J. Symbolic Computation (1987) **3**, 291–301

Using a Small Algebraic Manipulation System to Solve Differential and Integral Equations by Variational and Approximation Techniques

R. D. MILLS

Computing Science Department, University of Glasgow, Glasgow, UK

(*Received* 2 *December* 1985)

The microcomputer algebraic manipulation system MUMATH is used to implement the classical variational, Galerkin and least-squares techniques for solving boundary-value problems in differential equations and also for solving Fredholm integral equations. Examples are given which extend the precision of known results. The technique is presented as a general algorithm which can readily be implemented on other algebraic manipulation systems.

1. Introduction

Computer programs have been in existence since the fifties to manipulate polynomials, differentiate functions and solve equations. By the late sixties and early seventies programs were written which could integrate functions analytically. The late seventies saw the development of programs for the symbolic solution of differential and integral equations (see Golden, 1977; Stoutemyer, 1977; Bogen, 1979). Large and complex algebraic manipulation systems were devised to implement the algorithms which effect these processes, the best known being FORMAC, REDUCE2, MACSYMA and SCRATCHPAD.

There then opened up the exciting possibility of tackling many problems in applied mathematics and engineering which could be solved approximately by analytical methods but which require very large amounts of algebraic manipulation. The Rayleigh–Ritz, Galerkin and least-squares methods for solving boundary-value problems in differential equations are typical examples. For early work using computer algebra in these methods, see Miola (1974) and Andersen & Noor (1977). This application area involves the interaction of numerical and algebraic computation (see Ng, 1979). One of the first of such applications was the use of algebraic differentiation to determine the higher derivatives in the Taylor series method for solving initial-value problems in ordinary differential equations. For surveys of applications see, for example, Fitch (1979) and Brown & Hearn (1979).

Unfortunately the use of these large systems did not, and still has not, become very widespread and this has meant that the rate of progress in the application of algebraic manipulation techniques has been somewhat less rapid than might have been hoped. However, the ever-increasing use of microcomputers and the pioneering efforts of Stoutemyer and Rich (see the Microsoft MUMATH/MUSIMP Reference Manuals 1980) in implementing MUMATH on various 8-bit microprocessors have made symbolic manipulation techniques more widely available. See also the work of Fitch (1983) on

0747–7171/87/030291 + 11 \$03.00/0

implementing REDUCE on a Motorola 68000 based microcomputer. The present paper will demonstrate how the TRS80 Model I running MUMATH can solve many of the classical variational and approximation problems very effectively. Those wishing higher accuracy will have to gain access to, say, an IBM PC or a REDUCE2 system.

2. Ordinary Differential Equations

Self-adjoint second-order ordinary differential equations of the form

$$L[y] = (py')' - qy - f = 0 \tag{1}$$

satisfying boundary conditions

$$y = y_a \text{ at } x = a, \qquad y = y_b \text{ at } x = b \tag{2}$$

have the *variational formulation*

$$J[y] = \int_a^b (py'^2 + qy^2 + 2fy)\,dx = \text{minimum} \tag{3}$$

under the same conditions (2) (see Kantorovich & Krylov, 1958, p. 262). This property leads to the *Rayleigh–Ritz* technique for the approximate solution of (1), (2): determine constants a_i, $i = 1, 2, \ldots, n$ in the approximation

$$y \approx y_n = \sum_{i=1}^{n} a_i v_i(x) \tag{4}$$

which satisfies (2) and (3) with a suitable choice of *basis functions* $v_i(x)$. The equations for the a_i are then

$$\sum_{j=1}^{n} \alpha_{ij} a_j = -\beta_i, \qquad i = 1, 2, \ldots, n,$$

where

$$\alpha_{ij} = \alpha_{ji} = \int_a^b (pv_i'v_j' + qv_iv_j)\,dx$$

and

$$\beta_i = \int_a^b fv_i\,dx. \tag{5a, b, c}$$

In the *Galerkin method* we write the o.d.e. (1) as

$$M[y] = f \quad \text{where} \quad M \equiv L + f, \tag{6}$$

and if we use the same approximation (4) the Galerkin equations are

$$\int_a^b L[y_n]v_i\,dx = \int_a^b (M[y_n] - f)v_i\,dx = 0,$$

which amounts to forcing the error $L[y_n]$ to be orthogonal to all the functions v_i, $i = 1, 2, \ldots, n$. This approach gives the following equations for the constants a_i,

$$\sum_{j=1}^{n} \alpha_{ij} a_j = -\beta_i, \qquad i = 1, 2, \ldots, n,$$

where

$$\alpha_{ij} = \alpha_{ji} = -\int_a^b M[v_i]v_j\,dx \tag{7a, b, c}$$

and

$$\beta_i = \int_a^b f v_i \, dx.$$

If we integrate the first member of (5b) by parts, we see that equations (5) and (7) are equivalent. (The advantage of the Galerkin formulation is, of course, that it may be used in problems which do not admit of a variational formulation.)

