Computers & Structures Vol. 22, No. 1, pp. 1–12, 1986
Printed in Great Britain.

0045-7949/86 $3.00 + .00
Pergamon Press Ltd.

COMPUTER APPROACH TO THE R-FUNCTIONS METHOD OF SOLUTION OF BOUNDARY VALUE PROBLEMS IN ARBITRARY DOMAINS

J. KUCWAJ and J. ORKISZ
Technical University of Cracow, Poland

(*Received* 16 *February* 1983; *received for publication* 19 *April* 1983)†

Abstract—An analytically numerical method of solution of boundary-value problems is considered in arbitrary domains that may be concave and/or multiconnected. An essential feature of this so-called R-functions method (RFM) is a conversion of logical operations performed on sets (relevant to subdomains of which the considered domain is composed) into algebraic operations performed on elementary functions. The solution by the RFM is realized in two phases. In the first phase, an analytical formula for the so-called "general structure of solution" (GSS) is derived. GSS is a mapping that still contains undetermined function(s) but exactly satisfies all the prescribed boundary conditions. In the second phase, which is usually of numerical character, such function(s) is approximately evaluated by means of any suitable discrete method in order to satisfy the governing differential equation, which we consider or to minimize a relevant functional.

Numerous tedious analytical operations, especially differentiations of complicated elementary functions, are necessary to derive GSS. In the original version of the method these had to be manually performed. This prevented many potential users from applying the RFM. Thus the main object of this work is to use the symbolic programming in order to obtain a fully computerized approach to the R-functions method. Both GSS itself and the results of all required operations performed on it are automatically obtained by the computer in an analytical form and written as FORTRAN subroutines ready for use in calculations. The Tschebychev approximation of undetermined function(s) and the least squares procedure complete this approach. As implemented only a few simple dates are required from the user. Some numerical examples are presented and discussed here. Suggestions are made as to areas of further research.

The RFM may be applied to a wide class of linear and nonlinear boundary value problems in mechanics with the linear boundary conditions.

1. INTRODUCTION

Many physical and technical problems result in the solution of boundary value problems given either in a local or in a global formulation. Since exact solution of these problems is usually not possible, various approximate methods are applied. The analytical approach enables us to get a solution with only a few D.O.F., and its use is usually limited to regular domains. Numerical methods, although rather poor in the case of singular problems, will handle arbitrary domains. However, they usually involve a significant number of D.O.F. causing increased demands for both computer time and capacity.

Nowadays the analytical-numerical approach is gaining in popularity. Methods of this type have the advantage of yielding satisfactory results with relatively few unknowns (D.O.F.) even in cases of arbitrary domains. As examples the boundary element[2] and global element[4] methods may be quoted as well as so-called R-functions method (RFM). This last method has been the subject of our investigations which are presented here. It was introduced by V. L. Rvachev[21] and was successfully applied in the last decade by his group in the solution of various boundary-value problems of applied mathematics and engineering[22, 24–27]. The RFM is realized in two steps. In the first step which is of an analytical nature, the solution of the problem is determined in the form of a so-called "general structure of the solution" (GSS). GSS strictly satisfies all prescribed boundary conditions and still contains some undetermined free functions. In the second step, usually a numerical one, these free functions are found then by means of any suitable known method in order to satisfy the appropriate differential equations or to minimize a given functional.

Let us briefly explain the main idea of the method, using the example of deflections of a simply supported beam carrying a uniformly distributed load q and having prescribed boundary conditions $v(0) = 0$, $v(1) = \frac{1}{2}$. Instead of direct searches for deflections $v = v(x)$, we may look for the solution in the form

$$v = x(1-x)\Phi + \frac{1}{2}x = \omega\Phi + \phi. \tag{1}$$

Boundary conditions are always satisfied for any arbitrary undetermined function Φ, since $\omega = x(1-x)$ and $\phi = \frac{1}{2}x$ meet the homogeneous and non-homogeneous boundary conditions respectively. If we assume a quadratic representation for Φ and apply, e.g. the collocation method to the equation $v'' = -M(x)/\mathrm{EI}$ describing deflections of the beam with flexual stiffness EI, we may finally determine the function $\Phi = q(1^2 + 1x - x^2)/24\mathrm{EI}$.

In eqn (1) the expression $B = \omega\Phi + \phi$ is an example of GSS. Like in this case GSS always has to correspond to the type of boundary conditions prescribed and may be presented in the general form as $B = B(\Phi; \omega, \phi)$. The meaning of the ϕ, ω and ϕ functions is similar to the example presented above. The RFM gives us general indications of how to build the GSS for a specific class of problems. The theory of R-functions[21] enables us to form ω and ϕ

† The publisher regrets the delay in publishing this article.

functions for any 1-D, 2-D or 3-D domain of arbitrary shape. Multiconnected and concave domains are also admitted.

The GSS have been derived for several classes of problems. These are plate problems[21, 22], axisymmetric problems[26], torsion of bars[21], linear fracture mechanics[19], contact[24], heat conduction problems, and many other ones in engineering and applied physics[27]. The rigorous theoretical bases of the method have been presented in numerous papers and are summarized in the monographs[21, 25]. Some of the most important notions and definitions of the RFM and RF theory, necessary for our purposes, will be very briefly given in the next section.

Despite many intrinsic advantages of the RFM, which will be discussed later, its clear handicaps are that complex formulas for specific GSS are usually needed, and that laborious operations are required to be performed on these formulas. In the original version of the method[21, 25] these were done manually and the computer was used only in the second step of the method to perform the numerical calculations. This obviously prevented many potential users from applying the RFM.

In the present paper an attempt has been made, therefore, to computerize fully an approach to the RFM by means of symbolic programming. Using this method the GSS may be directly developed (step one) by the computer. The GSS is obtained then in the form of FORTRAN subroutines written in the source code and ready for direct applications in the second step of the method.

Examples of this technique applied to various boundary value problems will be presented in Section 4. Conclusions concerning current results will be presented and further research suggestions will be discussed in Section 5.

Symbolic programming has proven to be an efficient way of approaching many problems. In its initial applications only formal differentiation [6–8, 10] was carried out. Symbolic manipulations were later used to obtain solutions to several classes of problems is applied mechanics[6, 12], including nonlinear ones[12, 17]. Our preliminary experience [5, 9, 11, 19], shows that if symbolic operations are used, the RFM may be fully computerized in both the analytical and numerical parts. As implemented not numerous and relatively simple input data are required from the user. This removes the main obstacle preventing wider application of the method.

2. R-FUNCTIONS METHOD—BASIC NOTIONS AND INFORMATION

2.1 *Formulation of the problem*

Consider a boundary value problem given in a domain Ω either in the local formulation

$$Au = f \text{ in } \Omega \subset \mathbf{R}^n,\ u \in X(\Omega),$$

$$l_i u_i = \phi_i \text{ on } \partial\Omega_i \subset \partial\Omega, \quad i = 1, 2, \ldots, m, \quad (2)$$

$$\bigcup_{i=1}^{m} \partial\Omega_i = \partial\Omega,$$

or in a relevant global one

$$\int_\Omega F(u)\,\mathrm{d}\Omega + \int_{\partial\Omega} G(u)\,\mathrm{d}S = \min_u, \quad (3)$$

where A, F, l_i, G are operators defined over a space of functions determined in Ω and $\partial\Omega$ respectively.

