LA-UR 93-428

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

LA-UR--93-428

DE93 007326

TITLE A PROPOSED APPROACH FOR DEVELOPING NEXT-GENERATION COMPUTATIONAL ELECTROMAGNETICS SOFTWARE

AUTHOR(S): Edmund K. Miller
R. P. Kruger
Steve Moraites

SUBMITTED TO: ACES 93 Symposium

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO. 836 R4
ST. NO. 2629 5/81

A PROPOSED APPROACH FOR DEVELOPING NEXT-GENERATION COMPUTATIONAL ELECTROMAGNETICS SOFTWARE

E. K. Miller*, R. P. Kruger*, and
Steve Moraites**

*Los Alamos National Laboratory, USA
**Simulated Life Systems, Inc., Chambersburg, PA

PRESENTED AT THE 9'th ANNUAL
ACES
REVIEW OF PROGRESS
IN
APPLIED COMPUTATIONAL ELECTROMAGNETICS
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CA
MARCH 1991

A PROPOSED APPROACH FOR DEVELOPING NEXT-GENERATION COMPUTATIONAL ELECTROMAGNETICS SOFTWARE

E. K. Miller, Group MEE-3; R. P. Kruger, Group A-10
Los Alamos National Laboratory, Los Alamos, NM 87545
Steve Moraites, Simulated Life Systems, Inc., Chambersburg, PA

ABSTRACT

Computations have become a tool coequal with mathematics and measurements as a means of performing electromagnetic analysis and design. This is demonstrated by the volume of articles and meeting presentations in which computational electromagnetics (CEM) is routinely employed to address an increasing variety of problems. Yet, in spite of the substantial resources invested in CEM software over the past three decades, little real progress seems to have been made towards providing the EM engineer software tools having a functionality equivalent to that expected of hardware instrumentation. Furthermore, the bulk of CEM software now available is generally of limited applicability to large, complex problems because most modeling codes employ a single field propagator, or analytical form, of Maxwell's Equations. The acknowledged advantages of hybrid models, i.e., those which employ different propagators in differing regions of a problem, are relatively unexploited.

The thrust of this discussion is to propose a new approach designed to address both problems outlined above, integrating advances being made in both software and hardware development. After briefly reviewing the evolution of modeling CEM software to date and pointing out the deficiencies thereof, we describe an approach for making CEM tools more truly "user friendly" called EMSES (Electromagnetic Modeling and Simulation Environment for Systems, named selected in collaboration with Kenneth Siarkiewicz of RADC). This will be achieved through two main avenues. One is developing a common problem-description language implemented in a visual programming environment working together with a translator that produces the specific model description needed by various numerical treatments, in order to optimize user efficiency. The other is to employ a new modeling paradigm based on the idea of field propagators to expedite the development of the hybrid models that are needed to optimize computation efficiency. By nature of its design, EMSES will be highly modular, hence more portable, and will exploit progress being made in "scaleable" libraries to maximize performance in advanced parallel computational environments.

COMPUTATIONAL ELECTROMAGNETICS

The analysis and design of new materials, subsystems, and systems with specific electromagnetic requirements has led to research into the use of electromagnetic modeling codes on parallel processors. This research has been conducted for several years with the apparent conclusion that electromagnetic codes generally map well onto a wide range of machine architectures [Calalo (1987), Perlik and Moraites (1992), Russell and Rockway (1991), Davidson (1991)]. Both integral-equation (IE) and differential-equation (DE) methods map with high parallel efficiency onto such machines as the CM2 connection machine, the JPL hypercube, and the Cray YMP8. While no limitation of parallel architectures or parallel EM algorithms has been observed, several impediments to full exploitation of new machines have arisen.

A major limiting factor or impediment in achieving more useful and productive CEM CAD (Computer-Aided Design) tools remains the computation resource required, as parallel architectures at best offer quantitative speedups only in proportion to their increased throughput as opposed to qualitative speedups that alternate formulations might be hoped to provide. Or, as observed by Wandzura (1992) in a recent talk, the former is "evolutionary" while the latter would be "revolutionary." To illustrate both qualitatively and quantitatively the computation-resource problem, Fig. 1 shows how the computational requirements increase with modeling accuracy and the frequency of interest.

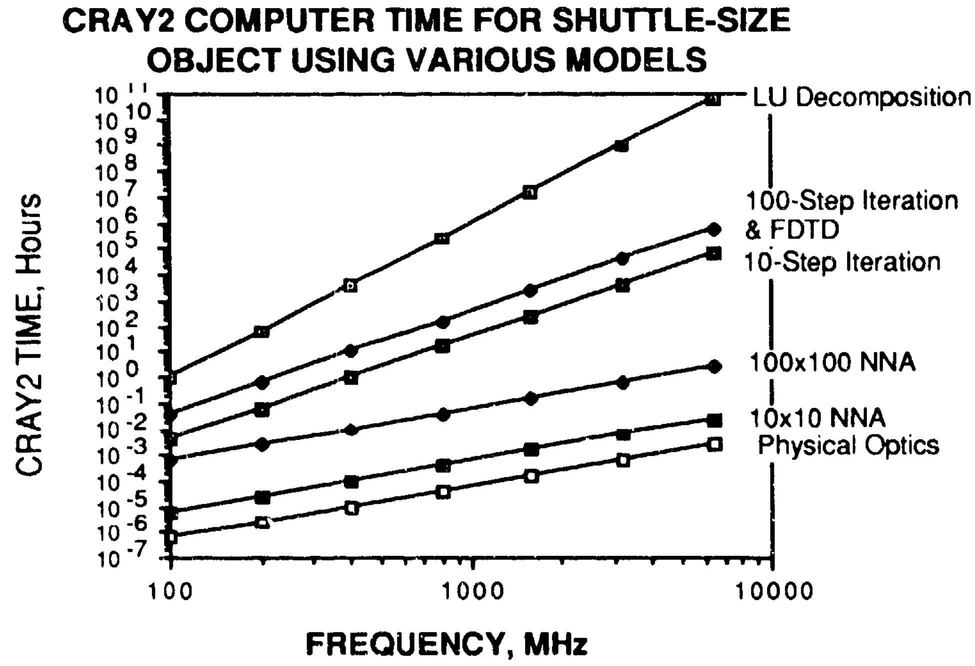

Figure 1: Illustrated here is the effect of frequency on the computational requirements [Miller (1991)]. In electromagnetics, an object size is measured in wavelengths, which is inversely proportional to frequency. Increasing frequency is equivalent to increasing the size of an object. A linear increase in either can cause a much larger rate of increase in the number of operations required to achieve a solution. The operation count for present and anticipated future models increases from the 2nd to as much as the 9th power of the frequency for three-dimensional problems, depending on the specific analysis method used. Note that the operation count for LU decomposition of an IE matrix increases as the 6th power of frequency for a surface-sampled object.

Another impediment to furthering large-scale CEM is that of preparing the input for a problem worthy of a teraflop computer and subsequently making use of the resultant massive amounts of data. Still a third impediment arises as a result of the improved performance of CEM tools where high-accuracy EM solutions for more complex problems such as low-observable targets might be nullified by effects of structural and thermal stress. The CEM software which is genuinely useful today must interface to at least thermal and structural analysis software for many applications. Finally, one of the most critical impediments to progress in CEM is multifaceted but is based primarily on the very rudimentary, one-of-a-kind, user interface that pervades present day CEM software. Rectifying this problem requires development of a standardized interface that provides a modeler access to all of the most widely used CEM tools. Overall, mitigating the impact of these impediments is best met by developing the integrated modeling infrastructure EMSES as illustrated generically in Fig. 2. A system such as EMSES is needed to not only open access to advanced CEM software to geographically remote users but also to permit continual and future additions, modifications, and program control while also providing other capabilities in such areas as verification and validation as is discussed further below.

