Asian Journal of Fuzzy and Applied Mathematics (ISSN: 2321 – 564X)
Volume 01– Issue 01, June 2013

A Fourth-Order Robust Numerical Method for Integro-differential Equations

Ali Filiz[1]

[1]Department of Mathematics
Faculty of Science
Adnan Menderes University
09010 Aydin-Turkey

ABSTRACT— ***In this paper linear Volterra integro-differential equations are discussed. Examples of these questions have been solved numerically using various methods for ODE (Ordinary Differential Equation) parts and quadrature rules for integral parts. Finally, a new fourth order routine is used for the numerical solution of the linear integro-differential equation***.

Keywords— Volterra integro-differential (integral) equation, θ-method, Runge-Kutta Methods, Truncation error, quadrature rule, fourth order.

---

1. INTRODUCTION

The term integral equation was apparently first used by Du Bois-Reymond in 1883. An integral equation is an equation in which the function to be determined appears under the integral sign. If we consider linear integral equations, that is, equations in which no nonlinear functions of the unknown function are involved. Consider the linear integro-differential equation of the form

$$(1) \qquad u'(t) = F\left(t, u(t), \int_0^t K(t,s)u(s)ds\right), \qquad u(0) = u_0, \quad t \geq 0.$$

Equation (1) can be solved numerically using various methods. In this paper $u(t_n)$ will denote the exact value of u at $t_n = t_0 + nh$. We shall use $\tilde{u}(t_n)$ or $\tilde{u}_n$ to denote an approximate value of u at t_n. However, in this paper we will define third and fourth order numerical methods for (1). Since the integral can not be determined explicitly, it may be approximated using familiar numerical integration methods. The Newton-Cotes integration formulas, which include the trapezoidal rule and Simpson's 1/3, are well suited here since they use nodes which were given in [8] and [2]. In section 2, we consider an elementary class of formulae for the numerical solution of integro-differential equation of first-order, based upon the θ-method. Recall that a functional equation in which the unknown function appear in the form of its derivatives as well as under the integral sign is called an integro-differential equation (see [12]).

In practice the equations encountered are rather more complex than these examples, and the Laplace transform technique is therefore primarily a theoretical tool.

2. THE NUMERICAL SOLUTION OF INTEGRO-DIFFERENTIAL EQUATIONS

In general formulae for numerical solution of integro-differential equations rely upon formulae for the underlying ODE (Ordinary Differential Equation), combined with auxiliary quadrature rules approximation of

$$(2) \qquad \tilde{z}(t_n) := h\sum_{j=0}^{n} \omega_{n,j} K(t_n, t_j)\, \tilde{u}(t_j) \approx \int_{t_0}^{t_n} K(t,s)\, u(s)\, ds.$$

If we adapt the θ-method to equation (1) in a natural manner, we obtain

$$(3) \qquad \tilde{u}_{n+1} = \tilde{u}_n + h\, \left((1-\theta)F(t_n, \tilde{u}_n, \tilde{z}(t_n)) + \theta F(t_{n+1}, \tilde{u}_{n+1}, \tilde{z}(t_{n+1}))\right),$$

where $\tilde{z}(t_n)$ defined in (2) and $\theta \in [0,1]$. If we define $\tilde{z}(t_n)$ for θ-method, we have

$$\tilde{z}(t_n) = \theta h\left(\omega_{n,0}K(t_n,t_0)\tilde{u}(t_0) + \sum_{j=1}^{n-1}\omega_{n,j}K(t_n,t_j)\tilde{u}(t_j) + \omega_{n,n}K(t_n,t_n)\tilde{u}(t_n)\right).$$

There are three important cases: θ=0, θ=1 and θ=0.5. If θ=0.5 then the equation (3) becomes

$$(4) \qquad \tilde{u}_{n+1} = \tilde{u}_n + h\left(\frac{1}{2}F(t_n,\tilde{u}_n,\tilde{z}(t_n)) + \frac{1}{2}F(t_{n+1},\tilde{u}_{n+1},\tilde{z}(t_{n+1}))\right).$$

The trapezoidal rule has local truncation error $O(h^3)$, if K and F sufficiently smooth and the convergence $O(h^2)$. If θ=0 then the equation (3) becomes

$$(5) \qquad \tilde{u}_{n+1} = \tilde{u}_n + h\ F(t_n,\tilde{u}_n,\tilde{z}(t_n)).$$

This method is then an explicit Euler method and has corresponding local truncation error (LTE) $O(h^2)$, and the convergence $O(h)$.