By the general structure of solution (GSS) of this problem we define a mapping $B: \mathcal{M} \to X(\Omega)$, that satisfies the following conditions:

$$l_i B(\phi) = \phi_i \text{ on } \partial\Omega_i, \quad i = 1, 2, \ldots, m\ \forall \phi \in \mathcal{M} \quad (4)$$

where $\mathcal{M}$ is a space of functions defined in Ω. We may also present the GSS in a more specific form $B = B(\Phi; \omega, \phi)$, where

$\omega: \mathbf{R}^n \to R$ is a mapping that defines the domain

$$\Omega = \{x \in \mathbf{R}^n : \omega(x) > 0\}, \quad \partial\Omega = \{x \in \mathbf{R}^n : \omega(x) = 0\}, \quad (5)$$

$\phi: \mathbf{R}^n \to R$ is a mapping defined in Ω, that strictly satisfies all prescribed boundary conditions.

Each GSS formula is relevant to the type of boundary conditions considered and may be relatively simple when expressed in terms of the ω, ϕ, Φ functions.

Let us consider two simple examples

1. Clamped plate

$$\nabla^4 u = f \text{ in } \Omega \text{ and } u = 0,\ \frac{\partial u}{\partial n} = 0 \text{ on } \partial\Omega.$$

The GSS in this case is $B = \omega^2\Phi$.

2. Membrane

$$\Delta u = f \text{ in } \Omega \text{ and } u = \phi \text{ on } \partial\Omega.$$

The GSS is $B = \omega\Phi + \phi$ as in the previously considered case of the simple beam.

2.2 *R-functions—some basic notions*

R-functions are real functions of many variables. The sign of each R-function is uniquely defined by the sign of its arguments. It will be shown that R-functions are related to the function of s-value logic. For our purpose, however, $s = 2$ and $s = 3$ are sufficient. Let us denote by B_3 the set $\{0, 1, 2\}$ and introduce the following operations on the set

$$a \vee b \equiv \max\{a, b\}, \quad a \wedge b \equiv \min\{a, b\}, \quad \bar{a} \equiv s - 1 - a, \quad a, b \in B_3 \quad (6)$$

that correspond to union, intersection and negation respectively. Using these operations as basic ones it is sometimes useful to introduce also (Fig. 1) some other known logical operations like Scheffer stroke, equivalence and implication

$$a/b \equiv \overline{a \wedge b}, \quad a \sim b \equiv (\bar{a} \vee b) \wedge (a \vee \bar{b}), \quad a \to b \equiv \bar{a} \vee b. \quad (7)$$

These definitions hold for both two-valued (Boolean) and three-valued logics.

Let us define now the mapping $F: \mathbf{B}_3^k \to B_3$ called the function of three-valued logic, and the function

$$S_3(x) = \operatorname{sgn} x + 1, \quad x \in R. \quad (8)$$

We may write now the following definition:

The R-function associated with a function F of

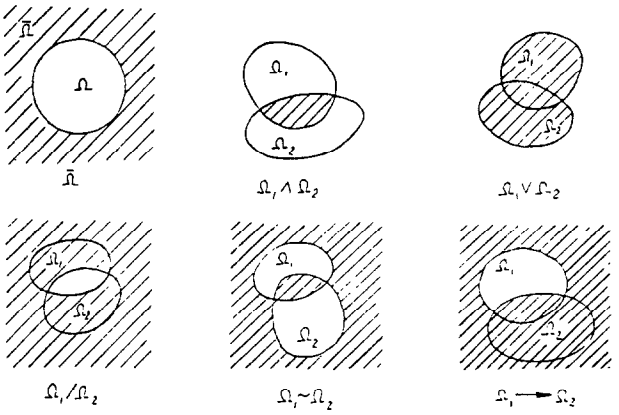

Fig. 1.

three-valued logic is the mapping $f:\mathbf{R}^k \to R$, such that for each $\mathbf{x} = (x_1, \ldots, x_k) \in \mathbf{R}^k$ holds

$$S_3[f(x_1, \ldots, x_k)] = F[S_3(x_1), \ldots, S_3(x_k)]. \quad (9)$$

Thus if relevant R-functions are found, logical operations (F) may be replaced by algebraic ones (f). Any R-function may be expressed in terms of only three basic R-functions associated with union, intersection, and negation. For negation we have

$$\bar{a} = -a \quad (10)$$

while the R-functions corresponding to union and intersection are defined by

$$S_3(aV_*^m b) \equiv S_3(a) V S_3(b) \text{ and } S_3(a\Lambda_*^m b) \equiv S_3(a) \Lambda S_3(b). \quad (11)$$

There are many functions that satisfy these conditions. Most often the following representations are used

$$\begin{aligned} aV_*^m b &= (a + b + \sqrt{(a^2 + b^2)})(a^2 + b^2)^{m/2}, \\ a\Lambda_*^m b &= (a + b - \sqrt{(a^2 + b^2)})(a^2 + b^2)^{m/2} \end{aligned} \quad (12)$$

or

$$\begin{aligned} aV_*^m y &= (a + b)(a^2 + b^2)^m + |a - b|(a - b)^{2m}, \\ a\Lambda_*^m b &= (a + b)(a^2 + b^2)^m - |a - b|(a - b)^{2m} \end{aligned} \quad (13)$$

both being functions of class C^m.

The R-functions enable us to get the ω and ϕ functions required to form a GSS in an arbitrary domain. The relation between R-functions and equations $\omega_i = 0$ of subdomains $\Omega_i \subset \mathbf{R}^n$, $i = 1, \ldots, k$ of a domain Ω in n-dimensional Euclidean space may be explained by the following theorem:

Let $-\omega_i:\mathbf{R}^n \to R$, $i = 1, \ldots, k$ is a mapping and $\Omega_i = \{\mathbf{x} \in \mathbf{R}^n:\omega_i(\mathbf{x}) > 0\}$ are corresponding open sets, so $\omega_i(\mathbf{x}) = 0$ if $\mathbf{x} \in \partial\Omega_i$, $\omega(\mathbf{x}) < 0$ if $\mathbf{x} \notin \bar{\Omega}_i$; $\sqrt{\Omega_1 \cap \ldots \cap \Omega_k} = \bar{\Omega}_1 \cap \ldots \bar{\Omega}_k - F:\mathbf{B}_3{}^k$ is an arbitrary function of three-valued logic, that defines the way the Ω domain is composed of the Ω_i subdomains $\Omega = \{\mathbf{x} \in \mathbf{R}^n:F[S_3(\omega_1(\mathbf{x})), \ldots, S_3(\omega_k(\mathbf{x}))] = 2\}$; it is assumed that F may be expressed in terms of unions, intersections, and negations; then: the function $\omega(\mathbf{x}) = f[\omega_1(\mathbf{x}), \ldots, \omega_k(\mathbf{x})]$ satisfies the conditions

$$\begin{aligned} &\omega(\mathbf{x}) > 0 \Leftrightarrow \mathbf{x} \in \Omega,\ \omega(\mathbf{x}) = 0 \Leftrightarrow \mathbf{x} \in \partial\Omega \\ &\omega(\mathbf{x}) < 0 \Leftrightarrow \mathbf{x} \notin \bar{\Omega}. \end{aligned} \quad (14)$$

This theorem enables us to replace logical operations F done on the sets Ω_i by algebraic operations performed on the ω_i functions. We do this in order to get the required function ω relevant to a given domain Ω of an arbitrarily complex form. This domain Ω is defined by the theory of set operations F applied to the known simple subdomains Ω_i, $i = 1, \ldots, k$ determined by the given equations $\omega_i = 0$; ω is an R-function associated with the function of logic F. Arguments of this R-function are functions of $\omega_1, \ldots, \omega_k$. In order to get ω it is sufficient to use the R-functions related to union, intersection, and negation only.