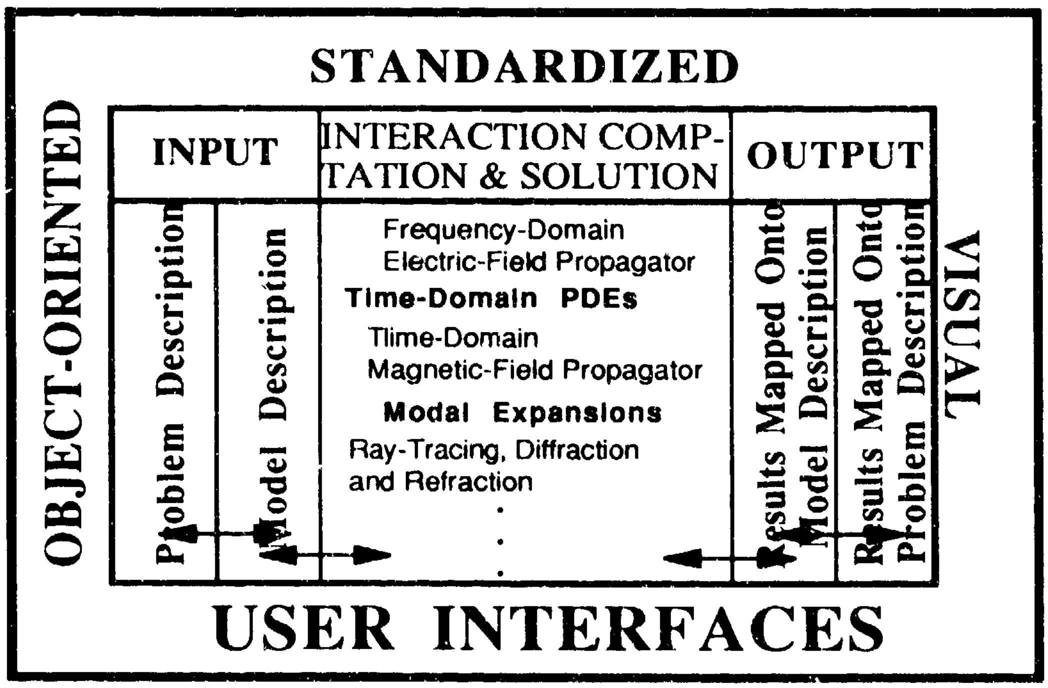

Figure 2: Conceptual block diagram for EMSES to illustrate its modularity with respect to using a field-propagator paradigm and its key components.

EMSES would be best developed as the scaleable software of the future in CEM for grand-challenge computing utilizing two complementary and parallel approaches. One approach will be to improve CEM software with respect to modeling performance and capabilities, and the other will be to better exploit continuing advances in computer hardware, especially in parallel, scaleable and distributed computing. Each area is discussed in turn below.

ADVANCES IN CEM MODELING SOFTWARE

Progress (i.e., solving bigger, more complex problems) in CEM is predicated on advances in solution speed and accuracy commensurate with advances in computing hardware on the one hand and user effort on the other. Driving user effort is the fact that the typical electromagnetics engineer, most often someone who has not written any of the software being used, must become familiar enough with several different software packages that each can be used with some minimum facility. Unfortunately, existing software usually has a limited interface and is generally incapable of incorporating improvements made to other functionally similar software. This situation might be unfavorably compared with the status of hardware instrumentation development. In the latter case, even the most complex instrumentation can be reliably used by someone familiar only with its functionality and application, i.e., the user is not required to know how to design an instrument in order to use it. We suggest that an equivalent approach is needed in designing the next generation of CEM modeling software, and that this might proceed by beginning with developing a model of the modeling process itself, for which the structure shown in Fig. 2 might provide a starting point.

Reducing Computational Complexity and Exploiting Special Hardware

Work to date on reducing model complexity (the number of operations required to achieve acceptable accuracy) spans the spectrum from being either primarily analytical to primarily numerical, or some intermediate combination thereof [Miller (1988) (1991)]. Among the more analytical approaches are the Fast Multipole Method [Engheta et al. (1992)] and the various high-frequency, asymptotic techniques [Stone (1990)]. In the former, the number of mutual interactions, M, needed to be included in an IE like model is reduced from of order N^2 to of order NlogN by representing the "faraway" interactions using multipole expansions together with a fast-Four transform (FFT). Asymptotic techniques such as the Geometrical Theory of Diffraction and a number of other variations reduce problem complexity by avoiding the need to solve for the current on an object. They instead deal only with the fields caused by specular reflection, refraction at edges, energy shedding on curved surfaces, or diffraction at dielectric interfaces.

Additional numerical approaches include those based on impedance-matrix localization [Canning (1990)]; spatial decomposition [Umashankar et. al. (1990)]; space segmentation [Wang and Ling (1991)]; diakoptics [Butler (1990)]; fast-propagation solutions [Miller and Gilbert (1991)]; various FFT-based procedures [Sarkar et. al. (1986)]; and multigrid methods [Kalbasi (1991)]. Almost all of these also use iterative matrix-solution techniques to reduce solution of a general matrix from being proportional to N^3 to of order IM, where I is the number of iterations required for an acceptably converged solution. All such techniques share the goal of reducing a model's complexity by altering the problem's description through using special basis and testing functions.

This divides the problem into parts that are more easily solved by taking advantage of special features the problem might possess to thereby develop a more efficient matrix solution. Other numerically oriented approaches for reducing complexity include the various time-domain DE models [Taflove (1988)] whose primary advantage is that, in their explicit formulation, they are solvable without matrix inversion.

As designing and building special computers for solving certain kinds of problems are becoming more practical, we are beginning to see a blurring between software and hardware design. This development might be typified by computers such as the WaveTracer computer [Miller (1990)] which was designed with particular kinds of DE models in mind. Upon noting the close analogy between signal processing and filtering and such DE models, new hardware paradigms are being developed, one example of which is represented by the "Wave Digital Filter," [Kuo and Levy (1990), Fettweis and Nitsche (1991)].

A Field-Propagator Paradigm for Electromagnetic Modeling

We note that at some point in the process, all electromagnetic modeling involves evaluating the fields caused by specified sources. When the sources are known, the problem is more straightforward, an example being to find the radiation pattern of an antenna. Most often the sources are unknown and are found as the solution of a boundary-value problem with boundary conditions imposed on the fields due to these sources. The source-field relationship, or field propagator, that is employed in this process may be based on:

1) the Maxwell curl equations written in differential or integral form to yield what are called finite-difference and finite-element models;

2) a Green's function and source integral to produce an IE or boundary-element formulation;

3) a mode-based description which leads to techniques such as the T-Matrix and Generalized Multipole Techniques or;

4) rays and diffraction coefficients which lead to an optics model.

The vast majority of present CEM modeling software is based wholly on using only one of these approaches for a particular problem. However, it is well known that the applicability of each is limited, and that models for more general problems should employ that field propagator which it is best suited to each subset of the overall problem. Effective implementation of such a hybrid model requires that a new modeling paradigm be employed that recognizes this need, since hybridization offers the only means by which larger and more complex problems can be successfully solved.

We suggest that CEM modeling-software development be reoriented so that field propagators are explicitly incorporated as its most basic ingredient. This means that, whatever the kind of modeling code is under consideration, the building blocks needed for its development and application are formulated and employed as field propagators. These propagators will be written as modular, scaleable, software-library elements that can be easily linked together in a systematic, dataflow oriented, and visual manner. This will greatly simplify developing the source-field relationship of a problem that is geometrically or electrically complex. The input to each propagator will be an appropriate source while its output will be a transformed field produced by a combination of source and propagator. The spatially (and possibly temporally) discrete set of field and source samples that result will generate sets of equations by imposing needed field continuity at common boundaries or in common regions of the separate propagators. Modeling a complex problem thus becomes a process of identifying the propagator types to be assigned to each region of the problem and the boundaries across, or volumes within, of propagator interaction. The computation then proceeds by assembling a set of equations for each spatial region. The collection of all such regions produces the final matrix, generally a combination of dense and sparse matrices because of using different propagators in different regions, that will model the entire problem.

