AD-A207 812

(2)

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S): ERA 89-3	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFOSR-TR- 89-0571
6a. NAME OF PERFORMING ORGANIZATION: Ecodynamics Research Associates, Inc. / 6b. OFFICE SYMBOL (If applicable): SC	7a. NAME OF MONITORING ORGANIZATION: AFOSR/NM
6c. ADDRESS (City, State, and ZIP Code): P. O. Box 8172, Albuquerque, NM 87198	7b. ADDRESS (City, State, and ZIP Code): Building 410, Bolling AFB, DC 20332-6448
8a. NAME OF FUNDING / SPONSORING ORGANIZATION: AFOSR / 8b. OFFICE SYMBOL (If applicable): NM	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F49620-88-C-0124DEF
8c. ADDRESS (City, State, and ZIP Code): Building 410, Bolling AFB, DC 20332-6448	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. 61102F / PROJECT NO. 2304 / TASK NO. A1 / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)
"DESIGN OPTIMIZATION OF SYSTEMS GOVERNED BY PARTIAL DIFFERENTIAL EQUATIONS"

12. PERSONAL AUTHOR(S)
Roache, Patrick J.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	Aug 88 to Jan 89	1989 March	49

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	shape optimization, lasers, adaptive grids, partial differential equations

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The results of the Phase I study on "Design Optimization of Systems Governed by Partial Differential Equations" are presented. The optimization algorithm used is the Pironneau-Polak method of feasible directions with Armijo step size, previously developed by the principal Consultant, Prof. E. Polak under AFOSR funding. This algorithm, and related ones developed by Prof. Polak, are uniquely applicable to practical engineering and science problems whose constraints are defined implicitly in terms of possibly discontinuous functionals of the solution to the PDE's. The objective (cost) and constraint functions are evaluated by execution, from the optimization code, of a separate (and complex) user-oriented PDE code; gradients are determined numerically. Feasibility is convincingly demonstrated by the design optmization of several practical laser electrode problems.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL: Lt. Col. Crowley	22b. TELEPHONE (Include Area Code): (202) 767-5028 / 22c. OFFICE SYMBOL: NM

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

FINAL REPORT

31 MARCH 1989

SBIR PHASE I CONTRACT F49620-88-C-0124DEF

"DESIGN OPTIMIZATION OF SYSTEMS GOVERNED BY PARTIAL DIFFERENTIAL EQUATIONS"

SUBMITTED TO: AIR FORCE OFFICE OF SCIENTIFIC RESEARCH
AFOSR/NM
ATTN: Major James Crowley
BUILDING 410, ROOM 209
BOLLING AFB, DC 20332-6448

SUBMITTED BY: ECODYNAMICS RESEARCH ASSOCIATES, INC.
P.O. BOX 8172
ALBUQUERQUE, NM 87198

PRINCIPAL INVESTIGATOR: Dr. Patrick J. Roache

3. IDENTIFICATION & SIGNIFICANCE OF THE PROBLEM OR OPPORTUNITY

The purpose of the proposed effort is to develop algorithms for the automated design optimization of physical systems (models) governed by partial differential equations (PDE's). The types of systems motivating the present effort are represented by the following two problems.

First Problem: electrode design optimization for lasers and switches, in 2D and 3D.

Second Problem: airfoil (2D) and wing (3D) design optimization.

The more immediate motivation is prompted by our previous work on the ELF codes (Refs. 1-5). These are user-interactive codes which calculate the electric field strength and related variables (e.g. energy deposition) in the cavity of lasers, pulsed power switches, and other electric field devices, using highly accurate boundary fitted coordinates and solution adaptive grid generation. These design tools allow the designer to perturb the device operating parameters and electrode shapes, producing 2D and 3D, steady or time-dependent solutions. The shapes are parameterized by a flexible family of blended superellipses, modified by blends with elementary circular arc electrodes and C^1 perturbations used for local field shaping. The ELF codes were developed over a five-year period, with principal funding from AFWL and AFOSR. They incorporate many of the adaptive grid algorithms developed under previous AFOSR funding (contract F49620-84-C-0079) and in fact have served as a test bed for these algorithms (Refs. 6-9). (See also list of publications of the proposed P.I.) The SDI Power Consortium centered at Auburn University has recently acquired the latest version of these codes, extended to include interior dielectrics, and is acting as a beta test site prior to full commercialization. (Six commercial sales of an earlier version already have been completed.)

The second problem, airfoil and wing design optimization, will be attempted in Phase II of this proposed work for only the low Reynolds number flow airfoil design in the cruise regime. The subject of low Re airfoils is enjoying a resurgence of interest from a range of possible applications, as witnessed by the recent conference proceedings edited by Mueller (Ref. 10).

The ELF codes are highly valuable design tools, but still require the intelligent user to obtain solutions for a variety of problem parameters, and to compare and interpret results from many calculations, in order to arrive at a design. It is difficult and time-consuming for the designer to systematically perform such a study, and we suspect that the designs so arrived at, while significantly improved over those obtained by traditional methods (and certainly more reliably analyzed), are still far from optimum. We propose to develop and test algorithms for automated design optimization using these codes.

The human user will still be the "designer" in the sense

of choosing the general configuration and desired operating characteristics of the device. But once the configuration and objective were determined, the algorithms will optimize the design within the configuration limits. (We are not proposing an "expert system" to aid in this initial configuration study during Phase I, but we perhaps will propose development of a simple expert system in the latter part of Phase III.)

The automation of the design optimization process is extremely important for any codes like these. While ELF is a powerful design tool for the laser designer, experience shows that the power of its analysis capabilities is difficult to exploit fully because of human limitations. While a human being can effectively optimize in one design parameter about an initial design point, he rather quickly saturates as the number of design parameters increases. For design optimization in an n-dimensional design parameter space with $n \geq 30$ (say 20 electrode geometry parameters and 10 more for laser physical characteristics), the designer is lost.

This situation is not unique to electrode design, of course, but applies to most fields of high technology, where powerful computational tools for analysis have been developed, but the optimization aspects of the analysis/design loop remain primitive or limited. In the area of transonic airfoil design, a solution constrained problem exists when a limit is placed on the local maximum Mach number; this problem has been successfully attacked by elementary parameter search techniques, but is feasible only for 5 or 6 parameters at most; Ref. 11. (The situation is aggravated for any highly integrated design; the most extreme example of which is the National Aerospace Plane.)

The automation of the search is, first of all, very demanding on the numerical techniques used. Each search point requires the solution of a 2D or 3D, possibly time-dependent, nonlinear PDE, preceded by the grid generation problem. Obviously, computational speed is required, yet speed must be sacrificed for robustness if these two desiderata are mutually exclusive. This consideration bears on the selection both of algorithms and of numerical tuning parameters for them, e.g. relaxation parameters, continuation methods, etc. Automatic truncation error convergence testing is also highly desirable, though perhaps not entirely necessary. These objectives suggest a multigrid approach for the solution methods.

Assuming for the moment that all these requirements can be met, we still have a difficult non-classical problem in nonlinearly constrained optimization which we describe as "solution constrained optimization". In a classical constrained optimization problem, we maximize some user-defined payoff function subject to linear or nonlinear constraints on the independent variables of the problem (such as the packaging constraints in the present problem class). But in our case, the optimization problem involves constraints on the solution or functionals of the solution itself, as well as constraints on

the independent variables.

For example, we solve the analysis problem using the ELF codes by solving, in completely general non-orthogonal boundary fitted coordinates, for the potential ϕ from the nonlinear elliptic equation,

$$\nabla \cdot (\sigma \nabla \phi) = 0$$

where σ is the conductivity, usually a nonlinear function of the electric field strength $E = |\nabla \phi|$ and other external parameters such as external ionization source strength (possibly itself a function of time), position, gas mixture, etc. The design optimization problem is then stated as follows. Subject to the (usual, classical) packaging constraints of geometry, we wish to vary the parameters of the electrode shape and ionization source in order to match some desired, user-specified solution functional such as energy deposition

$$D(x,y,z) = \sigma(x,y,z) \times E(x,y,z)^2$$

subject to a constraint on another solution functional, e.g.

$$\text{maximum of } E(x,y,z) < \text{ELIM}$$

where ELIM is some specified limiting maximum electric field strength to avoid streamer formation (i.e. spark breakdown). Whereas the classical constrained optimization problem involves optimizing some functional of the dependent variable $\phi(x,y,z)$ while constraining the independent variables (x,y,z), the present problem involves optimizing some functional of $\phi(x,y,z)$ while constraining yet another functional of $\phi(x,y,z)$, as well as (x,y,z).

This type of problem is little known in mathematical papers and books on optimization, but is in fact prototypical of many engineering design optimization problems. (Some previous studies related to this problem are reviewed below). Fortunately, it has been addressed rigorously and the field has been recently summarized by E. Polak (Ref. 12) of the Univ. of California at Berkeley, who has also developed a software implementation of his algorithms called DELIGHT. We consider this work to be of signal importance for the future of all engineering design optimization. Prof. Polak will be the principal Consultant on this proposed work.

PHASE I TECHNICAL OBJECTIVES AND TASKS

The Phase I proposal itemized the following seven tasks.

Task 1.

Installation of the DELIGHT optimization system and algorithms onto the Ecodynamics computer system and general familiarization.

Task 2.

Development of a stripped-down 2D version of ELF (ELFO) with the removal of many algorithmic options (e.g. solvers, grid generators, initial condition generators, etc.) and physical options (transients, externally applied circuits, nonlinear conductivities, internal dielectrics, etc.).

Task 3.

Development of algorithms and codes for the simplified ELFO code for the evaluation of partial derivatives of solution functionals with respect to design parameters by secant evaluations about a baseline case; this work includes examination of the scaling problem to avoid swamping of secant accuracy by round-off error and incomplete iteration error.

Task 4.

Incorporation of the above simplified ELFO with secant evaluations into DELIGHT to create DELIGHT.ELFO.

Task 5.

Exercise of the DELIGHT.ELFO system to evaluate accuracy and efficiency; refinement of algorithms; likely publication of new optimum electrode shapes for vacuum operation with realistic packaging constraints.

Task 6.

Formulation of the approach for more computationally intensive problems (nonlinear conductivities, transient and 3D cases, airfoils) including preliminary study of incorporating a multigrid search algorithm (with increasing resolution as the optimal condition is approached), and computer time projections for pre-determined levels of accuracy with considerations of increased computational power (Cray-2 and/or 16 processor Sequent).

Task 7.

Overall detailed plans for implementation in Phase II; preparation of final report and Phase II proposal.

PHASE I PROGRESS

The progress in the Phase I effort was very painful and irregular. Some of the early tasks were never accomplished but were superceded; in the end, the results of the Phase I study substantially exceeded the proposal, both in the difficulty of the specific problems solved and in the generality of the approach developed. After a difficult period at the bottom of the learning curve, the Phase I study clearly demonstrated the capability for optimizing realistic nonlinear problems, and developed the methodology for a general approach which does not require subroutinizing the PDE codes.

Significant software problems were encountered with the transfer of technology from the Consultant to Ecodynamics. During the initial trip by the Principal Investigator and the Senior Scientist to the Consultant's offices at Berkeley, the PI and SS benefitted greatly from essentially tutorial sessions with the Consultant. These discussions made it clear that our originally proposed task of converting DELIGHT.MIMO to our electric field problems was ill conceived. We did obtain a code utilizing the intended algorithms, in the hope of using that as a template for developing our own.

The first task was to convert this code to our computer systems at Ecodynamics and to interface it with our new Fortran codes. The candidates were a Sun using UNIX, an Acer (80386/7) using UNIX, or a MicroVAX using VMS. The code was written in C, and we did not have a C compiler on the MicroVAX. The machine of choice, by virtue of its dedicated use and speed, seemed to be the Acer. The programmer assigned the task of code conversion was quite familiar with Fortran, expert in data base systems, and conversant with UNIX and C. However, difficulty was experienced from the beginning.

Although some parts of the code yielded quickly, others became more and more convoluted. The SUN was invoked, and an expert consultant in C was brought in. It became clear that the C coding used in the driver was very non-standard and esoteric. The original code involved some 2000 lines of C code, and about 350 lines of Fortran interface. The Fortran interface could be used for elementary problems with simple algebraic expressions for the objective and constraint functions, but the "function" structure in non-standard C (involving extensive use of pointers) could not be extended to our goal of using a separate PDE code to evaluate objective and constraint functions. The method used was excellent for the original application of the Consultant, but

was orthogonal to our intended use with the ELF PDE code.

Arriving at this conclusion was made more difficult because of our lack of real expertise with C, the non-standard C practices used, and LPI C compiler errors on our system. There were problems with pointers, global variables, passing procedure names from Fortran to C, and getting the Fortran set-up to mesh with the C method. In a desperation effort, the programmer resorted to an attempt to convert all the code to C, with the intention of then re-writing the entire code in a more standard C structure and later converting it all to Fortran; this was finally judged impractical.

This frustrating experience used up a major portion of the Phase I time and budget; with some difficulty, the decision was made to drop the approach entirely, and a new approach was undertaken.

Retaining only the core C code for the quadratic minimization routines, the Consultant wrote a new optimization code and sample driver using the language of MATLAB. This was transferred to the Senior Scientist, who again had some difficulty getting it to run on the SUN, due to a significant change (really a reversal) in MATLAB syntax between the Consultant's MATLAB and our (older) version. Using MATLAB and a lingua franca to completely and unambiguously specify the algorithm, the Senior Scientist then translated this to a FORTRAN 77 code. We then had a model problem and code running on the SUN, and successfully converted it to the Acer.

We then proceeded to exercise the model code, and to pare it down to an even simpler problem with an obvious geometric interpretation. Various enhancements were made to the code, including use of iterative convergence criteria (on both the objective function and the parameter variation) instead of a fixed number of iterations, user-interactive problem specification, modularization and generalization of objective and constraint function evaluations, etc. These developments proceeded smoothly, with some errors due to the case-sensitivity option of the Fortran 77, necessitated by the use of mixed Fortran 77 and C code (for the quadratic minimization modules), C being inherently case sensitive. Also, the code development was difficult because the use of mixed C and Fortran code modules disabled the debugger in the LPI compilers.

In December 1988, progress was at last encouraging. We succeeded in learning about the actual performance of the algorithm on model problems. However, as the code development continued, adding simple but useful extensions (such as evaluating and outputting the number of gradient evaluations, extending modularity in anticipation of

numerical gradient evaluations, etc.) we reached a point where the entire code broke down, for no evident reason. From the beginning, we suspected a compiler error, but were unable to pin it down. We dropped back to an earlier version, built up the enhancements from a slightly different path and different code structure, and again made good progress until a certain level of code size or complexity was reached, at which point the entire code again broke down. The type of errors seemed to indicate segmentation problems, with innocuous operations (e.g. PRINT statements in the main program) destroying entirely local variables in subprograms. A third attempt was made to build up from a slightly different code architecture, and again the code broke down at a point close to the target.

Due to budgetary constraints and other time commitments for the Ecodynamics personnel involved, the work was abandoned for some time. However, during the conversion to the Acer UNIX machine of a related code (the ELF electric field codes previously developed by the PI and intended to be used as the prototypical PDE problem in this optimization work) a small but strange problem was observed. Most of the conversion was straightforward, but a single minor option (involving a first order correction to E-field due to applied external magnetic fields) would not work, and produced a random behavior reminiscent of our earlier optimization difficulties. The programmer was able to isolate the error in a simple model code. The LPI Fortran 77 was not retaining purely local variables (i.e., not in argument lists or COMMON blocks) in subprograms, unless they were contained in a DATA statement. We then found a compiler option in LPI Fortran 77 (the -saveall option) which corrects the error.

It turns out that this treatment of local (stack) variables as dynamic, rather than static, is part of the ANSI Fortran 77 standard. We treat this as a compiler error, and are of the opinion that it is a prime example of the harm caused by unstable language standards. It is not the standard on the SUN Berkely 4.2 UNIX F77. It of course is not the standard in Fortran 66 or any of its predecessors. It saved only a little storage, which is a concern curiously out of date. (It would have made more sense as the standard 20 years ago for small memory computers, but makes no sense now.) It is the standard in C and possibly other languages. Its capacity for harm in scientific programming is enormous. In the present case, it very nearly defeated the entire effort.

Once this compiler error was discovered and corrected, we made a last ditch effort on the project. We had proposed to develop a stripped down version of the 2D ELF codes, to be set up in a subroutine, which would be called by the optimization code. We then intended to pursue a slow and orderly progression of the problem difficulty. Shortly

before the initial meetings with the principal Consultant, the Principal Investigator and Senior Scientist met with the Consultant on electrode physics, W. M. Moeny. This team defined candidate optimality and constraint conditions for the electrode problems (see below).

However, the work done in chasing non-standard C-constructions, interfacing with Fortran, computer system conversions, and both C and Fortran 77 compiler errors had caused much more of our budget to be spent on programming support, and had used up most of the time and money on the project, with only a model geometric problem resulting. It clearly was not feasible to build a stripped down and subroutinized ELF code, and pursue the orderly development. We decided instead to tackle, at the eleventh hour, a more difficult problem which originally we had proposed addressing only in the Phase II effort, namely shape optimization of realistic electrode families including strong nonlinearities.

The (forced) utilization of mixed Fortran 77 and C codes had provided us with the capability of C to execute other processes. We configured our Fortran 77 optimization code modules which evaluated objective and constraint functions and gradients to write the search parameters to a file, then call a C subprogram, which in turn executed an ELF code. The ELF code read the parameters from the file, generated the boundary fitted coordinate system, obtained the solution to the PDE's, etc. The ELF code was also modified to evaluate numerically the objective and constraints functions, and write these to another file. Execution then proceeded back to the C code module, back to optimization, which then read the objective and constraint function values resulting from the last set of parameters. A flag was set by the optimization code at the beginning of each iteration in the file which is later written by the ELF code; if the ELF code did not successfully terminate and overwrite this flag, the optimization code then terminated.

The optimization code had to be converted to evaluate the gradients numerically, rather than analytically. This was done initially on the geometric model problems. The development went quite smoothly, and the results were quite good. However, we did learn what we might have anticipated; that the numerical noise associated with the gradient evaluation is not a problem early in the search (where in fact it often converges slightly faster than the analytical evaluation) but does lead to erratic and expensive wandering when the solution is approached and the convergence tolerances have been too casually set to unnecessarily low values.

At the same time that the numerical gradient evaluation was being developed, the programmer on the project converted the full 2D ELF codes from their original interactive design

to run under batch (obviously necessary for the optimization code to drive the ELF code without human intervention). This was accomplished using FORNAME, an Ecodynamics proprietary Fortran portable version of NAMELIST. (NAMELIST is not available on all Fortran compilers, e.g. Cray compilers, and is not covered in ANSI standards.) FORNAME reads input data from a user-written file which includes extensive documentation and which can be edited by any screen editor at hand. FORNAME was used to set up the base case parameters and the many other problem parameters which were not part of the optimization search, e.g. the nonlinear conductivity model, PDE solution algorithm options, options for cartesian or axisymmetric or radial geometries, etc. (The optimization search parameters are written by the optimization code not to a FORNAME file, but to a standard data file.) Some experience with the physical problem is required in this selection of a base case, as with the selection and definition of optimality criteria and constraint conditions.

PHASE I RESULTS: CANDIDATES FOR OPTIMALITY CRITERIA FOR THE ELF CODES

The following are candidates for optimality criteria for design optimization studies using the ELF electric field codes. These criteria are for the long range problems, and include not only the variable conductivity problem initially investigated in Phase I, but also interior dielectrics, singular problems, etc. The list is not exhaustive; indeed, the power of the proposed optimization methods is that the optimiality criteria and constraints can be tailored to the specific scientific or engineering application at hand.