To illustrate, consider a simple 2-D example of how to obtain an ω function for three quarters of a circle. For simplicity $m = 0$ was assumed. Using relations shown in Fig. 2, we may write

—definitions of subdomains

$$\Omega_i = \{(x, y) \in \mathbf{R}^2:\omega_i(x, y) > 0\},$$

—theory of set operations defining Ω and F

$$\Omega = (\Omega_1 \cup \Omega_2) \cap \Omega_3,$$

—corresponding algebraic operations

$$\omega = (\omega_1 V_* \omega_2)\Lambda_* \omega_3.$$

Hence we get

$$\omega = -x + y + \sqrt{(x^2 + y^2)} + 1 - x^2 - y^2 - \sqrt{((-x + y + (x^2 + y^2)^{1/2})^2 + (1 - x^2 - y^2)^2)}$$

and

$$\Omega = \{(x, y) \in \mathbf{R}^2:\omega(x, y) > 0\},$$

$$\partial\Omega = \{(x, y) \in \mathbf{R}^2:\omega(x, y) = 0\}.$$

The same bases are sufficient to get a function ϕ

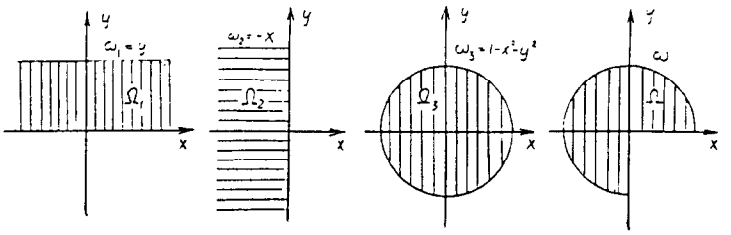

Fig. 2.

defined in Ω and strictly satisfying all boundary conditions ϕ_i prescribed on $\partial\Omega_i$, $i = 1, \ldots, l$. There are several possible ways to form such a ϕ function. The following formulas

$$\phi = \frac{\sum_{i=1}^{l} \omega_i^{-1}\phi_i}{\sum_{i=1}^{l} \omega_i^{-1}} \quad \text{and} \quad \phi = \sum_{i=1}^{l} \frac{\phi_i \prod_{\substack{j=1 \\ j\neq i}}^{l}}{\omega_i + \prod_{\substack{j=1 \\ j\neq i}}^{l} \omega_j} \tag{15}$$

proved[5, 20] to be useful in numerical calculations. The function ϕ is nonunique since, e.g. $\tilde{\phi} = \phi + \omega\Phi$ meets the same requirements for an arbitrary Φ nonsingular on $\partial\Omega$.

More complex formulas that may yield better results can be obtained if functions ω_i in the relation (15) are replaced by ω_i^0. Functions ω_i^0 are defined as a mapping $\mathbf{R}^n \to R$ that satisfies the equation

$$\partial\Omega_i^0 = \{x \in \mathbf{R}^n : \omega_i^0(x) = 0\}, \quad 1, \ldots, k \tag{16}$$

where $\partial\Omega_i^0 = \partial\Omega_i \cap \partial\Omega$ is the boundary common to Ω_i and Ω domains. Since ω_i is equivalent to $-\omega_i^2 \geqslant 0$, in order to find ω_i^0 we might use the relation

$$\omega_i^0 = (-\omega_i^2)\Lambda_*^m \Psi \Leftrightarrow \omega_i^0 = \omega_i^2 V_*^m \Psi. \tag{17}$$

Here $\Psi : \mathbf{R}^n \to R$ is any mapping satisfying the condition

$$\Omega_\Psi = \{\mathbf{x} \in R^n : \omega(\mathbf{x}) > 0\}, \tag{18}$$

where Ω_Ψ is defined by the relation $\partial\Omega_i^0 \equiv \Omega_\Psi \cap \partial\Omega_i$

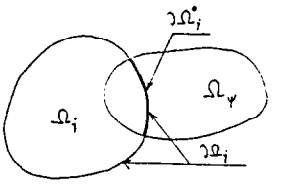

Fig. 3.

(Fig. 3). For sake of simplicity the formula

$$\omega_i^0 = (\sqrt{(\omega_i^2 + \Psi^2)} - \Psi)(\omega_i^2 + \Psi^2)^{m/2} \tag{19}$$

may also be used.

Let us return to our previous example and find functions ω_i^0 (Fig. 4). Each of them describes only a selected part $\partial\Omega_i^0$ of the total boundary $\partial\Omega_i$. Function ω_i^0 is equal to zero or else it is positive.

Thus we may get a function ϕ that meets the conditions (2). The same approach may be applied if instead of being imposed on a function u itself, boundary conditions are prescribed for its derivatives, e.g.

$$m_i \frac{\partial u_i}{\partial n} = \chi_i \text{ on } \partial\Omega_i \subset \partial\Omega, \quad i = 1, \ldots, m. \tag{20}$$

A more complicated situation arises when boundary conditions of both kinds (2) and (20) appear at the same time, e.g. in higher order or mixed problems.

For these problems it is useful to introduce the so-called normalized function ω_N. A function ω is

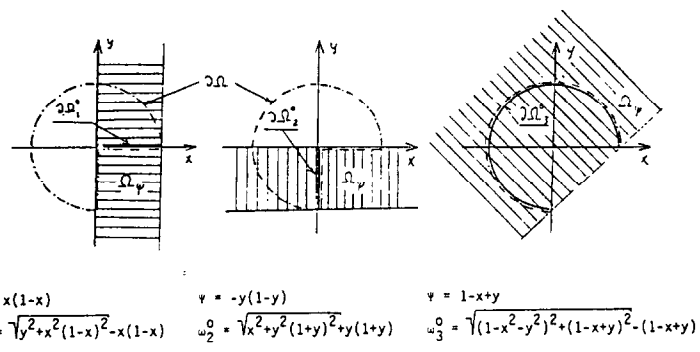

Fig. 4.

normalized to the k order if besides $\omega_{N|\partial\Omega} = 0$ it satisfies the conditions:

$$\left.\frac{\partial\omega_N}{\partial n}\right|_{\partial\Omega} \equiv 1, \left.\frac{\partial^i\omega_N}{\partial n^i}\right|_{\partial\Omega} \equiv 0, \quad i = 2, 3, \ldots, k, \quad (21)$$

where n is the normal direction oriented towards the domain Ω. A function normalized to the first order may be obtained from the formula

$$\omega_N = \omega[\omega^2 + |\text{grad}\,\omega|^2]^{-1/2}. \quad (22)$$

The following rule of differentiation of any function ζ taken on the boundary $\partial\Omega$ in the normal direction n holds:

$$\frac{\partial}{\partial n}(\omega_N\zeta) \equiv \frac{\partial\omega_N}{\partial n}\zeta + \omega_N\frac{\partial\zeta}{\partial n} \equiv \zeta.$$

Since the unit normal vector $\mathbf{n} = \text{grad}\,\omega_{N|\partial\Omega}$, we may also write

$$\begin{aligned}\frac{\partial\zeta}{\partial n}|_{\partial\Omega} &\equiv \text{grad}\,\zeta \cdot \mathbf{n}_{|\partial\Omega} \\ &\equiv \text{grad}\,\zeta \cdot \text{grad}\,\omega_{N|\partial\Omega} \\ &\equiv D_n\zeta_{|\partial\Omega},\end{aligned} \quad (23)$$

where the operator

$$D_n = \frac{\partial\omega_N}{\partial x}\frac{\partial}{\partial x} + \frac{\partial\omega_N}{\partial y}\frac{\partial}{\partial y} \quad (24)$$

of normal differentiation is introduced. The function $D_n\zeta$ is defined not only on $\partial\Omega$ but also in Ω.