We also suggest that this new approach should permit variations in the numerical treatment by expressing the propagators in a uniform and predefined way. Among the variations to be included would be basis and testing functions employed, model adaptation, and matrix-solution procedures. Allowable variations for a particular problem would depend on the resources available to the modeler on the distributed computational network. Thus, as we expand the computational network resources, the modeling algorithm can also grow and provide, for example, user choices to be made concerning numerical accuracy, spatial resolution, or the density of frequency and angle sampling.

The propagator paradigm approach proposed for EMSES is an inherently modular one. Propagator modules would provide the electric and/or magnetic fields or potentials needed for various single-propagator or multiple-propagator (hybrid) models. For example, one set of libraries would model the frequency-domain electric fields for filamentary, surficial and volumetric electric currents. Other library modules would employ differential, modal, and high-frequency propagators. These propagator modules can also be designed to provide the fields for various kinds of basis and testing functions as selected by the modeler using interactive decision aids.

As a specific example, the integration required to obtain a field involves summing weighted values of the IE kernel function. The subsequent integration required for the field testing involves another weighted summation of similar nature. Thus, the propagator evaluation for integral equations ultimately requires only weighted sums of kernel-function samples. Furthermore, the field samples required for the impedance matrix involve sampling a relatively well-defined parameter space. This provides the opportunity for pre-computing and storing fields in some suitable way so that much of the one-time computing cost of certain problem classes can be subsequently avoided. This approach is know as "function approximation" and "model-based parameter estimation" and is instrumental in reducing the cost of evaluating the Sommerfeld integrals needed for modeling an object near a planar interface by a factor of up to a thousand [Burke and Miller (1984)].

The process of computing fields in an IE context provides the coefficients for an interaction (impedance) matrix. We call this the "system" matrix for, depending on the formulation used, the coefficients will not all have the units of impedance, as they do for the electric-field IE. The system matrix is "assembled" by evaluating the fields of the various propagators that might be employed. Each row of the matrix arises from imposing some boundary condition or continuity condition at various points in the problem space. We note that this kind of modeling-code decomposition is well-suited for interfacing with a user decision aid.

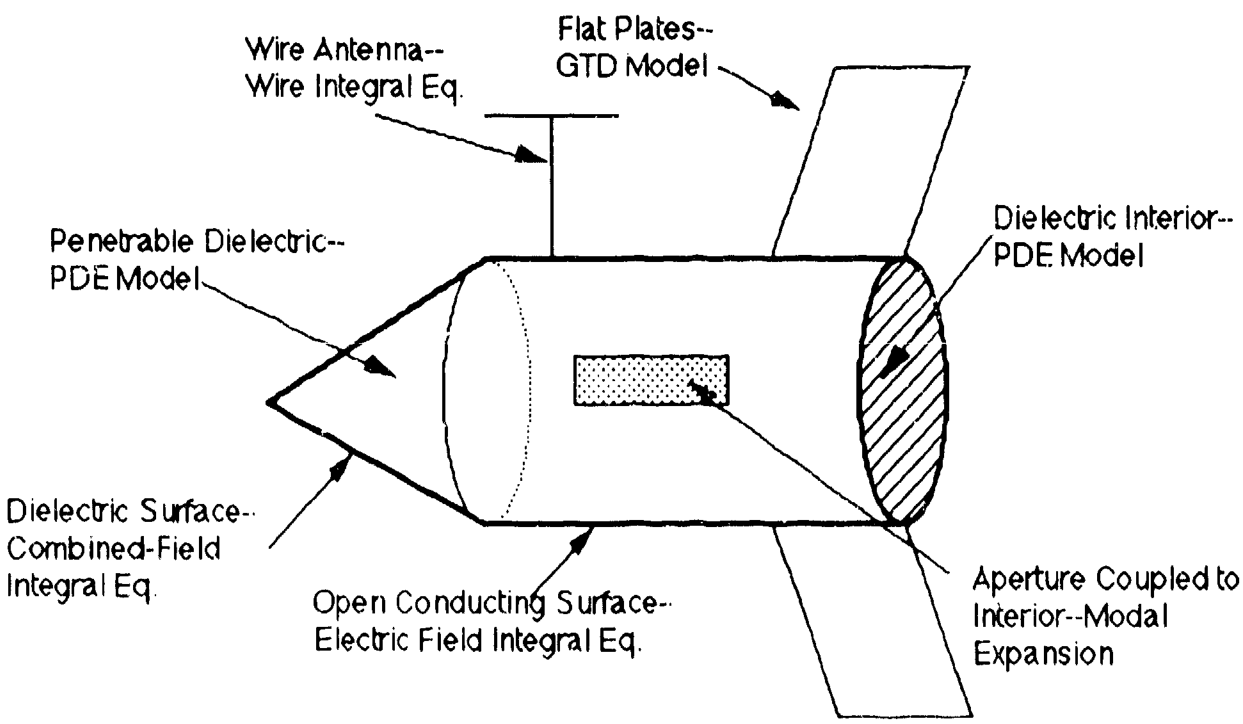

Figure 3. Example of generic problem for a propagator-based, hybrid approach could provide a more efficient model than would one based on a single field propagator.

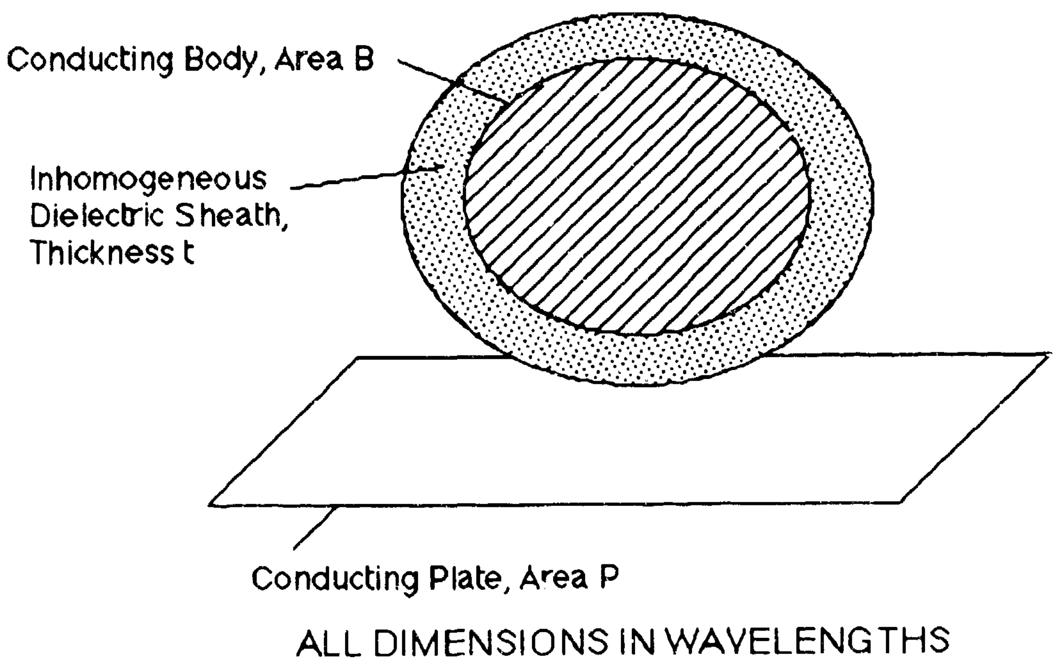

Figure 4a.