1. The optimization codes can be benchmarked by searching for the solution for the Rogowski electrodes. This problem has an analytic solution based on potential theory, generated by two straight electrodes, one infinite in extent (ground plane) and the other semi-infinite (in the left half-plane). The Rogowski electrode surface is that equipotential line which gives an enhancement factor EF = 1, where

$$EF = |E|_{max} / \text{nominal } |E|$$

$$= |\text{grad } O|_{max} / (O_{anode} - O_{cathode}) / gap$$

It may be formulated as an optimization problem where the cost function to be minimized is the value of the conformal variable normal to the electrode surface in the conformal system subject to the constraint of not exceeding EF = 1. This is a non-classical constrained optimization formulation since the constraint is on a functional of the solution.

The cost function may be evaluated analytically without any numerical solutions of PDE's required. (A Newton-Raphson

numerical solution for the analytical inverse may be required, but it is still economical.) This problem may be built up to incude numerical solution of PDE's on the conformal grid, and then to grid generation and non-conformal far-field boundary shapes and boundary conditions, to study the interaction of these truncation errors and modeling errors with the optimiza-tion search by reference to the analytical problem.

2. Minimize |Emax| with exceeding a specified packaging (circumscribed box) volume. This is the most common present use of the ELF codes, with applications to a wide range of laser and switch problems. It can be demanding of the human designer, but a reasonably satisfactory quasi-optimization can usually be achieved. For a specified package (not just package volume, but all package dimensions) this is a classical con-strained optimization problem since the constraint is on the dependent variables.

3. Minimize packaging volume (circumscribed rectangle). or actual electrode volume (and therefore weight) without exceeding a specified |E|max. This is the second most common application of ELF, with applications to a wide range of laser and switch problems, and is becoming more common. It is much more de-manding of the human designer, and is not a classical con-strained optimization problem since the constraint is on a functional of the solution. this is an excellent candidate for a significant practical application of the Phase I effort.

4. Same as #3 but with constraints not on volume but on dimensions (L, H, or L/H). This formulation has applications to an [Army] project for a Compact Light Weight Repetitive Pulsed Power (Laser) System.

5. Field Shaping Problems. Along a user-specified line in the space of the laser/switch cavity, match a target distribution of a specified solution variable. The possible variable distri-butions to be matched are |E|, electron number density ne, conductivity _, and Power Output = JE = _ E**2. Applications are to UV or E-Beam sustained High Energy Lasers.

6. Complications such as time-dependence and moveable electrodes would introduce weighting into the optimality criteria, and would be expensive. These are not foreseen in our Phase II study using ELF, but are feasible for Phase III applications.

7. Complications such as interior dielectrics, externally applied magnetic fields, tensor conductivity, more geometric parameters, etc. will complicate the solutions and likely affect the conditioning, but are not expected to alter the optimality criteria. These will be considered in Phase II.

8. Other optimality criteria such as local electron avalanching are too tenuous to consider at the present time, but are also feasible for Phase III applications.

PHASE I RESULTS: ELECTRODE DESIGNS

The optimization search is of course demanding on the robustness of PDE algorithms, both for grid generation and hosted equation solutions. Such PDE codes are more robust for fine grid resolutions (grid generation failures can occur when boundary features are not adequately resolved) but these of course are much more expensive. For demonstration purposes, we considered only problems with smoothly varying solutions, and discretized these in a modest 11x11 grid. More accurately resolved problems, say in a 101x101 grid, will of course require at least a factor of 100 increase in computer time (more if non-optimal PDE methods are used). However, the robustness is more challenged at coarse resolution, and we consider these feasibility demonstrations to be very convincing.

The physical problems and the ELF codes are described in detail in the reprint included as Appendix A. (The ELF codes have more recently been extended to include interior dielectrics including surface charge and surface conductivity, and time-dependent space charge.) The principal features of the three optimization problems are generally described as follows.

Problem 1. Rogowski family of electrodes (2 free parameters), linear PDE (vacuum solution), 3 constraints. The field equation solved is just Laplace's equation in cartesian coordinates for potential, tranformed to non-orthogonal boundary fitted coordinates. (See reprint in Appendix A for details. For this problem, the generation of the boundary fitted coordinate system is more difficult than the solution of the PDE on it.)

The objective is to minimize the packaging volume (circumscribed rectangle) of the upper electrode, subject to two geometry contraints on the electrode shape parameters (classical constraints) and one on a functional of the electric field (non-classical solution constrained optimization); namely, that the enhancement factor == ratio of maximum (over the cavity volume) of the electric field strength to the nominal electric field strength (applied potential difference divided by electrode gap) not exceed a user-specified value, in this case, 1.35. That is, the field enhancement due to electrode shapes should not exceed 35%.

The physical constraints on the geometric electrode parameters were such that the constraints were active at

solution, i.e. the minimum occured on a constraint boundary. With the convergecne tolerances set to 0.003, this search required 13 function evaluations and 4 gradient evaluations.

Problem 2. Same as problem 1, but with the geometric electrode constraint modified so that the constraints were not active at the solution, i.e. the solution was the same as the unconstrained problem.

This search required 14 function evaluations and 5 gradient evaluations.

Problem 3. Same as problem 2, but with the nonlinear conductivity turned on. The elliptic PDE being solved is now a nonlinear Poisson equation with nonlinear conductivity This involves an external source of ionization from an electron beam gun, involving exponential spatial decay of the source term, a table look up of Boltzman equation solutions for the particular gas mixture used, and implicit solution at every point in the PDE grid of an ODE for electron number density, nonlinearly coupled to the PDE field equation through dependence on the local (grid point location) value of the electric field = grad (potential).

This search required 16 function evaluations and 6 gradient evaluations. Sample plots of the solution are shown below.

Problem 4. Linear problem in radial coordinates (with y-->r) and the electrodes defined by blended super-ellipses further blended with an elementary electrode. (See reprint in Appendix A for details.) This problem involved 7 free parameters and 13 constraints.

The objective and constraints were interchanged compared to the earlier problems. The constraints involved a minimum useable volume of the lasing cavity and a maximum electrode volume, and the objective is to design the end of the electrodes (via blended super-ellipses further blended with elementary electrodes) so as to minimize the E-field enhancement factor. That is, the desire of the designer is still vaguely a small enhancement factor, but in Problems 1 to 3 this was a constraint; here it is the objective itself.

A sample design is shown below. This was run in a background mode with very tight convergence criteria (0.001 for the objective and all constraints) in order to conclusively demonstrate the convergence of the search. The search converged in 13 iterations, and required 159 function evaluations (each an ELF execution) and 13 gradient evaluations. Each gradient evaluation required perturbations in each of 7 parameters, i.e. 13x7 = 91 ELF executions. Thus

for both function and gradient evaluations, 250 ELF executions were required. Note, however, the strong effect of the convergence criteria interacting with the numerical noise, especially sensitive near the solution. 103 of the total 159 function evaluations occured in the last (13th) iteration. At the 12th iteration, all convergence criteria had been met except one, which had converged to 0.001036 instead of the required 0.001000. This trivial and arbitrary difference in the convergence cut-off increased the work from 140 ELF executions to 250.

During the entire 13 iteration solution path, 22 constraint violations were encountered. None were active at the solution; however, it is clear that the PDE code must be configured to be robust enough to give a reasonable answer (although not necessarily accurate) even when the constraints are violated. Several of the constraints are very close (to within the optimization algorithm's tolerance) to being active at the solution. It would make sense to then re-run the design optimization with these parameters set exactly to the constraint borders, correspondly reducing the number of solution parameters and constraints.

The initial base case involved identical electrode profiles for the upper and lower electrodes. However, the solution is not symmetric because of the radial geometry, the lower electrode being at a smaller radius. Because of this radial configuration, the maximum $|E|$ occurs on the lower electrode, and only the parameters for the lower electrode were varied in the optimizaiton search.

The design improvement accomplished by this optimization search may be noted from the following. An elementary electrode (straight section with circular tip) with the same packaging constraints shows an E-field enhancement factor of 39%. The initial base case design (using a simple elliptic tip) has a 28% enhancement. The final design has reduced this to 15%. It is anticipated that further improvements are possible with more general formulation of the electrode shape parameters and constraints formulation.

Further details on results will be presented at the Workshop on Shape Optimization organized by the principal Consultant, Prof. Polak, at University of California-Berkeley on 22-24 May 1989.

PHASE II EXTENSIONS

The likelihood of success of the Phase II work is clearly indicated by the results of the Phase I feasibility study presented above.

In the Phase II proposal, further development, verification, and application of the design optimization

methods proven feasible in the Phase I study is proposed. The further algorithmic development proposed includes use of multigrid methods to refine the PDE solution accuracy as the optimization search converges, use of non-structured (multiquadric) derivative methods combined with Domain Decomposition techniques and generalized to n-dimensions to evaluate gradients utilizing previous search points, possibly application of affine transformations to speed convergence, and use of more general curve and surface representation via Bezier curves for compatibility with user specification of base case designs via CAD/CAM. Verification and applications will be to shape optimization in laser electrode design and in airfoil design, and control system optimization in superalloy solidification.

REFERENCES

1. Roache, P. J., Steinberg, S. and Moeny, W. M., "Interactive Electric Field Calculations for Lasers", AIAA Paper 84-1655, AIAA 17th Fluid Dynamics, Plasma Physics, and Lasers Conference, Snowmass, Colorado, 25-27 June 1984.

2. Roache, P. J., Steinberg, S., Happ, H. J., and Moeny, W. M., "3D Electric Field Solutions in Boundary-Fitted Coordinates", Proc. 4th IEEE Pulsed Power Conference, Albuquerque, NM, 6-8 June 1983.

3. Roache, P. J., von Dadelszen, M. and Moeny, W. M. "Electric Field Calculations Using the ELF Codes", Proc. IEEE Pulsed Power Conference, Crystal City, DC, 10-12 June 1985.

4. Roache, P. J., "The ELF Codes: Electrode Design for Lasers and Switches", Invited Paper, Proc. CTAC-85 Conference, Melbourne, Australia, 25-28 August 1985.

5. Roache, P. J., Moeny, W. M., and von Dadelszen, M., "The ELF Electric Field Analysis Computer Codes. Vol. 1. Code Description and Specific Applications. Vol. 2. Users Manual." Tetra TR-85-016, Tetra Corp., Albuquerque, New Mexico, August 1985.

6. Roache, P. J. and Steinberg, S., "A New Approach to Grid Generation Using a Variational Formulation", AIAA Paper 85-1527, Proc. AIAA 7th Computational Fluid Dynamics Conference, University of Cincinnati, 15-17 July 1985.

7. Steinberg, S. and Roache, P. J., "Variational Grid Generation", Numerical Methods for Partial Differential Equations, Vol. 2, pp. 71-96, 1986.

8. Steinberg, S. and Roache, P. J., "Symbolic Manipulation and Computational Fluid Dynamics", Jour. Computational

Physics, Vol. 57, No. 2, Jan. 1985, pp. 251-284.

9. Steinberg, S. and Roache, P. J., "A Toolkit of Symbolic Manipulation Programs for Variational Grid Generation", AIAA Paper No. 86-0241, AIAA Aerospace Sciences Meeting, 6-9 Jan. 1986, Reno, Nevada.

10. Mueller, T. J. ed., "Proceedings of the Conference on Low Reynolds Number Airfoil Aerodynamics", UNDAS-CP-77B123, June 1985, University of Notre Dame, Notre Dame, Indiana.

11. Murman, E., personal comm.

12. Polak, E., "On the Mathematical Foundations of Nondifferentiable Optimization in Engineering Design", SIAM Review, Vol. 28, March 1987, pp. 21-89.

13. Roache, P. J. and Steinberg, S., "Application of a Single Equation MG-FAS Solver to Elliptic Grid Generation Equations (Sub-grid and Super-grid Coefficient Generation)", Proc. Second Copper Mountain Conference on Multigrid Methods, 31 March - 3 April 1985, Copper Mountain, Colorado, S. McCormick, ed., North Holland.

14. Sandgren, E. and Kral, L. D., "A Numerical Method for Solving Partial Differential Equations for Shape Optimization", to appear.

15. Haug, E. J. and Chun, Y. W., "Shape Optimization of a Solid of Revolution", Journal of Engineering Mechanics, Vol. 109, No. 1, Feb. 1983, pp. 30-46.

16. Braibant, V. and Fluery, C., "Shape Optimal Design Using B-Splines", Computer Methods in Applied Mechanics and Engineering, Vol. 44, 1984, pp. 247-267.

17. Bennett, J. and Botkin, M., "The Optimal Shape", Mechanical Engineering, August 1983.

18. Pinebrook, W. E. and Dalton, D., "Drag Minimization of a Body of Revolution through Evolution", Computer Methods in Applied Mechanics and Engineering, Vol. 39, 1983, p. 179.

19. Pinebrook, W. E. and Dalton, C., "The Evolution Strategy Applied to Drag Minimization of a Body of Revolution",

Mathematical Modelling, Pergammon Press, Vol. 4, 1983, p. 439.

20. Jameson, A., "Aerodynamic Design via Control Theory", ICASE Report 88-64, Nov. 1988.

21. Madych, W. R. and Nelson, S. A., "Multivariate Inter-

polation and Conditionally Positive Definite Functions II", J. Approx. Theory Appl., to appear.

22. Kansa, E., "A Comparative Study of Finite Difference and Multiquadric Schemes for the Euler Equations", Simulation, Nov. 1988, pp. 180-183.

23. Stead, S., "Estimation of Gradients from Scattered Data", Rocky Mountain J. Math., Vol. 14, 1984, pp. 265-279.

24. Hardy, R. L., "Multiquadric Equations of Topography and Other Irregular Surfaces", J. Geophys. Res., Vol. 176, 1971, pp. 1905-1915.

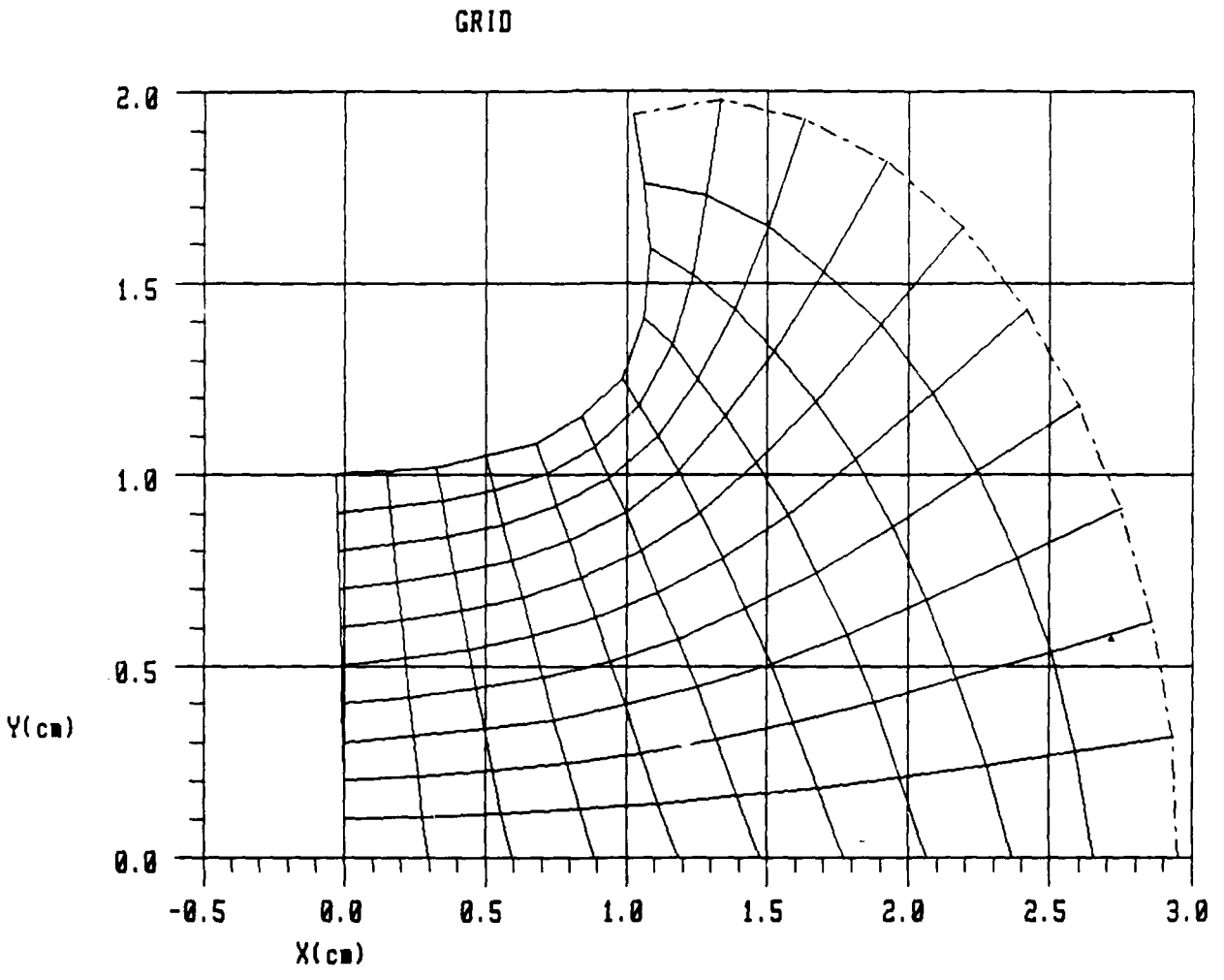


Figure 1-a.

Figure 1. Electrode problem #1. 2 parameters, 3 constraints, nonlinear.

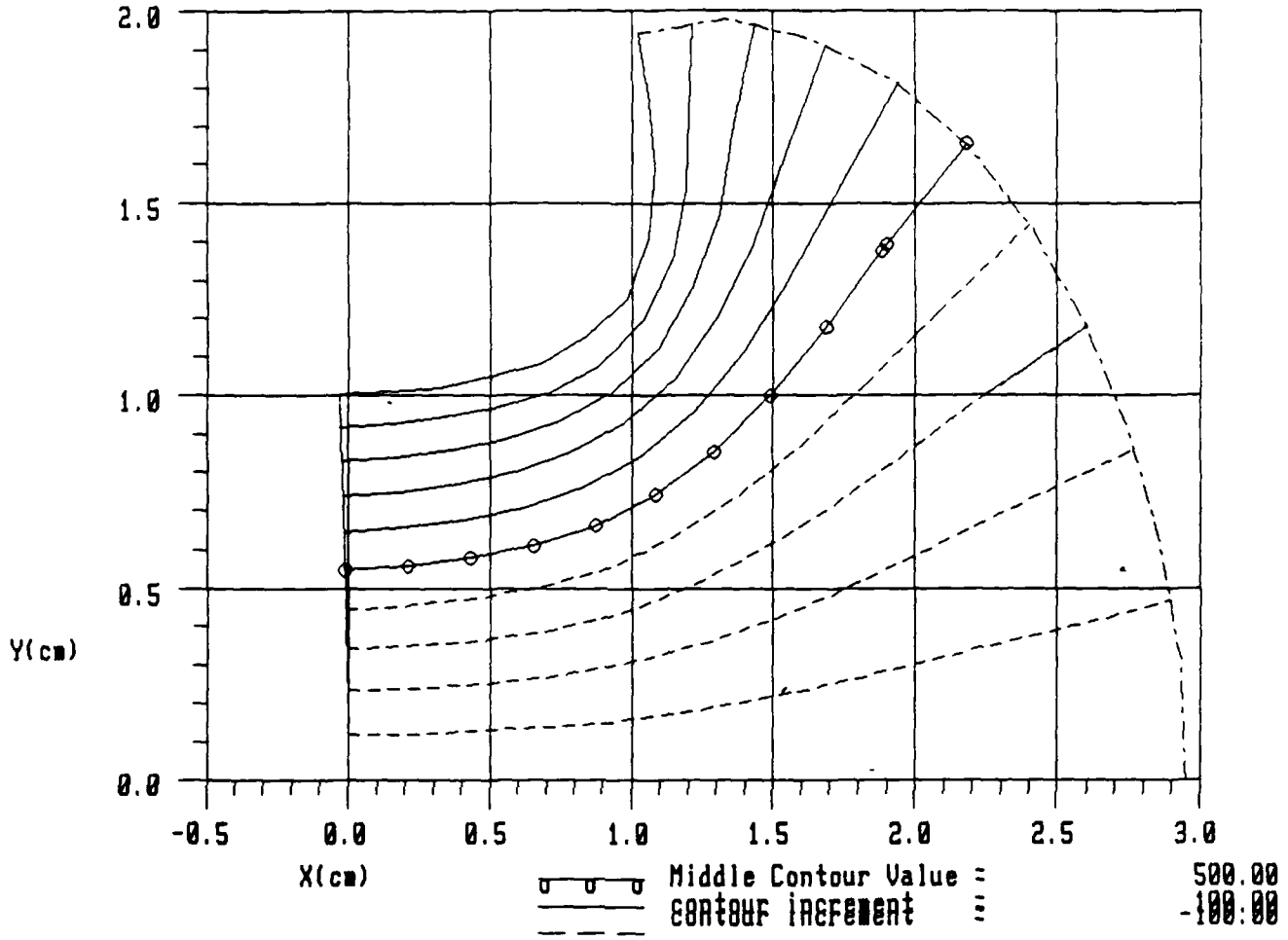

POTENTIAL CONTOURS
2.0
1.5
1.0
0.5
0.0
Y(cm)
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X(cm)
Middle Contour Value =
500.00
-100.00

Figure 1-b.

