1974

# Algorithms for Elliptic Partial Differential Equations: Metalgorithms and Selection

Elias N. Houstis
*Purdue University*, enh@cs.purdue.edu

Robert E. Lynch
*Purdue University*, rel@cs.purdue.edu

T. S. Papatheodorou

John R. Rice
*Purdue University*, jrr@cs.purdue.edu

## Report Number:

74-121

# Algorithms for Elliptic Partial Differential Equations:

# Metalgorithms and Selection

E. N. Houstis, R. E. Lynch, T. S. Papatheodorou and J. R. Rice

Purdue University

June, 1974
CSD TR 121

1. INTRODUCTION. The basic goal of this report is to formulate a framework in which to compare computational methods to solve elliptic partial differential equations. We believe that finite element methods are significantly superior to the classical and standard finite element methods and we expect that an algorithm comparison procedure as described here to provide solid and precise evidence to this effect. We assume the reader is familiar with these computational methods and we refer the reader to the report [1] by Birkhoff and Fix for an up-to-date presentation of finite element methods. We note that they mention the comparison of these two methods and indicate (page 12) the need for planned experiments on the effectiveness of various computational methods.

The next section briefly describes the problem domain considered. The third section describes the metalgorithms or computational components and organization for finite difference and finite element methods. We survey the possible choices of metalgorithm components at the end of this section and note that there are tens of thousands of them. We clearly cannot consider them all here and now. One of the major open questions in the area of algorithm selection is to decide how to judge whether significantly better algorithms have been overlooked. In Section 4 we briefly outline the abstract algorithm selection problem and then in Section 5 we propose a concrete selection procedure. This procedure is to select only between two algorithms, one for each of finite difference and finite elements. We have taken what we think of as the most direct, simple-minded versions of these two methods. The result method for finite elements has not been considered previously, but we feel that its simplicity makes it an attractive choice. We also describe the problem subclass to be used in the algorithm selection.

A comparative evaluation of the finite difference and finite element method for elliptic partial differential equations has been attempted by Schultz [7], Eisenstat and Schultz [2], [3], Rice [4], Birkhoff and Fix [1]. However we should notice that their results are based only on the asymptotic arithmetic operation count and thus their applicability is very restricted.

2. THE PROBLEM SPACE. In this section we describe the domain of problems to be considered.

a. General case. Assume $u = (u^1,\dots,u^n)$ satisfies the system of partial differential equations

$$G^i(x^1,\dots,x^m,u^1,\dots,u^1_{x^1},\dots,u^n_{x^m},u^1_{x^1x^1},\dots) = 0$$
$$i = 1,\dots,n$$

in a given domain and certain auxiliary conditions on the boundary of the domain. The problem is: Given $\varepsilon > 0$ then estimate $u$ within $\varepsilon$. We assume that the system of partial differential equations is elliptic i.e. every hyperplane is free at each point in the domain of definition. For a reasonable level of generality, we assume that the given domain $\Omega$, is open, connected subset of $R^m$ and that its boundary $\partial\Omega$ consists of a finite number of piece-wise smooth curves. Also we assume that the auxiliary conditions take the form of prescription of the values of the solution, or the values of the normal derivative of the solution, or a combination of the two, on the boundary of the domain.

This problem is extremely large and must be reduced considerably in any currently practical study of computational methods. The selection procedure discussed later specializes this space to linear problems and their choices a hopefully representative sample from this subspace.

b. Problem attributes. As seen from the definition of the Problem Space, each problem is determined by the following four attributes:

(i) the geometry of the domain of definition,

(ii) the differential operator,

(iii) the auxilliary conditions, and

(iv) the specified accuracy.

## 3. METALGORITHMS FOR FINITE DIFFERENCE AND FINITE ELEMENT METHODS.

According to Rice [6] the work metalgorithm means a framework or theory to study algorithms. A metalgorithm consists of a set of blocks or components (possibly in flowchart form) and it represents a class of algorithms, each of which has the form and attributes specified by the metalgorithm. Two metalgorithms are presented in the next two subsections and a survey of metalgorithm components is given in section 3.3.

### 3.1 Metalgorithm for finite difference methods.

This metalgorithm consists of the following components:

(i) a grid of points over the domain $\Omega$ that we call pivots if they lie in $\Omega$ or on $\partial\Omega$. We distinguish them to interior and boundary pivots according to whether their surrounding grid points are pivots or not.