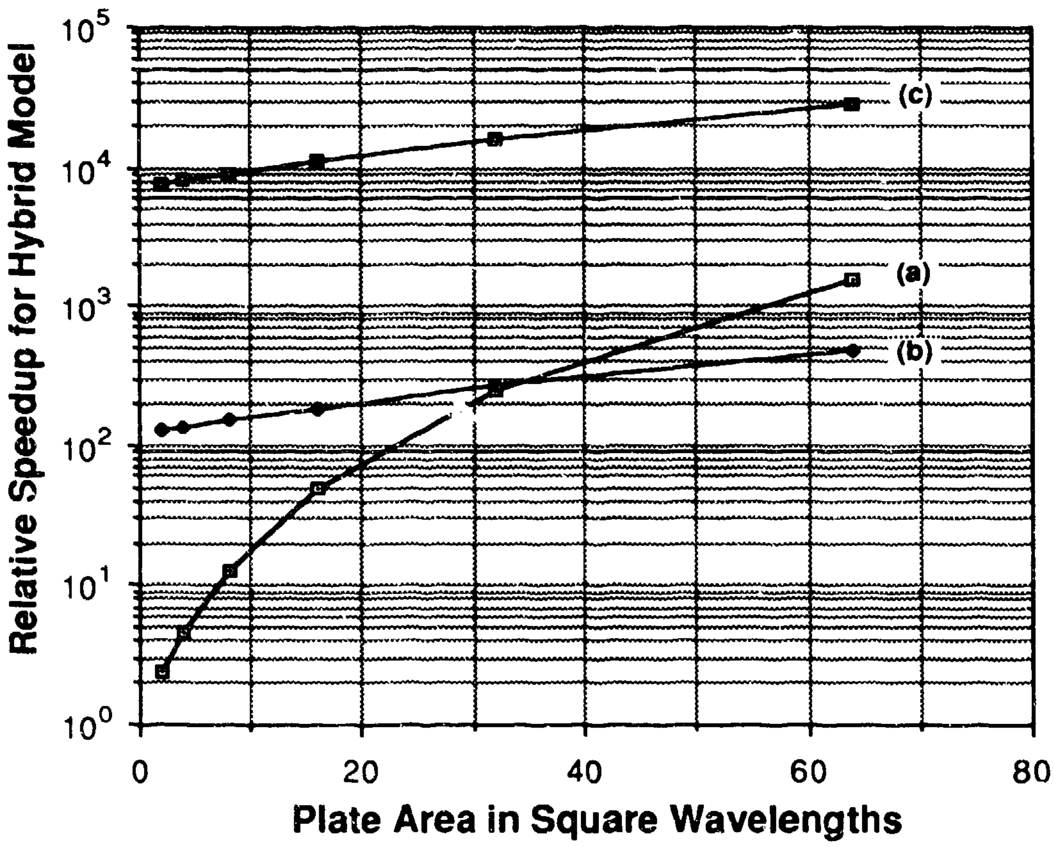

Figure 4b: Some results for a simple problem to illustrate the potential operation-count advantage of hybrid models. For the problem geometry depicted in Fig. 4a, the result for curve (a) applies where t = 0 (i.e., then is no sheath), and for (b) and (c) where an inhomogeneous sheath is present. Curve (a) demonstrates the relative speedup [given by $(1+P/B^3]$ achieving using a hybrid model [IE (IE) for the object with geometrical-theory of diffraction (GTD) for the plate-object interaction] over an IE model for both, when solving the impedance matrix using LU decomposition. Curve (b) shows the speedup [given by $(10t+1+P/B^2]$ achieved by a hybrid IE-GTD-PDE (partial-differential equation) model over an all-IE model using an iterative solution for both, when an inhomogeneous sheath covers the object. Finally, in curve (c) we demonstrate the speedup [given by $\{10B(10t+1+P/B)^2\}/t\}$ achieved by a hybrid Modal-GTD-PDE model over an all-IE, again solved using iteration. In the latter case, we sample only those surface fields whose mode numbers extend from 0.9 to 1.1 ka, where a is the effective sheath radius. We assume that the sampling density is 10 per wavelength in linear dimension (i.e., 100 per square wavelength and 1000 per cubic wavelength), that the object area is six square wavelengths, and for (b) and (c) that the integrated sheath thickness is one wavelength.

Developing Hybrid Models Using Field Propagators

After the modeler has developed a physical problem description for the application of interest, a decision aid will be used to analyze that description and provide a suggested list of propagators for each different portion of the problem. It will also help in selecting from among the set of available options, the modeling details that are best suited for each particular spatial region. For example, a large, smooth, conducting segment would best use an IE model that employs entire-domain bases, whereas a region of spatially varying dielectric would best employ a finite-element DE model. An illustration of this type of problem decomposition is shown in Fig. 3. A simplified example of the value of using such a method for a simple problem is shown in Fig. 4 where all curves show the speedup of the proposed type of algorithm under various conditions compared to an IE modeled using LU decomposition.

The properties of the system matrix for a problem will depend on whether the modeling is done in the time domain or frequency domain, and the kinds of propagators chosen. Therefore, the subsequent numerical solution of this matrix must reflect these differences. It is envisioned that EMSES will include LU decomposition, iterative, and various sparse-Matrix solution procedures which will provide solution options appropriate for given applications. For example, when an antenna problem is modeled, only one excitation or "right-hand side" is needed, so that an iterative solution would almost always be more efficient than LU decomposition. When a radar-cross section (RCS) is needed for many angles of incidence on the other hand, LU decomposition could be more efficient since an iterative solution may need approximately the same number of iterations for each new incidence angle. There may be some potential in this latter application, however to use the most recent solution as the starting point for a new angle of incidence and thus potentially reduce the number of iterations needed for convergence, which could make iteration more appropriate then. These kinds of options will be provided in EMSES to give the user a convenient, easily exercised menu of choices.

Model Adaptation for Error Control

Another potentially important means of improving computation efficiency is provided by adaptive methods. Almost all modeling in CEM currently employs predefined models where the number of unknowns is selected based on experience and modeling guidelines. Not until the computation is finished does the user normally obtain any quantitative indicator of how accurate or numerically converged are the results. If we determine that more spatial unknowns are needed to achieve the accuracy desired, the entire problem normally needs to be redone using a more refined model description.

Model adaptation in EMSES might be achieved by including a capability for checking model performance as the computations are being performed. A field propagator that is especially well-suited to making this feasible is one based on modal expansions of a field due to multipole sources. Present implementations of modal propagators do not require the usual surface-source discretization. They require only that the fields be sampled on boundary surfaces. Consequently, it is numerically efficient to solve a problem using a modal propagator for a given number of unknowns and then check boundary-field errors. If they are too large, new unknowns can be added and more field samples used in regions where the boundary errors are largest. This can be accomplished in a recursive fashion without discarding the first solution which serves as a starting point for the updated solution. Since there is no source-discretized approximation of the problem to be refined, the problem of using more sources and fields is greatly simplified. An IE model that employs entire domain (e.g., a Fourier series) over all or part of a problem boundary can be made adaptive in a similar fashion. The benefit of reducing the boundary error, which is normally the controlling factor in determining the overall solution accuracy, only enough to achieve the needed observable accuracy, will be substantial. This idea is illustrated conceptually in Fig. 5.

Verification and Validation

Aside from the work required by a user to prepare the input needed to exercise a computer model and access the output it produces, perhaps the greatest integrated effort associated with CEM is that of verifying code operation and validating the results produced. Verification is associated with determining that a modeling code produces results consistent with its design. Validation is concerned with establishing how well its results conform to physical reality. Both are ingredients essential to performing reliable modeling computations. The former is a necessary, but not sufficient, condition for acceptable code performance, while the latter determines how reliably a given code can be applied to physically meaningful problems.