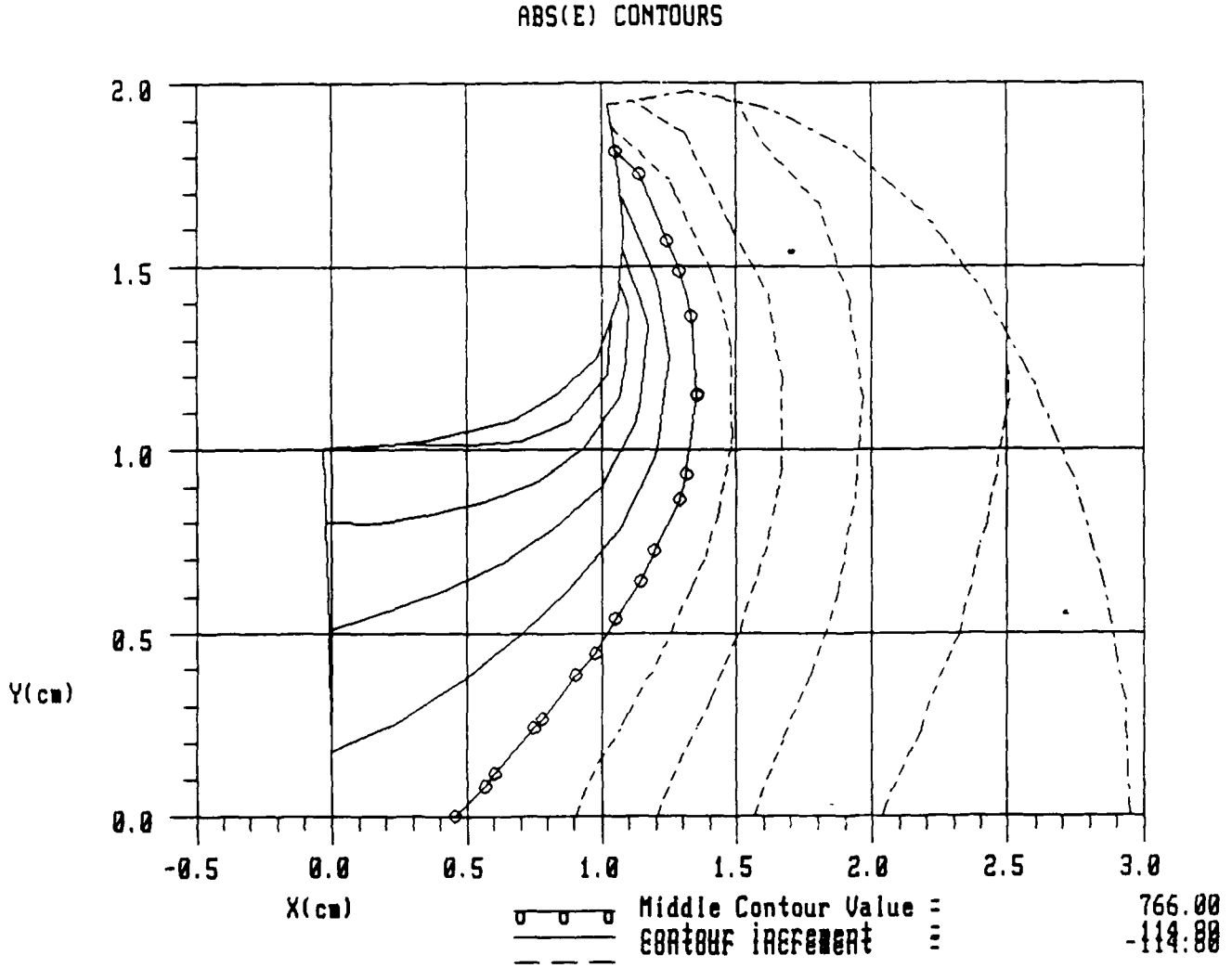

ABS(E) CONTOURS
2.0
1.5
1.0
0.5
0.0
Y(cm)
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X(cm)
Middle Contour Value =
766.00
-114.00

Figure 1-c.

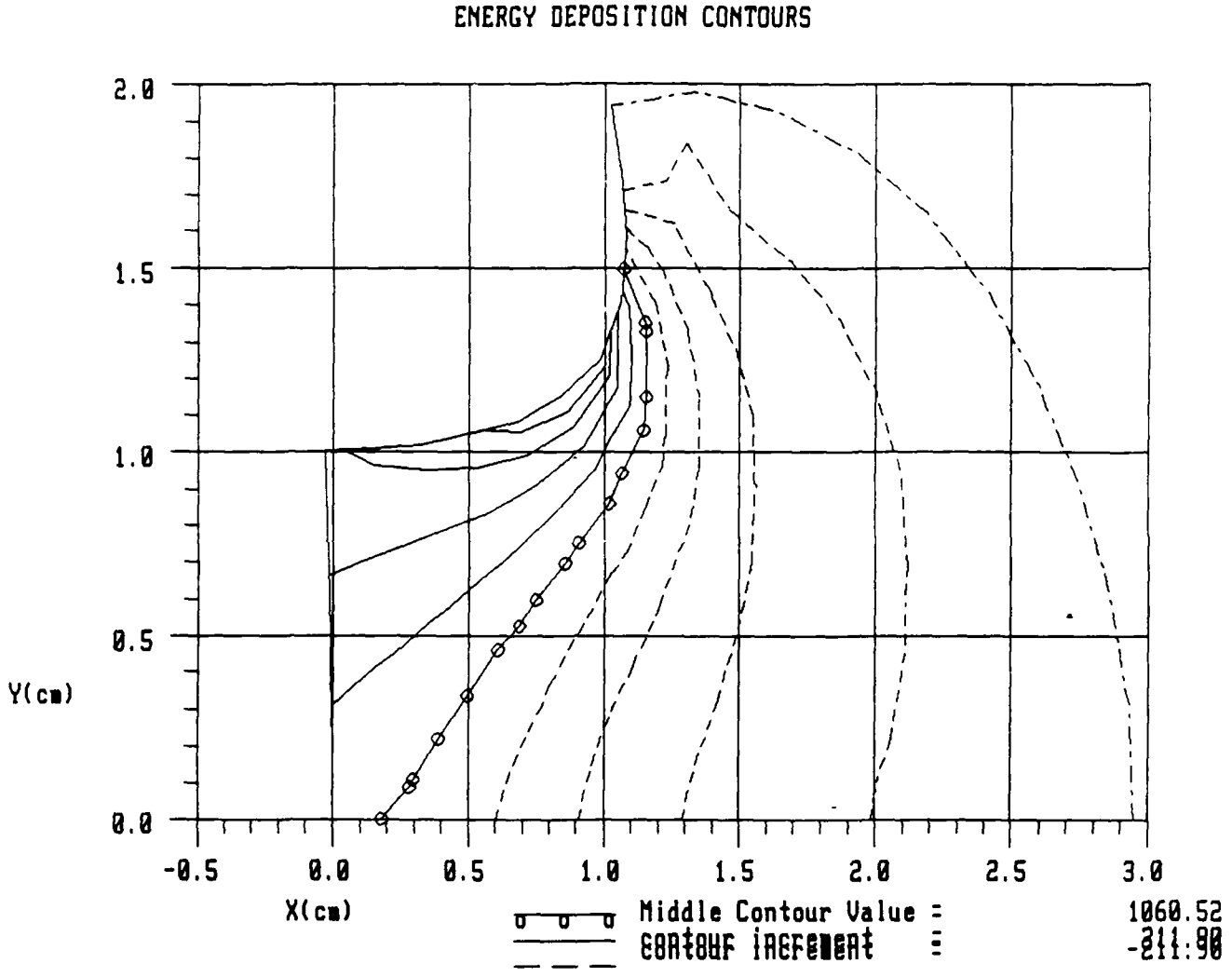

ENERGY DEPOSITION CONTOURS
2.0
1.5
1.0
0.5
0.0
Y(cm)
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X(cm)
Middle Contour Value =
1060.52
-211.90

Figure 1-d.

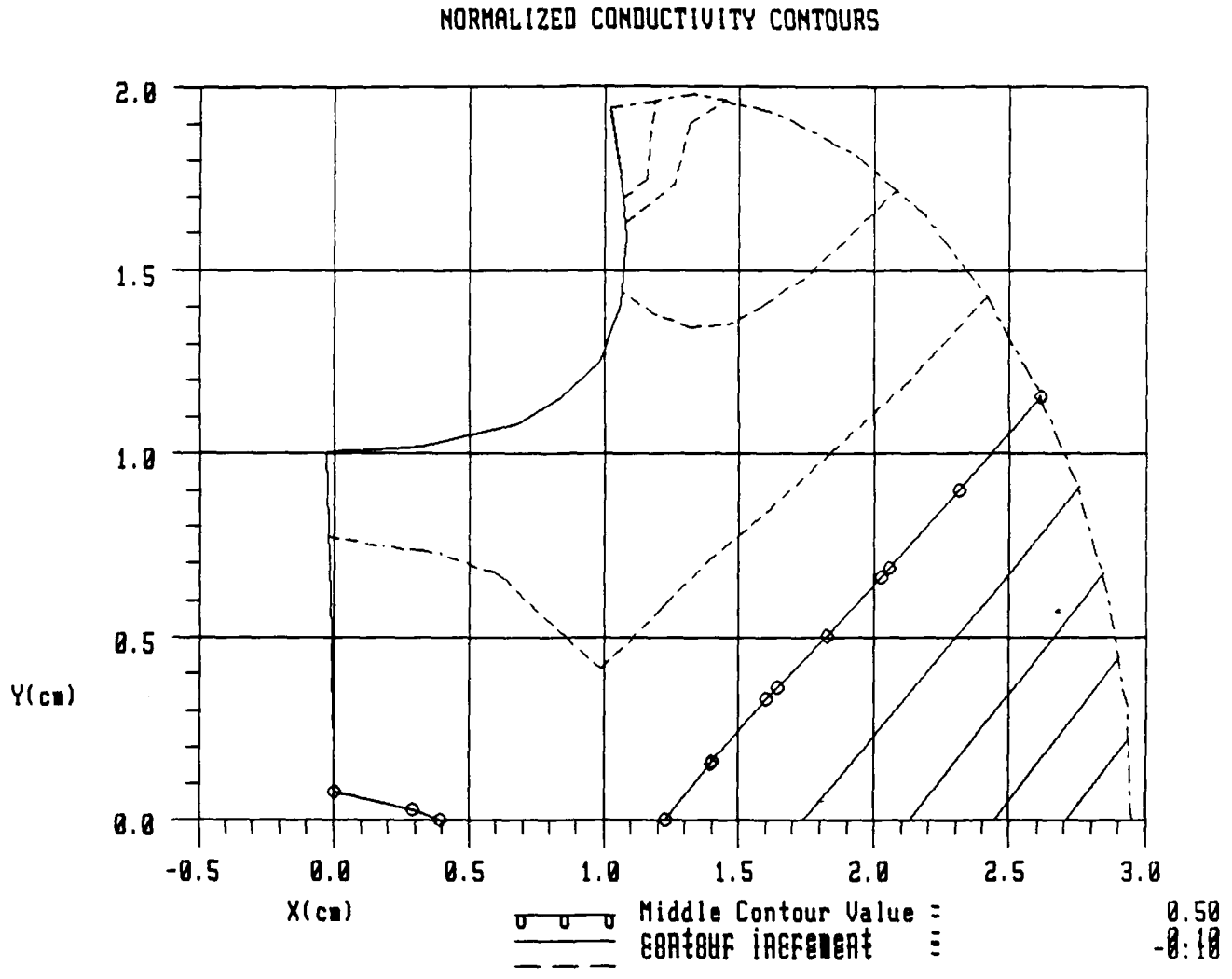

NORMALIZED CONDUCTIVITY CONTOURS
2.0
1.5
1.0
0.5
0.0
Y(cm)
-0.5
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X(cm)
Middle Contour Value =
0.50
Contour Increment =
-0.10

Figure 1-e.

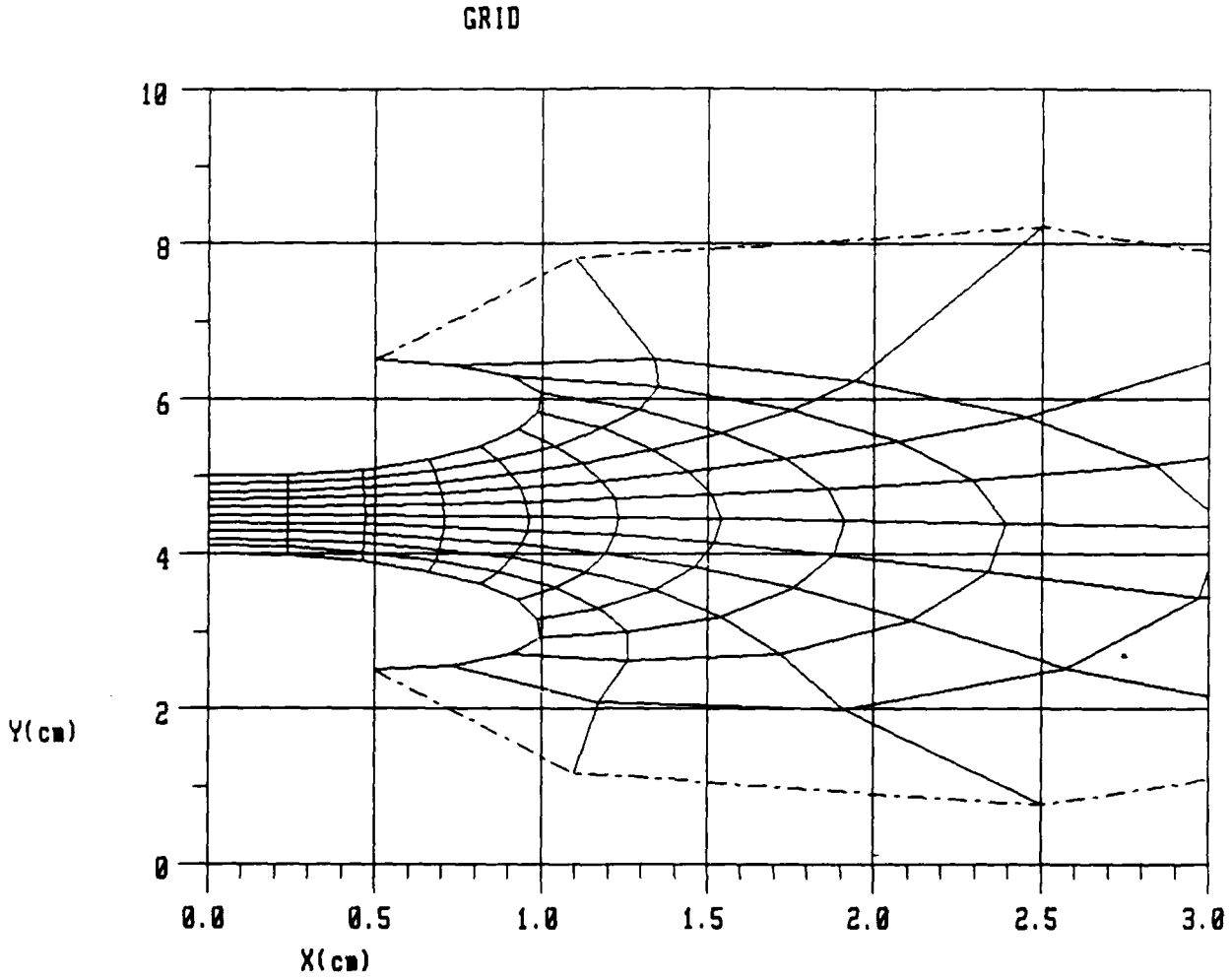


Figure 2-a.

Figure 2. Electrode Problem #4. Radial Coordinates. Initial Base Case.

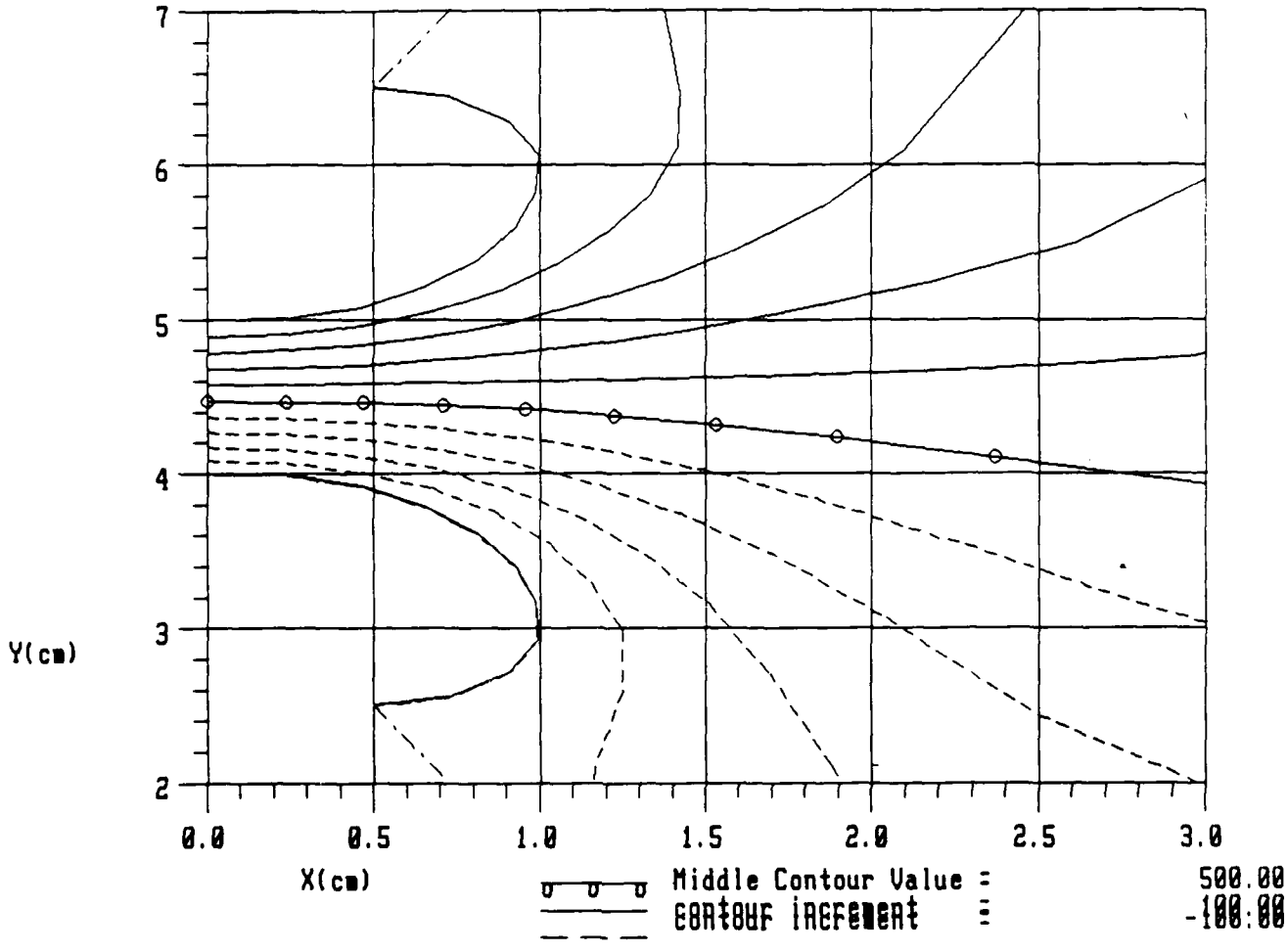

POTENTIAL CONTOURS
Y(cm)
X(cm)
7
6
5
4
3
2
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Middle Contour Value = 500.00
CONTOUR INCREMENT = -100.00

Figure 2-b.

