iterative method is akin to the banded Jacobi method, except that it uses a Seidel iteration technique. This method is a more coarsely granulated version of the banded Jacobi method and is suitable for a problem domain with unconnected substructures; good results were reported for this specific class of problems. The method is, however, not general enough for application to problems such as electrically large continuous bodies.

Conjugate gradient methods constitute the third class of iterative methods reviewed in this paper. Although Sarkar and Siarkiewicz [5] report faster convergence for this technique in comparison with stationary methods, Davidson [6], amongst others, reports slower convergence rates for the conjugate gradient method. The reason for this discrepancy possibly is a result of the fact that Sarkar investigated very simple problems (dipoles and square patches) that resulted in matrices that had only a few eigenvalues; for such matrices, the conjugate gradient method is known to converge quickly. Davidson applied the method to more realistic problems and reports that the solution times are even greater than those for LU-decomposition.

Recently, Canning [7] proposed an Impedance Matrix Localization (IML) method for MoM calculations. IML formulates sub-domain basis and weighting functions which exhibit strong directional characteristics. The radiation patterns of the basis and weighting functions are then made directional between the source and observation regions of a structure, resulting in an interaction matrix with many small elements. The small elements are set to zero, and the resulting sparse matrix is solved iteratively. The initial results presented for this technique are for simple two dimensional problems and it is difficult to compare to the SIM until results for more general three dimensional cases becomes available.

IV. Development of the Sparse Iterative Method (SIM)

The SIM consists of a sparse matrix that contains only those elements that have an interaction greater than some specified value. In the first implementation of the SIM, the elements making up the sparse matrix were chosen according to the distance between interacting segments. If the source and observation segments were further apart than a distance that we called the zero interaction distance, d_0, then their interactions were not included in the sparse matrix. An iterative scheme is then used to compensate for the couplings which were neglected. The accuracy of the solution obtained using the SIM can be forced to be as accurate as that obtained using LU-decomposition by using a sufficient number of iterations.

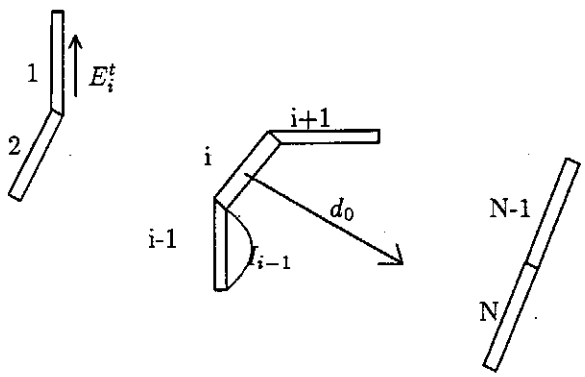

Fig. 1. Arbitrary oriented wire segments used to illustrate the development of the Sparse Iterative Method (SIM)

The following discussion concerning the development of the SIM is done with reference to figure 1. Initially, coupling between segments that are further apart than a distance, d_0, are omitted when filling the matrix, since coupling reduces with increasing separation. Alternatively, some measure of interaction other than a distance can be used to determine which matrix elements to include. The incorporation of other parameters, such as orientation, will probably improve the performance of the method.

Omitting the coupling between segments further apart than d_0 results in a sparse matrix, **S**; the non-zero elements represent strong coupling. Thus the **Z** matrix is broken into two submatrices

$$\mathbf{Z} = \mathbf{S} + \mathbf{D} \tag{9}$$

where the **D** matrix is the dense matrix of the interactions not in the sparse matrix **S**.

The matrix equation, $\mathbf{SI} = \mathbf{V}$, is solved for the currents, **I**, on the structure where **V** is the structure excitation. The current, **I**, will not satisfy the boundary conditions at all segments, since weaker interactions were disregarded and, as a result, some segment centers will have non-zero tangential E-fields. A modified LU-decomposition technique[9] is

used on S to take advantage of its sparsity, and our implementation did not use specialized storage mechanisms for the sparse matrix, as it is outside the scope of this work.