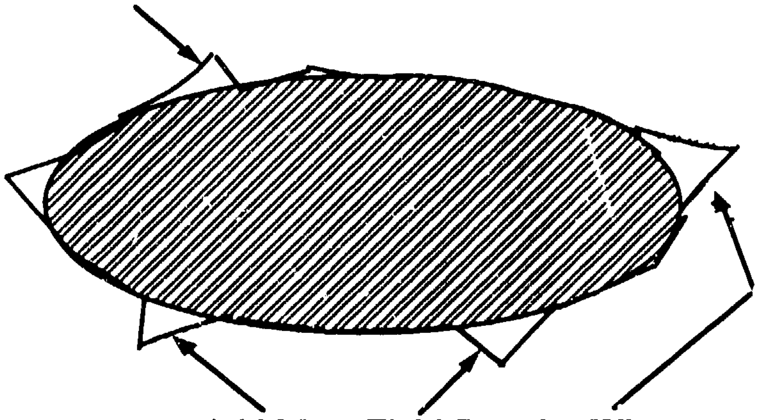

Figure 5: Conceptual example of using field sampling for model adaptation. Key ingredients for efficient adaptation are having available an appropriate error-evaluation procedure and a way of adding more unknown and field samples to the model.

Thus, computational checks would be advantageous at various points to establish quantitative measures of code performance with respect to both verification and validation. These checks will address the issues of:

1) moving codes between computers;

2) confirming continued valid operation of the code over time on a given computer, and;

3) giving guidance to the user concerning the validity of the computed results.

Computational models would ideally also include features that support "dialable" accuracy to permit an explicit trade-off between the cost of the computation and the accuracy of the results.

The first step in assessing computational accuracy stems from the two sources of error in any modeling exercise. These are the physical modeling error (E_p) which arises from approximating the physical problem of interest with some idealized mathematical representation, and the numerical modeling error (E_n) which occurs because only an approximate numerical solution is obtained to that idealized mode. Determining E_p will require access to measured data since few problems are modeled without employing some physical approximation such as representing a smoothly curved object by plane, triangular facets. Given adequate computational resources, E_n can always be made smaller than E_p. The essence of the verification and validation approach outlined here is to develop a protocol for systematically and consistently estimating E_n in response to the three points above.

There are a number of options that could be considered for this purpose but that are rarely utilized in the modeling codes now available. In connection with (1) and (2) above, for example, it would be advantageous to include a set of precomputed test cases, including the model input, results at various stages of the computation, and the final observables such as radar cross sections and/or thermal emissions. EMSES would then allow the user to automatically compare the results of running these test cases with their precomputed results using appropriate error norms to determine where any significant differences exist. Concerning (3), EMSES could also include a user option to exercise various validation checks that might range from checking far-field reciprocity, to evaluating boundary errors, or even comparing results from two different numerical models. Finally, EMSES could offer the modeler a quantitative "figure of merit" (FoM) which indicates how reliable the computed results might be.

It can be seen that verification and validation options range from being quite easily implemented to posing a research challenge. However, these issues will become increasingly essential as problem complexity and the associated total FLOP count continue to increase with faster computers.

Developing Problem Input

A useful metric in CEM is how large a problem measured in wavelengths or how many unknowns can be solved on a given computer architecture in a given amount of time. A measure like this is informative because it indicates for which size of problem the user might need to consider changing to a different computer platform. It also clearly demonstrates the state-of-the-art of present mainframe or supercomputers in terms of defining what a "large" but possibly solvable a problem actually is. For the present discussion, we will consider the number of unknowns solvable in one hour as the relevant measure.

This number is closely related to the effort needed to prepare the input data required for a computer solution. When the one-hour problem size involved only a few hundred unknowns, the effort needed to prepare the input manually was feasible. On current supercomputers, the one-hour IE problem size has passed 10,000 unknowns. The input effort has grown commensurately larger, reaching the point where manual data preparation becomes unfeasible. The need for computer assisted data preparation is even greater for DE equation models, where the number of spatial unknowns is proportionately larger because the model samples represent a volume of space rather than an enclosing surface.

Problem description for CEM actually occurs at two levels on the input side. The more elementary one is where the problem being modeled is described electrically and geometrically in the way required by the specific modeling code being used. For example, for a wire code the model description might include the two endpoints of each wire segment, its diameter, impedance loading, and connection information about wire segments attached to either end. A more advanced level, but one employed by few if any EM models today, is where the problem is described in engineering-oriented terms such as might be associated with engineering drawings. Translation of the latter, physical problem description (PPD) to a numerical model description (NMD) like the former, is then done by the user interface software. Implementing an approach like this for EMSES would have three distinct advantages:

1) The PPD would need to be developed only once;

2) The NMD can be developed interactively using an appropriate "translator" that follows guidelines needed by a particular modeling approach/library permitting human inspection and intervention where needed; and

3) The single PPD can be used to drive any modeling library component to make intermodel comparisons and solution presentation more consistent and accurate.

Visualizing the Model and the Results

Electromagnetics is one of the more mathematical and abstract sciences. There is little opportunity to view field and wave phenomena that are relevant to CEM. While it is true that we live in a visible light world, we can directly observe none of the phenomena that are important at the size-to-wavelength ratios that are of concern in CEM modeling. Yet visualization of the solutions that are obtained using CEM techniques is becoming more and more important.

One reason visual electromagnetics is needed is the growing complexity and size (number of unknowns, size to wavelength ratio) of problems being modeled. As already noted, it is relatively simple to confirm the correctness of the data needed to describe a problem consisting of an array of dipoles such as a logperiodic antenna. However, the physical behavior of even this simple problem, when the observables of interest are the antenna currents as a function of frequency or time or the angle dependent near and far fields, can be challenging to interpret without a graphical presentation. We believe that graphical presentation of results will become much more important for more complex problems where there is an even greater variety of parameters and variables to be observed. This need has led to an increased emphasis on scientific visualization in electromagnetics, [Miller et. al. (1981) (1988), Cole et. al. (1990)], to provide access to, and understanding of, the results of CEM modeling.

Visualization is also needed to ensure the correctness of problem-description and model-description data. Finding errors manually in numerical data that describes the complex interconnection of triangular facets used to represent a moderately complex conducting body is intimidating and error prone. Visual presentation of the model is the only effective way to inspect the input data. Visualizing the intervening steps in the computation process can also provide insight into the correctness of the numerical results and interpreting the physics being described. For example, in one application involving modeling an antenna near the earth-air interface, we found that a graphical plot of the impedance matrix showed a numerical "noise" on the smaller values of matrix coefficients. This demonstrated that 32-bit accuracy on our VAX computer was inadequate. Only when 64-bit computations were performed did this noise vanish. Plots of the inverse or admittance matrix for wires have similarly exhibited the problem's physics in ways not otherwise observable [Miller et. al. (1981)]. The EMSES environment would provide easy and convenient visualization of all aspects of modeling and results presentation as part of its computing infrastructure.

Combining CEM in a Multidisciplinary Library Interface

The kinds of structures whose electromagnetic properties are the result of tight specifications and advanced requirements cannot be analyzed without regard to other physical factors that affect shape, size, and material properties. Certain RCS reduction methods degrade severely when structures bend and deform. High gain antenna performance behaves similarly. Thus, as CEM capability progresses to the point where numerical design is feasible for the most advanced structures, a connection must be made to other disciplines such as thermal analysis and structural analysis in order to assess the design in the real world, which includes structural and thermal effects.