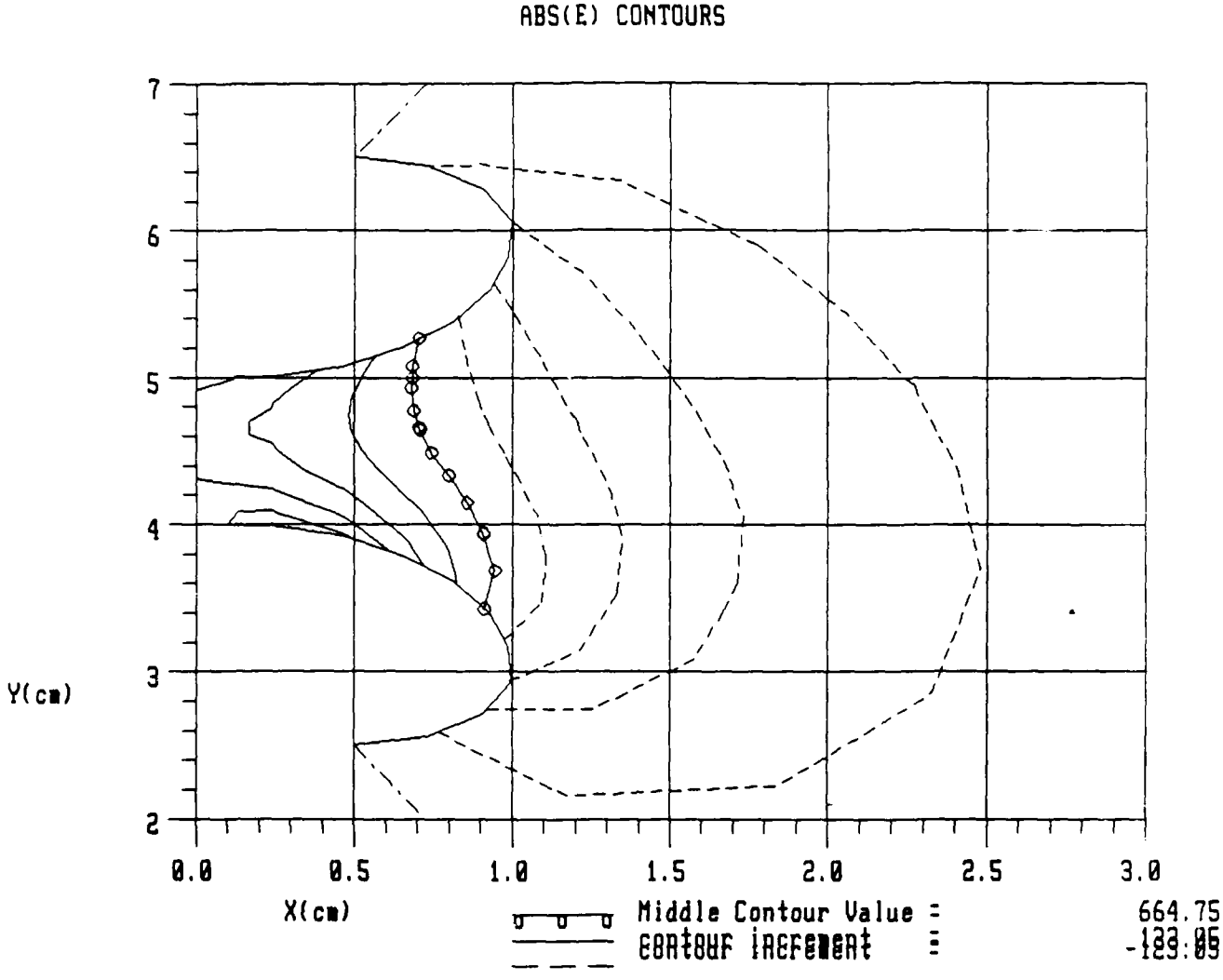

ABS(E) CONTOURS
Y(cm)
X(cm)
7
6
5
4
3
2
0.0
0.5
1.0
1.5
2.0
2.5
3.0
Middle Contour Value =
664.75
-133.05

Figure 2-c.

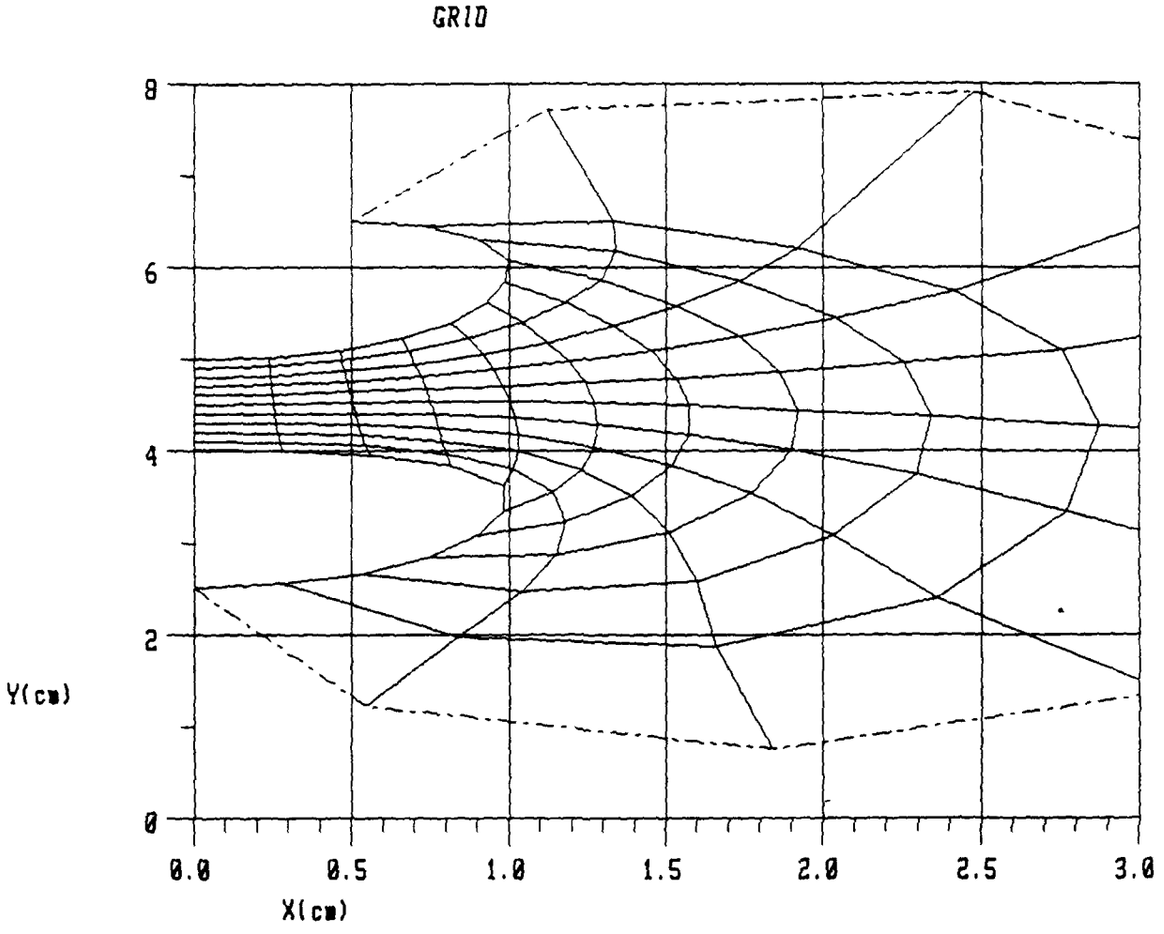


Figure 3-a.

Figure 3. Electrode Problem #4. Radial coordinates.
7 parameters, 13 constraints.

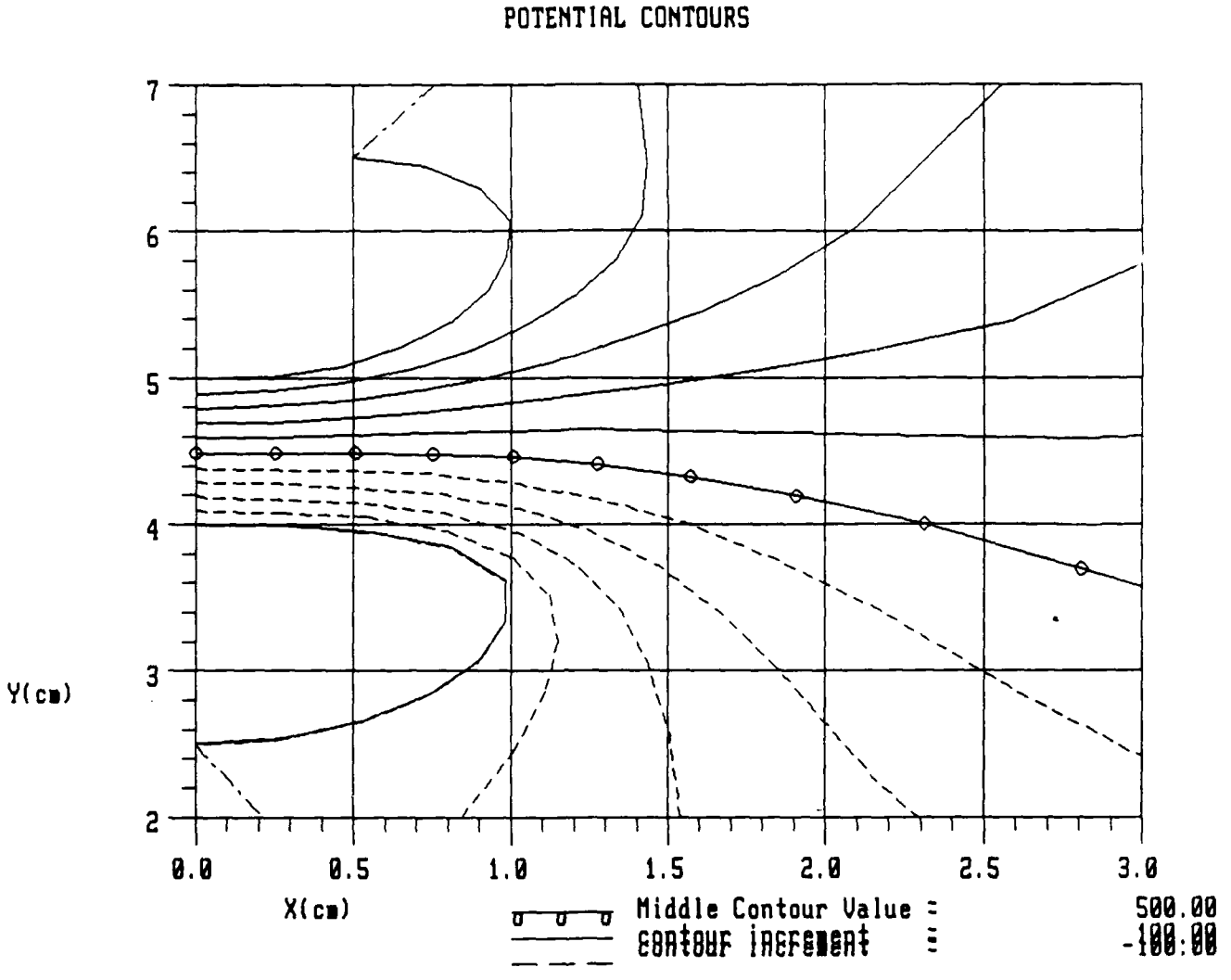

POTENTIAL CONTOURS
7
6
5
4
3
2
Y(cm)
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X(cm)
Middle Contour Value =
500.00
-100.00

Figure 3-b.

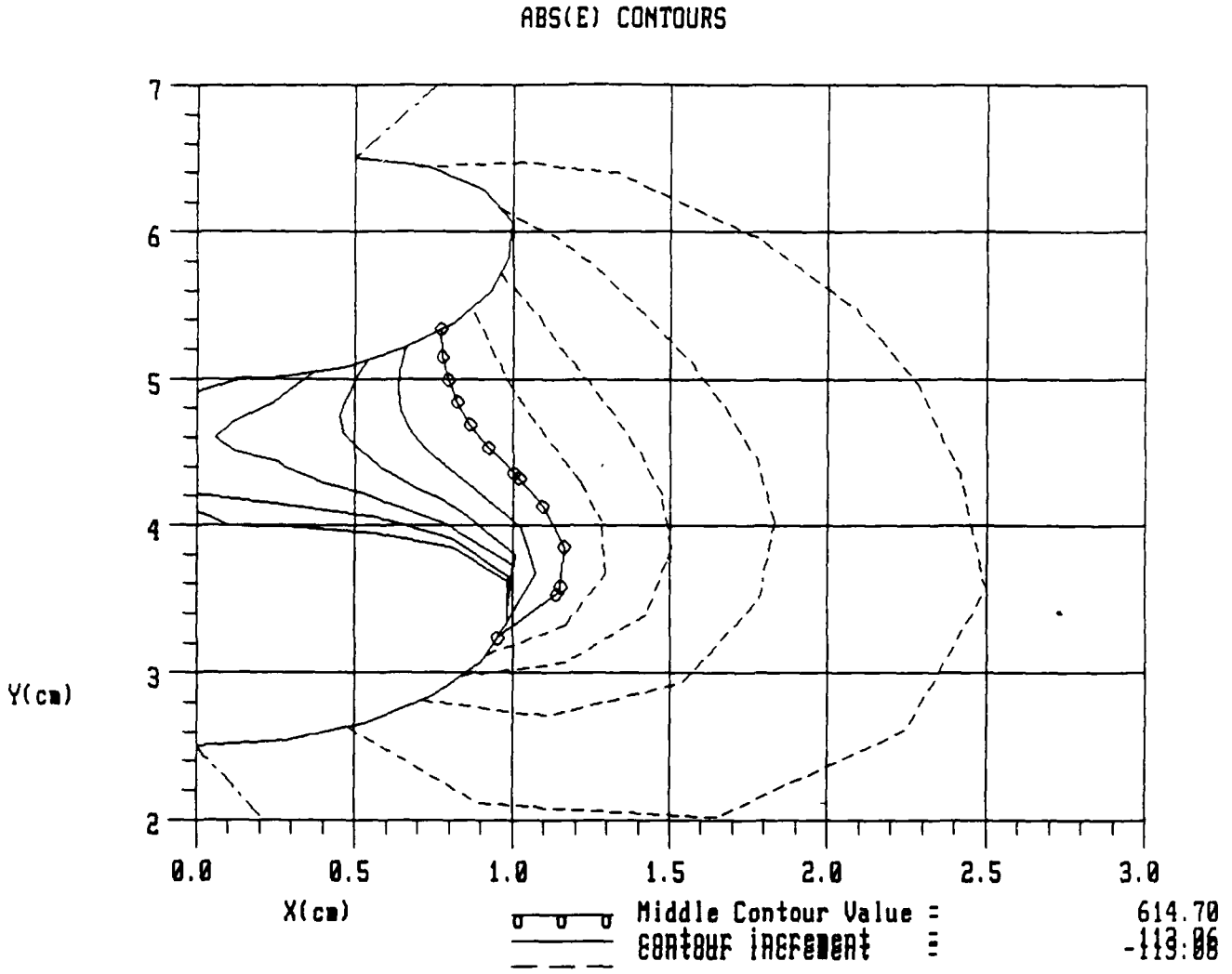

ABS(E) CONTOURS
7
6
5
4
3
2
Y(cm)
0.0
0.5
1.0
1.5
2.0
2.5
3.0
X(cm)
Middle Contour Value = 614.70
Contour Increment = -113.06

Figure 3-c.

SAMPLE FORNAME CASE FILE

The listing below provices a sample of a FORNAME case file set up to run the ELF electric field analysis codes in a batch mode. This FORNAME file is executed from the main program, and defines the overall physical parameters of the problem. A separate FORNAME file is called from the geometry routine to specify the electrode geometry parameters, and a third FORNAME file is called from the nonlinear conductivity routine which models the particular lasing gas and external sources of ionization, e.g. an electron beam gun. The FORNAME use can be configured to combine all three of these files into one, which is more convenient for a single run; we anticipate that a single file set-up will be used for the NUSC problems. However, there is some advantage to separating the different functionalities in the ELF codes, especially for automated optimization runs. For example, a particular optimization search would likely only change the geometric parameters in the second FORNAME case file. The complete set of files constitutes the entire documentation needed to run the code, including comment cards.