In the *Least-squares method* we find the constants a_i in (4) from the condition

$$J[y] = \int_a^b L^2[y_n] \, dx = \text{minimum}$$

or $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ (8a, b)

$$\int_a^b L \, \partial L/\partial a_i \, dx = 0,$$

i.e. we are minimising the integral of the square of the error over $[a, b]$. This condition gives the following equations for the constants a_i,

$$\sum_{j=1}^{n} \alpha_{ij} a_i = \beta_i, \qquad i = 1, 2, \ldots, n,$$

where

$$\alpha_{ij} = \alpha_{ji} = \int_a^b M[v_i] M[v_j] \, dx \qquad (9a, b, c)$$

and

$$\beta_i = \int_a^b f M[v_i] \, dx.$$

Clearly there is greater algebraic labour in setting up these equations than in (5) or (7), but the method is of general applicability like the Galerkin method.

USE OF SYMBOLIC COMPUTATION

The above solution techniques can be summarised in the following simple algorithm. For definiteness we refer to equations (5), but the algorithm is essentially the same for the other related techniques and problems considered in this paper.

ALGORITHM

(i) Construct the coefficients α_{ij} and β_i in *symbolic form* (i.e. in terms of i and j) from equations (5b) and (5c) with the assumed $v_i(x)$, using algebraic differentiation and integration.

(ii) Evaluate α_{ij} and β_i appropriately to set up equations (5a), taking advantage of the *symmetry* $\alpha_{ij} = \alpha_{ji}$.

(iii) Solve equations (5a) using arbitrary-precision rational arithmetic.

(iv) Print the final solution in symbolic form, e.g. as in (16).

When we come to the non-linear problem (22), we shall employ a numerical phase—in solving the system of non-linear algebraic equations (24a) for the constants a_i. That is, the step corresponding to (iii) is accomplished using floating-point computation, and so in this case the overall computation is hybrid.

We wish to emphasise here that a symbolic computation system enables us to extend the analytic phase of the computation far beyond what could be contemplated without such a system. This is especially true of problems involving a parameter (such as λ in the eigenvalue problem (18)) and non-linear problems where considerable algebraic simplification is required to form the non-linear algebraic equations prior to their solution. Also, in the symbolic phase all arithmetic is done exactly through the use of arbitrary-precision rational arithmetic, in contrast to ordinary floating-point computation which involves round-off error. In fact, in several of the large systems, output from the symbolic phase can be converted to (FORTRAN) expressions for incorporation in a program to perform numerical computation (see Ng, 1979).

To effect step (iii) above, it is necessary to use a symbolic matrix algebra package using arbitrary-precision rational arithmetic. The author wrote his own package in MUSIMP. This exercise was to gain experience with the MUSIMP language but also to overcome the problem that the TRS80 version of MUMATH does not come with such a package. (The CP/M version does.) In writing this package the author handled arrays as lists. Consequently, it was necessary to write functions to access and change the ith element of a list L (for a one-dimensional array). This was done in LISP fashion with the *recursive* functions

```
FUNCTION AREL(L,I),
  WHEN I=1, FIRST(L) EXIT,
  AREL(REST(L),I-1),
ENDFUN$

FUNCTION REPLCAREL(L,I,R),
  WHEN I=1, REPLACEF(L,R) EXIT,
  REPLCAREL(REST(L),I-1,R),
ENDFUN$
```

respectively, with obvious extensions to the (i, j)th element of a list of lists (for a two-dimensional array). The authors of the CP/M MUMATH matrix package took a different approach, based on the stack operations PUSH and POP.

We note finally here that step (i) cannot always be effected with the MUMATH system because of its limited integration capability. In these circumstances, the values of α_{ij} and β_i were computed directly by the system for each appropriate integer value of i and j.