EXPLOITING ADVANCES IN HARDWARE ARCHITECTURES AND ASSOCIATED SOFTWARE IN CEM MODEL DESIGN

Rationale for Emulating Hardware Design in Software Development

The present state of CEM may be compared with the situation that prevailed during the initial phases of the industrial revolution. Until machinery made it possible to produce more output per worker, there was little incentive to make interchangeable parts. Each craftsman produced a complete version of a given product. Its various parts though fulfilling the same function as the same part made by another worker, were not required to be interchangeable. However, when the economies of larger-scale production were fast becoming a possibility, it was soon recognized that continuing this kind of arrangement would largely offset the advantages that could otherwise be achieved. It was necessary, in the interest of production efficiency, that the creative control of individual workers be made subordinate to the benefits of standardization and interchangeability. This is a lesson that needs to be applied to software development.

The motivation for software scaleability is similar. While the production of analysis software in CEM continues unabated, designers are confronted by a bewildering array of modeling choices. Perhaps the most telling characteristic of the large majority of this software is the fact that each package requires the user to learn a new interface in spite of the fact that all these modeling tools involve a small number of the same basic steps. A major thrust of EMSES will be developing and implementing an integrated user interface to permit a designer to access and use it effectively.

Designing Software for Distributed and Parallel Architectures

Traditional multicomputing has relied on the close coupling of large numbers of homogeneous processors in hypercube distributed memory and shared-memory, bus-based interconnections. Recently, shared, distributed-memory architectures like the Kendal Square with interlocking rings of processors has extended the paradigm of closely coupled multicomputing that is a hybrid of the two earlier architectural types. Within this paradigm, machines have been developed that are either single instruction multiple data (SIMD) or Multiple instruction multiple data (MIMD). The recently announced CM-5 from Thinking Machines Inc., has confirmed the generality of the MIMD distributed memory approach as the dominant approach for the future of scalable closely coupled multiprocessors.

Closely coupled multicomputers have been made possible because of the high speed buses, rings, or hypercube communications mechanisms internal to these machines. These internal communications mechanisms have facilitated low-latency, high-bandwidth communications between processors which has made these machines efficient parallel processing computers. The recent development of very high bandwidth (800Mb/s) low latency crossbar switches which serve as local area network interconnects, and the promise of wide area network extensions of this technology, will enable the multicomputing paradigm to be extended to a much higher level of processor granularity and heterogeneity. Thus, in the near future, heterogeneous networks of architecturally diverse machines will be closely coupled over great physical distances.

Previous research at LANL, and elsewhere, has consistently shown that algorithm performance can be optimized when it is mapped to the hardware and software environment for which it is best suited. Thus, different algorithms map best to distinct architectures. Until the advent of HIPPI-based, high-speed crossbars, it was still often expedient to develop, test, and deliver a multiple-algorithm software system on a single parallel architecture fronted by a workstation network used as the user interfaces, even though the mapping of diverse algorithms to a single architecture was markedly suboptimal. The present and future potential of high-bandwidth, low-latency, crossbar networks, and their wide-area extensions will alleviate this restriction and make overall application-level optimization practical. As a result, high-performance, network-based multicomputing will be extended to include workstations, massively parallel machines, workstations with embedded accelerators, and conventional supercomputers as nodal processors on the network. The network will exploit both message passing and shared-memory capabilities and represents a hardware realization of a "virtual metacomputer".

It is recognized that the EMSES concept encompasses a wide spectrum of applications that will push the computational ability of existing computers. Because of this, it is clear that EMSES system will be designed from a very broad perspective. It will include mechanisms for distributed and parallel computing and interactive visualization. Furthermore, it must provide a working environment that encourages joint development by a geographically dispersed design and development groups. It must also support the rapid prototyping of new applications and enable the easy re-use of previously developed library software.

The key to the successful use of this virtual metacomputer will be the software-based programming environment and its underlying application level communications and control infrastructure which will permit the user to conveniently and seamlessly utilize this resource. EMSES would include this application-level software infrastructure to address the requirements of both the electromagnetic library software developer and the designer/user of this software. This would be accomplished both by integrating existing public domain, visual language, network, CAD, mesh-definition software, etc., and by enhancing these existing components to meet both the requirements of the electromagnetic library developer and the eventual designer/user. EMSES would also include scaleable and portable electromagnetics software which can be used by electromagnetics designers from a high level visual programming environment that shields these designers from the details needed to execute complex analyses over this networked virtual metacomputer.

It is no longer reasonable to expect each design and development team to write sophisticated network infrastructure software. EMSES would provide developers with a more abstract and powerful environment that links together existing sealable libraries of application software and "hides" the infrastructure details of this linkage from the user and would also provide the application level user with powerful CASE (Computer Aided Software Engineering) tools that facilitate the creation, compilation, and debugging of new library software. EMSES implementation would include a distributed and parallel software network infrastructure that can be programmed, monitored, and debugged as if it were contained within a single multicomputer.

Using the KHOROS System as a Basis for EMSES

One approach to realizing the distributed and parallel software development infrastructure for EMSES could be based on a public-domain system such as KHOROS whose design is illustrated in Fig. 6 [see Miller (1992)], which would satisfy the following seven design criteria:

1. Produce optimal and balanced computational performance on a heterogeneous computer network while transparently providing a software development CASE-tool environment that enables investigators to develop and test new library components while continuing to utilize a wide variety of existing software written in several high level languages.
2. Provide transparent access to and use of data and electromagnetic libraries over a network of different machine architectures.
3. Support an extended data-flow model of computation that is important for electromagnetic modeling.
4. Provide for network level control, communication, load balancing, fault-tolerant execution, and debugging.
5. Provide 3-dimensional object modeling and mesh generation capability using government owned BRLCAD system and its extensions or commercial solutions when applicable.
6. Provide a convenient means to interact with users who are not electromagnetic experts or programmers through a modern visual programming environment.
7. Be built using public domain, government owned, and/or easily affordable commercial software components based on open system principles.

KHOROS is built with the philosophy of being an open and extensible parallel and distributed system. It is the only open scientific computing environment that provides CASE tools for the creation, maintenance, and distribution of user contributed programs. These integrated tools are utilized by a developer to create KHOROS-compatible libraries. KHOROS provides three levels of compatibility:

1) Process interface: This minimal level of compatibility allows the developer to integrate in an existing set of executable programs. The developer only needs to edit/configure the user interface of the visual language and to interactively create a graphical user interface for each program. No software development or compiling is required for this level of integration.

2) Procedure interface: This level of compatibility allows the developer to integrate or develop a library of procedures of functions. The developer interactively creates a specification file for each procedure that is used as input to a code generator. The code generator acts as a programmers apprentice to automate the creation of all user interface code. This level of integration allows the developer to utilize the source configuration and maintenance tools provided in KHOROS.

3) Procedure interface and data structure: This highest level of compatibility also allows the developer to utilize the reusable libraries of the KHOROS system.