(ii) a processor that generates a set of algebraic equations from the operator equations

(iii) a processor that generates a set of algebraic equations from the auxiliary conditions

(iv) an equation solver that solves the system of interior pivot equations of (ii) and boundary pivot equations of (iii) and

(v) a processor that performs a measurement of the results and terminates the algorithm.

The computations represented by this metalgorithm consist of the determination and/or execution of these 5 components in the sequence listed.

3.2 Metalgorithm for finite element methods. This metalgorithm consists of the following components:

(i) partition

(ii) approximate space

(iii) operator equations

(iv) auxiliary conditions

(v) an equation solver that solves the equations generated by components (iii) and (iv),

(vi) measurement of results.

The first component subdivides the domain of interest into finite elements, the second one generates a finite set of basis functions, the third forms a set of algebraic equations from the operator equations, the fourth forms a set of algebraic equations from the auxiliary conditions, the fifth solves the system of algebraic equations resulted by the components (iii), (iv) and the sixth performs a measurement of the results and terminates the algorithm. The computations represented by this metalgorithm consist of the determination and/or execution of these 6 components in the sequence listed.

3.3 Survey of Metalgorithm Components. We present a list of possible choices of the components of the two metalgorithms cited above. We start with the components of the finite difference metalgorithm:

(i) <u>grid</u>

1. uniform
2. graded
3. special cases near singularities
4. mesh boundary

(ii) <u>interior pivot equations</u>:

1. 5-point star
2. 9-point star
3. special schemes
4. higher order difference approximations

(iii) <u>boundary pivot equations</u>:

for the treatment of Dirichlet boundary conditions in the plane

1. interpolation of degree zero
2. interpolation of degree one
3. interpolation of higher degree
4. unsymmetrical 5 point star or higher order

for the treatment of normal derivative boundary conditions

1. 5-point star
2. 9-point star
3. 11-point formulae
4. higher order differences and rectangular boundaries
5. higher order differences and curved boundaries

(iv) <u>equation solver</u>

1. Gauss elimination
2. nested dissection
3. SOR
4. Block and line SOR
5. SSOR

(v) measurement of results:

1. timing of parts of the computation
2. estimation of error
3. computation of solution at non-mesh points
4. other properties of the solution
5. other properties of computation

We note that there over $4 * 4 * 4 * 5 = 320$ combinations of these components, not counting the possibilities for the final one. When one realizes that there are further possibilities not mentioned here plus a variety of significant variations of each component, one concludes that there are at least 10,000 distinct "methods" for finite differences. This does not count seemingly trivial variations in program implementation that do, in fact, affect the computations significantly. It is clear that we cannot consider all these possibilities now.

Next we present a list of possible choices of the components for the finite element metalgorithm

(i) Partition

1. rectangles (uniform, graded, special cases near singularities, special mesh boundary)
2. triangles
3. isoparametric elements (triangular and quadrilateral)
4. tetrahedra
5. "brick" type elements

(ii) Approximation space

1. One dimensional piecewise polynorials

| Element type | Smoothness | Joining conditions |
|---|---|---|
| Linear | $C^0$ | 1 |
| Quadratic | $C^0$ | 1 |
| Cubic Hermite | $C^1$ | 2 |
| Cubic Splines | $C^2$ | 3 |

2. Two dimensional triangular elements

| Element type | Smoothness | Joining conditions |
|---|---|---|
| Linear | $C^0$ | 3 |
| Quadratic | $C^0$ | 6 |
| Cubic | $C^0$ | 10 |
| Cubic $Z_3$ | $C^0$, $u_x$ $u_y$ continuous at vertices | 10 |
| Restricted $Z_3$ | As above, plus equal coefficients of $x^2y$ and $xy^2$ | 9 |
| Quintic | $C^1$, $u_{xx}$, $u_{xy}$ $u_{yy}$ continuous at vertices | 21 |
| Reduced quintic | $C^1$, $u_{xx}$, $u_{xy}$, $u_{yy}$ continuous at vertices | 18 |
| Cubic macrotriangle | $C^1$ | 12 |
| Quintic | $C^2$, $u$, $u_x$, $u_y$ $u_{xx}$, $u_{xy}$, $u_{xy}$ continuous at vertices | 18 |