Thus a function ϕ that satisfies boundary conditions (2) and (20) on $\partial\Omega$ and is regular enough in Ω may be found as a sum $\phi = \phi_I + \phi_{II}$. The first one $\phi_I = \tilde{\phi} - \omega_N D_n\tilde{\phi}$ fulfills the condition (2) and homogeneous conditions (20) while the second one $\phi_{II} = \omega_N\chi$ satisfies these conditions vice versa. Hence

$$\phi = \tilde{\phi} + \omega_N(\chi - D_n\tilde{\phi}) \quad (25)$$

where $\tilde{\phi}$ and χ are functions that meet only requirements (2) and (20) respectively and may be found using formula (15).

Only the most typical and simple types of boundary conditions were considered here; much more information may be found in the papers[21, 25].

2.3 *The R-function method*

The method is carried out in two steps. In the first step the GSS is derived in the form of an analytical formula $B(\Phi; \omega, \phi)$. This formula depends on the given boundary conditions and shape of the domain. When written in terms of ω, ϕ, and Φ functions GSS may be used for a wide class of problems having the same boundary conditions. Such general structures of solutions (GSS) have been obtained for a variety of problems of applied mechanics[19–22, 24–27]. These will not be discussed here. In order to find a particular form of GSS the functions ω and ϕ should be determined for a specific domain. This may always be accomplished using the R-function approach described above. In the second step, which is of a numerical nature, an undetermined function Φ is discretized, e.g. in the form

$$\Phi = \sum_{k=1}^{M} \Phi_k N_k(\mathbf{x}), \quad k = 1, \ldots, M \quad (26)$$

or the finite difference approach may be used. Since the boundary conditions are always satisfied, the discretization may be done in the cartesian product in the $\mathbf{R}^n$ space, e.g. in a rectangle (Fig. 5) or rectangular solid in 2-D and 3-D bodies respectively. Approximation over such a region is relatively simple and many known methods may be applied. Application of spline functions or orthogonal polynomials, e.g. Tschebychev ones $N(x) = T(x_1)T(x_2)\ldots T(x_3)$ are particularly useful.

It is expected, that the number of unknowns (D.O.F.) may be relatively small since the analytical part of the GSS should already contain the required singularities caused by both the boundary conditions and the shape of the domain.

An approximate solution of the boundary value problem (2) or (3) is assumed now in the form $u = B(\Phi_1, \ldots, \Phi_M; \omega, \phi)$ with unknown parameters $\Phi_k, k = 1, \ldots, M$. These parameters may be found by using any one of a number of appropriate numerical methods. The techniques that have been the most frequently used to solve the resulting overdetermined system of simultaneous algebraic equations have been the collocation and the least squares methods. This approach has proven to be useful [5, 9, 21, 22, 24] since it does not require the differentiation of complicated formulas and good accuracy of the results is preserved.

2.4 *Singularity and GSS*

According to its definition GSS always satisfies all prescribed boundary conditions. In order to improve the final solution, however, we may sometimes enforce additional requirements on the GSS. Thus in cases in which the asymptotic behaviour of the solution is known in a neighborhood of the boundary or at a given point, we may build this behaviour into the GSS. These asymptotic solutions are often singular. It is convenient then to present GSS in the form $B = B_0 + \psi B_1$, where ψ is a factor involving the required type of singularity, and B_0 and B_1 are nonsingular parts of GSS; the first term being responsible for all nonhomogeneous boundary conditions, while the second for the undetermined function Φ. We next examine a few examples of such a procedure.

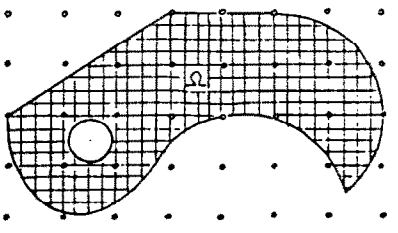

Fig. 5.

In linear elastic fracture mechanics we have[19] GSS

$$B = B_0 + \omega_N^2 \Phi \prod_{j=1}^{n} r_j^{-1/2} \tag{27}$$

hence

$$\psi = \prod_{j=1}^{n} r_j^{-1/2}. \tag{28}$$

The singularity $r_i^{-1/2}$ depends on the distance r_i from each crack tip i.

In the case of contact problems involving a body of polygonal shape involved we get[21]

$$\psi = \omega_N^{-1/2} \prod_{j=1}^{n} r_j^{\nu_j - \frac{1}{2}}, \quad \nu_j = \nu(\alpha_j). \tag{29}$$

Two types of singularities are included here. The first one $\omega_N^{-1/2}$ is referred to a neighborhood of an edge of the boundary $\partial\Omega$. The second one $r_j^{\nu_j - \frac{1}{2}}$ corresponds to a vicinity in the distance r_j from the jth corner of the polygon; $\nu(\alpha_j)$ is a known function of the polygon angle α_j.

We may consider the Laplace equation

$$\nabla^2 u = -2 \text{ in } \Omega, \quad u = 0 \text{ on } \partial\Omega \tag{30}$$

in order to examine the singularity caused by corners in an Ω domain.

The singularity appears at these points where a tangent to $\partial\Omega$ does not exist, i.e. $\partial\Omega \notin C^1$. The type of singularity depends on the angle α (Fig. 6):

$$\begin{aligned} u &= Cr^2 + 0(r^2) && \text{for } 0 < \alpha < \pi/2, \\ u &= Cr^2 \ln r + 0(r^2 \ln r) && \text{for } \alpha > \pi/2, \\ u &= Dr^{\pi/\alpha} + 0(r^{\pi/2}) && \text{for } \pi/2 < \alpha < 2\pi, \\ u &= Dr && \text{for } \alpha = \pi, \end{aligned} \tag{31}$$

where C and D are constants.

Generally, if we know that at a given point $(x_0, y_0) \in \bar{\Omega}$ the solution behaves like $0(r_0^{\alpha})$, in order to obtain the required type of asymptotic behaviour we may replace an ω_N function by

$$\omega^* = \omega_N \rho^{\alpha - 1} \tag{32}$$

where $r = \rho(x, x_0)$; here ρ denotes the distance between two points.

The effectiveness of such an approach where the singularity is incorporated into the GSS in advance was investigated in some numerical examples presented in Section 4.

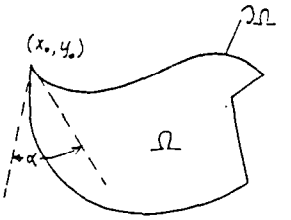

Fig. 6.

3. COMPUTER APPROACH

3.1 *Introduction*

In the original version of the RFM [21, 22, 24–27] the GSS was essentially formed manually. This required many laborious operations. Only the numerical part was left to the computer. In the present paper symbolic programming has been used in order to derive GSS in an analytical form as well as to accomplish all operations required by an operator L corresponding to a boundary value problem. Finally a relevant source code is automatically generated. It is written in the form of FORTRAN subroutines, that are ready for use in the numerical step of the method. Thus a fully computerized approach (Fig. 7) to the RFM may be obtained, leaving only a few input data to be prepared by the user.