EXAMPLE. We first consider a well-known special case of (1), viz.

$$y''+y+x = 0 \tag{10}$$

with boundary conditions $y(0) = y(1) = 0$ (see Kantorovich & Krylov (1958, p. 269) and Collatz (1960, p. 220)). This problem has the exact solution

$$y = \sin(x)/\sin(1)-x. \tag{11}$$

Both of these authors give the Ritz–Galerkin solutions for $n = 2$ using the basis functions

$$v_i = x-x^{1+i}, \tag{12}$$

viz.

$$y \approx y_2 = 71x/369-8x^2/369-21x^3/123, \tag{13}$$

for which the absolute error (in relation to the exact solution $y(x)$) $|e(x)| < 0{\cdot}0004$ in $[0, 1]$. Collatz also gives the least-squares solution for $n = 2$. Rational arithmetic was used throughout. The present author used this problem as a test case for his programs and successfully reproduced (13). Evidently (12) is not the best form for the basis functions as the series expansion of the exact solution (11) proceeds in odd powers of x. Consequently, this author used

$$v_i = x - x^{2i+1}, \tag{14}$$

and thereby found the $n = 2$ Ritz–Galerkin solution

$$y_2 = 657x/3488 - 345x^3/1744 + 33x^5/3488, \tag{15}$$

which has $|e| < 6 \times 10^{-6}$ in [0, 1]. He determined also the $n = 5$ solution

$$\begin{aligned} y_5 = {} & 112104387771x/595049363456 - 117858958525x^3/595049363456 \\ & + 2946473915x^5/297524681728 - 70153985x^7/297524681728 \\ & + 5844685x^9/1785148090368 - 52003x^{11}/1785148090368, \end{aligned} \tag{16}$$

which has $|e| < 7 \times 10^{-14}$ in [0, 1]. Bearing in mind that (16) was *computed and printed automatically in a few minutes* on a TRS80 Model I running MUMATH, we see at once the power of even a small algebraic manipulation system for the present class of problems: few indeed would seriously contemplate the algebraic labour involved in obtaining this result by hand.

The corresponding results for the least-squares method were obtained but we do not reproduce them here. We note only that the overall computing time for the $n = 5$ case was over twice that of the Ritz–Galerkin solution (16) and the maximum absolute error was slightly greater ($|e| < 2 \times 10^{-13}$).

EIGENVALUE PROBLEMS

Here we replace the non-homogeneous term f in (1) by $-\lambda y$ and consider the *homogeneous* boundary conditions

$$y = 0 \quad \text{at} \quad x = a \quad \text{and} \quad x = b.$$

If we seek the solution in the form (4), we find the Ritz–Galerkin equations are then

$$\sum_{j=1}^{n} (\alpha_{ij} - \lambda \gamma_{ij}) a_j = 0, \qquad i = 1, 2, \ldots, n$$

with

$$\alpha_{ij} \text{ as defined in (5b) or (7b)} \tag{17a, b, c}$$

and

$$\gamma_{ij} = \int_a^b v_i v_j \, dx.$$

Equations (17) are a system of n homogeneous equations in n unknowns which have a non-trivial solution only if the determinant of coefficients is zero. On expansion this determinant yields a polynomial of degree n in λ whose n roots are approximations to the first n eigenvalues of the differential-equation eigenvalue problem.

EXAMPLE. We consider the well-known problem

$$\begin{aligned} y'' + \lambda y &= 0 \\ y(-1) = y(1) &= 0, \end{aligned} \tag{18a, b}$$

whose lowest eigenvalue is $\lambda_1 = (\pi/2)^2$ with corresponding eigensolution $\cos(\pi x/2)$. We assume the basis functions to have the form

$$v_i(x) = (1 - x^2) x^{2(i-1)} \tag{19}$$

(i.e. selecting the *even* eigensolutions of (18)). Kantorovich & Krylov (1958, p. 297) give the *characteristic equation* for $n = 2$ and $n = 3$, viz.

$$\begin{aligned} \lambda^2 - 28\lambda + 63 &= 0 \qquad (\lambda_1 = 2{\cdot}467437) \\ 4\lambda^3 - 450\lambda^2 + 8910\lambda - 19305 &= 0 \qquad (\lambda_1 = 2{\cdot}46740111). \end{aligned} \tag{20a, b}$$