3. Two dimensional rectangular elements

| Element type | Smoothness | Joining conditions |
|---|---|---|
| Bilinear | $C^0$ | 4 |
| Biquadratic | $C^0$ | 9 |
| Restricted biquadratic | $C^0$ | 8 |
| Ordinary bicubic | $C^0$ | 16 |
| Hermite bicubic | $C^{1,1}$ | 16 |
| Splines of degree $k - 1$ | $C^{k-2,k-2}$ | $k^2$ |
| Hermite, degree $k-1 = 2q-1$ | $C^{q-1,q-1}$ | $k^2$ |
| Serendipity $p > 2$ | $C^0$ | $4p$ |

(iii) Operator equation: Criteria used to derive approximate algebraic equations

1. Ritz method
2. Galerkin
3. least-squares
4. Collocation

(iv) Auxiliary condition equations

1. Approximate spaces that satisfy the auxiliary conditons
2. Approximate spaces that satisfy the auxiliary conditions on a piecewise linear approximation of the boundary.
3. Approximate spaces that satisfy the auxiliary conditions on a piecewise Lagrange interpolation of the boundary
4. Approximate spaces that satisfy the auxiliary conditions on a piecewise Hermite interpolation of the boundary
5. Approximate spaces that satisfy the auxiliary conditions on a modified boundary
6. Approximate the auxiliary conditions on the boundary or on a modified boundary

(v) <u>Equation solver</u>

1. Gauss elimination

2. Nested Dissection

3. Iteration

(iv) <u>Measurement of results</u>

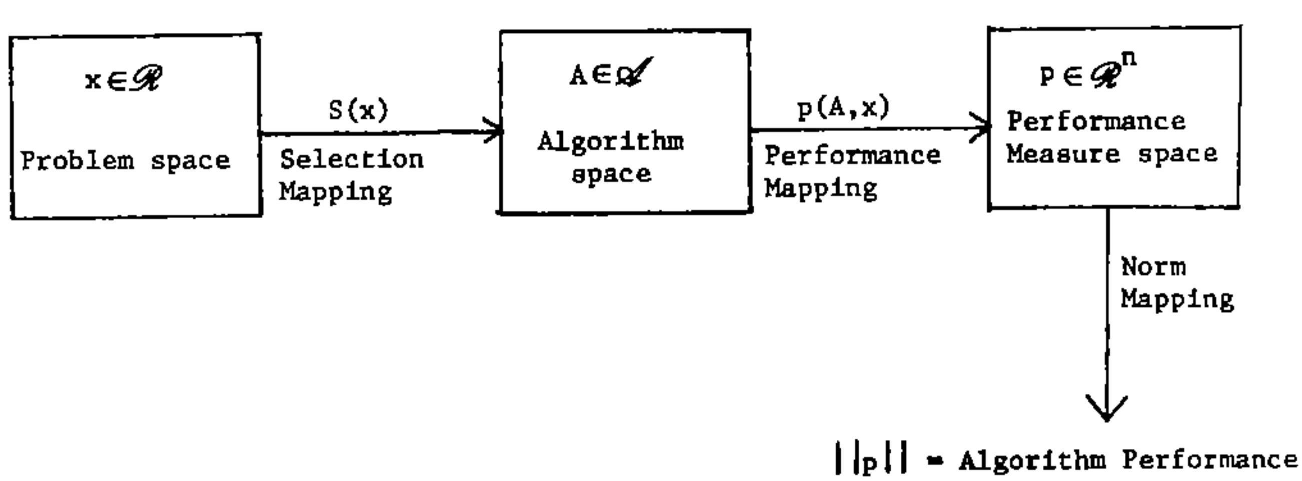
Same as for finite differences.

We may estimate the number of realizations of this metalgorithm in the same way as before. We obtain at least 5 * 8 * 4 * 6 * 3 = 2880 basic possibilities which lead to at least 100,000 significantly distinct choices for a finite element computational method (computer program).

## 4. THE ALGORITHM SELECTION PROBLEM

Following Rice [5] we present a "Basic MODEL", state the "Algorithm Selection Problem" and some selection criteria.

