TECHNICAL NOTES AND SHORT PAPERS

Tests of Parlett's ALGOL Eigenvalue Procedure *Eig* 3

By George E. Forsythe

The ALGOL eigenvalue procedure *Eig* 3 in Section 14 of Parlett [1] was transcribed into Extended Algol [2] for the Burroughs B 5000 and used to obtain the eigenvalues of several matrices.

The procedures were altered as follows:

1. Formal changes were made to conform to the Burroughs character set for ALGOL. For example,

a. All lower case letters were changed to capitals.

b. As a result, the label L in procedure *Triangle* was changed to LL, to avoid conflict with the integer l.

c. V was changed to OR.

2. Two minor changes were made to conform to syntactic differences between B 5000 ALGOL and true ALGOL:

a. Each label was declared in the head of its innermost block.

b. The heading of procedure *Evaluate* was deleted. The body was labeled Hy, and this block was substituted for Parlett's procedure statement labeled Hy.

c. The procedure *mod* was deleted, and the expression $mod\ (j - 1, 3)$ was replaced by $(J - 1)\ MOD\ 3$, valid in B 5000 ALGOL.

3. To complete the program ALGOL bodies were written for the procedures *comsqrt* and *scale*.

4. In the absence of an overflow procedure, *overflow* was changed to a local Boolean variable in the body of procedure *Laguerre*, but set permanently to **false**.

5. Two irrelevant types of change were made in certain expressions:

a. In some places a factor like $y \uparrow 2$ was changed to $y \times y$.

b. Some inequalities of form $a > b$ were reversed to read $b < a$.

6. The texts of some comments were changed.

In the absence of overflow none of these changes should affect the program at all. Separate run-time indications assured us that we did not have overflow.

The B 5000 procedure *EIG* 3 was used to obtain the eigenvalues of matrices A and B in Section 14 of Parlett [1]. All computed eigenvalues were correct to within 2×10^{-7}. The compiling time from a card-deck input was approximately 43 seconds, including a test program. The running time was approximately 8 seconds for each matrix, excluding output time.

James Varah of Stanford University used *EIG* 3 to obtain the eigenvalues of 100 matrices of order 10 generated in a certain random manner. For each matrix,

Received November 1, 1963. This work was sponsored by the Office of Naval Research under Contract Nonr 225(37), and by the National Science Foundation under grant NSF-GP948. Machine time was furnished by the Burroughs Corporation. The author wishes to thank Beresford Parlett and W. M. McKeeman for assistance.

the trace was compared with the sum of the computed eigenvalues, and found to differ by only a few units in the last place.

A copy of the B 5000 program and its output have been deposited in the UMT file.

Computer Science Division
Stanford University
Stanford, California 94305

1. B. Parlett, "Laguerre's method applied to the matrix eigenvalue problem," *Math. Comp.*, v. 18, 1964, p. 469–485.
2. Burroughs Corporation, Equipment and Systems Marketing Division, "Extended Algol reference manual for the Burroughs B 5000," Detroit, 32, Michigan, November, 1962.

Differential Approximation Applied to the Solution of Convolution Equations

By Richard Bellman, Robert Kalaba and Bella Kotkin

1. Introduction. In the course of constructing some mathematical models of physiological processes connected with cancer chemotherapy [1], we have encountered functional equations containing convolution terms. Equations of this type are unpleasant computationally because of the storage, and thus time, requirements for solution. In some cases, these storage requirements could exceed present capabilities and thus seriously impede numerical solution.

We wish to present a new approach to this problem using the technique of differential approximation. To illustrate the method, we shall consider the equation

$$u(t) = f(t) + \int_0^t e^{-(t-s)^2} u(s)\, ds. \tag{1.1}$$

2. Polynomial Approximation and Extensions. A classical problem, which owes its inception to a control process associated with the Watt steam engine (see [2]), is that of obtaining a polynomial which deviates the least from a given function, where the deviation is measured by an assigned norm.

If we recognize that a polynomial $p_n(t) = a_0 + a_1 t + \cdots + a_n t^n$ is a solution of the linear differential equation

$$\frac{d^{(n+1)}u}{dt^{(n+1)}} = 0, \tag{2.1}$$

then we see immediately that this problem is a particular case of the more general problem of finding an equation

$$\frac{d^{(n+1)}u}{dt^{(n+1)}} + a_1(t)\frac{d^n u}{dt^n} + \cdots + a_n(t)u = 0, \tag{2.2}$$

Received September 6, 1963. This investigation was supported in part by Public Health Service Research Grant Number RG-9608, from the Division of General Medical Sciences, National Institutes of Health.