If θ=1 then the equation (3) becomes

$$(6) \qquad \tilde{u}_{n+1} = \tilde{u}_n + h\ F(t_{n+1},\tilde{u}_{n+1},\tilde{z}(t_{n+1})).$$

This method is a backward- Euler method and has corresponding LTE $O(h^2)$, and the convergence $O(h)$. Of course, whereas we have defined approximations $\tilde{z}(t_n)$ in terms of quadrature rules that reflect the underlying ODE method, it is in principle possible to "mix and match". Thus we can combine (2) with the approximation (3). The combinations of the formulae can be chosen on the basis of orders of convergence. The above discussion was based on the method. There are two directions in which this method can be generalized. The first involves adapted linear multistep methods (LMM) for ODE's and second involves adapting Runge-Kutta methods. In each case, we shall require to approximate terms

$$\tilde{z}(t_n) := h\sum_{j=0}^{n}\omega_{n,j}K(t_n,t_j)\ \tilde{u}(t_j) \approx \int_{t_0}^{t_n}K(t,s)u(s)ds \quad \text{at selected values at t.}$$

Equation (1) can be solved numerically using various methods. In this paper we will focus on third and fourth order numerical methods for (1). The integral can not be determined explicitly; it may be approximated using familiar numerical integration methods.

Table 1: Table of orders of convergence Example 3.1 of (1).

ODE/Quadrature	***Rectangle(Left End)***	***Trapezoidal***	***Rectangle(Right End)***
Forward Euler	$O(h)$	$O(h)$	$O(h)$
Trapezoidal	$O(h)$	$O(h^2)$	$O(h)$
Backward Euler	$O(h)$	$O(h)$	$O(h)$

The Newton-Cotes integration formulae, which include the trapezoidal rule and Simpson's 1/3, are well suited here since they use nodes which were previously calculated:

$$\int_{t_0}^{t_n}K(t,s)u(s)ds \approx\ h\sum_{j=0}^{n}\omega_{n,j}K(t_n,t_j)\ \tilde{u}(t_j),$$

where $\omega_{n,j}$ are the appropriate coefficients for the composite integration schemes chosen. A combination of integration method may be used. Thus, the composite explicit Euler-Newton Cotes scheme (with first order accuracy) applied to (1) becomes

$$\tilde{u}_{n+1} = \tilde{u}_n + h\ F(t_n,\tilde{u}_n,\tilde{z}(t_n)).$$

Simpson's 1/3 rule requires that n, the number of subintervals dividing $[t_0, t_n]$ be even. Therefore, Simpson's 1/3 rule can not be used at each step. When n is odd, one method is to use Simpson's 1/3 rule on $[t_0, t_{n-1}]$ and trapezoidal rule on $[t_{n-1}, t_n]$, adding the results to approximate the integral on $[t_0, t_n]$. Another method is to use the trapezoidal rule $[t_0, t_1]$ and Simpson's 1/3 rule thereafter.

3. THE FOURTH ORDER NUMERICAL ROBUST ROUTINE OF LINEAR INTEGRO-DIFFERENTIAL EQUATIONS

The explicit finite difference method given in (5) as applied to (1) easily extended to more accurate predictor-corrector method. The predictor step uses (5) to obtain $\tilde{u}^{\kappa}_{n+1}$, which is followed by the corrector step, which uses higher order trapezoidal method

$$(7) \qquad \tilde{u}_{n+1} = \tilde{u}_n + h\left(\frac{1}{2}F(t_n, \tilde{u}_n, \tilde{z}(t_n)) + \frac{1}{2}F(t_{n+1}, \tilde{u}^{\kappa}_{n+1}, \tilde{z}(t_{n+1}))\right).$$

This procedure is sometimes referred to as modified Euler method (second order Runge-Kutta-RK2) and is one order magnitude more accurate than the explicit method.