The crux of our solution technique is to apply artificial sources to the structure with values that are equal and opposite to the tangential error fields on each segment. These sources compensate for the coupling that has been neglected between those segments that are further than d_0 apart. The error fields can be described by a vector of length, N, called $\mathbf{E}^e$. Mathematically, application of the artificial sources can be achieved by subtracting the error fields, $\mathbf{E}^e$ from the excitation vector, $\mathbf{V}$. New currents are then calculated by solving $\mathbf{SI} = \mathbf{V} - \mathbf{E}^e$. This procedure is performed iteratively until the solution converges.

The iterative technique described above can be expressed in concise mathematical notation as follows:

Set $\mathbf{V}_0 = \mathbf{V}$ and iterate:

$$\mathbf{SI}_k = \mathbf{V}_k \tag{10}$$

$$\mathbf{E}^e_k = \mathbf{DI}_k \tag{11}$$

$$\mathbf{V}_{k+1} = \mathbf{V}_k - \mathbf{E}^e_k \tag{12}$$

This procedure can be continued until the number of iterations, k, results in a converged solution for $\mathbf{I}_k$. Convergence is not usually easy to define without knowledge of the accurate answer. Two measures of convergence were used:

The first is the Predicted Relative Error (PRE) as defined by [8]. For the k-th iteration, the PRE is defined as

$$PRE_k = (IRE_k)^2 / IRE_{k-1} \tag{13}$$

where IRE is the Iterative Relative Error (IRE) for iteration k and is given by:

$$IRE_k = \frac{\|\mathbf{I}_k - \mathbf{I}_{k-1}\|}{\|\mathbf{I}_k\|} \tag{14}$$

- the Euclidean vector norm is implied.

The other convergence measure used was the normalised residual error, R^2 which is given by

$$R^2 = \left(\frac{\|\mathbf{E}^e_k\|}{\|\mathbf{E}^e_0\|} \right)^2 \tag{15}$$

as defined in [6]

The computer time requirements for the SIM are given by equation (16).

$$t \propto AN^2 + BN^x + kCN^2 + \text{other smaller terms} \tag{16}$$

As in equation (1), the AN^2 term is associated with calculating the interaction elements for the **S** and **D** matrices. The BN^x term is related to the time required to factor the sparse matrix, **S**, where x ranges from 1–3, depending on matrix sparsity. When large sparse matrices are factored x approaches 1 . The kCN^2 term is related to the time required to calculate the E-tangential errors using equation (11) for k iterations. In this case the C coefficient is quite small since it only involves a single complex multiplication and addition.

The SIM method hence produces results in times proportional to N^2. The SIM resulted from careful consideration of the physical mechanisms at play in the method of moments. It is interesting to note that the final method bears a similarity to the Banded Jacobi Method [8], where the updating of the RHS of equation (10) is done in exactly the same manner. The major advantage of the SIM is that a sparse, rather than a banded, matrix is employed. As such, the method is not constrained to examples which are easily renumbered. The SIM should also result in either less dense matrices to factorise, or, for similar densities, a lower number of iterations to reach convergence.

V. Illustration of the Operation of the Sparse Iterative Method

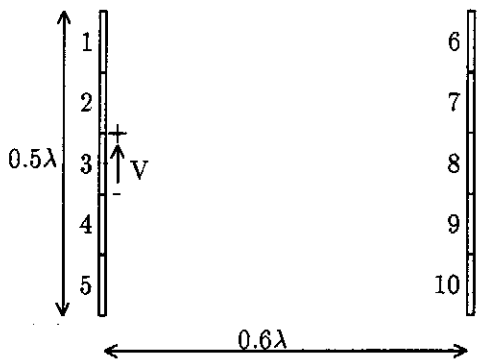

Fig. 2. Two dipole example to illustrate the operation of the Sparse Iterative Method

The SIM is best illustrated using a two dipole example (figure 2). Each dipole consists of five 0.1λ segments and are spaced 0.6λ apart. The zero interaction distance, d_0, was set to 0.5λ, and the absolute wavelength was 10 m (29.98 MHz). The segment numbering is indicated in figure 2; the first dipole was excited with a 1 V applied E-field source in the center (segment 3). This example is hence simplified, in the sense that all the interactions on each dipole are included in the S-matrix, but no interactions between the two dipoles are included, since they are spaced at more than d_0 from each other. The format of the $\mathbf{SI}_0 = \mathbf{V}$ equation is shown in figure 3.