The absolute error of the last approximation to λ_1 is already less than 10^{-8} in magnitude. For $n = 4$ and $n = 5$ the present author found

$$\begin{aligned} \lambda^4 - 308\lambda^3 + 21021\lambda^2 - 360360\lambda + 765765 &= 0 \\ 2\lambda^5 - 1365\lambda^4 + 229320\lambda^3 - 12530700\lambda^2 + 198402750\lambda - 416645775 &= 0. \end{aligned} \tag{21a, b}$$

Solving these equations by the Newton–Raphson method, we find $\lambda_1 = 2{\cdot}4674011002730$ and $2{\cdot}4670411002723397$ with absolute errors of less than 7×10^{-13} and 5×10^{-17} in magnitude respectively. The corresponding eigensolutions differ in magnitude from $\cos(\pi x/2)$ by less than 2×10^{-7} and 6×10^{-10} respectively in $[-1, 1]$. These are extremely good results. As is well known, the accuracy of the higher frequency solutions becomes progressively poorer, though for $n = 4$ and $n = 5$ the second eigenvalue was found to be accurate to 3 and 5 decimals respectively from (21a, b).

The characteristic equation was obtained from (17) by MUSIMP functions written by the author. These functions are based on the *recursive* Laplace expansion of a determinant in terms of its cofactors (an approach not recommended for large n because of the $n!$ growth in the number of arithmetic operations). This step had to be taken, however, since the determinant evaluator in the MUSIMP matrix algebra package is based on Gauss elimination and will not produce the characteristic equation directly (because the MUMATH system does not cancel common factors in numerators and denominators unless these appear explicitly). This shortcoming is not present on the more sophisticated algebra systems or later versions of MUMATH.

NON-LINEAR EQUATIONS

The Ritz and Galerkin methods give rise here to sets of non-linear algebraic equations for the constants a_i. The formation of these equations is somewhat more complicated and their solution is more difficult to obtain.

We illustrate with the example

$$\begin{aligned} y'' &= 3y^2/2 \\ y(0) = 4, &\quad y(1) = 1 \end{aligned} \tag{22a, b}$$

considered by Collatz (1960, p. 212) which has the exact solution $y = 4/(1+x)^2$. With the basis

$$\begin{aligned} y_n &= 4 - 3x + \sum_{i=1}^{n} a_i v_i \\ v_i &= x - x^{i+1} \end{aligned} \tag{23a, b}$$

the Ritz–Galerkin equations reduce to the system of non-linear equations

$$\sum_{j=1}^{n} \left(\alpha_{ij} - \sum_{k=1}^{n} \beta_{ijk} a_k \right) a_j - \gamma_i = 0, \qquad i = 1, 2, \ldots, n,$$

where

$$\alpha_{ij} = \int_0^1 v_i[v_j'' - 3(4-3x)v_j]\,dx,$$

$$\beta_{ijk} = (3/2)\int_0^1 v_i v_j v_k\,dx \qquad \text{(24a, b, c, d)}$$

and

$$\gamma_i = (3/2)\int_0^1 (4-3x)^2 v_i\,dx.$$

Collatz obtained the two non-linear equations corresponding to $n = 2$ above and gave the appropriate solution $a_1 = -7{\cdot}07004$, $a_2 = 2{\cdot}72044$. The relative error of this approximation is less than 2% in magnitude everywhere in [0, 1]. The present author obtained the equations for $n = 3$ and $n = 4$ using MUMATH. For brevity, only the solutions are quoted here:

$$\begin{array}{ll} a_1 = -9{\cdot}840060 & a_1 = -11{\cdot}1862943 \\ a_2 = 7{\cdot}342806 & a_2 = 11{\cdot}3823795 \\ a_3 = -2{\cdot}318631 & a_3 = -7{\cdot}0419785 \\ & a_4 = 1{\cdot}8941640. \end{array} \qquad \text{(25a, b)}$$

These sets of non-linear equations were solved without difficulty by the Newton–Raphson method for systems of equations. Solutions (25a, b) have relative errors which are respectively less than 0·25% and 0·06% in magnitude everywhere in [0, 1].

We note that it is tempting to proceed in MUMATH without first effecting the algebraic reductions (24), i.e. allowing the system to square out and collect all the terms before differentiating with respect to the a_i. If this approach is adopted, the growth in the size of the intermediate expressions as n becomes large soon uses up all the available memory of a small microcomputer. (This growth problem is, of course, true, to a lesser degree, of the linear problems especially when using the least-squares method.) Consequently, it is always advisable with a small system to perform as much preliminary simplification as possible of the type exemplified by equations (24).