Such a computer approach is realized in two steps. In the first one an automatic generation of GSS $B(\Phi; \omega, \phi)$ as well as the value $LB(\Phi; \omega, \phi)$ of an operator L on this GSS is done in the form of relevant FORTRAN subroutines. This is achieved by means of symbolic programming, i.e. generation by the computer itself of the pertinent strings of chars. In the case considered these strings represent expressions that are used in order to obtain the text of required FORTRAN subroutines for functions ω, ϕ, B, LB, etc.

When these programs are ready, we come to the second numerical step of the method. Then as yet undetermined function Φ is discretized in terms of M unknown parameters Φ_i, $i = 1, \ldots, M$. Various methods may be applied in order to find first the unknown parameters and second the approximate solution of the boundary value problem considered.

3.2 *Symbolic operations*

Subsequent steps of the symbolic programming applied are shown in the flowchart (Fig. 8) and will be briefly described below.

1. Introduced is alphanumeric data corresponding to theory of set operations performed on given subdomains Ω_i, in order to form a required Ω domain. They are written in a FORTRAN code using conversions described in columns 2, 3, 6 and 7 of Table 1.

Table 1.

	Operations				Arguments			
Symbol	Set	Theory	R-functions		Set	Theory	R-functions	
1	2	3	4	5	6	7	8	9
Math.	$\cap$	$\cup$	$\wedge_*$	$\vee_*$	Ω	Ω_i	ω	ω_i
FORTRAN	*	+	AIL	SUM	F	F(I)	F	F(I)

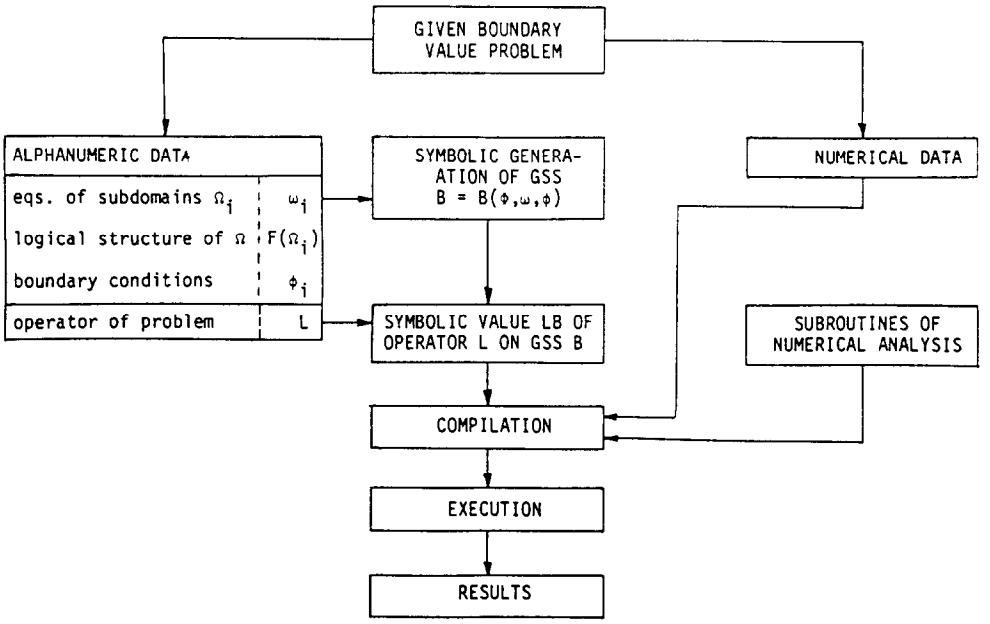
GIVEN BOUNDARY VALUE PROBLEM
ALPHANUMERIC DATA
eqs. of subdomains Ω_i | ω_i
logical structure of Ω | $F(\Omega_i)$
boundary conditions | ϕ_i
operator of problem | L
SYMBOLIC GENERATION OF GSS B = B(φ,ω,φ)
NUMERICAL DATA
SYMBOLIC VALUE LB OF OPERATOR L ON GSS B
SUBROUTINES OF NUMERICAL ANALYSIS
COMPILATION
EXECUTION
RESULTS

Fig. 7.

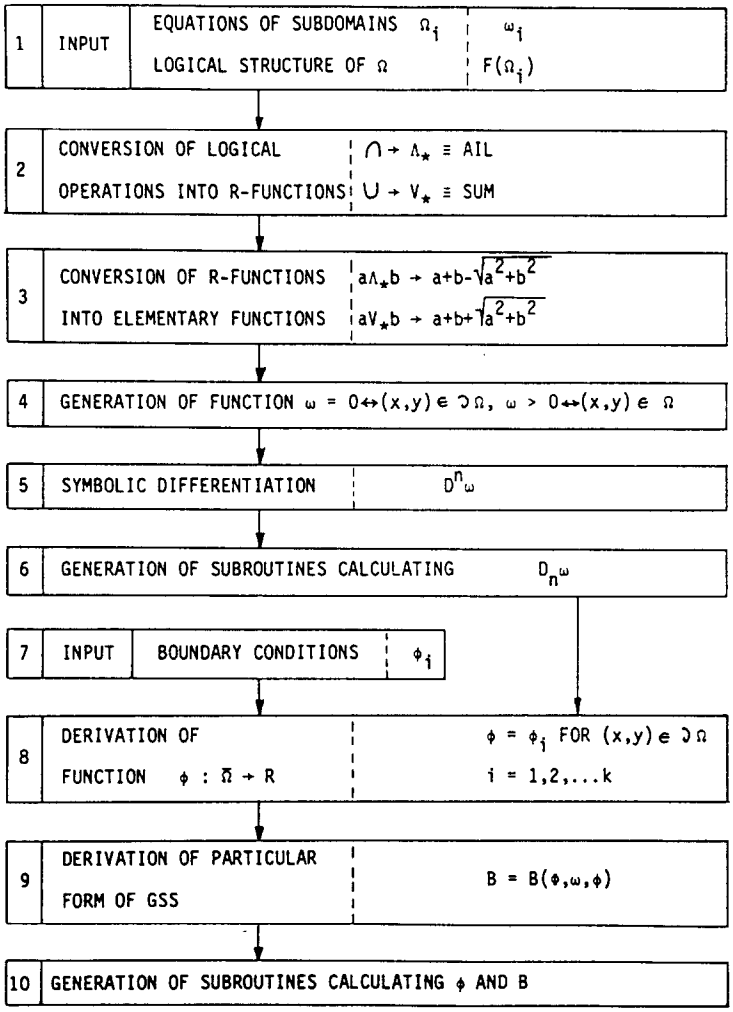
1 INPUT EQUATIONS OF SUBDOMAINS Ω_i | ω_i
LOGICAL STRUCTURE OF Ω | $F(\Omega_i)$
2 CONVERSION OF LOGICAL OPERATIONS INTO R-FUNCTIONS $\cap \to \wedge_* \equiv$ AIL, $\cup \to \vee_* \equiv$ SUM
3 CONVERSION OF R-FUNCTIONS INTO ELEMENTARY FUNCTIONS $a\wedge_* b \to a+b-\sqrt{a^2+b^2}$, $a\vee_* b \to a+b+\sqrt{a^2+b^2}$
4 GENERATION OF FUNCTION $\omega = 0 \leftrightarrow (x,y) \in \partial\Omega$, $\omega > 0 \leftrightarrow (x,y) \in \Omega$
5 SYMBOLIC DIFFERENTIATION $D^n\omega$
6 GENERATION OF SUBROUTINES CALCULATING $D_n\omega$
7 INPUT BOUNDARY CONDITIONS ϕ_i
8 DERIVATION OF FUNCTION $\phi : \bar{\Omega} \to R$ $\phi = \phi_i$ FOR $(x,y) \in \partial\Omega$, $i = 1,2,\ldots k$
9 DERIVATION OF PARTICULAR FORM OF GSS $B = B(\phi,\omega,\phi)$
10 GENERATION OF SUBROUTINES CALCULATING φ AND B

Fig. 8.