The first iteration will solve for the current on the left dipole (segments 1–5), but since interactions to the right dipole are omitted, no current will be induced on segments 6–10. Figure 4 shows this current in the bottom graph.

Figure 4 also shows the tangential E-field error, versus position on the antennas, for each iteration. After the first iteration, no E-field error is present on the left dipole—all interactions on that dipole were represented in the S-matrix, and the boundary conditions satisfied by normal method of moments operation. Large E-field errors exist on the parasitic dipole, since no interactions to that antenna were taken into account. These values are then subtracted from the excitation vector for the next estimate for current,

$$\begin{bmatrix} Z_{1,1} & Z_{1,2} & Z_{1,3} & Z_{1,4} & Z_{1,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{2,1} & Z_{2,2} & Z_{2,3} & Z_{2,4} & Z_{2,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{3,1} & Z_{3,2} & Z_{3,3} & Z_{3,4} & Z_{3,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{4,1} & Z_{4,2} & Z_{4,3} & Z_{4,4} & Z_{4,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{5,1} & Z_{5,2} & Z_{5,3} & Z_{5,4} & Z_{5,5} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Z_{6,6} & Z_{6,7} & Z_{6,8} & Z_{6,9} & Z_{6,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{7,6} & Z_{7,7} & Z_{7,8} & Z_{7,9} & Z_{7,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{8,6} & Z_{8,7} & Z_{8,8} & Z_{8,9} & Z_{8,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{9,6} & Z_{9,7} & Z_{9,8} & Z_{9,9} & Z_{9,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{10,6} & Z_{10,7} & Z_{10,8} & Z_{10,9} & Z_{10,10} \end{bmatrix} \begin{bmatrix} I_0(1) \\ I_0(2) \\ I_0(3) \\ I_0(4) \\ I_0(5) \\ I_0(6) \\ I_0(7) \\ I_0(8) \\ I_0(9) \\ I_0(10) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Fig. 3. Format of $\mathbf{SI_0} = \mathbf{V}$

$$\begin{bmatrix} Z_{1,1} & Z_{1,2} & Z_{1,3} & Z_{1,4} & Z_{1,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{2,1} & Z_{2,2} & Z_{2,3} & Z_{2,4} & Z_{2,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{3,1} & Z_{3,2} & Z_{3,3} & Z_{3,4} & Z_{3,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{4,1} & Z_{4,2} & Z_{4,3} & Z_{4,4} & Z_{4,5} & 0 & 0 & 0 & 0 & 0 \\ Z_{5,1} & Z_{5,2} & Z_{5,3} & Z_{5,4} & Z_{5,5} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & Z_{6,6} & Z_{6,7} & Z_{6,8} & Z_{6,9} & Z_{6,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{7,6} & Z_{7,7} & Z_{7,8} & Z_{7,9} & Z_{7,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{8,6} & Z_{8,7} & Z_{8,8} & Z_{8,9} & Z_{8,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{9,6} & Z_{9,7} & Z_{9,8} & Z_{9,9} & Z_{9,10} \\ 0 & 0 & 0 & 0 & 0 & Z_{10,6} & Z_{10,7} & Z_{10,8} & Z_{10,9} & Z_{10,10} \end{bmatrix} \begin{bmatrix} I_1(1) \\ I_1(2) \\ I_1(3) \\ I_1(4) \\ I_1(5) \\ I_1(6) \\ I_1(7) \\ I_1(8) \\ I_1(9) \\ I_1(10) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 - E_0^1(1) \\ 0 - E_0^1(2) \\ 0 - E_0^1(3) \\ 0 - E_0^1(4) \\ 0 - E_0^1(5) \end{bmatrix}$$

Fig. 5. Format of $\mathbf{SI_0} = \mathbf{V}$, after first iteration.