3. Partial Differential Equations

The previous techniques may readily be applied to partial differential equations.

POISSON'S EQUATION

We consider the well-known elliptic boundary-value problem (torsion problem)

$$\begin{aligned} \nabla^2 u &= -2 \quad \text{in } S \\ u &= 0 \quad \text{on boundary of } S, \end{aligned} \qquad \text{(26a, b)}$$

where S is the square $-1 \leqslant x \leqslant 1$, $-1 \leqslant y \leqslant 1$ (see Kantorovich & Krylov, 1958, p. 281).

With the basis

$$u_n = \sum_{i=1}^{n} a_i v_i(x, y) \qquad (27)$$

we find the Ritz–Galerkin equations for the constants a_i are

$$\sum_{j=1}^{n} \alpha_{ij} a_j + \beta_i = 0, \qquad i = 1, 2, \ldots, n,$$

where

$$\alpha_{ij} = \alpha_{ji} = \int_{-1}^{1}\int_{-1}^{1} \left[\frac{\partial v_i}{\partial x}\frac{\partial v_j}{\partial x} + \frac{\partial v_i}{\partial y}\frac{\partial v_j}{\partial y}\right] \mathrm{d}x\,\mathrm{d}y \quad \text{(Ritz)}$$

$$= -\int_{-1}^{1}\int_{-1}^{1} v_i \nabla^2 v_j \,\mathrm{d}x\,\mathrm{d}y \quad \text{(Galerkin)} \tag{28a, b, c}$$

and

$$\beta_i = -2\int_{-1}^{1}\int_{-1}^{1} v_i \,\mathrm{d}x\,\mathrm{d}y.$$

With the v_i as

$$v_i = (x^2-1)(y^2-1)(x^2+y^2)^{i-1}, \tag{29}$$

which satisfies the boundary conditions, Kantorovich & Krylov give the $n = 1$ and $n = 2$ solutions

$$\begin{aligned} u_1 &= 5(x^2-1)(y^2-1)/8 \\ u_2 &= (x^2-1)(y^2-1)(1295/2216 + 525(x^2+y^2)/4432) \end{aligned} \tag{30a, b}$$

respectively. The present author obtained the $n = 3$ and $n = 4$ solutions with the MUMATH system, corresponding to

$$\begin{aligned} a_1 &= \frac{10061391}{16987888} & a_1 &= \frac{26947213979609}{45802341113840} \\ a_2 &= \frac{537075}{8493944} & a_2 &= \frac{10106985607719}{91604682227680} \\ a_3 &= \frac{3942939}{67951552} & a_3 &= \frac{-9630261958317}{183209364455360} \\ & & a_4 &= \frac{24088961157261}{366418728910720} \end{aligned} \tag{31a, b}$$

respectively. Analytical solutions to this problem are well known (see, for example, Kantorovich & Krylov, 1958, p. 283), so we find for $n = 3$, $|e| < 0{\cdot}021$ and for $n = 4$, $|e| < 0{\cdot}005$ throughout S. We found that the symmetry of this problem made it more efficient to use the Ritz form of (28b) than the Galerkin form for the evaluation of the coefficients α_{ij} with MUMATH.

BIHARMONIC PROBLEMS

Next, consider the problem (vertically loaded clamped plate)

$$\begin{aligned} \nabla^4 u &= 1 \quad \text{in } R \\ u &= 0 \\ \partial u/\partial \nu &= 0, \quad \text{on boundaries of } R \end{aligned} \tag{32a, b, c}$$

where R is the annulus $1 \leqslant r \leqslant 2$, $0 \leqslant \theta \leqslant 2\pi$, r, θ are plane polar coordinates and ν is the normal (see Rektorys, 1975, p. 302).