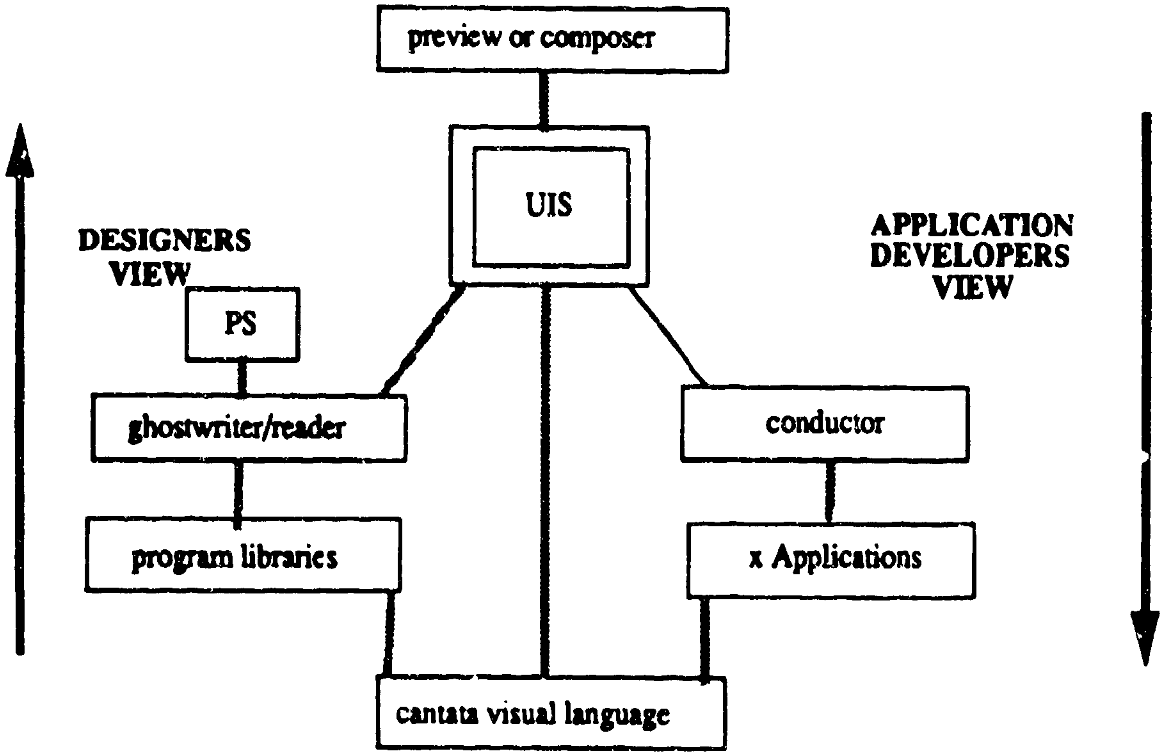

Figure 6: Block diagram of the KHOROS system and CASE tools.

Application developers work from top to bottom in Fig. 6. They use the various tools provided by the KHOROS environment to extend the capabilities of the system, add new routines to be accessed by the visual language, or create interactive graphical applications. End users/designers (shown at the bottom of the Figure) use the visuai language to create custom solutions to their CEM problems.

Preview, composer, ghostwriter and conductor represent specific User Interface Development tools that are provided within the KHOROS software structure. The library of data processing algorithms and the X applications are supported by utility and development libraries. The User Interfaces Specification (UIS) and Program Specification (PS), along with the libraries, act as input to the KHOROS tools which then generate programs. These programs include many of the KHOROS tools themselves, in addition to programs created by application developers using the KHOROS system. All applications developed using KHOROS may be referenced from within Cantata, the visual language programming environment.

"Distributed processing" in KHOROS is currently supported by the ability to manually specify remote machines upon which to execute individual KHOROS programs. The capability to do distributed processing is implemented via employment of remote data transport mechanisms and automatic process scheduling. With distributed processing, one needs a method to execute jobs remotely, as well as a mechanism to transport data back and forth from the remote machine.

KHOROS uses various data-transport mechanisms for local and remote communication. Local-transport mechanisms include shared memory, files, pipes, and streams; remote-transport mechanisms include Sockets and TLI (System V Transport Layer Interface). Custom data transport mechanisms such as HIPPI protocols to support high-speed CM-2-to-Cray communication have been implemented. With the use of remote data transport, the ability to get input from and output to remote machines is implemented. The data transport and distributed processing capability can be taken advantage of either from the cantata visual language, or from individual command line executions of KHOROS programs.

In operation, KHROS provides various kinds of visualization, including windows that display input and output data sets and the computational modules themselves which are shown as "glyphs." Each glyph represents either a process or a data source. The glyphs are connected by lines representing data transport between the different processes, and can be arranged by the user into different data-flow configurations to accomplish various kinds of computations. The modeler thus works in a mode similar to that of a hardware designer who, beginning at the gate, circuit, and chip level, constructs larger circuits and boards from elementary modules to achieve specified design goals. By storing commonly used combinations of modules, some computational "circuits" can be used over and over again to avoid duplicating past effort. Furthermore, by adhering to well-defined design rules, the modeler need not be bother with most of the minutia that characterizes most modeling now. Instead, the modeler can concentrate on conducting electromagnetic experiments on the computer by connecting together the required software components in much the same way that an experimentalist performs various experiments by using available hardware components.

COMPARISON WITH OTHER ONGOING CEM RESEARCH

The preponderance of CEM software developed to date consists of research programs intended to accomplish specific goals. A much smaller set of modeling codes has been developed in industry and at government laboratories for sponsors. A few of the codes originating in this fashion have become widely distributed because they have been well documented and supported and are available at a small cost and with few restrictions. A short, but representative, list of examples include:

NEC [Breakall, Burke, and Miller (1985)]
EM-TRANAIR [Bussoletti, et. al. (1988)]
ESP [Newman and Pozar (1978)]
EMPAC [Wilton, et. al. (1989)]
FERM [Lee, Shnidman, and Lichauco (1987)]
GEMACS [Siarkiewicz (1988)]
JUNCTION [Wilton and Hwu (1989)]
MININEC [Rockway, et. al. (1988)]
PATCH [Johnson, Wilton, and Sharp (1988)]
RCS BSC V2 [Marhefka and Brinkley (1988)]
SPEX [Ludwig (1986)]
TSAR [Ray (1991)]

A still smaller set of codes have been developed in the commercial arena. These are usually available without restriction, but can be quite expensive. Most work to date has concentrated on traditional code design where a single computational model is developed for application to a limited set of suitable problems.

One exception to this rule is GEMACS (Generalized Electromagnetic Model for the Analysis of Complex Systems). This software represents an early effort to develop an integrated modeling environment that was expected to evolve eventually into a package that offers a variety of modeling options. GEMACS was originally developed at BDM Corporation with continuing support provided from Rome Air Development Center. It contains several different kinds of models, including frequency-domain integral equations and time-domain differential equations, but seems not to have gained as wide acceptance in the CEM community as, for example, the NEC package. One reason for this may be that GEMACS is a large (approaching 150,000 lines of code) package with limited modularity and portability. It was also developed prior to the advent of distributed and parallel computing, so that porting it to these new computing environments would require major changes. While GEMACS offers some capability for hybrid modeling, it now seems relatively limited in scope compared with evolving requirements. However, GEMACS provides a number of valuable lessons learned which will be valuable in designing a package like EMSES.

Another, more recent, modeling package is EMPACK. It is being developed by a Wilton and his students at the University of Houston. Their goal is to develop modular CEM tools suitable for a wide variety of applications. Although some of the concepts behind EMPACK are attractive, its eventual realization as a solid, user-friendly package is not assured. In addition, EMPACK addresses only the CEM part of the problem and does not address the scaling of this software to a distributed highly parallel computing environment.

Much of the current work in CEM is devoted to adapting existing models to parallel machines. A typical example is Davidson (1991), in which NEC (Numerical Electromagnetics Code) was ported to a 32-transputer PC-based system. Other work is targeted at reducing the operation count of modeling by developing new techniques [for example, Gurel and Chew (1990), Kalbasi (1991)] or refining or approximating existing models [Butler (1990), Canning (1990)]. There are no efforts we are aware of that develop the field-propagator approach proposed for EMSES. While a fair amount of work has been done on developing more convenient and automatic procedures for preparing the computer-model descriptions, there also are no comprehensive efforts that target the problem-model-description approach we propose. Verification and validation has also recently received increasing attention recently [Miller (1989)].

Concluding Comments

Taken as a whole, we believe that it is absolutely essential to develop as a next-generation CEM software package an integrated system such as EMSES. This is necessary both to better exploit evolving computer hardware and systems, and to provide the more productive environment for analysts and designers which will permit them to concentrate on electromagnetics issues rather than computer and numerical issues as is now so often the case. We envision that, following an approach such as proposed here in the form of EMSES, next-generation CEM tools should provide a computational capability to electromagnetics designers and analysts that is functionally equivalent to the measurement capability now expected by electromagnetics experimentalists.