The fourth order classical Runge-Kutta method (RK4) can also be adapted to the numerical solution of (1). Stepping from $\tilde{u}_n$ with step-size h to obtain $\tilde{u}_{n+1}$, the RK4 method as applied to this problem may be written as

$$k_1 = hF(t_n, \tilde{u}_n, \tilde{z}(t_n)\ ),$$

$$\tilde{u}^b_{n+1/2} = \tilde{u}_n + \frac{k_1}{2}, \qquad k_2 = hF(t_{n+1/2}, \tilde{u}^b_{n+1/2}, \tilde{z}_{n+1/2}\ ),$$

$$k_2 = hF(t_{n+1/2}, \tilde{u}^b_{n+1/2}, \tilde{z}_n + \frac{h}{4}\left[\tilde{u}_n + \tilde{u}^b_{n+1/2}\right]),$$

$$\tilde{u}^c_{n+1/2} = \tilde{u}_n + \frac{k_2}{2}, \qquad k_3 = hF(t_{n+1/2}, \tilde{u}^c_{n+1/2}, \tilde{z}_{n+1/2}\ ),$$

$$k_3 = hF(t_{n+1/2}, \tilde{u}^c_{n+1/2}, \tilde{z}_n + \frac{h}{4}\left[\tilde{u}_n + \tilde{u}^c_{n+1/2}\right]),$$

$$\tilde{u}^p_{n+1} = \tilde{u}_n + k_3, \qquad k_4 = hF(t_{n+1}, \tilde{u}^p_{n+1}, \tilde{z}_{n+1}\ ),$$

$$k_4 = hF(t_{n+1}, \tilde{u}^p_{n+1}, \tilde{z}_n + \frac{h}{6}\left[\tilde{u}_n + 4\tilde{u}^c_{n+1/2} + \tilde{u}^p_{n+1}\right]),$$

$$(8) \quad \tilde{u}_{n+1} = \tilde{u}_n + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right).$$

In this example, the trapezoidal rule is used to approximate $\tilde{z}(t_n) \approx \int_{t_0}^{t_n} K(t,s)u(s)ds$ on $[t_n, t_{n+1/2}]$ in calculating, k_2 and k_3 while Simpson's 1/3 rule is used on $[t_n, t_{n+1}]$ in calculating k_4. If desired, the trapezoidal rule may be used on $[t_0, t_n]$ (gives second order accuracy); the trapezoidal rule and Simpson's 1/3 rule (giving third order accuracy, see Table 3) may be used on $[t_0, t_n]$.

In order to get fourth order accuracy the integral term must be evaluated more accurately on $[t_n, t_{n+1/2}]$ in calculating k_2 and k_3, as shown in (9) and (10) below. While (10) is used $[t_n, t_{n+1}]$ in calculating k_4 accurately (better than trapezoidal rule). Simpson's method I may be used on $[t_0, t_n]$ or Simpson's method II (see these methods [9]) may be used on $[t_0, t_n]$. If we interpolating on $\tilde{u}_{n-1}$, $\tilde{u}_n$, $\tilde{u}_{n+1/2}$ (special formulae required for the first step, for example we can use (8)) Lagrange's formula for points t= -1, 0, 1/2 gives

$$u(t) = \frac{1}{h^2}\left[\frac{2}{3}t(t-\frac{h}{2})u_{-1} - 2(t+h)(t-\frac{h}{2})u_0 + \frac{4}{3}t(t+h)u_{1/2}\right].$$

If we integrate the expression between 0 and h/2, we get

(9) $$\int_0^{h/2} u(s)ds \approx h(-\frac{1}{72}u_{-1} + \frac{7}{24}u_0 + \frac{2}{9}u_{1/2}).$$

Similarly, we can find t= -1, 0, 1

(10) $$\int_0^{h} u(s)ds \approx h(-\frac{1}{12}u_{-1} + \frac{2}{3}u_0 + \frac{5}{12}u_1).$$