Assuming circular symmetry, so that

$$\nabla^2 \equiv \frac{d^2}{dr^2} + \frac{1}{r}\frac{d}{dr}$$

and taking

$$u_n = \sum_{i=1}^{n} a_i v_i(r), \tag{33}$$

we find the Ritz–Galerkin equations for the a_i are ($'$ denoting differentiation with respect to r)

$$\sum_{j=1}^{n} \alpha_{ij} a_j - \beta_i = 0, \qquad i = 1, 2, \ldots, n,$$

where

$$\begin{aligned} \alpha_{ij} = \alpha_{ji} &= \int_0^{2\pi}\int_1^2 (v_i'' + v_i'/r)(v_j'' + v_j'/r) r \, dr \, d\theta \quad \text{(Ritz)} \\ &= \int_0^{2\pi}\int_1^2 v_i \nabla^4 v_j r \, dr \, d\theta \quad \text{(Galerkin)} \end{aligned} \tag{34a, b, c}$$

and

$$\beta_i = \int_0^{2\pi}\int_1^2 v_i r \, dr \, d\theta.$$

With

$$v_i = (r^2 - 1)^2 (4 - r^2)^2 r^{2(i-1)}, \tag{35}$$

which satisfies the boundary conditions, we find the $n = 1$ and $n = 2$ approximations correspond to

$$a_1 = \frac{7}{19968} \qquad \begin{aligned} a_1 &= \frac{1043}{1088832} \\ a_2 &= \frac{-35}{181472} \end{aligned} \tag{36a, b}$$

respectively. Equation (36b) agrees with the result given by Rektorys. The $n = 3$ and $n = 4$ approximations were computed by the author with MUMATH and these correspond to

$$\begin{aligned} a_1 &= \frac{24433}{14593472} & a_1 &= \frac{710317319}{295661116096} \\ a_2 &= \frac{-15305}{21890208} & a_2 &= \frac{-227133005}{147830558048} \\ a_3 &= \frac{121}{1368138} & a_3 &= \frac{29721483}{73915279024} \\ & & a_4 &= \frac{-2119975}{55436459268} \end{aligned} \tag{37a, b}$$

respectively. The exact solution is again known (see Rektorys, 1975, p. 304) so we find for $n = 3$ $|e| < 0{\cdot}0001$ and for $n = 4$ $|e| < 0{\cdot}000025$ everywhere in R. We again found the Ritz form better than the Galerkin form for computing the coefficients α_{ij} with MUMATH.

4. Integral Equations

Our final example is a Fredholm integral equation. Many such equations admit of a variational formulation (e.g. see Collatz, 1960, p. 487). We will, however, use the Galerkin approach to solve the special case

$$y(x)-\frac{1}{4}\int_0^{\frac{1}{2}\pi} xty(t)\,\mathrm{d}t = \sin x - \frac{1}{4}x \tag{38}$$

taken from Delves & Walsh (1974, p. 98) where it is solved by an entirely different method. (The kernel is *separable*, and so we can easily construct the exact solution, $y(x) = \sin x$.)

Using the basis (4), we find the Galerkin equations are

$$\sum_{j=1}^{n} \alpha_{ij}a_j - \beta_i = 0, \qquad i = 1, 2, \ldots, n,$$

where

$$\alpha_{ij} = \alpha_{ji} = \int_0^{\frac{1}{2}\pi}\left[v_j(x)-\frac{1}{4}\int_0^{\frac{1}{2}\pi} xtv_j(t)\,\mathrm{d}t\right]v_i(x)\,\mathrm{d}x \tag{39a, b, c}$$

and

$$\beta_i = \int_0^{\frac{1}{2}\pi}(\sin x - x/4)v_i(x)\,\mathrm{d}x.$$

As in Delves & Walsh we take

$$v_i(x) = x^{2i-1}, \tag{40}$$

bearing in mind the power series expansion of the true solution.

This example is harder for MUMATH in several ways. First, if we retain π as a symbolic entity throughout, the intermediate expressions become prohibitively long for MUMATH when $n > 2$. Consequently, we found it necessary to give π an appropriate rational approximation at the start of the computation. Even if this is done, the intermediate rational numbers very quickly increase in size and soon exhaust the main memory of the TRS80 Model I. Use of the floating-point package suggested by the authors of MUMATH (see Douglas, 1982) does not solve this problem either as the full rational representation of any number is still retained in memory. To overcome these difficulties, the author wrote a MUSIMP function TRUNC which simply truncates a given number of the least significant digits in the numerator and denominator of a long rational fraction. At least four more digits were always retained than the number of significant figures quoted in the final results, so as to counter any build up of error in the intermediate computations. Lastly, we have to integrate products of powers of x and $\sin x$ which requires either the use of the INTMORE package in the CP/M MUMATH system or the writing of a special function. The former uses up valuable main memory. The author used both techniques as a check. (It is, of course, recognised that all these difficulties would easily be overcome on a large system such as REDUCE2.) The following approximations were found with MUMATH,

$$\begin{aligned}
y_1 &= 24x/\pi^3\\
y_3 &= 0{\cdot}99977x-0{\cdot}16583x^3+0{\cdot}00757x^5\\
y_5 &= 1{\cdot}000000002x-0{\cdot}166666632x^3+0{\cdot}008333148x^5\\
&\quad -0{\cdot}000198154x^7+0{\cdot}000002618x^9.
\end{aligned} \tag{41a, b, c}$$