A. <u>BASIC MODEL</u>. We describe the basic abstract model by the following diagram



where we denote by

$\mathscr{P}$ = Problem space or collection

x = Member of $\mathscr{P}$, problem to be solved

$\mathscr{A}$ = Algorithm space on collection

A = Member of $\mathscr{A}$, algorithm applicable to problems from $\mathscr{P}$

S = Mapping from $\mathscr{P}$ to $\mathscr{A}$

$\mathscr{R}^n$ = n-dimensional real vector space of performance measures

p = Mapping from $\mathscr{A} \times \mathscr{P}$ to $\mathscr{R}^n$ determining performance measures

$\| \;\|$ = Norm on $\mathscr{R}^n$ providing one number to evaluate on algorithm's performance on a particular problem

B. The objective of "Algorithm Selection Problem" is to determine $S(x)$ so as to have high algorithm performance. Two of the most obvious selection criteria are the following:

Best Selection Choose that selection mapping $B(x)$ which gives maximum performance for each problem:

$$\|p(B(x),x)\| \geq \|p(A,x)\| \qquad \text{for all} \quad A \in \mathscr{A}$$

Best Selection from a Subclass of Mappings and Problems. One is to choose just one algorithm from a subclass $\mathscr{S}_0$ to apply to every member of a sublclass $\mathscr{P}_0 \subset \mathscr{P}$. Choose that selection mapping $S^*(x)$ from $\mathscr{S}_0$ which minimizes the performance degradation for all members of $\mathscr{P}_0$:

$$\max_{x \in \mathscr{P}_0}[\|p((B(x), x)\| - \|p(S^*(x),x)\|] \leq \min_{S \in \mathscr{S}_0} \max_{x \in \mathscr{P}_0}[\|p(B(x),x)\| - \|p(S(x),x)\|]$$

where as an Algorithm Subclass we use later a small number of programs.

Next, as performance measures we choose the computer time, (or arithmetic operation as a machine independent measure) and memory required.

Finally, the selection mapping is a constant in our later discussion.

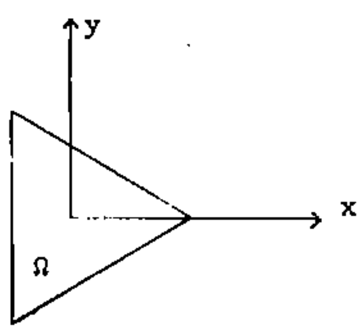
## 5. SPECIFIC CHOICES PROPOSED FOR A SELECTION

### 5.1 The Problem Subclass

In this section we describe ten partial differential equations that we might consider.

The list given below provides only the first three attributes of the Problem. In all cases we use three different specified accuracies $\varepsilon = 10^{-3}, 10^{-6}, 10^{-9}$

I. The torsion problem for an equilateral triangle

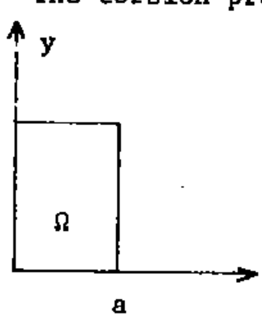



Operator eq. $u_{xx} + u_{yy} = -2$ on $\Omega$

Auxiliary cond. $u = 0$ on $\partial\Omega$

true solution: $u = \frac{1}{18}(2\sqrt{3}x + 1)(\sqrt{3}x - 3y - 1)(\sqrt{3}x + 3y - 1)$

II. The torsion problem for a bar of solid rectangular cross section

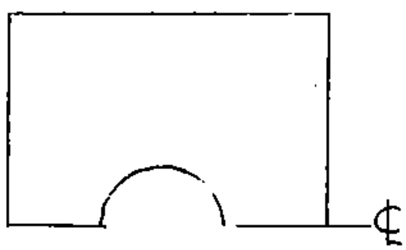



Operator eq. $u_{xx} + u_{yy} = -2$ on $\Omega$

Auxiliary cond. $u = 0$ on $\partial\Omega$

III. Torsion of a hollow bimetalic shaft.

Operator eq. $\frac{\partial}{\partial x}(\frac{1}{G}\frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(\frac{1}{G}\frac{\partial u}{\partial y}) + 2\theta = 0$




where

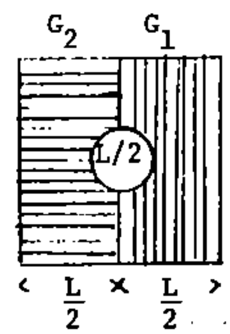
u is the stress function

G is the shear modulus

$\theta$ is the angle of twist per unit length of the shaft

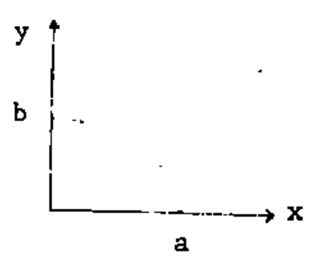
Auxiliary condition

u = 0 on the external boundary




$G_2/G_1 = 3$

IV. Torsion problem for an elliptical shaft




Operat. eq. $u_{xx} + u_{yy} = 0$ on $\Omega$

Auxil. eq. $u = \frac{1}{2}(x^2 + y^2)$ on $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$

true sol. $u = \frac{1}{2}\frac{a^2 - b^2}{a^2 + b^2}(x^2 - y^2) + \frac{a^2 b^2}{a^2 + b^2}$