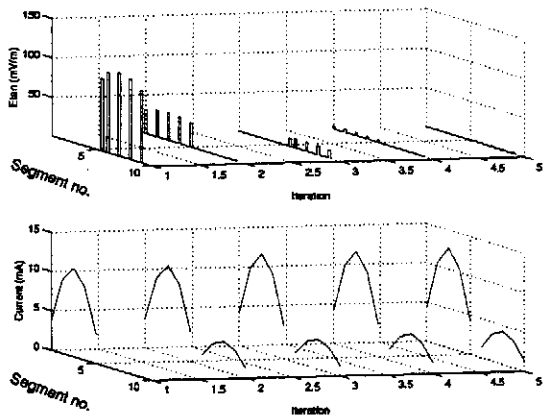

Fig. 4. The current and E_{tan} error on the two dipole array example after every iteration

yielding the matrix equation shown in figure 5

Solving with this modified excitation vector yields the current and E-tangential error results for iteration 2 (see figure 4). A current is now established on the parasitic dipole with zero E-tangential errors. The reason why the tangential E-field errors are zero is because they were counter-acted by error sources that were equal and opposite to the tangential errors that were found in the previous iteration. In this iteration, the effect that the parasitic dipole current has on the source dipole is neglected, hence there will be E-tangential errors on the source dipole. A new excitation vector is formulated to negate these errors. Figure 4 illustrates how this process results in a current distribution which satisfies the boundary conditions after a few iterations.

VI. Results

This section commences with results for linear dipoles of varying electrical length. The results for this rather trivial problem demonstrate the computational advantages inherent in the SIM, compared to normal LU-decomposition. Thereafter, results for a $2\lambda^2$ square grid (544 segments) and a simple missile structure (410 segments) are presented. These more realistic examples verify the operation of the SIM for complex problems and demonstrate the computational time advantage of the SIM over LU-decomposition. The SIM computational speed advantage will increase with an increase in problem size.

All computer time results are for a Intel 486 processor running at 33 MHz and equipped with 256 kByte cache memory. The two methods both run within the same C++ NEC2 equivalent program with a software switch to select whether LU-decomposition or the SIM was to be used. The redesigned program [1] was compiled using the GCC public domain C++ compiler with optimisation. No manual effort was made to optimise either of the two routines numerically or computationally.

A. Dipoles

The dipole length was continually adjusted to ensure that segment length is maintained at $\lambda/10$. The 500 seg-

TABLE I

TABLE OF RESULTS FOR DIPOLE PROBLEMS (ALL SIMULATIONS WERE DONE WITH INTERACTION DISTANCE OF 2λ)

segments	k	S-den -sity(%)	S-factor (s)	SIM-iterate (s)	SIM Total (s)	LU-factor (s)
100	1	36	1	0.25	1.25	3
200	4	19	1	2	2	35
300	5	13	1	5	6	87
400	6	10	1	10.2	11.2	206
500	7	8	3	17.8	20.8	403
600	8	6.5	4	28.8	32.8	697
700	9	5.6	5	44.1	49.1	1106
800	9	5.0	7	58	65	1650
900	10	4.4	9	82	91	2350
1000	11	4	10	112	122	3224

ment dipole, for instance, will be 50λ in length. The dipoles were all fed on the center (or just off center) segment with an applied field voltage source. Table I gives detailed results for these simulations. The matrix fill times are not included—these are the same for LU-decomposition and the SIM. Figure 6 shows the computer time comparison between LU-decomposition and the SIM. The two components in the SIM solution, sparse matrix factoring and iteration time, are also shown in figure 6.

