A Method for Solving Nonlinear Volterra Integral Equations of the Second Kind*

By Peter Linz**

Abstract. The approach given in this paper leads to numerical methods for Volterra integral equations which avoid the need for special starting procedures. Formulae for a typical fourth-order method are derived and some numerical examples presented. A convergence theorem is given for the method described.

1. Introduction. In this paper we consider the numerical solution of the equation

$$(1) \qquad y(x) = g(x) + \int_0^x K(x, t, y(t))dt .$$

Here $y(x)$ is the unknown function which is to be determined in some interval $0 \leq x \leq a$. The function $g(x)$ and the kernel $K(x, t, y)$ are given and are assumed to satisfy the conditions:

(1) $g(x)$ is continuous and bounded in $0 \leq x \leq a$,

(2) $K(x, t, y)$ is bounded and uniformly continuous in x and t for all finite y and $0 \leq t \leq x \leq a$,

(3) $K(x, t, y)$ satisfies a uniform Lipschitz condition

$$|K(x, t, y_1) - K(x, t, y_2)| \leq L|y_1 - y_2|$$

for all $0 \leq t \leq x \leq a$.

It is well known that under these conditions Eq. (1) has a unique solution [8, p. 42].

In practice, approximate solutions to Eq. (1) are frequently obtained by finite-difference methods, such that approximate values for $y(x)$ are computed at certain discrete points (mesh-points) of the range. Several approaches exist for developing such numerical methods. The first consists of replacing the integral in (1) by some numerical quadrature using values at the mesh-points and satisfying the resulting equation at these points. The system of equations thus obtained can be solved, one point at a time. Discussions of this approach have been given by Fox and Goodwin [2], Noble [5], and Mayers [4]. In general, special starting procedures are needed to obtain the solution at the first few points. Some typical starting procedures were given by Noble [5] and Day [1].

A different class of approximation formulae is based on an extension of the Runge-Kutta methods to Eq. (1). This has been studied in detail by Pouzet [6], [7]. Runge-Kutta methods are self-starting, but tend to be complicated and inefficient and hence of limited practical use.

A third approach uses numerical quadrature, but the computations are arranged such that several values of y are obtained at the same time. This leads to what are

---

generally called block-by-block methods, and it is with these that we are concerned here. The block-by-block approach was first suggested by Young [9] in connection with product integration techniques. The present paper extends this idea to develop methods which are generally useful. The main advantage of this approach is due to the fact that no special starting procedures are needed; thus the methods are simple to use and switching step-size presents no problem.

2. Block-by-Block Methods. The basic interval $[0, a]$ is divided into steps of width h, such that $x_i = ih$ and $Nh = a$. The approximate solution will be defined at the mesh-points x_i and denoted by Y_i, such that Y_i is an approximation to $y(x_i)$. We rewrite Eq. (1) in the form

$$y(x_n) = g(x_n) + \int_0^{x_{pm}} K(x_n, t, y(t))dt + \int_{x_{pm}}^{x_n} K(x_n, t, y(t))dt ,$$

where p is some integer and $m =$ integer part of n/p. If the values $Y_0, Y_1, \cdots, Y_{pm}$ are known, then the first integral can be approximated by standard quadrature methods. The second integral is estimated by a quadrature rule using values of the integrand at $t = x_{pm}, x_{pm+1}, \cdots, x_{p(m+1)}$. Since the values of Y at these points are unknown, we have a system of p simultaneous equations

$$(2) \qquad Y_n = g(x_n) + h \sum_{i=0}^{mp} w_{ni} K(x_n, x_i, Y_i) + h \sum_{i=0}^{p} w'_{ni} K(x_n, x_{mp+i}, Y_{mp+i}) ,$$

for $n = mp + 1, mp + 2, \cdots, p(m + 1)$, where w_{ni} and w'_{ni} depend on the quadrature rule used. For sufficiently small h the system (2) has a unique solution which can be determined by iteration (or directly, if the system is linear). Thus, a "block" of p values of Y is obtained simultaneously.

For example, using $p = 2$ the integration over $[0, x_{2m}]$ can be accomplished by Simpson's rule, and the integral over $[x_{2m}, x_n]$ by using a quadratic interpolation of the integrand at the points $x_{2m}, x_{2m+1}, x_{2m+2}$. Then (2) becomes

$$Y_{2m+1} = g(x_{2m+1}) + \frac{h}{3} [K(x_{2m+1}, x_0, Y_0) + 4K(x_{2m+1}, x_1, Y_1) +$$
$$\cdots + K(x_{2m+1}, x_{2m}, Y_{2m})]$$
$$(3) \qquad + \frac{5h}{12} K(x_{2m+1}, x_{2m}, Y_{2m}) + \frac{2h}{3} K(x_{2m+1}, x_{2m+1}, Y_{2m+1})$$
$$- \frac{h}{12} K(x_{2m+1}, x_{2m+2}, Y_{2m+2}) ,$$

$$Y_{2m+2} = g(x_{2m+2}) + \frac{h}{3} [K(x_{2m+2}, x_0, Y_0) + 4K(x_{2m+2}, x_1, Y_1) +$$
$$(4) \qquad\qquad \cdots + 4K(x_{2m+2}, x_{2m+1}, Y_{2m+1})$$
$$+ K(x_{2m+2}, x_{2m+2}, Y_{2m+2})] .$$