Operat. eq. the same

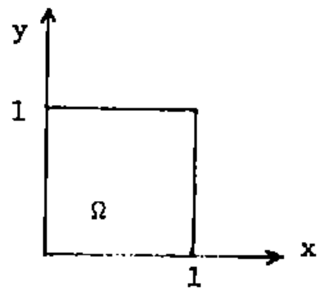
Auxil. eq. $\frac{\partial u}{\partial \nu} = y \cos(x, \nu) - x \cos(y, \nu)$

on $\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1$ and

u = 0 on the axis of coordinates

true solution: $u = -\frac{a^2 - b^2}{a^2 + b^2}\, xy$

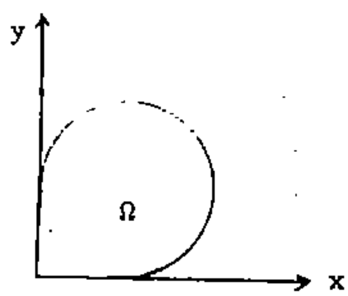
V. Steady temperature distribution in a rod in which heat was generated according to exponential law




Operator equation: $u_{xx} + u_{yy} + \frac{1}{2}(1 + e^{u}) = 0$ on $\Omega$

Auxiliary condition: $u = 0$ on $\partial\Omega$

VI. Torsion problem for a circular shaft



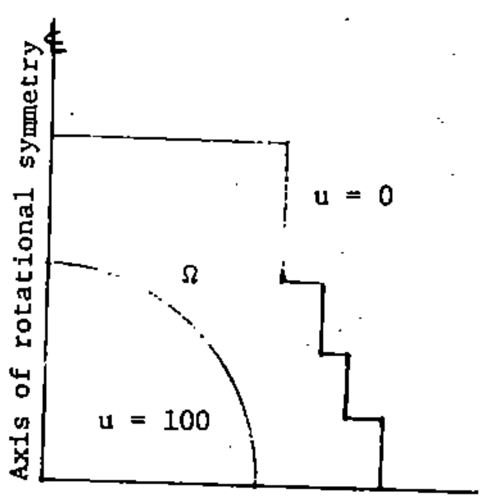

Operator eq. $u_{xx} + u_{yy} = 0$ on $\Omega$

Auxiliary cond.: $\frac{\partial u}{\partial \nu} = \frac{1}{R} \frac{\sin\phi - \cos\phi}{3 + 2\sin\phi + 2\cos\phi}$

where $\tan\phi = (y - R)/(x - R)$

true solution: $u = \tan^{-1} y/x$

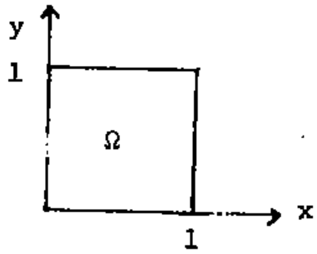
VII. Axi-symmetric heat flow.




Operat. eq. $\frac{\partial}{\partial r}\left(rk \frac{\partial u}{\partial r}\right) + \frac{\partial}{\partial z}\left(rk \frac{\partial u}{\partial z}\right) = 0$ on $\Omega$