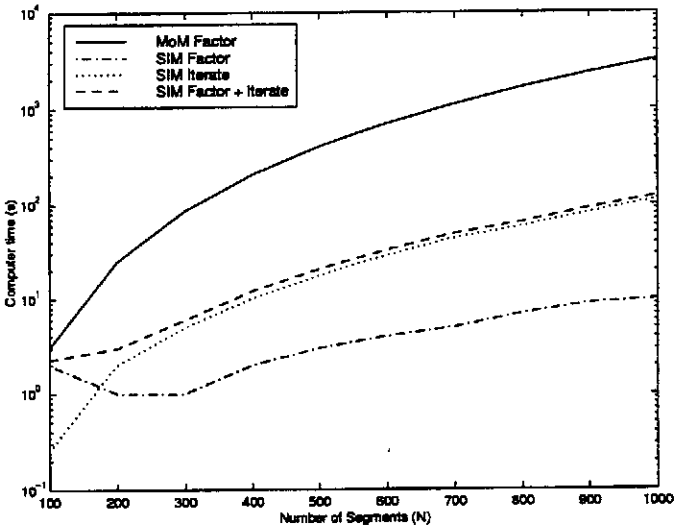

Fig. 6. Computer time comparison between the SIM and LU-decomposition solutions on a Intel 486-33 computer. SIM results are broken down in terms of the component for sparse matrix factoring and for total iteration time.

The values in figure 6 were obtained using a constant $d_0 = 2\lambda$. There is a tradeoff between the time spent on factoring the sparse matrix and the time associated with iteration to obtain the final solution. Figure 7 indicates the computer times required for factoring **S**, for iteration and the total solve time, all plotted versus d_0.

Figure 7 indicates that the SIM factor time increases with increasing d_0, since the matrix density increases. The iteration time, on the other hand, reduces with increasing

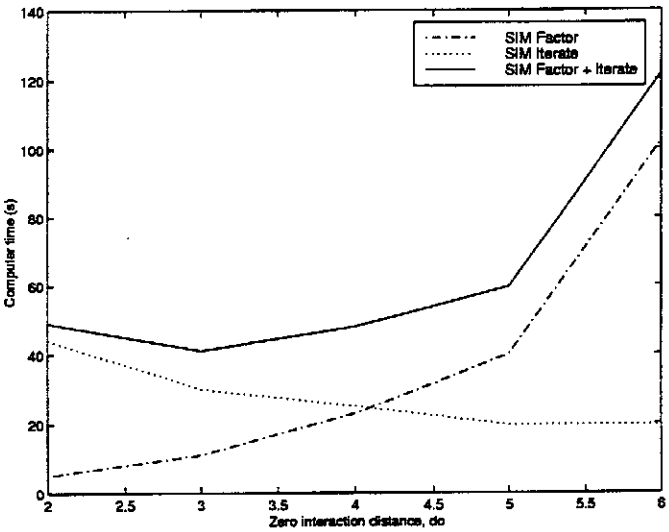

Fig. 7. Computer time versus d_0 for factoring, iteration and the combination in the SIM for a 700 segment dipole

d_0, since more interactions are taken into account in the **S**-matrix and requires fewer corrections to achieve convergence. Given that the factor time increases faster than the corresponding reduction in iteration time with increasing d_0, it is clear that the optimum is close to the minimum d_0 required for stable solutions.

For problems with a large number of unknowns, table I shows a dramatic decrease in solution time using the SIM, as opposed to LU-decomposition. The speed-up factor at 1000 segments is around 30, and this value will continue to increase for larger problems, since LU-decomposition follows a N^3 time dependence whereas the SIM is essentially limited by a N^2 time dependence. The following two sections show results for grid problems.

B. Grid Patch

A wire grid patch, with dimensions and segment numbering similar to an example presented in the GEMACS manual [8], was analysed - the geometry is shown in figure

TABLE II
TABLE OF RESULTS FOR GRID PATCH

$d_o(\lambda)$	S-density	k	PRE (%)	S-factor	SIM-iterate	SIM Total	LU-factor
0.9	39	4	0.29	458	25	483	733
0.65	24.2	3	0.64	285	20	305	733
0.4	10.6	8	0.88	170	42	212	733

8. It was hoped to compare SIM to GEMACS using other geometries, but the program is unfortunately not available due to US military restrictions.