For example if

$$\Omega = \Omega_1 \cap (\Omega_2 \cup \Omega_3) \cup \Omega_4 \cup \Omega_5$$

the data are:

$$F(1)*(F(2) + F(3)) + F(4) + F(5).$$

Expressions F(I) corresponding to ω_i functions describing subdomain Ω_i, must be given by the user.

2. Logical operations introduced above are transformed into R-functions according to the following rules (Table 1)

$$\mathcal{S}A1 + A2\mathcal{S} \rightarrow SUM(\mathcal{S}A1\mathcal{S}, \mathcal{S}A2\mathcal{S}),$$

$$\mathcal{S}A1*A2\mathcal{S} \rightarrow AIL(\mathcal{S}A1\mathcal{S}, \mathcal{S}A2\mathcal{S}), \qquad (33)$$

$$\mathcal{S}F(K)\mathcal{S} \rightarrow F(K).$$

An expression placed between chars $\mathcal{S}$ is understood to be subjected to further analysis. In order to explain this procedure let us continue the previous example:

$\mathcal{S}F(1)*(F(2) + (F(3)) + F(4) + F(5)\mathcal{S}\rightarrow$

$SUM(\mathcal{S}F(1)*(F(2) + F(3))\mathcal{S}, \mathcal{S}F(4) + F(5)\mathcal{S})\rightarrow$

$SUM(AIL(\mathcal{S}F(1)\mathcal{S}, (\mathcal{S}F(2) + F(3)\mathcal{S})),$

$SUM(\mathcal{S}F(4)\mathcal{S}, \mathcal{S}F(5)\mathcal{S}))\rightarrow$

$SUM(AIL(F(1), SUM(F(2), F(3))), SUM(F(4), F(5))).$

The sequence of these transformation is presented in Fig. 9.

3. R-functions are transformed into elementary functions. R-functions of both types (12) and (13) have been used, e.g. for (12) we have

$$\mathcal{S}SUM(A1, A2)\mathcal{S} \rightarrow (\mathcal{S}A1\mathcal{S} + \mathcal{S}A2\mathcal{S}$$
$$+ SQRT((\mathcal{S}A1\mathcal{S})**2$$
$$+ (\mathcal{S}A2\mathcal{S})**2))*SQRT((\mathcal{S}A1\mathcal{S})**2$$
$$+ (\mathcal{S}A2\mathcal{S})**2)**M, \qquad (34)$$
$$\mathcal{S}AIL(A1, A2)\mathcal{S} \rightarrow (\mathcal{S}A1\mathcal{S} + \mathcal{S}A2\mathcal{S}$$
$$- SQRT((\mathcal{S}A1\mathcal{S})**2 + (\mathcal{S}A2\mathcal{S})**2)*$$
$$*SQRT((\mathcal{S}A1\mathcal{S})**2 + (\mathcal{S}A2\mathcal{S})**2)**M,$$
$$\mathcal{S}F(K)\mathcal{S} \rightarrow F(K).$$

Similarly in Step 2 consecutive expressions are analyzed and subsequently transformed into the simplest ones starting from the extreme left.

4. A source FORTRAN code for a subroutine of type FUNCTION is generated using results of the previous three steps.

5. Derivatives $D^n\omega$, if required by GSS, are found by the formal differentiation. Standard functions as well as expressions built by arithmetical operations are found by a subroutine POCH of symbolic differentiation given before in [10]. An option has recently been provided for partial derivatives $\partial^{i+j}/\partial x^i \partial y^j$ of R-functions Λ_* and V_* as well (Table 2).

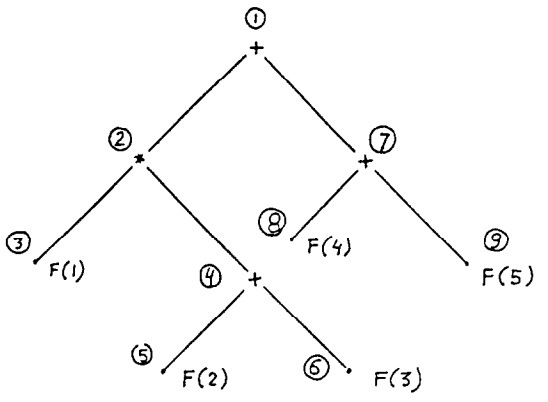

Fig. 9.

Table 2.

Math. symbol	Λ_*	$\vee_*$	$\frac{\partial^{i+j}\Lambda_*}{\partial x^i \partial y^i}$	$\frac{\partial^{i+j}\Lambda_*}{\partial x^i \partial y^i}$
FORTRAN symbol	PØØ	SØØ	P„I"„J"	S„I"„J"

The following transformations are performed

$$\mathcal{S}P_{\prime\prime}I''_{\prime\prime}J''(A1, A2)\mathcal{S} \rightarrow P_{\prime\prime}I + 1''_{\prime\prime}J''(A1, A2)*(\mathcal{S}A1\mathcal{S})$$
$$+ P_{\prime\prime}I''_{\prime\prime}J + 1''(A1, A2)*(\mathcal{S}A2\mathcal{S}) \qquad (35)$$

for P and the same are carried out for S. Expressions obtained this way contain names of R-functions and their derivatives. Arbitrary R-functions Λ_* and V_* may then be used. Due to their complexity, first and second order derivatives of R-functions (12) have been found, programmed, and stored in the library. In the case of R-functions (13) derivatives are automatically generated using the POCH subroutine and the formula

$$\frac{\partial^{i+j}}{\partial a^i \partial b^j}[|a-b|(a-b)^{2m}] = (-1)^j(2m+1)(2m)$$
$$\cdots(2m-i-j+2)|a-b|(a-b)^{2m-i-j}. \qquad (36)$$

6. Proceed as in Step 4.

7–10. In the current version of the program a general form of GSS, i.e. expressed in terms of Φ, ω and ϕ functions, has to be given by the user. Using this GSS, given boundary conditions (Step 7), formulas type (15) and (22) presented in the previous section as well as the rules introduced in the Steps 1–6 of this algorithm—function ϕ is derived (Step 8). It is followed by a particular form of GSS (Step 9). Finally relevant FORTRAN subroutines are obtained (Step 10) like in Steps 4 and 6.

Once more formal operations are used when the symbolic value LB of a given operator L is found on the GSS B derived above.

A set of computer testing programs was constructed using standard FORTRAN code and based on the approach presented here. These programs that are oriented to applying the RFM to a wide class of boundary value problems in applied mechanics are under current development now.

4. NUMERICAL RESULTS

Presented here are some numerical results obtained by our actual computer programs. They were mainly oriented to testing both the RFM and the programs. The other results obtained by use of symbolic programming in the RFM are also mentioned.

1. Torsion of a bar of squared cross-section was considered. Thus $\Delta u_{|\Omega} = -2$, $u_{|\partial\Omega} = 0$ while GSS $B = \omega\Phi$. Tschebychev's approximation $N_{ij} = T_i(x)T_j(y)$, $i,j = 1, 2, 3$ was assumed with the total number of $n = 9$ D.O.F. The number of collocation points was variable. Then the value u_0 in the midpoint was found (Table 3).