Therefore the Runge-Kutta formulae become $n \geq 2$ (for starting values we can use equation (8))

$$k_1 = hF(t_n, \tilde{u}_n, \tilde{z}(t_n)\ ),$$

$$\tilde{u}^b_{n+1/2} = \tilde{u}_n + \frac{k_1}{2}, \qquad k_2 = hF(t_{n+1/2}, \tilde{u}^b_{n+1/2}, \tilde{z}_{n+1/2}\ ),$$

$$k_2 = hF(t_{n+1/2}, \tilde{u}^b_{n+1/2}, \tilde{z}_n + h\left[-\frac{1}{72}\tilde{u}_{n-1} + \frac{7}{24}\tilde{u}_n + \frac{2}{9}\tilde{u}^b_{n+1/2}\right]),$$

$$\tilde{u}^c_{n+1/2} = \tilde{u}_n + \frac{k_2}{2}, \qquad k_3 = hF(t_{n+1/2}, \tilde{u}^c_{n+1/2}, \tilde{z}_{n+1/2}\ ),$$

$$k_3 = hF(t_{n+1/2}, \tilde{u}^c_{n+1/2}, \tilde{z}_n + h\left[-\frac{1}{72}\tilde{u}_{n-1} + \frac{7}{24}\tilde{u}_n + \frac{2}{9}\tilde{u}^c_{n+1/2}\right]),$$

$$\tilde{u}^p_{n+1} = \tilde{u}_n + k_3, \qquad k_4 = hF(t_{n+1}, \tilde{u}^p_{n+1}, \tilde{z}_{n+1}\ ),$$

$$k_4 = hF(t_{n+1}, \tilde{u}^p_{n+1}, \tilde{z}_n + h\left[-\frac{1}{12}\tilde{u}_{n-1} + \frac{2}{3}\tilde{u}_n + \frac{5}{12}\tilde{u}^p_{n+1}\right]),$$

(11) $$\tilde{u}_{n+1} = \tilde{u}_n + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right).$$

Table 3 shows the fourth order accuracy obtained with this formula. In Example 3.1, we have used Runge-Kutta methods and numerical quadratures, θ-method, the trapezoidal rule and Simpson's rules and their combinations.

Example 3.1: Consider a first order Linear Volterra integro-differential equation of the form

(12) $$u'(t) = \mu + \lambda\int_0^t k(t-s)\, u(s)\, ds, \quad t \geq 0; \qquad u(0) = u_0.$$

This equation (12) has analytical solution $u(t) = \frac{\mu}{\lambda}\sinh(\sqrt{\lambda}\ t) + u_0\ \cosh(\sqrt{\lambda}\ t)$ when k (t-s) =1.

Case (i): If we choose $\lambda = -1,\ \mu = 0$ and u_0 =1, we obtain $u(t)$ = cos(t).

Case (ii): If we choose $\lambda = -1,\ \mu = 1$ and u_0 =0, we obtain $u(t)$ =sin(t).

In this case, differentiation gives $u''(t) = \lambda u(t)$ and the equation reduces to second order ODE. If the kernel of convolution type ($K(t,s) = k(t-s)$), and $\int_0^\infty |k(\sigma)|\,d\sigma < \infty$, we can solve (1), with suitable initial conditions, by Laplace transforms (see [6], [5] and [1]). We use the trapezoidal rule for the integral term. The errors found are given Table 2 and Table 3, where ***error=|true value – approximate value***|. Unless otherwise indicated, in this paper, error means absolute error. Table 2 is consistent with the property that the order of the error is $O(h^2)$.

Table 2: Example 3.1 solved using (3) with θ=0.5, $\mu = 1,\ \lambda = -1,\ u_0 = 0,\ t_{\max} = 1$ gives $O(h^2)$.

t	*Error with h=0.1*	*Error with h=0.05*	*Error with h=0.025*
0.2	1.6310E-04	4.0821E-05	1.0208E-05
0.4	3.0658E-04	7.6728E-05	1.9187E-05
0.6	4.1212E-04	1.0313E-04	2.5790E-05
0.8	4.6393E-04	1.1608E-04	2.9027E-05
1.0	4.4987E-04	1.1254E-04	2.8139E-05

Table 3: Errors in the solution of (12) with;