It is not a trivial task to set up a FORNAME file, and it is not recommended for use while the code is still in an early developmental stage. However, once the code input is stable and the FORNAME file is built, it is very easy to modify copies to produce new cases, and it is particularly conducive to documentation of runs. Significantly, the FORNAME file may be opened and edited with ANY text editor with which the user is familiar. Thus, the user-friendly environment is not dependent on any particular hardware or software system, unlike mouse-driven menu systems.

```
!
! File ELF_OPT1.FNM
! This is the ELF Forname data file, case OPT-1-Base, 28 JAN 1989.
! The purpose of this run is to establish a base case for
! optimization.
! Set up by Paul Sery, Ecodynamics, Inc. (505) 262-0440.
!

! ENTER A RUN TITLE  (MAXIMUM 32 | CHARACTER):
 title = OPT-1-Base

!
!  ENTER  MCYL, NLIN:
!     MCYL =0 FOR CARTESIAN, =1 FOR RADIAL, =2 FOR AXISYMMETRIC
!     NLIN =0 FOR CONSTANT CONDUCTIVITY SIGMA,
!          =1 FOR VARIABLE CONDUCTIVITY SIGMA.
!           (PRE-SET VALUES WERE 0, 1)
 mcyl = 0
 nlin = 1

!  ENTER  NFSIG :
!     NFSIG DETERMINES VARIABLE CONDUCTIVITY ROUTINE;
!        =0 GIVES GLOW DISCHARGE MODEL, FSIG0,
!        =1 GIVES THEOPHANIS ET AL. MODEL, FSIG1,
!        =2 GIVES XENON FLASHLAMP MODEL, FSIG2, ETC.
!           (PRE-SET VALUE WAS 0 )
 nfsig = 0

!  ENTER NELEC:
!     NELEC DETERMINES ELECTRODE GEOMETRY ROUTINE;
!        =0 GIVES BLENDED SUPER-ELLIPSES, ELEC0,
!        =1 GIVES SKEWED BOX, ELEC1,
!        =2 GIVES ROGOWSKI FAMILY, ELEC2,
!        =3 GIVES CONCENTRIC ELLIPSES, ELEC3, ETC.
!        =4 GIVES FIRST MVD GEOMETRY, ELEC4, ETC.
!           (PRE-SET VALUE WAS 0)
 nelec = 2

!  ENTER NSFLG :
!     NSFLG.GT.0 ALLOWS USER ENTRY OF SURFACE PARAMETERS.
!      (PRE-SET VALUE WAS 0)
 nsflg = 1

!  ENTER NSOLVR, NSOLVRG :
!     SETS SOLUTION METHOD FOR PHI, GRID:
!        = 0 FOR HOPSCOTCH SOR, = 1 FOR POINT SOR, = 2 FOR GEM.
!          (PRE-SET VALUES WERE 0, 0)
 nsolvr  = 0
 nsolvrg = 0

!  ENTER  NGFLG, NP2 :
!     NGFLG.GT.0 ALLOWS USER ENTRY OF GRID GENERATION PARAMETERS,
!     NP2 SETS GRID SOLVER, =0 FOR HOMOGENEOUS, =1 FOR GRAPE,
!        = 2 FOR MODIFIED THOMAS, =3 FOR EXTENDED THOMAS.
!          (PRE-SET VALUES WERE 0, 0)
 ngflg = 0
 np2   = 0

!  ENTER NADAPT, NADAPTF, EMAXAD :
!     NADAPT.GT.0 ADAPTS BOUNDARY POINTS TO SOLUTION IN
```

```
!     NADAPT+1 GRIDS, STOPPED EARLY IF EMAX CONVERGES TO
!     EMAXAD.
!     FOR NADAPTF = 1, ADAPTS TO COMBINATION OF
!                      MAXIMUM E-VAL AND CURVATURE,
!                 = 2, SIGMAS,
!                 = 3, ENERGY DEPOSITION,
!                 = 4, PRE-ADAPTS TO BSGEBD,
!                      RE-SETTING NADAPT TO 0.
 nadapt  = 0
 nadaptf = 1
 emaxad  = 1

!  ENTER  NTIML = NO. TIME STEPS,
!     NTIML = 0 FOR STEADY-STATE ONLY)
!             (PRE-SET VALUE WAS 0)
 ntiml = 0

!  ENTER  TIML, CWTIM, NTIMBC, NSPACEQ :
!     TIML    = TIME LIMIT  [SECONDS],
!     CWTIM   = 0 FOR IMPLICIT, =1 FOR EXPLICIT,=1/2 FOR TRAPEZOIDAL,
!     NTIMBC  = 0 FOR          FIXED BOUNDARY VALUES AND SHAPE,
!             = 1 FOR TIME DEPENDENT BOUNDARY VALUES AND FIXED SHAPE,
!             = 2 FOR  "       "          "      "    AND SHAPE.
!               (REQUIRES GRID RE-GENERATION AT EVERY TIME STEP.
!     NSPACEQ = 1 TURNS ON CALCULATION OF SPACE CHARGE EFFECT.
!               (PRE-SET VALUES WERE 3.0E-5, 0.0, 0, 0 )
! *** NOTE ***  these values are used only if NTIML >= 0 (see above).
 timl    = 3.E-5
 cwtim   = 0.
 ntimbc  = 0
 nspaceq = 0

!  ENTER NEXCIR; PLEASE:
!     NEXCIR = 1 TO TURN ON EXTERNAL CIRCUIT EQUATION SOLUTION
!              (PRE-SET VALUE WAS 0)
! *** NOTE ***  these values are used only if NTIML >= 0 (see above).
 nexcir = 0

!  FOR EXTERNAL CIRCUIT EQUATION, ENTER:
!     INITIAL PHI [VOLTS], RESISTANCE [OHMS],
!     CAPACITANCE [FARADS] =
!     EXPHII, EXRES, EXCAP; PLEASE:
!       (PRE-SET VALUES WERE 1000.0, 1000.0, 0.0001)
 exphii = 1000.
 expres = 1000.
 excap  = .0001

!  ENTER  PHIANO, PHICATH [VOLTS]
!     (THESE ARE SCALING FACTORS GENERALLY, AND FOR THE NOMINAL,
!     ELECTRODE CONFIGURATIONS, ARE ACTUAL ANODE/CATHODE VALUES.
!       (PRE-SET VALUES WERE 1000.0, 0.0)
 phiano  = 1000.
 phicath = 0.

!  ENTER NDIEL; PLEASE:
!     NDIEL =  0 FOR NO INTERIOR DIELECTRIC,
!     NDIEL = +1 FOR AN INTERIOR DIELECTRIC ON RIGHT (OR ABOVE),
!             -1                             LEFT (OR BELOW).
!             (PRE-SET VALUE WAS 0)
 ndiel = 0
```

```
!  FOR INTERIOR DIELECTRIC, ENTER:
!     RELATIVE PERMITTIVITY RPERM
!        (RPERM <= 0 TO REMOVE DIELECTRIC MATERIAL BUT TO JUST USE
!         THE INTERIOR DIELECTRIC BOUNDARY AS A GRID CONTROL SURFACE).
!     SURFACE RESISTANCE SURSIG [OHMS/METER]
!        (NOTE SURSIG=0 IS NOT MEANINGFUL IF NLIN=1)
!     SURFACE CHARGE * FREE SPACE PERMITTIVITY = SURCHG
!     IDEL,JDEL = I OR J VALUE OF INTERFACE
!        (ONE & ONLY ONE .NE. 0, AND AT LEAST 2 IN FROM BOUNDARY)
!        ( < 0 SELECTS AUTOMATIC DETERMINATION OF IDEL,JDEL)
!        (PRE-SET VALUES WERE RPERM, SURSIG, SURCHG, IDEL, JDEL =
!         3.5, 0.0, 0.0, -1, 0)
 rperm  = 3.5
 sursig = 0.
 surchg = 0.
 idel   = -1
 jdel   = 0

!  ENTER FILE SUFFIX TO SAVE GRID
!     AND ELECTRODE PARAMETERS ON OUTPUT FILE, PLEASE
!       ( e.g. enter 31 to SAVE to file FOR031.DAT)
!       ( enter 0 to not SAVE)
 idfilr = 31

!  WRITE TO OUTPUT FILES,
!       ( e.g. enter 32 to SAVE to file FOR031.DAT)
!       ( enter 0 to not SAVE)
 idfilr = 32

! End of File
```

AIAA'84

AIAA-84-1655
Interactive Electric Field Calculations for Lasers

P. J. Roache and S. Steinberg,
Ecodynamics Research Associates Inc., Albuquerque, NM, and
W. M. Moeny, Tetra Corp., Albuquerque, NM

AIAA 17th Fluid Dynamics, Plasma Dynamics, and Lasers Conference

June 25-27, 1984/Snowmass, Colorado

INTERACTIVE ELECTRIC FIELD CALCULATIONS FOR LASERS[1]

by

Patrick J. Roache[2]

W. M. Moeny[3]

and

Stanly Steinberg[4]

Abstract

The goal of the computational effort described herein is to develop computer codes for rapidly and accurately modeling the electric fields in the cavity of lasers and switches. The designer is able to interactively perturb the laser operating parameters and/or electrode geometry, and quickly obtain new solutions. The codes use automatic generation of solution-adaptive boundary-fitted coordinate systems, and solve two- and three-dimensional problems in both steady-state and time-dependent modes.

Introduction and Overview

The design of electrodes for lasers and switches is well-defined only for unrealistically idealized conditions. The frequently used Rogowski electrode shapes are "optimal" only in the sense of producing an enhancement factor of unity, i.e., the electric field strength is no where greater than the nominal value. More importantly, the solution is based on vacuum conditions and is not a complete specification, i.e., the Rogowski shape is not closed, and must be completed by some (usually arbitrary) closure such as blending with a radius, etc. The same is true of the Chang electrodes.

The computer codes described herein address the realistic electrode design problem, including non-vacuum operation and complete electrode specification with "packaging" constraints of overall size. Using efficient finite difference methods in boundary-fitted coordinates, the ELF (ELectric Field) code makes it practical to design the electrode geometry and laser operating parameters during user-interactive sessions on a VAX computer. A single code is used for all 2-D calculations, both steady state and time dependent. Options are available for either the planar, axisymmetric, or radial electrode geometries. Boundary conditions and boundary shapes may be time dependent; in particular, an external circuit equation is provided so that electrode potential may be calculated as part of the solution, dependent on the integrated current through the cavity, rather than being specified a priori. The geometry and conductivity calculations are modularized so that they may be readily modified by the user.

Automatic grid generation is performed interactively using elliptic generating equation techniques. As an option, a solution adaptive grid generation technique is used to adapt the grid to the solution (either in the steady state solution, or within an intra-time-step iteration for a time-dependent problem) so as to increase the resolution of the maximum electric field strength (always an important design parameter) and the accuracy.

The code accounts for externally controlled or self-sustained glow discharges or other plasmas, such as arcs, by modifying the nonlinear conductivity. Various empirical formulations for (steady) electrical conductivity have been evaluated using the code. While somewhat useful, this approach has now been abandoned. A more fundamental approach has been adopted, that of solving for the conductivity by time integration of the ordinary differential equations for electron number density equation at each mesh point in the 2D or 3D grid, coupled nonlinearly to the local E-field. The electron drift velocities, etc., are obtained by table look-up of Boltzmann code solutions performed beforehand (i.e., noninteractively) for the particular gas mixture used. Code applications have included pulsed electric CO_2 lasers and a xenon flashlamp. For xenon flashlamp calculations and for streamer calculations, the temperature is also solved at each node point by time integration of an energy equation. Calculations have shown some insight into streamer formation in plasma discharges, the unsteady development of a self-sustained glow discharge, and lensing effects due to nonuniformities in external ionization sources.

For many cases studied, we find the electric field solutions to differ significantly from vacuum calculations, indicating that the commonly used Rogowski solutions and Chang solutions for the electrode shapes are far from optimal for important classes of problems. The true optimal geometry is, in fact, a strong function of the laser physics and of the operating conditions whenever significant physics are involved in the conductivity. Also, different lasers may have different optimality criteria; e.g., an electron-beam laser may be designed to give nearly uniform energy deposition in the cavity, whereas a self-sustained CO_2 laser may be designed to minimize the local extrema of the electric field strength, subject to external packaging geometry constraints, so as to minimize arcing and maximize the operating voltage.

Three-dimensional calculations are done in a separate code and are used only to design the roll-off of the electrodes in the third dimension so as not to produce a locally high electric field due to 3-D effects.

[1]Work supported partially by the U.S. Air Force Weapons Laboratory, the U.S. Air Force Office of Scientific Research, and the U.S. Army Research Office.

[2]Ecodynamics Research Associates, Inc., P. O. Box 8172, Albuquerque, New Mexico 87198.

[3]Tetra Corporation, 1325 San Mateo S.E., Albuquerque, New Mexico 87108.

[4]Ecodynamics Research Associates, Inc., P. O. Box 8172, Albuquerque, New Mexico 87198. Also, Professor, Department of Mathematics and Statistics, University of New Mexico, Albuquerque, New Mexico 87106.

GEOMETRY DESCRIPTION

We generally assume a "smart user" who can modify our Fortran subroutine to define his own family of electrode geometries. However, we also have built two such routines which appear to have a wide range of applicability.

The first geometry subroutine is based on the common Rogowski electrode family, defined by the parametric equations in Ω

$$x/g = \Omega + e^{\Omega} \cos\psi \qquad (1a)$$

$$y/g = \psi + e^{\Omega} \sin\psi. \qquad (1b)$$

The inter-electrode gap is proportional to g, and Ω = constant lines (orthogonal to ψ-lines) are used for the far-field computational boundary. The parameter value $\psi = 1/2$ on the upper electrode and $\psi = 0$ on the lower gives the most familiar "optimal" Rogowski shape, which asymptotically approaches a vertical line and gives a unity enhancement of the E-field; that is, with $E_{nominal}$ = electrode voltage/gap width, the enhancement factor = Emax/Enominal ≡ EF = 1. The upper and lower shapes and sizes are defined independently. This subroutine produces a fairly realistic, though not a "completed" family of electrodes. Since a simple closed-form vacuum solution exists, this routine has been valuable in code validation [Refs. 12,27] and in testing for the effect of the computational far-field boundary.

The second geometry routine is based on blended super-ellipses. The super-ellipse is defined (in the first quadrant) by the equation

$$(x/x_1)^p + (y/y_1)^p = 1 \qquad (2)$$

where x_1 and y_1 are the dimensions of the circumscribed rectangle of the inner electrode surface. For $p = 2$, a simple ellipse is obtained. For $p \geq 2$, the surface is smooth and tangent to the circumscribed rectangle at the end points. As $p \to \infty$, the super-ellipse approaches the circumscribed rectangle. Each electrode inner surface is then obtained as a linear blend of two super-ellipses of powers p_L and p_R with the same semi-major and -minor axes x_1 and y_1. They are blended by a factor w_L; in polar coordinates (r,θ) at the same θ, the r's are blended as

$$r = w_L \cdot r_L + (1-w_L) \cdot r_R \qquad (3)$$

This blended super-ellipse surface can be shifted, so that the initial section is a straight line, and can be further blended with an elementary electrode formed by a straight section and a radius. The inner surface can be further modified by a blended perturbation, which may be used for E-field shaping (e.g. to accomplish pre-ionization). The perturbation is a shifted sinewave, raised to a power p_p. As p_p becomes large, the perturbation approaches a rectangle, but is smooth and blends to the basic shape with continuous first derivatives for finite p_p, avoiding sharp corners. The back side or outer surface is formed by a simple radius and a straight section.

BOUNDARY-FITTED COORDINATE GENERATION

The solution for the potential ϕ in the laser cavity is obtained by solving the nonlinear elliptic equation

$$\nabla \cdot \sigma \nabla \phi = 0 \qquad (4)$$

where the conductivity σ is in general a nonlinear function of the electric field strength E where $E = \nabla\sigma$. We need accurate solutions in nonrectangular geometries and, in particular, we need to accurately solve the E-field on the boundaries. It is sometimes possible to obtain such solutions using a Cartesian coordinate system with partial cells near the irregular boundaries. However, it is well-known that such a treatment often leads to numerical instabilities and/or numerical inaccuracies. We have chosen the approach of using a numerically generated boundary-fitted coordinate system, following Thompson, et al. [Refs. 1 and 2]

In the boundary-fitted coordinate system, the physical domain (x,y) is transformed to a regular, rectangular coordinate system in coordinates (ξ,η). In order to ensure that no coordinate lines cross in the transformed (ξ,η) plane, the coordinate transformation is generated by the solution of a coupled pair of nonlinear elliptic equations for x and y given by

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} + J^2(P x_{\xi} + Q x_{\eta}) = 0 \qquad (5a)$$

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} + J^2(P y_{\xi} + Q y_{\eta}) = 0 \qquad (5b)$$

where

$$\alpha \equiv x_{\eta}^2 + y_{\eta}^2 \qquad (5c)$$

$$\beta \equiv x_{\xi}x_{\eta} + y_{\xi}y_{\eta} \qquad (5d)$$

$$\gamma \equiv x_{\xi}^2 + y_{\xi}^2 \qquad (5e)$$

$$J = x_{\xi}y_{\eta} - x_{\eta}y_{\xi}.$$

(J is the Jacobian of the transformation.)

These equations are solved for $x(\xi,\eta)$ and $y(\xi,\eta)$, with the boundary values on x and y being specified by the boundary shape and the desired mesh density at the boundaries. Second-order accurate finite difference equations are used throughout. P and Q are functions chosen to control the grid at interior points (see below).

Note that all calculations, including the calculation for the generation of the coordinates, are performed in the (ξ,η) plane rather than the physical (x,y) plane. Thompson, et al. [Ref. 1] use a point SOR iteration scheme to solve the coordinate system transformation, but this proves to be somewhat inefficient for high mesh resolution.

The approach used in the current work is to solve the system of coordinate equations using semidirect nonlinear solution techniques

previously developed for fluid dynamics applications by a present author (Roache). This system is well suited to the semidirect method because the two nonlinear equations for $x(\xi,\eta)$ and $y(\xi,\eta)$ couple only at interior points. (This contrasts to the boundary coupling of the Navier-Stokes equations, which slows convergence. (See, e.g., Refs. 3 thru 5.) The equations are first linearized about some initial guess for the solution. The linear equations so generated are generally variable coefficient, nonseparable, linear elliptic equations. These are solved using the GEM code, which is based on marching methods for linear equations [Refs. 5 thru 10]. These methods are the only practical ones (outside of brute force direct Gaussian elimination) for solving such nonseparable elliptic equations directly rather than iteratively. A sequence of linear solutions is then used in a quasi-Picard iteration to solve the nonlinearity. If the geometry gives an unfavorable cell aspect ratio for the marching method, point or hopscotch SOR iterative algorithms are used as the linear solver. For most cases studied, the GEM code, using some key double precision calculations for the VAX computers [Ref. 11] are more robust and much faster, especially for high resolution across the cavity.

The speed of convergence of these nonlinear equations for the coordinate systems depends somewhat on the accuracy of the initial guess. For a mild problem in which the electrode shapes are a slight distortion from rectangular, an adequate initial guess is obtained by simple linear interpolation between boundary values of x and y in the (ξ,η) plane. In these cases, the coordinate system transformation is solved typically in less than 10 nonlinear quasi-Picard iterations. For more severe problems, adequate initial guesses are difficult to achieve. (For example, even for a simple U-shaped cavity, linear interpolation gives a folded coordinate system with a negative Jacobian.) For these cases, it is necessary to use nonlinear continuation methods. One technique, due to Maliska [Ref. 29], is to build up the coefficients in Equation (5) from the linear decoupled case of $\alpha = \beta = 1$, $\gamma = 0$. This technique is quite simple to implement for typical electrode geometries; for further details, see Ref. 12. A more recently developed method of geometric boundary continuation appears to be more robust and requires less decision-making from the user. The continuation process is automated, and requires only that the final electrode shapes be such that the quadrilateral formed by the end points of the electrodes have positive volume. See Ref. 30 for details.

INTERIOR MESH CONTROL BY NONHOMOGENEOUS TERMS

Three different approaches are used for the evaluation of the nonhomogeneous terms P and Q in the grid-generating equations. The simplest and, in many cases, satisfactory approach is to simply set $P = Q = 0$ everywhere. This produces the method first used by Winslow [Ref. 14] in one of the early grid generation papers, originally described and used only for triangular mesh problems. It is well-known that this primitive approach gives a poor mesh for some geometries, particularly when concave surfaces are present, since the interior mesh lines avoid the boundary in these cases (e.g., see Coleman, Ref. 15). However, for most cases of interest in laser electrode and switch design, concave electrodes are not of interest and this simple approach actually packs more lines in the quasi-normal direction just where needed, in the region of highest (convex) curvature where the E-field is highest. The method also has the advantage of being easier to converge than nonhomogeneous cases. Furthermore, when the nonlinear equations are converged to a solution, the solution is always a valid grid, i.e., there are no crossings of coordinate lines (presuming that the boundary specification is consistent) whereas injudicious use of P and Q can scramble the grid.

The second code option is to directly control the spacing and angle of the coordinates for the first line off the electrode surface, using the algorithms of Steger and Sorenson [Refs. 16 and 17]. For example, the user may specify the angle to give normal coordinates at the surface and a small initial spacing. (In fluid dynamics calculations, the spacing can be specified to resolve estimated boundary layer thickness.) The algorithm also allows for the specification of certain tuning parameters which control the extent of the surface control out into the interior region. Note that it is not true, as might be naively expected, that it is always advantageous to have orthogonal coordinates, as demonstrated by Coleman [Ref. 15]. However, it is usually advantageous to have coordinates orthogonal at least at the boundary (if the conditions of the geometry at the adjacent bounding surfaces do not preclude this locally). The form of the control functions in this algorithm involves spatial exponentials, and we have found a difficult sensitivity to the initial condition and the user-input parameters. Although we have generated some good grids with this algorithm, the option is not recommended generally because it often fails to converge in the semidirect nonlinear solution procedure. (Note, however, that the GRAPE code [Ref. 17] based on this algorithm is widely circulated and extensively used, and Sorenson has published a variety of impressive grid generation solutions for split-flap airfoils, etc.)