For the last two results, $|e| < 0{\cdot}0016$ and 3×10^{-8} respectively everywhere in $[0, \pi/2]$. The three-term approximation differs only slightly from that given in Delves & Walsh. It may be of interest that the five-term approximation is only very slightly less accurate than the best five-term polynomial approximation of $\sin x$ in $[0, \pi/2]$ (see, for example, Lyusternik *et al.*, 1965, p. 91).

5. Concluding Remarks

It has been shown how the MUMATH algebraic manipulation system running on one of the first widely available microcomputers can very effectively solve many of the classical variational, Galerkin and least-squares problems in differential and integral equations. Solutions are obtained using rational arithmetic. In all cases, comparisons were made with known exact solutions.

We remark finally that the present work exemplifies the combined use of algebraic and numerical computation as a powerful tool in Applied Mathematics and Engineering. It is hoped that the paper might help to stimulate further growth in this important area.

This work was completed while the author was on Study Leave from the University of Glasgow. He is indebted to the Editor and Referees for helpful comments.

References

Andersen, C. M., Noor, A. (1977). Symbolic manipulation techniques for vibration analysis of laminated elliptic plates. In: *Proc. 1977 MACSYMA Users' Conference*, pp. 161–175. NASA Publication CP-2012, NASA, Washington, USA.

Bogen, R. A. (1979). Addendum to "Analytically solving integral equations by using computer algebra". *ACM Trans. Math. Software* **5**, 234–237.

Brown, W. S., Hearn, A. C. (1979). Applications of symbolic algebraic computation. *Comput. Phys. Commun.* **17**, 207–215.

Collatz, L. (1960). *The Numerical Treatment of Differential Equations* (3rd edn). Berlin: Springer-Verlag.

Delves, L., Walsh, J. (eds) (1974). *Numerical Solution of Integral Equations*. Oxford University Press.

Douglas, B. (1982). Copernica Mathematica. *80 Micro* **6**, 412–426.

Fitch, J. (1979). A survey of symbolic computation in physics. In: (Ng, E. W., ed.) Symbolic and Algebraic Computation. *Springer Lec. Notes Comp. Sci.* **72**, 30–41.

Fitch, J. (1983). Implementing REDUCE on a microcomputer. In: (van Hulzen, J. A., ed.) Computer Algebra. *Springer Lec. Notes Comp. Sci.* **162**, 128–136.

Golden, J. P. (1977). MACSYMA's symbolic ordinary differential equation solver. In: *Proc. 1977 MACSYMA Users' Conference*, pp. 1–10. NASA Publication CP-2012, NASA, Washington, USA.

Kantorovich, L. V., Krylov, V. I. (1958). *Approximate Methods of Higher Analysis*. Groningen: Noordhoff.

Lyusternik, L. A. *et al.* (1965). *Handbook for Computing Elementary Functions*. Oxford: Pergamon Press.

Microsoft Consumer Products (1980). *MUMATH/MUSIMP Instruction Booklet (for the TRS80)*. Bellvue, Washington, USA.

Microsoft (1980). *The MUMATH/MUSIMP Symbolic Mathematics System Reference Manual*. Bellvue, Washington, USA.

Miola, A. (1974). The use of symbolic computation in solving free boundary problems by the Ritz–Galerkin method. In: Proc. EUROSAM '74, pp. 103–104. *SIGSAM Bull.* **8**.

Ng, E. W. (1979). Symbolic-numeric interface: a review. In: (Ng, E. W., ed.) Symbolic and Algebraic Computation. *Springer Lec. Notes Comp. Sci.* **72**, 330–345.

Rektorys, K. (1975). *Variational Methods in Mathematics, Science and Engineering*. Dordrecht: Reidel.

Stoutemyer, D. (1977). Analytically solving integral equations by using computer algebra. *ACM Trans. Math. Software* **3**, 128–146.