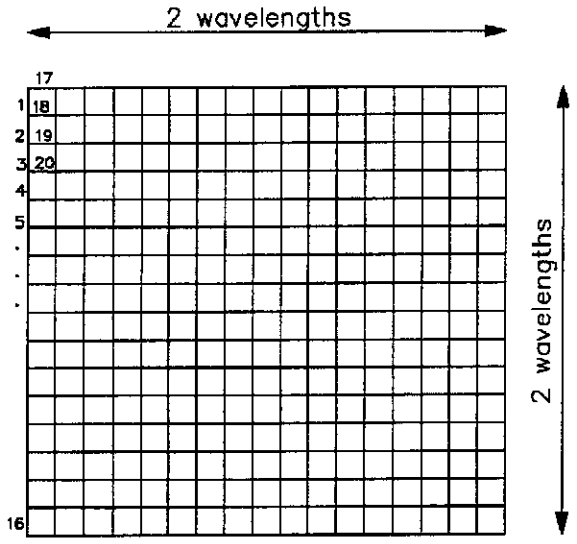

Fig. 8. The 544 segment wire grid patch problem

The grid was excited with a plane wave with a direction of propagation broadside to the patch and a polarisation of 45° relative to the patch edges. Table II shows the comparison between the SIM and LU-decomposition or various problem parameters. In this case, iterations were continued until a $PRE < 1\%$ was achieved. This value is quite accurate for most electromagnetic problems [8], but typically, values close to machine precision were obtained with 3 or 4 more iterations.

The speed increase for this problem, relative to LU-decomposition, for an interaction distance of 0.4λ, is approximately 3-4 times faster. This speed-up is much less than the approximately 20 times increase for a dipole problem with the same number of segments. Wire grid problems have a higher number of segments within the zero-interaction distance; the sparse matrix will therefore be considerably denser for a grid problem when compared to a linear dipole. As the size of the grid problems increase, so the difference between the execution time of the SIM and the LU-decomposition based techniques will become larger. The SIM is hence not inherently less successful with grid problems, but rather just requires electrically larger problems for its full potential to be achieved. Our current implementation on a 486 personal computer does not allow larger problems to be evaluated at this stage.

C. Missile

A missile example drawn from [6] was used to illustrate the simulation of a more complex problem. Figures 9

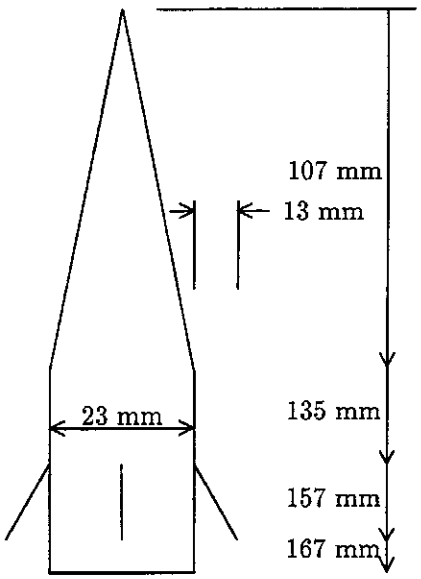

Fig. 9. The geometry of the missile geometry used as test case at a frequency of 3 GHz

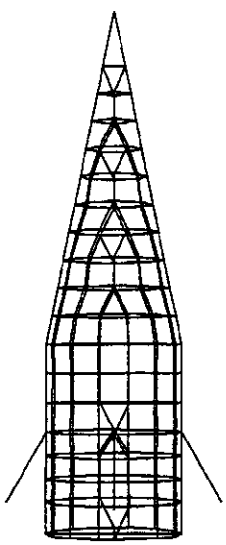

Fig. 10. The gridded 410 segment missile problem

and 10 show the actual and segmented geometry. Segments which were approximately 0.1λ in length were used, and

TABLE III

TABLE OF RESULTS FOR MISSILE EXAMPLE

$d_o(\lambda)$	S-density	k	PRE (%)	S-factor	SIM-iterate	SIM Total	LU-factor
0.35	25.7	6	0.65	42	13	55	194

wire radii were chosen to ensure a wire surface area equal to twice the actual surface area of the missile. Applied E-field voltage sources were placed on the 4 slanting wires and the 410 segment problem was analysed using both LU-decomposition and the SIM.