Errors in the Solutions (12) for Various Methods ($\mu = 1,\ \lambda = -1,\ u_0 = 0,\ t_{\max} = 1$)									
	Exact	(A) Explicit		(B) Implicit		(C) RK4 and SimpTrap		(D) RK4 and Simp II	
t	Solution	h=0.025	h=0.0125	h = 0.025	h = 0.0125	h=0.025	h=0.0125	h = 0.025	h = 0.0125
0.1	0.0998334	5.7228e-05	2.9907e-05	6.7538e-05	3.2485e-05	3.9424e-09	5.5109e-10	7.3484e-10	3.3681e-11
0.2	0.1986693	2.3829e-04	1.2166e-04	2.5828e-04	1.2666e-04	1.7562e-08	2.3103e-09	1.0650e-09	5.4422e-11
0.3	0.2955202	5.3976e-04	2.7348e-04	5.6822e-04	2.8059e-04	4.0594e-08	5.2430e-09	1.3730e-09	7.3910e-11
0.4	0.3894183	9.5579e-04	4.8236e-04	9.9091e-04	4.9114e-04	7.2586e-08	9.2908e-09	1.6503e-09	9.1611e-11
0.5	0.4794255	1.4782e-03	7.4414e-04	1.5176e-03	7.5399e-04	1.1290e-07	1.4373e-08	1.8888e-09	1.0702e-10
0.6	0.5646425	2.0965e-03	1.0535e-03	2.1375e-03	1.0638e-03	1.6073e-07	2.0385e-08	2.0813e-09	1.1967e-10
0.7	0.6442177	2.7983e-03	1.4043e-03	2.8375e-03	1.4141e-03	2.1509e-07	2.7207e-08	2.2209e-09	1.2913e-10
0.8	0.7173561	3.5692e-03	3.6030e-03	4.0738e-06	1.7975e-03	2.7488e-07	3.4696e-08	2.3018e-09	1.3503e-10
0.9	0.7833269	4.3930e-03	2.1999e-03	4.4178e-03	2.2061e-03	3.3884e-07	4.2695e-08	2.3191e-09	1.3705e-10
1.0	0.8414710	5.2524e-03	2.6280e-03	5.2641e-03	2.6309e-03	4.0562e-07	5.1034e-08	2.2691e-09	1.3496e-10
	Exact	(E) RK2 and Trap		(F) RK4 and Trap		(G) RK4 and TrapSimp		(H) RK4 and Simp I	
t	Solution	h=0.025	h=0.0125	h = 0.025	h = 0.0125	h=0.025	h=0.0125	h = 0.025	h = 0.0125
0.1	0.0998334	5.1818e-06	1.2955e-06	7.1956e-09	1.9741e-09	2.3161e-09	2.4629e-10	7.3484e-10	3.3686e-11
0.2	0.1986693	1.0208e-05	2.5522e-06	6.2993e-08	1.6502e-08	7.8435e-09	8.9431e-10	1.0657e-09	5.4461e-11
0.3	0.2955202	1.4926e-05	3.7317e-06	2.1827e-07	5.6291e-08	1.6477e-08	1.9315e-09	1.3753e-09	7.4015e-11
0.4	0.3894183	1.9187e-05	4.7971e-06	5.2191e-07	1.3354e-07	2.8050e-08	3.3373e-09	1.6552e-09	9.1811e-11
0.5	0.4794255	2.2852e-05	5.7133e-06	1.0198e-06	2.5971e-07	4.2334e-08	5.0837e-09	1.8973e-09	1.0734e-10
0.6	0.5646425	2.5790e-05	6.4478e-06	1.7539e-06	4.4524e-07	5.9044e-08	7.1356e-09	2.0941e-09	1.2014e-10
0.7	0.6442177	2.7883e-05	6.9711e-06	2.7614e-06	6.9939e-07	7.7842e-08	9.4512e-09	2.2388e-09	1.2976e-10
0.8	0.7173561	2.9027e-05	7.2572e-06	4.0738e-06	1.0300e-06	9.8344e-08	1.1983e-08	2.3255e-09	1.3585e-10
0.9	0.7833269	2.9136e-05	7.2844e-06	5.7162e-06	1.4432e-06	1.2012e-07	1.4677e-08	2.3491e-09	1.3808e-10
1.0	0.8414710	2.8139e-05	7.0351e-06	7.7067e-06	1.9437e-06	1.4271e-07	1.7477e-08	2.3057e-09	1.3619e-10