Equations (3) and (4) constitute a valid method for the numerical solution of Eq. (1). However, it is not completely general since it requires the use of the kernel outside the region $0 \leqq t \leqq x \leqq a$. If the kernel does not exist outside this region or if it is ill-behaved near the edges, numerical difficulties may arise. This objection can be overcome by the following modification. We write

$$\int_{x_{2m}}^{x_{2m+1}} K(x_{2m+1}, t, y(t))dt \simeq \frac{h}{6} K(x_{2m+1}, x_{2m}, Y_{2m})$$

$$+ \frac{2h}{3} K(x_{2m+1}, x_{2m+1/2}, Y_{2m+1/2}) + \frac{h}{6} K(x_{2m+1}, x_{2m+1}, Y_{2m+1}) .$$

Now $Y_{2m+1/2}$ is not known, but may be estimated by quadratic interpolation

$$Y_{2m+1/2} \simeq \tfrac{3}{8} Y_{2m} + \tfrac{3}{4} Y_{2m+1} - \tfrac{1}{8} Y_{2m+2} .$$

The resulting equations then are

$$Y_{2m+1} = g(x_{2m+1})$$

$$+ \frac{h}{3} [K(x_{2m+1}, x_0, Y_0) + 4K(x_{2m+1}, x_1, Y_1) + \cdots + K(x_{2m+1}, x_{2m}, Y_{2m})]$$

(5)
$$+ \frac{h}{6} K(x_{2m+1}, x_{2m}, Y_{2m}) + \frac{2h}{3} K(x_{2m+1}, x_{2m+1/2}, \tfrac{3}{8} Y_{2m}$$

$$+ \tfrac{3}{4} Y_{2m+1} - \tfrac{1}{8} Y_{2m+2})$$

$$+ \frac{h}{6} K(x_{2m+1}, x_{2m+1}, Y_{2m+1}) ,$$

(6)
$$Y_{2m+2} = g(x_{2m+2}) + \frac{h}{3} [K(x_{2m+2}, x_0, Y_0) + 4K(x_{2m+2}, x_1, Y_1)$$

$$+ \cdots + K(x_{2m+2}, x_{2m+2}, Y_{2m+2})] .$$

3. Theoretical Results.

Definition 1. Let $Y_0(h)$, $Y_1(h)$ $\cdots$ denote the approximation obtained by a given method using step-size h. Then a method is said to be convergent if and only if

$$|Y_i(h) - y(x_i)| \to 0 , \quad \text{for } i = 0, 1, 2, \cdots, N$$

as $h \to 0$, $N \to \infty$, such that $Nh = a$.

Definition 2. A method is said to be of order q if q is the largest number for which there exists a finite constant C such that

$$|Y_i(h) - y(x_i)| \leqq Ch^q , \quad i = 0, 1, \cdots, N ,$$

for all $h > 0$.

Our aim is to show that method (5), (6) converges and to establish its order of convergence. We need the following lemma.

LEMMA. *If*

$$|\xi_n| \leqq A \sum_{i=0}^{n-1} |\xi_i| + B , \quad A > 0 , \quad B > 0 ,$$

then $|\xi_n| \leqq B(1 + A)^n$.

The proof follows immediately by induction. As a corollary we have that, if $A = hK$ and $x = nh$, then

(7)
$$|\xi_n| \leqq Be^{Kx} .$$

The main result then follows immediately.

THEOREM. *The approximation method given by Eqs.* (5) *and* (6) *is convergent and its order of convergence is four.*

Proof. Let $\epsilon_i = y(x_i) - Y_i$. Then, using Eqs. (1), (5), (6) and the Lipschitz condition, we can find a constant C such that

$$|\epsilon_{2m+1}| \leqq hC \sum_{i=0}^{2m} |\epsilon_i| + hC|\epsilon_{2m+1}| + hC|\epsilon_{2m+2}| + |R_{2m+1}| \, ,$$

$$|\epsilon_{2m+2}| \leqq hC \sum_{i=0}^{2m} |\epsilon_i| + hC|\epsilon_{2m+1}| + hC|\epsilon_{2m+2}| + |R_{2m+2}| \, ,$$

where the R's represent the quadrature errors involved in integrating $K(x, t, y(t))$ using the given quadrature rule. Setting $\|\epsilon_m\| = \max_{i=1,2} |\epsilon_{2m+i}|$, and $R = \max_i |R_i|$, we have, from (7),

$$\|\epsilon_m\| \leqq (1 - 2hC)^{-1} R \exp [Cx_n/(1 - 2hC)] \, .$$

For sufficiently smooth K and y, $R = O(h^4)$, thus

$$\|\epsilon_m\| = O(h^4)$$

and the theorem is proven.