The third code option is to generate the nonhomogeneous P and Q terms from the surface shape, following Thomas and Middlecoff [Refs. 18,19]. The source terms are chosen to have the form

$$P = \overline{P}(\xi,\eta)(\xi_x^2 + \xi_y^2) \qquad (6a)$$

$$Q = \overline{Q}(\xi,\eta)(\eta_x^2 + \eta_y^2) \qquad (6b)$$

The shape of the adjoining boundaries and the surface itself give an (approximate) evaluation for the $\overline{P}$ and $\overline{Q}$ terms on the boundaries, which are then one dimensionally interpolated in logical space ($\overline{P}$ in η and $\overline{Q}$ in ξ) into the interior. We have followed this approach, with some slight differences in the detailed implementation. Thomas evaluates the surface P and Q terms from elemental vector relations in the surface, assuming (locally) that normal grid derivatives are zero. (See also the discussion of Thomas' approach by Thompson in Ref. 20.) We prefer to evaluate the surface $\overline{P}$ and $\overline{Q}$ directly from the grid generation equations. For the surface at

$\eta = 1$, the second derivatives such as $x_{\eta\eta}$ are not set equal to zero, but are interpolated in from the $\xi = 0$ and $\xi = 1$ boundaries. The tangential derivative terms are evaluated from the specified boundary distributions (see below). For the cross-derivatives, temporary data set of the x and y coordinates is set up near each boundary, by interpolation from the adjacent bounding surfaces. (The interpolated grid points are not actually used except for the numerical evaluation of $x_{\xi\eta}$ and $y_{\xi\eta}$ in the surface $\bar{P}$ and $\bar{Q}$. For example, if the surface under consideration is highly concave, the four interpolated points in the quasi-normal direction will likely produce a local crossing of grid lines, i.e., an invalid coordinate system. But we use these points only locally and as a convenience in evaluating the cross-derivatives in the grid generation equations and not as the final grid.) This approach allows us to obtain a smooth and consistent evaluation of $\bar{P}$ and $\bar{Q}$ terms, with less restrictive assumptions about the behavior near boundaries. For most cases, the resulting grid is expected to be close to that of the original Thomas and Middlecoff method. Like the original, the idea extends readily to 3-D bounding surfaces as well, and it likewise produces P and Q terms which do not vary so radically locally, contrasted to the Steger-Sorenson algorithm. Consequently, the grid solution converges better in semidirect iterations, although with less local control of the grid parameters.

With either the second (Steger-Sorenson) or third (Thomas and Middlecoff) algorithm for the P and Q terms, a first grid is established using P = Q = 0, and the nonlinear iterations are started from this solution.

The matrix structure used for the grid solution procedure also allows the user to specify grid position precisely at interior boundaries. (The GEM solution method will fail, so hopscotch SOR is used.) This may be used to fit the grid to the boundaries of interior dielectrics or resistive electrodes, for example. However, the result will generally be a grid which is not smooth, resulting in locally high truncation error. Either the second or third option on the P and Q terms can be used to obtain some degree of grid smoothness through the interior boundary, but there are convergence difficulties. It is not known if these difficulties are fundamental or are due simply to coding errors, but at the present time this option is not operational.

SOLUTION-ADAPTIVE GRID GENERATION

The accuracy of calculations in general boundary-fitted grids can be enhanced if the grid can somehow be adapted to the solution. There are a variety of approaches to this problem; see, for example, the papers in Refs. 13, 21 and 22. However, none of these address the problem of adapting the boundary distribution of grid points, but instead concentrate on the problem of adaptation at interior points. In electrode calculations, we find the boundary distribution to be more critical. (See also Holst and Brown, Ref. 31, for inviscid transonic airfoil calculations using boundary adaptation.)

Higher accuracy and finer resolution of the maximum E-field value is obtained by a solution-adaptive strategy. A first solution is obtained in a grid whose boundary point distribution is set only by local arc length and curvature considerations. (For example, the boundary grid points may be equidistributed in arc length, or may be concentrated more near regions of high local surface curvature.) Next, a second grid is constructed so as to concentrate more grid lines near large E-field and gradients of E. (This involves an interpolation problem, since the old E-solution and the new interpolated solution do not have the same domain of definition.) The process can be repeated and converges to a final grid rapidly.

In our earlier work [Refs. 27,28] we adapted only on the maximum E-field value on the electrode surfaces. For long and nearly uniform electrodes, this algorithm does a poor job, packing too many points in a nearly uniform region, at the expense of regions with more solution structure. The presently used adaptation function is more complicated, allowing for a good deal of fine-tuning and experimentation. Along any of the boundaries, the user can specify three adaptation parameters p_f, p_p, and p_e which then pack surface points according to the equation for the surface packing factor

$$S_{pf} = \left\{1 + \left|\frac{p_f}{1-p_f}\right| \left|p_e E_n + (1-p_e)E_c\right|\right\}^{p_p} \quad (7)$$

where E_N is the local normalized E-field = E/Emax, and E_c depends on the gradient of E. Following a recommendation of Eiseman (personal communication) we evaluate E_c not just as $\phi_{\xi\xi}$, but as the complete solution curvature

$$E_c = \frac{\phi_{\xi\xi}}{(1+\phi_\xi^2)^{3/2}} \quad (8)$$

The grid point surface distribution is then obtained by equidistribution in the weighted arc length $\bar{s}$ where

$$\bar{s}(s) = \int_0^s S_{pf}\, ds \quad (9)$$

For $p_f = 0$, we have $S_{pf} = 1$, and the nodes are equidistributed in arc length with no solution adaptation. From the viewpoint of resolution of E_{max}, we would adapt only on E itself. From the viewpoint of truncation error, we would adapt only on E_c; however, this procedure is unstable if the grid iteration is continued, and even for one grid adaptation the procedure is too sensitive. Likewise for large values of p_p.

SOLUTION OF THE STEADY POTENTIAL EQUATION IN BOUNDARY-FITTED COORDINATES

The transformed nonlinear potential equation is now to be solved in the (ξ,η) plane. However, the form of the equation changes drastically, because the coordinate transformation used is nonconformal. In particular, cross derivatives of the form $\partial^2\phi/\partial\xi\partial\eta$ are generated where none existed in the physical plane. This introduces no inaccuracy in the solution (unless the grid is highly skewed) but does require that the solution method used to solve the equations be able to treat a general 9-point operator. The

marching methods for elliptic equations are capable of handling this generality, although at some penalty in computer time compared to the 5-point operator.

As expected, the convergence is sensitive to the nature of the nonlinearity in conductivity σ. We originally used a simple update in a quasi-Picard iteration, which is adequate for fairly strong nonlinearities, typically giving convergence in 10 to 15 iterations. However, calculations for highly nonlinear conductivity models (e.g., xenon flashlamp calculations) require underrelaxation and may require as many as 40 iterations.

STEADY-STATE EMPIRICAL MODELS FOR CONDUCTIVITY

In our original work, the steady state ionization S of an external electron-beam gun was modeled empirically, following Ref. 23, by the following equation.

$$S = \exp(-\gamma x) \operatorname{atan} \frac{y+a}{x} - \operatorname{atan} \frac{y-a}{x} \qquad (10)$$

where $\gamma = E/2 \cdot V$. The electron beam has a voltage V and a width 2a located at x = 0 between y = -a and y = +a. From the same reference, the nonlinear conductivity σ is given by

$$\sigma = C \cdot E^{0.45} S^{0.5}, \; C = O(1) \qquad (11)$$

In all, four different models for the conductivity were originally developed for the electron-beam lasers; they are of varying accuracy and are sensitive to the geometry of the laser electrodes. Although some of these were useful (particularly that of Ref. 23 above) for steady problems, the approach has generally been found to be of inadequate accuracy and has been abandoned in favor of a more fundamental approach.

We have more recently developed the self-sustained or externally-sustained glow discharge kinetics model described below, which is generally applicable to all geometries (and applies as well to the three-dimensional problem), and has the further generality of being a time-dependent model.

TIME-DEPENDENT SOLUTIONS

Time-dependent solutions are obtained for the problem of a transient buildup of the nonlinear conductivity. (We have not addressed the much more difficult problem associated with true electromagnetic transients with a characteristic time related to the speed of light.) The time-dependent conductivity is solved by temporal integration of the number-density equation. In our most frequently used model, this equation is

$$\frac{dn_e}{dt} = S(t) + \alpha_t n_e - \alpha_a n_e - \alpha_r n_e n_i \qquad (12)$$

where n_e and n_i are the electron and ion number densities, S is the (possibly time dependent) source, e.g., from an electron-beam gun or ultraviolet source, α_t is the Townsend coefficient, α_a is the attachment coefficient, and α_r is the recombination coefficient. The source S decays exponentially in space and may be ramped on/off in time.

This Equation (12) is nonlinearly coupled to the elliptic solution for the E-field, but only locally; i.e., the number-density equation involves no spatial derivatives. The time-integration routine has options for fully explicit differencing (fastest, first-order accurate), trapezoidal weighting (second-order formal accuracy, more accurate for slowly varying solutions), or fully implicit differencing (more robust, first-order accurate). The initial solution at time zero is always obtained by the fully implicit method, so as to assure self-consistent initial conditions. This initial solution is usually the most difficult to obtain because of the inadequacy of the initial linearization. The adaptive grid procedure can be used within an intra-time-step iteration.

Time dependency is also allowed in the boundary shape and the boundary values. In the simplest cases, these can be specified as arbitrary functions of time in the user-supplied (or modified) subprogram for the geometry description. A more useful option allows the user to specify external circuit characteristics; then the code integrates these, along with the internally calculated current in the laser cavity, to provide the potential boundary conditions for the transient start-up calculation. The capability of moving the boundary shape has been utilized in initial studies of streamer formation.

A more elaborate conductivity model was used for transient calculations of a xenon flashlamp. It involves a highly nonlinear coupling with the temperature field, also solved by integration of a time-dependent energy equation, with simple empirical relations for electron drift velocity.

$$\frac{dn_e}{dt} = \alpha_t n_e + \alpha_r \left[N^2 \left(\frac{n_e^*}{N} \right)^2 - n_e^2 \right] \qquad (13a)$$

where n_e^* is the equilibrium electron number density, and

$$\frac{n_e^{*2}}{N^2} = \frac{Q}{1+Q} \qquad (13b)$$

$$Q = \frac{C_1}{P} T^{2.5} e^{-(qV_1/kT} \qquad (13c)$$

$$\alpha_r = C_2 n_e (T/1000)^{-9/2} \qquad (13d)$$

$$\log \alpha_t = C_3 \log(E/N) + C_4 \qquad (13e)$$

$$V_d = C_5 (E/N) \qquad (13f)$$

$$\frac{dT}{dt} = \frac{1}{C_v} \left\{ \sigma E^2 - C_6 T^2 \right\} \qquad (13g)$$

$$\sigma = Q \cdot n_e \cdot V_d / E \qquad (13h)$$

where V_1, q, k, and C are various constants. Note that radiation losses have not been

accounted for above, so that the equations do not have an accurate steady state form. (Steady state operation of such a device is not of interest.)

Between the main program modules which use σ, and the user-modifiable subprogram modules which model σ, is an important subprogram module which scales σ. Scaling is critical, especially on the short word length VAX computer, both at the high and low ends. Scaling on the high end is accomplished by normalizing σ over the region of computation, noting that the solution of the potential Equation (4) is invariant to a spatially constant factor. We evaluate $\sigma_{scale} = \max \sigma$, store $\bar{\sigma}_{ij} = \sigma_{ij}/\sigma_{scale}$ and replace Equation (4) by

$$\nabla \cdot \bar{\sigma} \nabla \phi = 0 \qquad (14)$$

which has the same ϕ solutions as Equation (4). Then σ_{scale} is used to reconstitute $\sigma_{ij} = \sigma_{scale} \cdot \bar{\sigma}_{ij}$ where needed, for calculations of current density = $E\sigma$ and energy deposition = σE^2. Scaling on the low end to avoid indeterminancies in the matrix equation for Equation (14) is accomplished by limiting the minimum scaled $\bar{\sigma}$ to 10^{-3}. The ϕ-solution is affected little by thus limiting a three order-of-magnitude range in σ, and singular behavior is averted as the vacuum condition of $\sigma = 0$ is approached; vacuum solutions are obtained with $\sigma_{ij} = 1$ everywhere and $\sigma_{scale} = 0.$,

The code allows for interactive user input of all the parameters of the conductivity model; e.g., one may vary the magnitude or time-dependent ramping on/off for the external source of ionization, or alter the source offset, initial temperature, etc., as well as the electrode geometry parameters described above.

DATA PROCESSING

Our philosophy is strongly inclined toward user interaction. The methods used are fast enough so that the coarse-grid solutions roll out with little user patience required. (For example, for a 16x16 mesh, grid generation typically takes less than 10 CPU seconds on the VAX 780 with hardware floating point arithmetic, and the potential solution typically requires less than 20 seconds using the glow discharge models for conductivity.) The one exception is the preparation of detailed contour plots of the solution functionals. At the instructions of the user, these data are output to permanent files for postprocessing. However, interactive raster plots of the grid can be obtained, and the code prompts the user for tabulations of potential, E-field, conductivity, energy deposition, and when calculated) the electron number density and temperature. The code also calculates and displays the maximum E-field, the total current through the cavity, and maximum conductivity, and allows raster plots of E along the upper and lower electrodes. Convergence history data, of purely numerical interest, are also displayed for both the grid generation procedure and the potential solution procedure.

THREE-DIMENSIONAL CALCULATION METHODS

Three-dimensional calculations are done in a separate code, and are used only to design the rolloff of the electrodes in the third dimension so as not to produce locally high electric field due to 3D effects. Accuracy is again achieved by using nonorthogonal boundary-fitted coordinates, in which the electrode boundaries always lie along coordinate lines. This greatly increases the accuracy, but also increases the complexity of the formulation, especially in 3D. The complexity of the transformed problem is impressive, due to the generation of cross-derivative terms and variable coefficients which come from the transformation process.

We have used computer Symbolic Manipulation to address the problem of this complexity. Details of the approach are given in Refs. 24 and 25.

Briefly, the approach uses the computer Symbolic Manipulation code MACSYMA to perform mathematical operations of a logical nature. These are not floating point calculations, but rather operations, such as the chain rule differentiation, substitution and grouping of algebraic terms, etc.

We have used the VAX computer-based Symbolic Manipulation code VAXIMA to analytically perform the coordinate transformation of the hosted equations, to substitute the finite difference equations, to gather the coefficients together, and to write the FORTRAN code for the finite difference stencil. The matrices defining the stencil are then passed to a "canned" solver (the spatial marching methods in two dimensions or hopscotch SOR methods in three dimensions) and the solution is obtained without the user writing FORTRAN code. The same procedure is followed for the generation of the three-dimensional grid using the elliptic grid generation techniques.

The complexity of the resulting equations for both the potential equation and the coordinate transformation equations themselves preclude their complete description in this paper. Note that the simple constant coefficient Poisson equation in three dimensions, which results in a constant coefficient 7-point operator when differenced in Cartesian coordinates, transforms to a generally nonconstant coefficient, 19-point operator, in the general nonorthogonal three-dimensional coordinates. However, using the expanded (nonconservation) form of the variable coefficient Poisson equation, we can make use of symmetry and antisymmetry properties of the operator to reduce the stencil to 10 unique coefficient arrays. Advantage is taken of the virtual memory capabilities of the VAX computer family to store and manipulate these large arrays.

THREE-DIMENSIONAL CODE VALIDATION

The potential for errors in either the problem formulation or the encoding procedure always exist in complex codes, such as three-dimensional boundary-fitted codes. The need for validation was emphasized in the present work because Symbolic Manipulation was used; the resulting "psychological distance" from the work made it less likely to be satisfied with the superficial plausibility exercises based on intuitive ideas of acceptable levels of absolute errors.

We validated the codes by choosing a continuum solution for the class of problems treated by the code, and monitoring the convergence to this continuum solution by systematic truncation error convergence testing over a sequence of grid sizes. This procedure, though well-known in theory, is seldom followed in practice, especially for elliptic equations.

The selected form of the solution was chosen to ensure that all the derivative terms in the operator are exercised and that there is nonzero truncation error for finite mesh spacing even without the transformation of coordinates. Similar procedures were followed for both the potential equation code and the grid generation code, the latter being more difficult to validate because of the need for a nonlinear solution. A range of stretching parameters and continuum equation coefficients were examined in a grid successfully halved three times from 5^3 to 33^3. Details are found in Ref. 24. The near constancy of the maximum truncation error divided by the mesh spacing squared indicates that the coding is correct and that the solution is actually second-order accurate.

Symbolic Manipulation was also used to evaluate the formulation of the transformed zero-gradient boundary conditions, which are used on symmetry boundaries and as far-field boundaries on the non-electrode computational surfaces.

THREE-DIMENSIONAL CODE PERFORMANCE

The grid generation procedure is straightforward for simple three-dimensional shapes for which an initial guess of the coordinate system, based on linear interpolation in each of the three coordinate directions, is adequate. In this case, adequacy requires a nonnegative Jacobian of the transformation. However, for slightly more difficult electrode shapes, the linear interpolation results in a "tangled" mesh with negative Jacobians. In these cases, it will be necessary to use the continuation methods described earlier in 2D in order to obtain a reasonable solution.

Also, the present codes do not address the more difficult problem of generation of the grid on the nonplanar surfaces of the three-dimensional problem. For a general surface, these surface grids are presently defined by the user in a subroutine which describes the three-dimensional geometry of the laser cavity. The code has an option for generating the 3D surface grid by simple scaling of the 2D grid at constant z-planes. This is practical only for relatively simple three-dimensional cavity shapes; for the 3D rolloff calculations discussed here, this approach is entirely adequate.

Once an adequate three-dimensional grid has been generated, the solution of the nonlinear potential equation in the interior is no more difficult than in the two-dimensional case, except for the necessarily larger computer times involved. The marching methods do not work well in 3D (Refs. 5 and 6), so a hopscotch SOR method is currently used.

Both for the more difficult grid generation problem and for the solution of the potential equation, interactive solutions are practical only in relatively coarse grids, for example, a 5^3 or an 11^3 grid. These are obtained in only a few minutes on a VAX 780 computer running interactively.

REPRESENTATIVE CALCULATIONS

Representative calculations for a variety of 2D and 3D problems are shown in the figures. Brief preliminary results have been previously presented in Refs. 26 thru 28.

Figures 1 thru 7 show the results of calculations in a 16x16 grid for a planar geometry based on the Rogowski electrode family with $\psi_1 = -0.3$, $\psi_2 = +0.7$. ($\psi_1 = 0$ gives a lower electrode consisting of a straight line, and $\psi_1 = -\psi_2$ gives a symmetric reflection. $\psi_2 = 1/2$ is the "optimal" Rogowski upper electrode, which asymptotically approaches a vertical line. $\psi_2 = 1$ gives the singular limit of a slit upper electrode.)

Figure 1 is for a vacuum calculation in a non-adapted grid, with the boundary distribution of grid points set by equidistribution of arc length. The enhancement factor $EF \equiv E_{max}/E_{nominal}$ has a value EF = 1.25 for this case. However, the sharp peak in the numerical solution of surface E-field vs. arc length s, indicates inadequate resolution of the peak at the upper electrode. This plot, and the contour plot of the E-field, indicate an adequately resolved solution on the lower electrode. The (fictitious) total current (calculated with an arbitrary scaling factor of $\sigma = 1$) is calculated primarily for the information on truncation error. Since no (fictitious) current flows through either the left boundary (symmetry) or the far-field boundary ($\phi_n = 0$), the continuum solution must give a balance between the current in and out at the two electrode surfaces. The numerical solution, which is not conservative [Ref. 32] does not satisfy this condition exactly. The deviation in this case is 0.9%. (A large value of the deviation, perhaps greater than 10%, is an indication of inadequate grid refinement.)