2. The problem

$$\Delta u = f \text{ in } \Omega \subset \mathbf{R}^2, \quad u_{|\partial\Omega} = 0 \text{ on } \partial\Omega \qquad (37)$$

was considered. It was assumed that

$$f = -2\pi^2 \sin \pi x \sin \pi y, \quad \Omega = [0, 1] \times [0, 1];$$

exact solution is $u = \sin \pi x \sin \pi y$. Approximate solutions $\tilde{u}$ were obtained for various ω functions:

a. $\omega = (1 - x)x(1 - y)y$,

b. $\omega = [(1 - x)x] \Lambda_* [(1 - y)y]$,

c. $\omega = (1 - x) \Lambda_* x \Lambda_* (1 - y) \Lambda_* y$.

The relative error $\epsilon = (\tilde{u} - u)/u$ expressed in percent is given in Table 4.

An essential influence of ω function was observed. Two types of R-functions (12) and (13) were also examined. The first of them gave better results.

3. Problem (37) was also solved for $f = -2$ and the domain shown in Fig. 10 (torsion of an elastic bar). The influence of singularity given by eqn (31) for

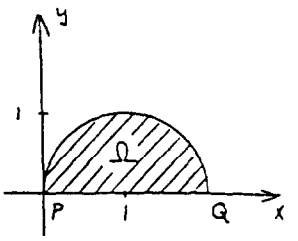

Fig. 10.

$\alpha = \pi/2$ was examined. Three cases were investigated:

(a) singularity was not taken into account either in P or in Q;

(b) the asymptotic behaviour of the solution in the point P was included in the approximate solution;

(c) singularity was subtracted out of both points P and Q.

Results obtained for u in the point (1, 0.5) were as follows: *a*, 0.191, *b*, 0.215, *c*, 0.228.

4. The next example is a problem of the same kind, but with domain Ω shown in Fig. 11. The influence of singularity in the point K where $\alpha = 3\pi/2$ was investigated. The result in the point P (1.5, 1) was $u_P = 0.0675$. if singularity was taken into account, and $u_P = 0.1356$ if not. Since the exact solution in this case is not known, in order to evaluate the results we may use the principle of the minimum of harmonic functions.

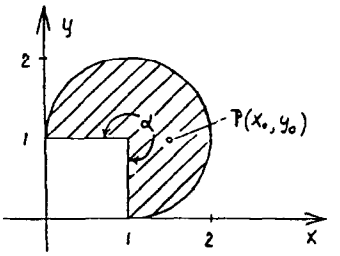

Fig. 11.

Table 3.

Number of collocation points m	$20 \times 20 = 400$	$25 \times 25 = 675$	$30 \times 30 = 900$	$35 \times 35 = 1275$	$38 \times 38 = 1444$
Value of u_0	0.154594	0.146572	0.142635	0.140023	0.140167

Table 4.

Item	Point	Solution case a	b	c	Exact
Value u		0.999	0.929	0.770	1.000
error ϵ%	(0.5, 0.5)	−0.0004	−7.01	−23.0	—
Value u		0.706	0.636	0.489	0.707
error ϵ%	(0.25, 0.25)	−0.006	−10.1	−30.7	—

Let $\Delta u = -A$ in $\Omega \subset R^2$, $A = \text{const}$. $u_{|\partial\Omega} = 0$, $P(x_0, y_0) \in \Omega$ and $r = (A/4)[(x - x_0)^2 + (y - y_0)^2]$ then

$$\max_{\partial\Omega} r \geqslant u(x_0, y_0) \geqslant \min_{\partial\Omega} r.$$

Hence in our case we have $1.125 \geqslant u_P \geqslant 0.125$. This inequality holds only for the first result, i.e. when asymptotic solution was included in the approximate solution.

Other applications of the computerized approach to the RFM have been recently carried out and presented in separate papers. All of them are oriented to deal with domains of arbitrary shapes including concave and multiconnected regions. Thus the following problems were considered:

—static analysis of clamped plates of various shapes[3]. The results were in good agreement with those done earlier by the original version of the method[22] as well as with ones obtained by the Boundary Element and Finite Difference methods[3].

—Evaluation of stress intensity factors in Linear Elastic Fracture Mechanics[19, 20]. The global approach using the principle of minimum of total elastic energy was applied.

—Automatic generation of geometrical characteristics (e.g. stiffness and mass matrices) of special elements of an arbitrary shape[5].

—Identification of internal boundary and external points of the given domain[11]. An application was made to an automatic mesh generation for FEM and FDM. The same option is included now in the actual package of computer programs for the RFM.

The first of the applications mentioned above was mainly done because of testing purposes and comparison with the original, manual version of the RFM. The next three present new fields where this method has not been used yet.

5. FINAL REMARKS

A fully computerized approach to the R-function method (RFM) of solution of boundary value problems in domains of arbitrary shape was investigated. Thus both the so-called general structure of the solution (GSS) B and the value LB of any given differential operator L defined on this B may be found by the computer in an analytical form and obtained as subroutines written in a source code. These are automatically used later on together with other additional programs for further numerical analysis of a considered problem. In the original version of the RFM[21, 25] both B and LB have to be derived by manual, usually very tedious operations. Despite many successes of the RFM this inconvenience has been one of the main reasons preventing many potential users from applying this powerful method, especially when compared with the versatility of the FEM. Now this barrier seems to have been overcome.

The following was accomplished in the present paper besides the brief outlines of the theory of the RFM:

—a general idea of a fully computerized approach to the method based on symbolic programming; corresponding algorithms were also described;

—development of subroutines that perform the main steps of these algorithms; they are given in the form of a package of experimental programs;

—results of numerous tests; they concern both the method itself as well as the computer programs developed; further applications in some fields of computational mechanics and in numerical analysis are also mentioned.

So far our investigations have been mainly experimental in character. Their primary object was to find a computerized approach to the RFM and to gain some experience in its applications. It was also an opportunity to examine the RFM itself, since when fully computerized this method might be a powerful tool of computational mechanics. Then the following disadvantages of the RFM were observed:

—final results are highly dependent on the ω and ϕ functions assumed; at this time it is an open question as to which is the best choice;

—GSS B obtained for any particular domain Ω is usually expressed by complex formulas; very tedious operations, especially when manually performed, are needed in order to find value LB for a given operator L. These operations may be carried out by the computer when symbolic programming is applied but the formulas still remain lengthy;

—expressions of the 0/0 and of finite value may appear in points of intersection of boundaries $\partial\Omega$, and $\partial\Omega_{i+1}$, $i = 1, \ldots, k$.

Actual computer programs, due to their experimental character, have some temporary limitations such as:

—only 2-D problems and ones described by operators not higher than second order may be considered;

—a general form of GSS appropriate to boundary conditions considered like $B = \omega\Phi + \phi$ still have to be given by the user.

On the other hand, numerous advantages of the method may be stated:

—domains of arbitrary shapes, multiconnected and concave cases are admitted;

—boundary conditions are always satisfied, numerical solution may be sought, therefore, in a domain which is the cartesian product containing given domain Ω;

—both linear and nonlinear problems with linear boundary conditions may be considered;

—if an asymptotic solution of the problem is known, it may be incorporated in advance in the GSS; singular cases are also included;

—in the numerical part of the method any discrete approach is admissible;

—there is a possibility of a significant decrease in the number of D.O.F. when compared with numerical methods such as FEM and FDM;

—fully computerized version of the method is possible with only a few basic data required of the user.

The following conclusions may be made:

—though wide comparisons with such well known and widely applied methods like the FEM, FEM or BEM are not available yet, the RFM seems to be both promising and worthy of further investigation as a general method of solution of boundary value problems. In its original version it has already been successfully applied to a variety of problems [21, 22, 24–27]. However, its potential versatility, competiveness, and efficiency can significantly be increased once a fully computerized version is obtained.