4. Numerical Examples.

Example 1.

$$y(x) = 1 + x - \cos x - \int_0^x \cos (x - t)y(t)dt \, .$$

Solution. $y(x) = x$.

The absolute values of ϵ_i for various step-sizes are given in Table 1.

<div align="center">Table 1</div>

x	$h = .2$	$h = .1$	$h = .05$
.2	5.80×10^{-8}	3.21×10^{-8}	1.94×10^{-9}
.4	1.98×10^{-6}	1.16×10^{-7}	7.17×10^{-9}
.6	4.11×10^{-6}	2.40×10^{-7}	1.50×10^{-8}
.8	6.40×10^{-6}	3.98×10^{-7}	2.45×10^{-8}
1.0	9.89×10^{-6}	5.61×10^{-7}	3.52×10^{-8}
1.2	1.19×10^{-5}	7.39×10^{-7}	4.63×10^{-8}
1.4	1.61×10^{-5}	9.16×10^{-7}	5.75×10^{-8}
1.6	1.74×10^{-5}	1.09×10^{-6}	6.85×10^{-8}
1.8	2.20×10^{-5}	1.25×10^{-6}	7.84×10^{-8}
2.0	2.21×10^{-5}	1.39×10^{-6}	8.77×10^{-8}

Example 2 (From [1]).

$$y(x) = 3 + 2x - \int_0^x [2(x - t) + 3]y(t)dt \, .$$

Solution. $y(x) = 4e^{-2x} - e^{-x}$.

The errors are shown in Table 2.

5. Concluding Remarks.
The method which we have established here constitutes a convenient algorithm for solving nonlinear Volterra equations of the

second kind which is computationally efficient and of high accuracy. While only one method was explicitly given, it is clear that the approach can be generalized for arbitrary order. A general statement of the convergence theorem can be found in [3]. Also, the method can easily be extended to a system of simultaneous Volterra equations.

TABLE 2

x	$h = .2$	$h = .1$	$h = .05$
.2	3.26×10^{-3}	7.08×10^{-5}	4.38×10^{-6}
.4	1.54×10^{-3}	9.26×10^{-5}	5.73×10^{-6}
.6	4.52×10^{-4}	9.04×10^{-5}	5.60×10^{-6}
.8	1.30×10^{-4}	7.82×10^{-5}	4.83×10^{-6}
1.0	1.94×10^{-4}	6.29×10^{-5}	3.89×10^{-6}
1.2	8.05×10^{-4}	4.82×10^{-5}	2.98×10^{-6}
1.4	2.26×10^{-4}	3.55×10^{-5}	2.20×10^{-6}
1.6	4.24×10^{-4}	2.53×10^{-5}	1.56×10^{-6}
1.8	1.38×10^{-4}	1.74×10^{-5}	1.08×10^{-6}
2.0	1.94×10^{-4}	1.15×10^{-5}	7.12×10^{-7}

University of Wisconsin Computing Center
Madison, Wisconsin 53706

1. J. T. DAY, "A starting method for solving nonlinear Volterra integral equations," *Math. Comp.*, v. 21, 1967, pp. 179–188. MR **36** #6168.
2. L. FOX & E. T. GOODWIN, "The numerical solution of non-singular linear integral equations," *Philos. Trans. Roy. Soc. London Ser. A.*, v. 245, 1953, pp. 501–534. MR **14**, 908.
3. P. LINZ, *The Numerical Solution of Volterra Integral Equations by Finite Difference Methods*, MRC Technical Summary Report #825, Mathematics Research Center, University of Wisconsin, Madison, Wis., 1967.
4. D. F. MAYERS, *Numerical Solution of Ordinary and Partial Differential Equations*, Pergamon Press, Oxford and Addison-Wesley, Reading, Mass., 1962, Chapters 13, 14. MR **26** # 4488.
5. B. NOBLE, "The numerical solution of nonlinear integral equations and related topics," in *Nonlinear Integral Equations*, edited by P. M. Anselone, University of Wisconsin Press, Madison, Wis., 1963. MR **28** #4321.
6. P. POUZET, *Étude en Vue de leur Traitment Numérique d'Équations Intégrales et Intégro-Différentielles du Type de Volterra pour des Problèmes de Conditions Initiales*, Thesis, University of Strassbourg, 1962.
7. P. POUZET, "Méthode d'intégration numérique des équations intégrales et intégro-différentielles du type de Volterra de seconde espèce. Formules de Runge-Kutta," *Symposium on the Numerical Treatment of Ordinary Differential Equations, Integral and Integro-differential Equations*, (Rome, 1960), Birkhäuser Verlag, Basel, 1960, pp. 362–368. MR **23** #B601.
8. F. G. TRICOMI, *Integral Equations*, Pure and Appl. Math., vol. V, Interscience, New York, 1957. MR **20** #1177.
9. A. YOUNG, "The application of approximate product-integration to the numerical solution of integral equations," *Proc. Roy. Soc. London Ser. A*, v. 224, 1954, pp. 561–573. MR **16**, 179.