Figure 2 repeats the case of Figure 1, but with a solution-adaptive grid. The calculated EF is now decreased slightly from 1.25 to 1.22. The peak is also better resolved. Note that the current deviation is now increased to 1.3%, probably indicating an overall increase in the truncation error, although this is considered preferable because of the increased resolution of E_{max}. Figure 3 is for an even more strongly adapted grid, in which the deviation has increased to a marginally acceptable level of 4%, and the calculated EF is reduced to 1.19.

Figure 4 is a non-vacuum steady-state calculation for the same electrode, using the flow discharge model for conductivity, and the same non-adapted grid as Figure 1. The nonlinear conductivity has increased the calculated EF of Figure 1 from 1.25 to 1.59. Neither of these solutions accurately resolve the peak, but the comparison is believed to at least qualitatively indicate the large effect of the nonlinear conductivity, even at this operating voltage of 1KV.

Figure 5 shows the effect of higher voltage on a steady-satate calculation. Using the same

electrodes, grid, and nonlinear conductivity model as Figure 4, the operating voltage is increased from 1 to 3 KV/cm. For a vacuum calculation, the operating voltage level simply scales the entire E-field solution, with no effect on the enhancement factor EF. In the present case, the effect of nonlinear conductivity is such that the calculated EF increases significantly, from 1.59 to 2.07. Again, these results should only be interpreted qualitatively, pending a high resolution solution.

Figures 6 and 7 present the early transient calculations for the same electrodes, conductivity model, and operating voltage of 1 KV/cm as Figure 4. The self-consistent initial condition at time = 0 (usually the most difficult solution in the time-dependent calculations) is shown in Figure 6. The calculated EF = 1.38, and increases to 2.31 at time = 0.5 usec in Figure 7. This is indeed a large effect of nonlinear time-dependent conductivity compared to EF = 1.25 for the vacuum calculation in the same grid, although these results can be interpreted only qualitatively without systematic grid refinement studies.

Figure 8 is a vacuum calculation for a simple Rogowski electrode with $\psi_1 = 0$, $\psi_2 = 0.7$. (It may also be interpreted as the upper half of a symmetric electrode pair.) The purpose here is to present an overlay of the fixed (non-adapted) grid (shown by dotted lines) and an adapted grid, showing how the grid is more densely clustered where the E-field needs higher resolution.

Figure 9 demonstrates the use of the geometry subroutine based on blended super-ellipses. The upper electrode formed by a simple ellipse on the inner surface of thickness 1.0 cm; the entire electrode is "packaged" in a box of dimensions 4.5 cm x 1.5 cm. The lower electrode is formed by a blend between p = 2 (left) and p = 15 (right) super-ellipses, with a left weighting factor $w_L = 0.7$. The inner surface thickness is 1.0 cm, and the "package" is dimension 5.0 cm x 2.0 cm. The irregularity in the last contour line of the E-field indicates loss of accuracy near the far-field computational boundary, due to the highly stretched grid in this region. This is not considered significant, since we are not interested in the solution in this region. The calculated enhancement factor EF is only 1.02 for this electrode, which is not optimized. For more stringent "packaging" constraints, we would expect EF to be higher, and the optimization problem would become progressively sharper. However, the indication is that this electrode family is preferable, from practical considerations, to the Rogowski and Chang electrodes.

Figure 10 is for elementary electrodes formed by a straight section and a radius for the inner surface. (The outer electrode surface, or back side, is also formed the same way in all the electrodes in this family, but the inner and outer radii are not necessarily equal.) The calculated EF = 1.19 for this case.

Figure 11 shows the same electrode as Figure 9, with a blended perturbation added to the lower electrode. The perturbation is 5% in amplitude, and extends from 2.5 cm to 3.5 cm, with a perturbation shape factor $p_p = 2.0$. The gross effect on the contour plots of the E-field is shown, but of course the details near the perturbation would require high resolution on the same scale as the perturbation.

Figure 12 simply gives an example of one set of 3D surfaces which we have used for full 3D calculations. The 3D ELF code is not nearly as highly developed as the 2D code in its options and user interface, and the grid is not adaptive in the third direction. However, transient as well as steady state calculations may be performed.

FUTURE WORK

In the near future, we intend to change the 3D solver used from the hopscotch SOR method to a multigrid technique. Other plans for future work include improved solution for nonlinear conductivity using a Monte Carlo solution, charge-separation effects near the electrodes, use of a fourth-order accurate deferred correction solution to both improve accuracy and to automate the truncation-error convergence testing, modeling of resistive electrodes, and automation of the design process.

The automation of the design process would involve nonlinearly constrained optimization of the inverse design problem. That is, the user would specify families of design parameters and design goals (e.g., the normalized energy deposition distribution in the laser cavity and/or the maximum E-field) and the code would design the electrode shape, subject to packaging restraints, etc.

The actual production of the designed electrodes is accomplished by numerically controlled (NC) machining. Presently, the coordinates are passed to the NC by microcomputer, requiring $O(10^5)$ coordinates to achieve fidelity. We are attempting to build the geometry algorithm into the interpolation basis of the NC software, so that only $O(10^2)$ geometry parameters need be passed, perhaps by configuring the Vax computer as a terminal.

Finally, we are presently examining the feasibility of applying these techniques to the full Maxwell equations with variable properties, considering true electromagnetic transients with a characteristic time related to the speed of light.

ACKNOWLEDGEMENT: Alan Lampson of Tetra Corp. was responsible for the glow discharge modeling.

REFERENCES

1. Thompson, J. F., Thames, F. C., and Mastin, C. W., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Field Containing Any Number of Arbitrary Two-Dimensional Bodies," J. Computational Physics, Vol. 15, No. 3, July 1974, pp. 299-319.

2. Thompson, J. F., Thames, F. C., and Mastin, C. W., "Boundary-Fitted Curvilinear Coordinate Systems for Solution of Partial Differential

Equations on Fields Containing Any Number of Arbitrary Two-Dimensional Bodies," NASA CR-2729 July 1977.

3. Roache, P. J., "The Split NOS and BID Methods for Steady State Navier-Stokes Equations," in Lecture Notes in Physics Series, Vol. 35, Springer-Verlag, Berlin, 1975, pp. 347-352.

4. Roache, P. J., "The LAD, NOS, and Split NOS Methods for the Steady State Navier-Stokes Equations," Computers and Fluids, Vol. 3, 1975, pp. 179-195.

5. Roache, P. J., Semidirect and Marching Methods for Partial Differential Equations, to appear.

6. Roache, P. J., "Marching Methods for Elliptic Problems: Part I, "Numerical Heat Transfer," Vol. 1, 1978, pp. 1-25.

7. Roache, P. J., "Marching Methods for Elliptic Problems: Part II," Numerical Heat Transfer, Vol. 1, 1978, pp. 163-181.

8. Roache, P. J., "Marching Methods for Elliptic Problems: Part III, Numerical Heat Transfer, Vol. 1, 1978, pp. 183-201.

9. Roache, P. J., "GEM Solutions of Elliptic and Mixed Problems with Nonseparable 5- and 9-Point Operators," Proc. LASL Conference on Elliptic Solvers, 30 June - 2 July 1980, Santa Fe, New Mexico, Academic Press, M. Schultz, ed., pp. 399-403.

10. Roache, P. J., "Performance of the GEM Codes on Nonseparable 5- and 9-Point Operators," Numerical Heat Transfer, Vol. 4, No. 4, 1981, pp. 395-408.

11. Roache, P. J., "Additional Performance Aspects of the GEM Codes," to appear in Numerical Heat Transfer.

12. Roache, P. J., "Semidirect/Marching Solutions and Elliptic Grid Generation," in Ref. 13, pp. 729-737.

13. Thompson, J. F., ed., Numerical Grid Generation Elsevier Science Publishing Co., New York.

14. Winslow, A. M., "Adaptive Mesh Zoning by the Equipotential Method," UCID-19062, 1981.

15. Coleman, R. M., "Generation of Boundary-Fitted Coordinate Systems Using Segmented Computational Regions," in Ref. 13, pp. 633-636.

16. Steger, J. L., and Sorenson, R. L., "Automatic Mesh-Point Clustering Near a Boundary in Grid Generation with Elliptic Partial Differential Equations," J. Comp. Phys., Vol. 33, No. 3, December 1979, pp. 405-410.

17. Sorenson, R., "A Computer Program to Generate Two-Dimensional Grids about Airfoils and Other Shapes by the Use of Poisson's Equation," NASA TM-81198, May 1980.

18. Thomas, P. D. and Middlecoff, J. F., "Direct Control of the Grid Point Distribution in Meshes Generated by Elliptic Equations," AIAA Journal, Vol. 18, No. 6, Jun. 1980, pp. 652-656.

19. Thomas, P. D., "Construction of Composite Three-Dimensional Grids from Subregion Grids Generated by Elliptic Systems," AIAA 5th Computational Fluid Dynamics Conference, Palo Alto, California, 1981. See also, Thomas, P. D., "Numerical Generation of Composite Three-Dimensional Grids by Quasi-Elliptic Systems," in Ref. 13, pp. 667-686.

20. Thompson, J. F., "Elliptic Grid Generation," in Ref. 13, pp. 79-106.

21. Ghia, K. N., and Ghia, U., eds., Advances in Grid Generation, ASME FED, Vol. 5, 1983.

22. Babuska, I., Chandra, J., and Flaherty, J. E., eds., Adaptive Computational Methods for Partial Differential Equations, SIAM, Philadelphia.

23. Theophanis, G. A., Jacob, J. H., and Sackett, S. J., "Discharge Spatial Nonuniformity in E-Beam Sustainer CO_2 Lasers," Journal of Applied Physics, Vol. 46, No. 5, May 1975, p. 2329.

24. Steinberg, S., and Roache, P. J., "Symbolic Manipulation and Computational Fluid Dynamics" to appear in Journal of Computational Physics.

25. Roache, P. J., and Steinberg, S., "Symbolic Manipulation and Computational Fluid Dynamics" Proceedings of the AIAA 6th Computational Fluid Dynamics Conference, Danvers, Massachusetts, July 13-15, 1983. To appear in AIAA Journal.

26. Roache, P. J., Moeny, W. M., and Filcoff, J. A., "Computational Solutions in Body-Fitted Coordinates of Electric Field in Externally Sustained Discharges," in Proceedings of the 3rd IEEE Pulsed Power Conference, Albuquerque, New Mexico, 1981.

27. Roache, P. J., Happ, H. J., and Moeny, W. M., "Unsteady 2D Electric Field Modeling with High Accuracy on Conductor Surfaces," in Proceedings of the 4th IEEE Pulsed Power Conference, Albuquerque, New Mexico, 6-8 June 1983.

28. Roache, P. J., Steinberg, S., Happ, H. J., and Moeny, W. M., "3D Electric Field Solutions in Boundary-Fitted Coordinates," Proc. 4th IEEE Pulsed Power Conference, Albuquerque, New Mexico, 6-8 June 1983.

29. Maliska, C. R., A Solution for Three-Dimensional Fluid Flow Problems in Nonorthogonal Coordinates, Ph.D. Dissertation, University of Waterloo, 1981.

30. Roache, P. J., "A Geometric Boundary Continuation Method for Grid Generation", to be submitted.

31. Holst, T. L. and Brown, D., "Transonic Airfoil Calculations Using Solution-Adaptive Grids," AIAA Journal, Vol. 21, No. 2, February 1983, pp. 304-306.

32. Roache, P. J., Computational Fluid Dynamics Hermosa Publishers, Albuquerque, 1976.

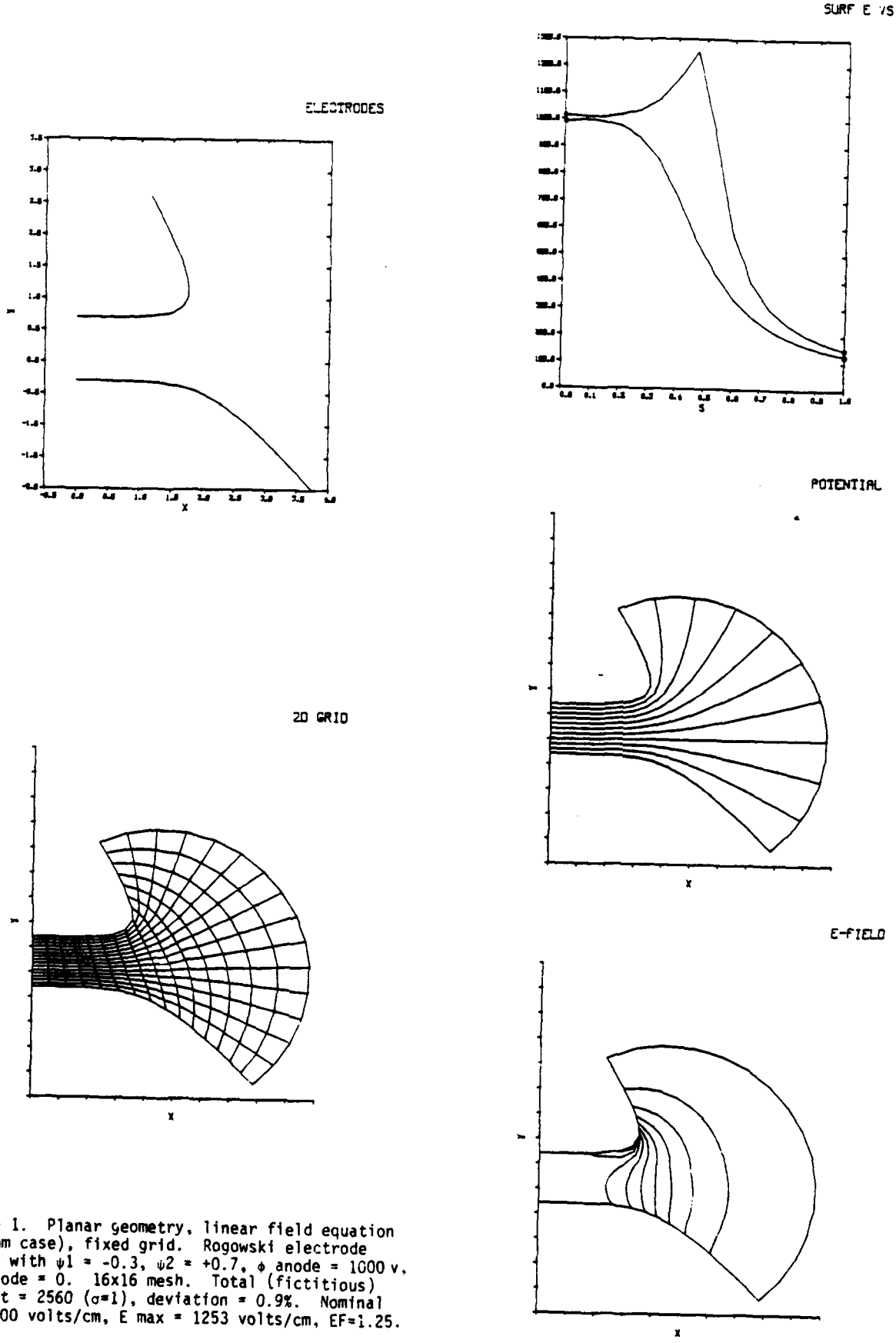
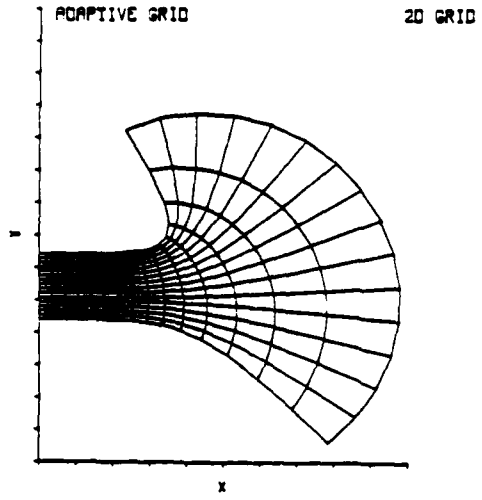
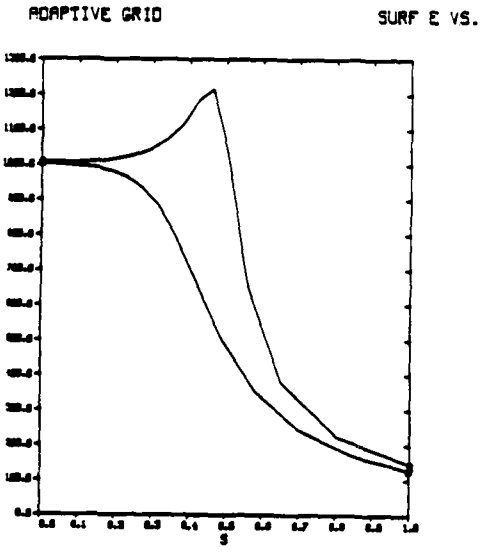


Figure 1. Planar geometry, linear field equation (vacuum case), fixed grid. Rogowski electrode family with $\psi 1 = -0.3$, $\psi 2 = +0.7$, ϕ anode = 1000 v, ϕ cathode = 0. 16x16 mesh. Total (fictitious) current = 2560 (σ=1), deviation = 0.9%. Nominal E = 1000 volts/cm, E max = 1253 volts/cm, EF=1.25.





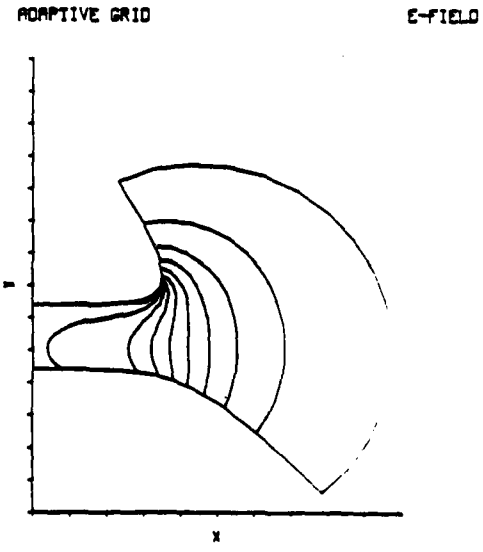
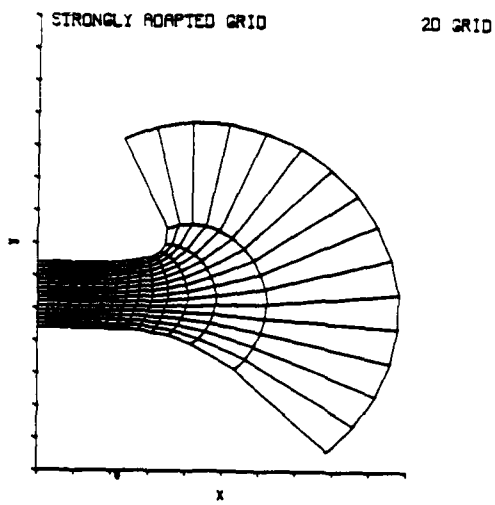
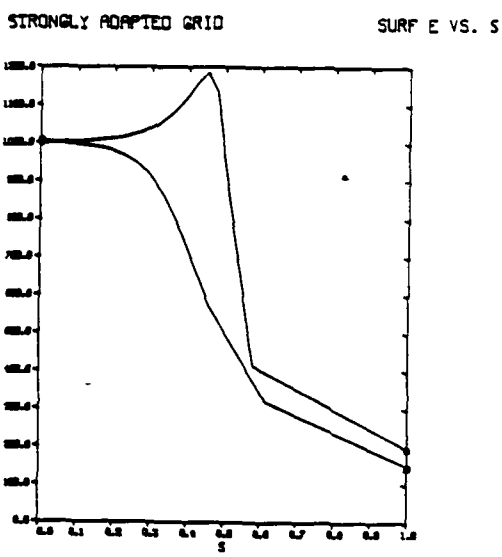


Figure 2. Same conditions as Fig. 1, solved in a solution-adaptive grid. Total (fictitious) current = 2540 (σ=1), deviation = 1.3%. EF = 1.22.