—Since the actual computer approach to the RFM

presented here is still in an experimental stage, many questions such as the efficiency and applicability of the method remain unanswered as yet. Therefore despite the fact that results obtained so far are very encouraging, further intensive investigations are needed.

Some reasonable directions of such research are listed below.

Development of the RFM itself:

—fundamental mathematical studies in order to find the most appropriate ω and ϕ functions for a problem and domain considered;

—studies on asymptotic solutions (especially singularities);

—combination of RFM with other methods, especially the finite difference method (FDM) with arbitrary irregular meshes[13, 14].

Development of the software:

—lifting actual restrictions, e.g. admission of 3-D problems and of operators of a higher order than two; further automatization like that of the derivation of the general case of GSS; self-elimination of 0/0 expressions;

—permanent investigation to improve, develop and expand the current experimental programs into broadly based production programs;

—a computer approach to the so-called structural version of RFM when the complex domain is divided into some subregions;

—adaptation of the computerized RFM to the solution of nonstationary and nonlinear problems.

Applications. First of all many testing boundary value problems should be solved in order to:

—gain experience;

—find the best choice of ω and ϕ functions;

—check the usefulness of including asymptotic solutions;

—examine the efficiency and applicability of the method, e.g. to determine the best field of application;

—compare the computer version of the RFM with the manual approach and also to other methods, especially the FEM and FDM;

—determine the most suitable numerical approach in the second part of the method.

New fields of application may be also hoped for, e.g. nonlinear problems, nonstationary, and 3-D ones.

There are obviously many questions that should be answered. One may expect, however, that there are fields of applications where the computerized RFM, especially when combined with the FDM at irregular meshes, could be competitive with the FEM.

Besides boundary values problems, the RFM may also be used as an auxiliary tool in computational problems, e.g. automatic mesh generation[11], generation of trial functions required in many approximate method (Ritz, Galerkin) or shape functions[5] used in the FEM.

REFERENCES

1. W. F. Ames, *Numerical Methods for Partial Differential Equations*. Academic Press, New York, San Francisco (1977).
2. C. A. Brebbia, *The Boundary Element Method for Engineers*. Pentech Press, London (1981).
3. W. Cecot and J. Orkisz, Comparison of some analytical–numerical methods using static analysis of plates of arbitrary shapes (in Polish). *Rozpr. Inz.* 4, **31**, 459–471 (1983).
4. L. M. Delves and C. A. Hall, An implicit matching principle for global element calculations. *J. Inst. Math. Applic.* **23**, 223–234 (1979).
5. M. Korzeniowski and J. Orkisz, An application of the R-functions method to an automatic generation of characteristics of finite elements of arbitrary shape (in Polish). *Proc. 5th Nat. Conf. Comput. Meth. in Structural Mechanics, Karpacz* 6–9 *May* 1981, Vol. 1, pp. 235–244 (1981).
6. Ka-Ching Wong, *Symbolic Computing, Numerical and Computer Methods in Structural Mechanics*. Academic Press, New York (1975).
7. J. Kucwaj, Symbolic computer differentiation in nonlinear boundary value problems of mechanics (in Polish). *Proc. 4th Nat. Conf. Comput. Meth. in Structural Mech., Koszalin*, Vol. 4, pp. 391–398 (1979).
8. J. Kucwaj, Application of symbolic programming to the automatic generation of Jacobi's matrix in the Newton–Raphson method (in Polish). *Mechanika i Komputer*, **4**, Ed. PWN Warsaw (1981).
9. J. Kucwaj and J. Orkisz, Computer approach to an analytical–numerical method of the solution of boundary value problems in mechanics. *Proc. 6th Int. Conf. Structural Mech. Reactor Tech., Paris, August* 17–21, Paper M 3/9 (1981).
10. J. Kucwaj and J. Orkisz, A FORTRAN subroutine of symbolic differentiation (in Polish). *Mechanika i Komputer* **5**, Ed. PWN Warsaw (1982).
11. J. Kucwaj and J. Orkisz, Identification of inner, outer and boundary points of domains of an arbitrary shape (in Polish). *Mechanika i Komputer* **5**, Ed. PWN Warsaw, pp. 143–149 (1982).
12. J. Kucwaj, J. Orkisz and W. Tworzydlo, Application of symbolic programming to an automatic solution of non-linear equations of membrane shells by the generalized FDM (in Polish). *Arch. Inz. Lad.* (1982).
13. T. Liszka and J. Orkisz, The finite difference method at arbitrary irregular grids and its application in applied mechanics. *Comput. Structures* **11**, 83–95 (1980).
14. T. Liszka and J. Orkisz, The finite difference method for arbitrary irregular meshes— a variational approach to applied mechanics problems. *Proc. 2nd Int. Cong. Num. Meth. Engng G.A.M.N.I.* 2, Paris (Dec. 1980).
15. S. G. Mikhlin and K. L. Smolitskiy, *Approximate Methods for Solution of Differential and Integral Equations*. American Elsevier, New York (1967).
16. A. K. Noor and S. J. Andersen, Computerized symbolic manipulation in structural mechanics—progress and potential. *Comput. Structures* **1/2**, 10 (1979).
17. A. K. Noor and C. M. Andersen, Computerized symbolic manipulation in nonlinear finite element analysis. *Symp. Computational Meth. in Nonlinear Structural and Solid Mech.*, Washington (1980).
18. J. M. Ortega and W. G. Poole, Jr., *An Introduction to Numerical Methods for Differential Equations*. Pitman, Marshfield, Massachusetts (1981).
19. J. Orkisz, W. Rachowicz and M. Wnuk, Evaluation of stress intensity factors by the R-functions method. *Int. J. Engng. Fracture Mech.* 107–126 (1984).
20. W. Rachowicz, Evaluation of stress intensity factors by the boundary integral and R-functions methods (in Polish). M.Sc. thesis in the Technical University of Cracow, Cracow (1981).
21. V. L. Rvachev, *Methods of Algebraic Logic in Mathematical Physics* (in Russian). Ed. Naukova Dumka, Kiev (1974).
22. V. L. Rvachev, L. V. Kurpa, N. G. Sklepus and L. A. Utshishvili, *R-Functions and Their Applications to Solution of Plates of a Complex Form* (in Russian). Kiev (1973).
23. V. L. Rvachev, G. P. Mankov and V. V. Fedko, *On a Technique Applied to Obtain Computer Programs for Differentiation of Superposition of Functions* (in Rus-

sian). Dopovidi Akademii Nauk of Ukrainian SSR 1, 71–73 (1981).
24. V. L. Rvachev and B. S. Procenko, *Contact Problems of the Theory of Elasticity with Non-Classical Domains* (in Russian). Ed. Naukova Dumka, Kiev (1977).
25. V. L. Rvachev and V. A. Rvachev, *Nonclassical Methods of Approximation Theory in the Boundary Value Problems* (in Russian). Ed. Naukova Dumka, Kiev (1979).
26. V. L. Rvachev and N. S. Sinekop, Structures of the difference type solutions of the axisymmetric problems for elastic bodies of finite dimensions. *Applied Mathematics and Mechanics*, 646–649 (Sept. 1982). Translation of PMM, U.S.S.R. 5, 45, 862–866 (1981).
27. V. L. Rvachev and A. P. Slesarenko, *Algebra of Logics and Integral Transformations in Boundary Value Problems* (in Russian). Ed. Naukova Dumka, Kiev (1976).