REFERENCES

Bussoletti, J.E., E T. Johnson, K.W. Sidwell, B.L. Everson, D.P. Young, R.H. Burkhart, and S.S. Samant (1988), EM-TRANAIR: A Computer Program for the Solution of Maxwell's Equation in Three Dimensions," Boeing Military Airplane Company, AFWAL-TR-87-3082.

Breakall, J.K., G.J. Burke, and E.K. Miller (1985), "The Numerical Electromagnetic Code (NEC)," in Proceedings of 6th Symposium and Technical Exhibition on Electromagnetic Compatibility, Zurich, March 5-7, pp. 301-308.

Burke, G.J. and E.K. Miller (1984), "Modeling Antennas Near to and Penetrating a Lossy Interface", IEEE Transactions on Antennas and Propagation, AP-32, pp. 1040-1049.

Butler, C.M. (1990), "Diakoptic Theory and the Moment Method," in digest of IEEE Antennas and Propagation Society Symposium, pp. 72-75.

Calalo, R. et. al. (1987) "Hypercube Matrix Computation Task," JPL Publication 87-18, Jet Propulsion Laboratory, Pasadena, CA

Canning, E X. (1990), "The Impedance Matrix Localization (IML) Method for Moment-Method Calculations," IEEE Antennas and Propagation Magazine, 32, pp. 18-28.

Cole, R.J., E.K. Miller, S. Chakrabarti, and G. Prasad (1990), "Learning About Fields and Waves using Visual Electromagnetics," IEEE Transactions on Education, 33, February, pp. 81-94.

Davidson, D.B (1991), "Parallel Algorithms for Electromagnetic Moment Method Formulations," Ph.D. Dissertation, U. of Stellenbosch, Stellenbosch, South Africa.

Engheta, N., W.D. Murphy, V. Rokhlin, and M.S. Vassiliou (1992), "The Fast Multipole Method (FMM) for Electromagnetic Scattering Problems," to be published in IEEE Transactions Antennas and Propagation.

Fettweis, A. and G. Nitsche (1991), "Numerical Integration of Partial Differential Equations Using Principles of Multidimensional Wave Digital Filters," J. of VLSI Signal Processing.

Johnson, W. A, D. R. Wilton, and R.M. Sharpe (1988), Patch Code Users Manual, Sandia National Laboratory, SAND87-2991.

Kalbasi, Khalil (1991), "DSP-based Multilevel Iterative Technique for the Moment Method Solution of Large Electromagnetic Problems," Ph.D. Dissertation, U. of Kansas, Lawrence.

Kuo, C.C. Jay, and B.C. Levy (1990), "Discretization and Solution of Elliptic PDEs--A Digital Signal Processing Approach," IEEE Proceedings, 12, pp. 1808-1842.

Lee, S., D.A. Shnidman, and E A. Lichauco (1987), "Numerical Modeling of RCA and Antenna Problems," Lincoln Laboratory, Massachusetts Institute of Technology, ESD-TR-87-036.

Ludwig, A.C. (1986), "A Comparison of Spherical Wave Boundary Value Matching Versus Integral Equation Scattering Solutions for a Perfectly Conducting Body," IEEE Antennas and Propagation Society Transactions, AP 34, pp. 857-65.

Marhefka, R.J., and T.J. Brinkley (1988), "Comparison of Methods for Far Zone Scattering from a Flat Plate and Cube," Applied Computational Electromagnetics Society Journal, Vol. 3, No. 2.

Miller, E.K. (1986), Editor of "Time-Domain Measurements in Electromagnetics", Van Nostrand Reinhold, New York, NY.

Miller, E.K. (1988), "A Selective Survey of Computational Electromagnetics," Invited Tutorial-Review Paper, IEEE Transactions Antennas and Propagation, pp. 1281-1305.

Miller, E. K. (1990), "PCs for AP," Column in IEEE AP-S Magazine, Vol. 32, No. 3, June.

Miller, E. K. (1991), "Solving Bigger Problems--By Reducing the Operation Count and Increasing the Computation Bandwidth," invited article, IEEE Proceedings, October.

Miller, E. K. (1992), Description of Khoros system in "PCs for AP," Column in IEEE AP-S Magazine, Vol. 32, No. 2, February.

Miller, E.K., F.J. Deadrick, G.J. Burke and J.A. Landt (1981), "Computer-Graphics Applications in Electromagnetic Computer Modeling, Electromagnetics," 1, pp. 135-153.

Miller, E. K, R. Merrill, and R.J. Cole (1988), "Computer Movies for Education," IEEE Transactions on Education, May, 58-68, Education Transactions Best Paper Award for 1988.

Miller, E.K. and M.A. Gilbert (1991), "Solving the Helmholtz Equation Using Multiply-Propagated Fields," International J. of Numerical Modeling in Engineering, 4, pp. 123-138.

Newman, E.H. and D.M. Pozar (1978), "Electromagnetic Modeling of Composite Wire and Surface Geometries," IEEE Transactions Antennas and Propagation, AP-26, pp. 784-788.

Perlik, A.T., and Moraites, S.C. (1992, to be published), "Electromagnetic Wave Analysis using FDTD and its Implementation on the Connection Machine," Chapter 8, Progress in Electromagnetics Research, PIER 7/8, Elsevier Press

Ray, S. (1991), in "PCs for AP" column by E.K. Miller, in IEEE Antennas and Propagation Society Magazine, Vol. 33, No. 6, December.

Rockway, J.W., J.C. Logan, D.W.S. Tam, and S.T. Li (1988), "The MININEC System: Microcomputer Analysis of Wire Antennas," Artech House, Boston.

Russell, L.C. and J.W. Rockway (1991), "Parallel Processing Applied to Computational Electromagnetics," Technical Report 1450, Naval Ocean Systems Center, San Diego, CA.

Sarkar, T.K., E. Arvas, and S.M. Rao (1986), "Application of FFT and the Conjugate Gradient Method for the Solution of Electromagnetic Radiation from Electrically Large and Small Conduction Bodies," IEEE Transactions Antennas and Propagation, AP-34, pp. 635-640.

Siarkiewicz, K.R. (1988), "GEMACS - The Present and the Future," Proceedings of 4th Annual Review of Progress in Applied Computational Electromagnetics.

Stone, W.R. (1990), "Radar Cross Sections of Complex Objects," IEEE Press.

Taflove, A. (1988), "Review of the Formulation and Applications of the Finite-Difference Time-Domain Method for Numerical Modeling of Electromagnetic Wave Interactions with Arbitrary Structures," Wave Motion, 10, pp. 547-582.

Umashankar, K, S. Nimmagadda, and A. Taflove (1990), "Application of Integral Equation and Method of Moments for Electrically Very Large Scatterers Using Spatial Decomposition Technique," in digest of IEEE Antennas and Propagation Society Symposium, pp. 76-79.

Wandzura (1992), Oral presentation on the Fast-Multipole Method at Workshop on Computational Electromagnetics, Rome Laboratory, Hanscom AFB, MA, December.

Wang, T. M. and H. Ling (1991), "Electromagnetic Scattering from Three-Dimensional Cavities via a Connection Scheme," IEEE Antennas and Propagation, AP-39, pp. 1505-1513.

Wilton, D. R, and S.U. Hwu (1989), "JUNCTION: A Computer Code for the Computation of Radiation and Scattering by Arbitrary Conducting Wire/Surface Configurations," Proceedings of 5th Annual Review of Progress in Applied Computational Electromagnetics, pp. 43-51.

Wilton, D. R, S.V. Yesantharao, C.M. Cheng, and R.M. Sharpe (1989), "EMPACK: Building Block Subroutines for the Method of Moments," Proceedings of 5th Annual Review of Progress in Applied Computational Electromagnetics, pp. 34-42.