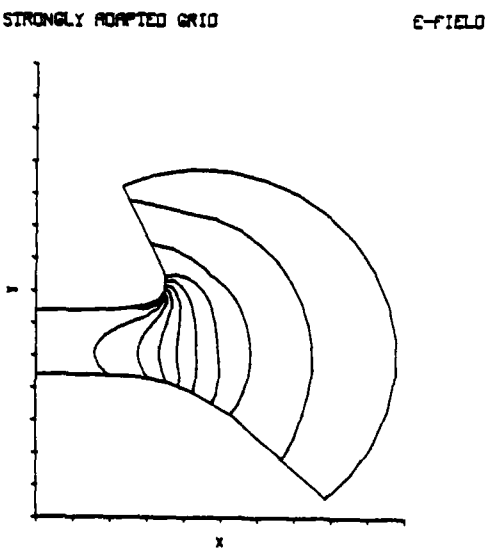
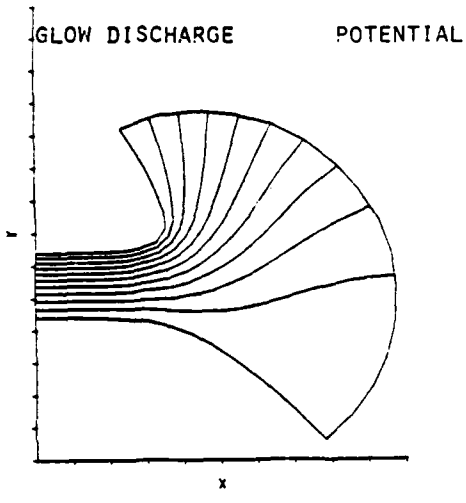
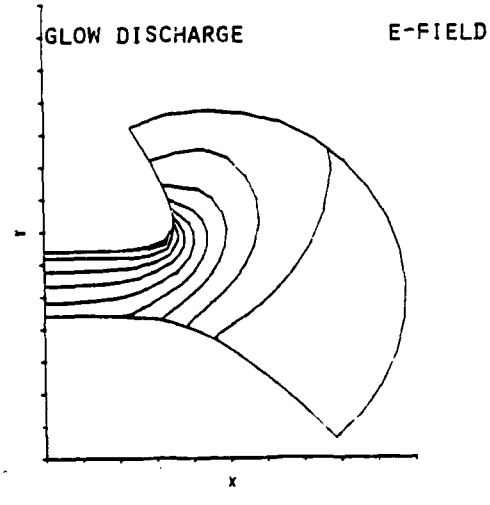


Figure 2. Same conditions as Fig. 2, strongly solution-adaptive grid. Total (fictitious) current = 2620(σ=1), deviation = 4.0%, EF = 1.19.





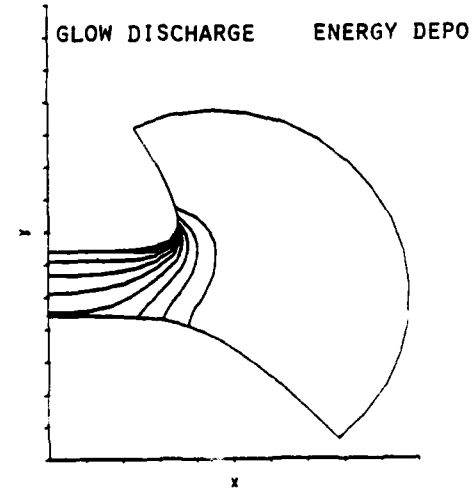
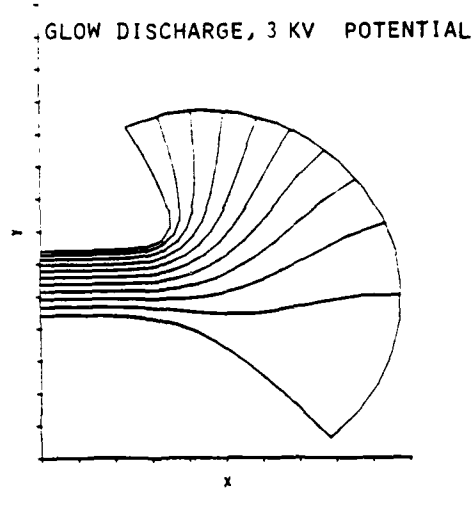
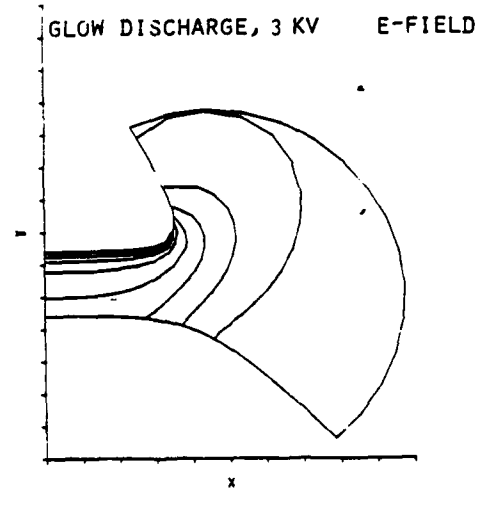


Figure 4. Same conditions as Fig. 1, with nonlinear field equation (glow discharge model). ϕ anode = 1KV, total current = 2.9, deviation = 0.6%, EF = 1.59.





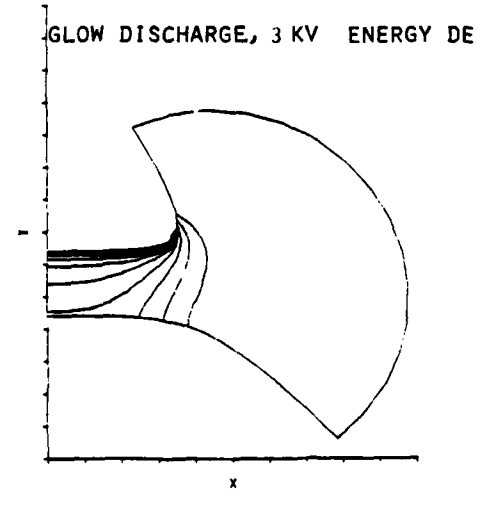
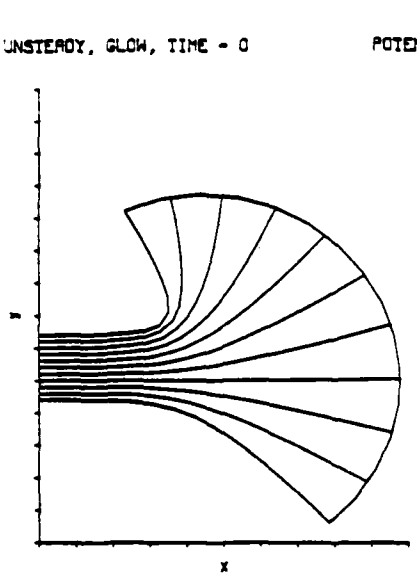
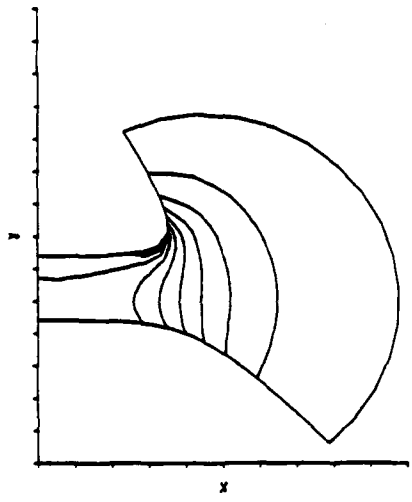


Figure 5. Same conditions as Figure 4, with ϕ anode = 3KV, total current = 17.0, deviation = 2.4%, EF = 2.07.





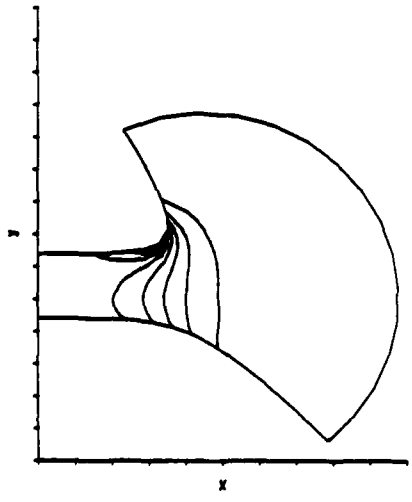
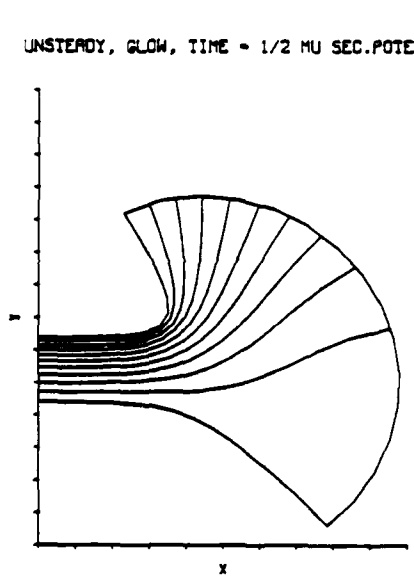
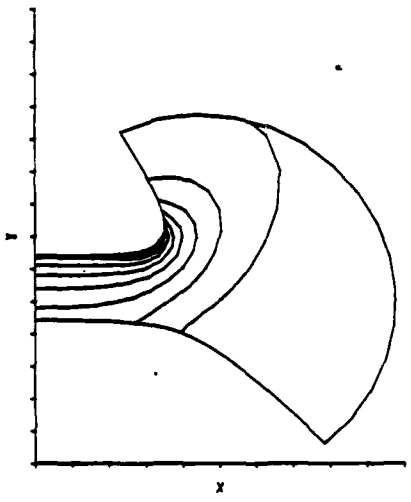


Figure 6. Same conditions as Fig. 4, transient calculation at time = 0. Total current = 0.009, deviation = 0.8%, EF = 1.38.





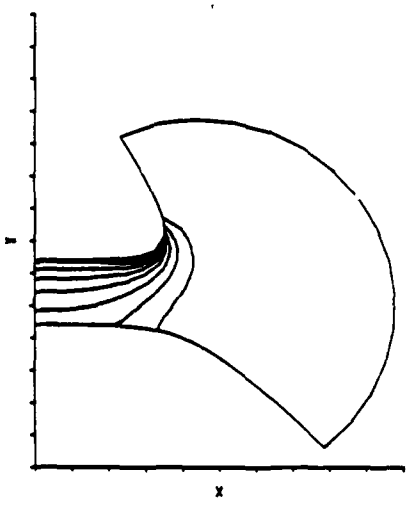


Figure 7. Same conditions as Fig. 6, transient calculation at time = 0.5 μ sec. Total current = 1.39, deviation = 0.4%, EF = 2.31.

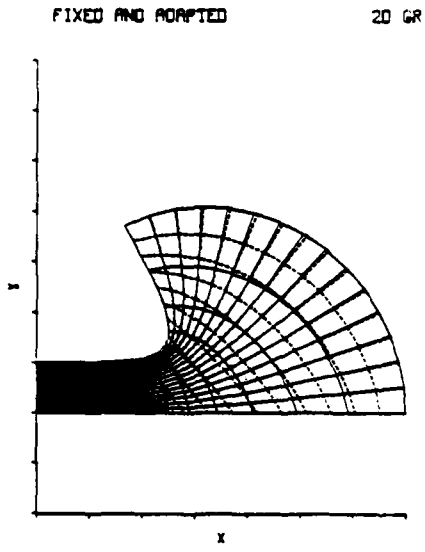


Figure 8. Same conditions as Fig. 1, with Rogowski electrode family with $\psi 1 = 0$, $\psi 2 = 0.7$. Total (fictitious) current = 3660 (σ=1), devation = 0.9%. EF = 1.23.

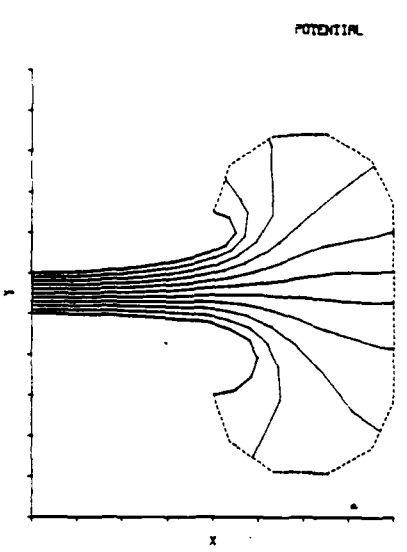
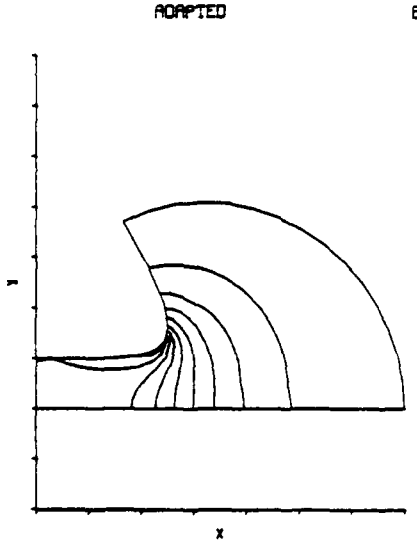
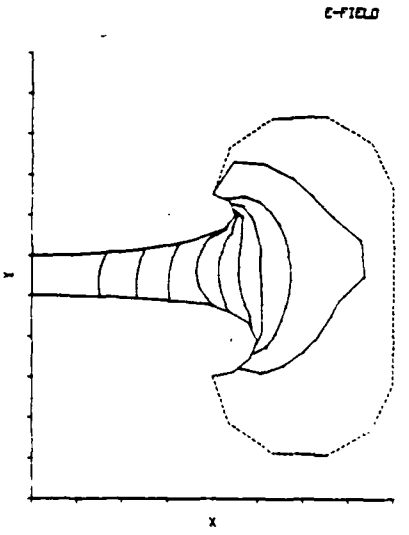


Figure 9. Same conditions as Fig. 1, with electrodes based on blended super-ellipses EF = 1.02.





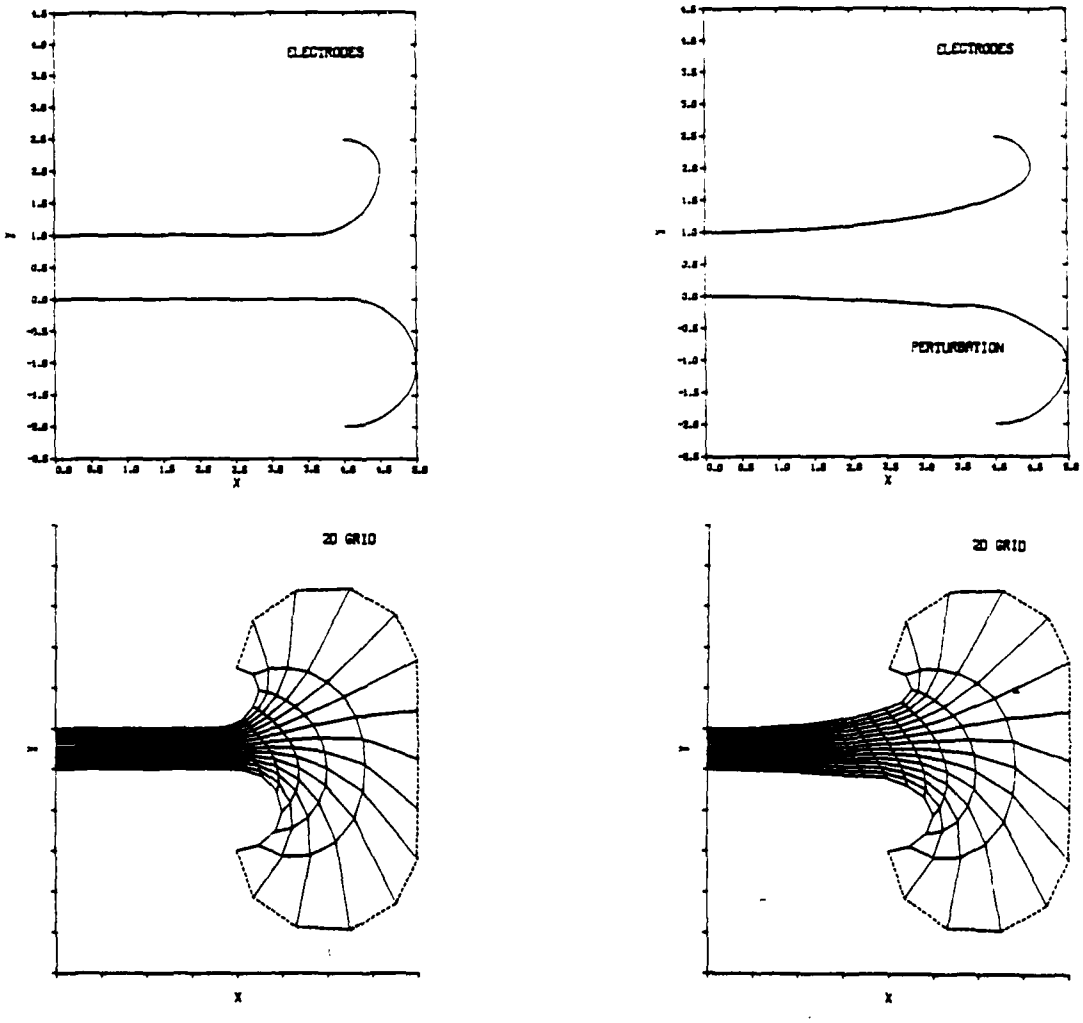


Figure 10. Same conditions as Fig. 1, with electrodes based on elementary electrode shapes consisting of straight lines plus radii. EF=1.19.

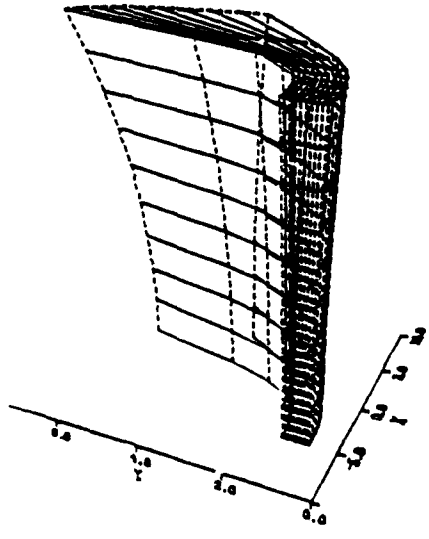


Figure 12. Example of 3D surfaces for electric field calculations.

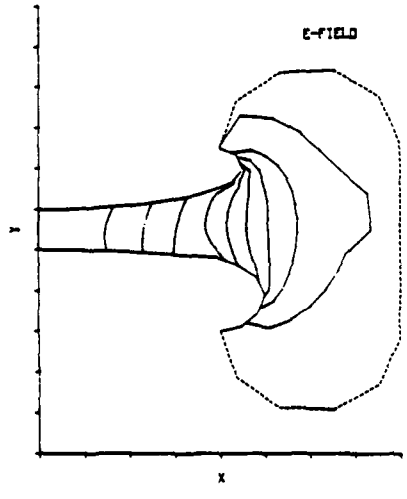


Figure 11. Same conditions as Fig. 9, with electrodes based on blended super-ellipses plus a smooth perturbation on the lower electrode.