Table III gives the pertinent values for parameters and times for this problem. The SIM once again produced result about 4 times faster than LU-decomposition. The points raised in the previous section concerning the density of the sparse matrix also apply to this problem.

To illustrate the difference in the results produced by the two techniques, the input impedances from both methods were recorded.

LU-decomposition: 16.2458-j0.06050 Ω
SIM: 16.2432-j0.05737 Ω

VII. CONCLUSION

A sparse iterative method for replacing LU-decomposition in MoM problems has been presented; the method is based on the formulation of a sparse matrix through neglecting those matrix elements which represent coupling smaller than a predefined value.

Correction for neglecting many small couplings is then performed by calculating the tangential E-field at match points, and applying these as equal and opposite excitations to the structure. Performing this procedure iteratively results in a solution which is forced to satisfy the boundary conditions at the match points, up to any specified degree of accuracy - including that of LU-decomposition. Even when enough SIM iterations are performed such that the currents are as accurate as those obtained from LU-decomposition, the SIM still executes faster.

The convergence of the method is somewhat dependant on the particular geometry, but in all cases a 10^{-3} limit on $E_{\tan}$ guaranteed convergence. It was shown that in some cases this can be relaxed to 10^{6}.

Simple comparative cases were presented to indicate computer time related aspects, and some more complex two- and three-dimensional cases were also analyzed to demonstrate the general applicability of the SIM.

REFERENCES

[1] D. C. Nitch, *A serial and parallel design of NEC2 to demonstrate the advantages of the object-oriented paradigm in comparison with the procedural paradigm.* PhD thesis, University of the Witwatersrand, PO Box 3, Johannesburg, 2050, South Africa, 1993.

[2] G. J. Burke and A. J. Poggio, "Numerical electromagnetics code (nec) - method of moments. parts I-III.," tech. rep., Naval Ocean Systems Center (NOSC), January 1981.

[3] T. T. Ferguson, T. H. Lehman, and R. J. Balestri, "Efficient solution of large moments problems: Theory and small problem results," *IEEE Transactions on Antennas & Propagation*, vol. AP-24, pp. 230–235, March 1976.

[4] P. J. Baldwin, A. Boswell, D. Brewster, and J. Allwright, "Iterative calculation of ship-borne hf antenna performance," *IEE Proceedings-H*, vol. 138, pp. 151–158, April 1991.

[5] T. K. Sarkar, K. R. Siarkiewicz, and R. F. Stratton, "Survey of numerical methods for solution of large systems of linear equations for electromagnetic field problems," *IEEE Transactions on Antennas & Propagation*, vol. AP-29, pp. 847–856, November 1981.

[6] D. B. Davidson, *Parallel Algorithms for Electromagnetic Moment Method Formulations.* PhD thesis, University of Stellenbosch, 1991.

[7] F. X. Canning, "The impedance matrix localization (iml) method for moment-method calculations.," *Antennas and Propagation Magazine*, pp. 18–30, October 1990.

[8] D. L. Kadlec and E. L. Coffey, "General electromagnetic model for the analysis of complex systems (GEMACS). engineering manual. (version 3)," Tech. Rep. RADC-TR-83-217, Vol II (of three), Rome Air Development Center (RBCT), 1983.

[9] K. S. Kundert and A. Sangiovanni-Vincentelli, *Sparse User's Guide, A Sparse Linear Equation Solver, Version 1.3a.* Department of Electrical Engineering and Computer Sciences; University of California, Berkeley, April 1988.