$k$ is the conductivity

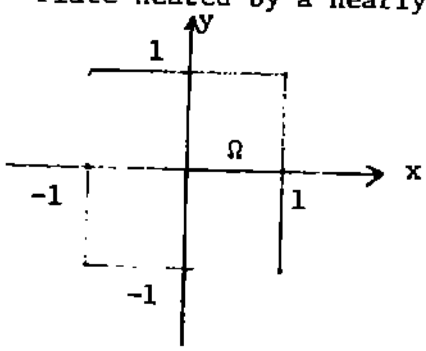
VIII. Heat conduction problem




Operator eq.: $u_{xx} + u_{yy} = -1$ on $\Omega$

Auxiliary cond.: $u = \frac{\partial u}{\partial \upsilon}$ on $\partial\Omega$

X. Plate heated by a nearly oblong element.




Operator equat.: $u_{xx} + u_{yy} = \frac{1}{[1+(.4x)^6][1+(.7y)^8]}$

Auxiliary cond.: $u = 0$ on $\partial\Omega$

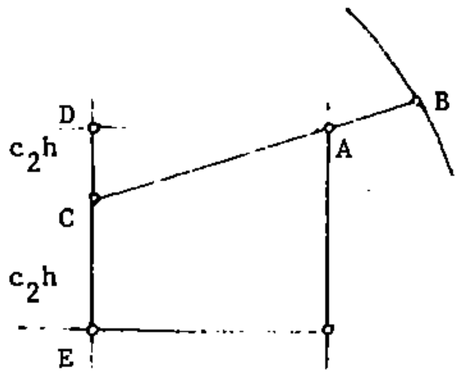
## 5.2 The Finite Difference Method

In this section we specify a choice for the components of the finite difference metalgorithm

a. Grid: uniform grid

b. Interior pivot equations: we require the quadratic Lagrange interpolant of the solution at 5-point star net to satisfy the operator equation at the center point of the star. Notice that it leads to five point star difference approximation to a linear eliptic operator.

c. Boundary pivot equations: First, for Dirichlet boundary conditions we use a linear interpolation operator to find the approximate solution at boundary pivots using the points on the boundary that lie on the mesh lines. Second, for Neumann boundary conditions of the form $A_1(s)u + A_2(s)\frac{\partial u}{\partial v} = A_3(s)$ on $\partial\Omega$, we use a non-symmetric five-point scheme described in the following way




$$1 \le c_1 \le \sqrt{2}, \quad c_2 \ge 0, \ c_3 \ge 0$$

$$c_2 + c_3 = 1$$

for the boundary pivot A we can write down the difference equation

$$A_1(B) \quad V(A) + A_2(B) \frac{1}{c_1 h} \quad \frac{c_3 \quad V(D) + c_2 \quad V(E)}{c_2 + c_3} - \quad V(A) \quad = A_3(B)$$

where V denotes the approximate solution.

d. Equation solver: we choose Profile Gaussian elimination. We realize that various iterative methods may be superior to Gaussian elimination. They are, however, more difficult to use in the variety of problems we consider because of unknown rates of convergence and relaxation parameters. We expect to be able to make an a posteriori analysis which indicates the results one would obtain by replacing Gaussian elimination by iterative methods.

## 5.3. The Finite Element Method

In this section we specify a choice for the components of the finite element method

a. We subdivide the domain of definition in rectangles and in some cases we overlap the boundary by a finite number of rectangles.

b. Approximate space: we choose the bicubic Hermite rectangular element

c. Operator approximation criteria: Collocation. In order to determine part of the degrees of freedom of the approximate solution we assume that it satisfies the operator equation at four Gaussian points inside each rectangular element.

d. Auxiliary conditions approximation criteria: Collocation. The rest of the degrees of freedom of the approximate solution i.e. $4S_b + 4$ or $4S_b$, where $S_b$ = number of boundary sides, are determined by requiring the approximate solution to satisfy the auxiliary conditions. For each piece of the boundary we use 8 points except in the case of non closed boundary in which case 10 points are used for two particular boundary sides.

e. As solver of the collocation equations we again choose the profile Gaussian elimination.

f. The same measures as in the finite difference method.

## REFERENCES


1. Garrett Birkhoff and George J. Fix, [1974], Higher-order linear finite element methods. AEC and ONR report, May, 1974.

2. S. C. Eisenstat and M. H. Schultz, [1972], Computational aspects of the finite element method, Research report number 72-1, Yale University dept. of Computer Science.

3. S. C. Eisenstat and M. H. Schultz, [1973], Complexity of partial differential equations. Proceedings of conference on Complexity of sequential and parallel numerical algorithms.

4. J. R. Rice, [1973], On approximation theoretic methods for operator equations. Unpublished manuscript.

5. J. R. Rice, [1974], The algorithm selection problem. Abstract Models, CSD TR 116, May 1974.

6. J. R. Rice, [1973], A metalgorithm for adaptive quadrature to appear in J. Assoc. Comp. Mech.

7. Martin H. Schultz, [1972], The Computational Complexity of Elliptic Partial Differential Equations. Proceedings of IBM Symposium in Complexity of Computer Computations.