Table 3. Errors in the solution of (12) with;

(A) The explicit ($\theta = 0$) method and the trapezoidal rule (gives error O(h));

(B) The implicit ($\theta = 1$) method and the trapezoidal rule (gives error O(h));

(C) The fourth order Runge-Kutta method (RK4) and Simpson's 1/3 rule and the trapezoidal rule (gives error $O(h^3)$);

(D) The fourth order Runge-Kutta method (RK4) and Simpson's method II (gives error $O(h^4)$);

(E) The second order Runge-Kutta method (RK2) and the trapezoidal rule (gives error $O(h^2)$);

(F) The fourth order Runge-Kutta method (RK4) and the trapezoidal rule (gives error $O(h^2)$);

(G) The fourth order Runge-Kutta method (RK4) and the trapezoidal rule and Simpson's 1/3 rule (gives error $O(h^3)$);

(H) The fourth order Runge-Kutta method (RK4) and Simpson's method I (gives error $O(h^4)$);

4. CONCLUSION

In fact, after this numerical calculation we were expecting order of $O(h^4)$. In view, it seems to be true because of the truncation error for the classical fourth order Runge-Kutta (RK4) and Simpson's rules are $O(h^4)$. Thus, we found the expected $O(h^4)$. Numerical order of convergence is also calculated:

$$Ord = \frac{\ln(Error_1) - \ln(Error_2)}{\ln(2)}$$

We expected that Ord=4. Obtained theoretical results are confirmed by numerical experiments.

5. REFERENCES

[1] Asanov, A., Uniqueness of the solution of systems of convolution-type Volterra integral equations of the first kind, In: Inverse problems for differential equations of mathematical physics (Russian), pp. 26-34, 155, Novosibirsk: Akad. Nauk SSSR sibirsk. Otdel. Vychil. Tsentr., 1978.

[2] Baker, C. T. H., The Numerical Treatment of Integral Equations, Clarendon Press; Oxford University Press, 1977.

[3] Baker, C. T. H., G A Bochorov, A Filiz, N J Ford, C A H Paul, F A Rihan, A Tang, R M Thomas, H Tian and D R Wille., Numerical Modelling By Retarded Functional Differential Equations, Numerical analysis report, Manchester Centre for Computational Mathematics, No:335, ISSN 1360-1725, 1998.

[4] Baker, C. T. H., G A Bochorov, A Filiz, N J Ford, C A H Paul, F A Rihan, A Tang, R M Thomas, H Tian and D R Wille., Numerical modelling by delay and Volterra functional differential equations, In: Computer Mathematics and its Applications- Advances and Developments (1994-2005), Editor: Elias A. Lipitakis, LEA Publishers, Athens, Greece, 2006.

[5] Bellman, R., A survey of the theory of the boundedness stability and asymptotic behaviour of solutions of linear and non-linear differential and difference equations, Washington, 1949.

[6] Cooke, K. L., Functional differential equations close to differential equation, Amer. Math. Soc., 72, 285-288, 1966.

[7] Filiz, A., On the solution of Volterra and Lotka-Volterra type equations, LMS supported One Day Meeting in Delayed Differential Equation (Liverpool, UK), 12th March 2000.

[8] Filiz, A., Numerical solution of some Volterra integral equations, PhD Thesis, University of Manchester, 2000.

[9] Linz, P. Analytical and Numerical Methods for Volterra Equations, SIAM, Philadelphia,1985.

[10] Volterra, V., (1931)., Le¸cons sur la Th´eorie Math´ematique de la Lutte Pour la Vie, Gauthier-Villars, Paris, 1931.

[11] Volterra, V., Theory of Functionals and of Integro-Differential Equations, Dover, New York, 1959.

[12] Volterra, V., Sulle equazioni integro-differenziali della teoria dell'elastica, Atti della Reale Accademia dei Lincei 18 (1909), Reprinted in Vito Volterra, Opera Mathematiche; Memorie e Note Vol. 3. Accademia dei Lincei Rome, pp. 288-293, 1957.