Research Article *http://dx.doi.org/10.4314/mejs.v13i2.2*

Eighth order Predictor-Corrector Method to Solve Quadratic Riccati Differential Equations

Wase Kassahun, Alemayehu Shiferaw, and Solomon Gebregiorgis*
Department of Mathematics, Faculty of Natural Sciences, Jimma University, Jimma, Ethiopia (*solomonggty@gmail.com).

ABSTRACT

In this paper, the eighth-order predictor-corrector method is presented for solving quadratic Riccati differential equations. First, the interval is discretized and then the method is formulated by using Newton's backward difference interpolation formula. The stability and convergence of the method have been investigated. To validate the applicability of the proposed method, two model examples with exact solutions have been considered and numerically solved. Maximum absolute errors are presented in tables and figures for different values of mesh size h and the present method gives better results than some existing numerical methods reported in the literature.

Keywords: Predictor-corrector method, Riccati differential equations, Stability analysis.

1. INTRODUCTION

The Riccati differential equation is named after the Italian nobleman Count Jacopo Francesco Riccati (1676–1754). The book of Reid (1972) contains the fundamental theories of Riccati equation, with applications to random processes, optimal control, and diffusion problems. Beside important engineering and science applications that today are known as the classical proved, such as stochastic realization theory, optimal control, robust stabilization, and network synthesis, the newer applications include such areas as financial mathematics (Biazar and Islami, 2010). Nonlinear deferential equations are essential tools for modeling many physical situations, for instance, spring mass systems, resistor-capacitor-induction circuits, bending of beams, chemical reaction, pendulums, motion of rotating mass around body and so on.

Riccati equation is a basic first-order nonlinear ordinary differential equation. It has the form $\frac{dy}{dx} = p(x) + q(x)y + r(x)y^2$ which can be considered as the lowest order nonlinear approximation to the derivative of a function in terms of the function itself. It is assumed that $y(x)$, $p(x)$, $q(x)$ and $r(x)$ are real functions of the real argument x. It is well known that solutions to the general Riccati equation are not available and only special cases can be treated (Ince, 1956). Even though the equation is nonlinear, similar to the second order inhomogeneous linear ordinary differential equations one needs only a particular solution to find the general solution (Anas et al.,

Submitted: *21st June 2020* ***Revised and Accepted:*** *6th December 2020*

2010). This problem has attracted much attention and has been studied by many authors. Tan and Abbasbandy (2008) employed the analytic technique called Homotopy Analysis Method (HAM) to solve a quadratic Riccati equation. Mukherje and Roy (2012) presented the solution of Riccati equation with variable co-efficient by differential transformation method. Batiha (2015) applied the multistage variational iteration method as a new efficient method for solving quadratic Riccati differential equation. Gemechis File and Tesfaye Aga (2016) presented fourth order Runge-Kutta method for solving quadratic Riccati differential equations. Vinod and Dimple (2016) presented Newton-Raphson based modified Laplace Adomian decomposition method for solving quadratic Riccati differential equations. Gemadi et al. (2017) presented fifth order predictor corrector method for solving quadratic Riccati differential equation. Fateme and Esmaile (2017) presented approximate solution for quadratic Riccati differential equations by Bezier curves method. Ghomanjani and Khorram (2017) presented method of Bezier curves, by developing the Bezier polynomial of degree n. Ghomanjani and Shateyi (2020) presented an effective algorithm for solving quadratic Riccati differential equation based on Genocchi polynomials.

Hence, there are many continuous attempts to get a method that yields more accurate results. Therefore, the purpose of this study is to formulate a more accurate and stable method for solving quadratic Riccati differential equation than some existing methods in the literature.

2. DESCRIPTION OF THE METHODS

2.1 Description of the Method

Consider the quadratic Riccati differential equation of the form

$$\frac{dy}{dx} = p(x) + q(x)y + r(x)y^2 , \; y(x_0) = \alpha , \; x_0 \le x \le x_f . \tag{1}$$

Where, $p(x)$, $q(x)$ and $r(x)$ are continuous with $r(x) \neq 0$ and x_0, x_f, α are arbitrary constants for $y(x)$, which is unknown function. To describe the scheme, we denote the problem in equation (1) as:

$$\frac{dy}{dx} = f\left(x, y\right). \tag{2}$$

And divide the interval $\left[x_0, x_f\right]$ into N equal sub intervals of mesh length h and the mesh points given by $x_i = x_0 + ih;\; i = 1, 2, ..., N$. Then $h = \dfrac{x_f - x_0}{N}$

Where, N is positive integer.

Integrating equation (2) on interval $\left[x_i, x_{i+1}\right]$, we obtain:

$$y\left(x_{i+1}\right) = y\left(x_i\right) + \int_{x_i}^{x_{i+1}} f\left(x, y\right) dx \quad . \tag{3}$$

To derive the method, we approximate $f\left(x, y\right)$ by Newton's backward difference interpolation polynomials.

2.1.1. Description of Predictor Method

Taking t data values $\left(x_i, f_i\right), \left(x_{i-1}, f_{i-1}\right), \left(x_{i-2}, f_{i-2}\right), ..., \left(x_{i-t+1}, f_{i-t+1}\right)$, we fit the Newton's backward difference interpolating polynomial of degree $t-1$ and we get:

$$P_{t-1}\left(x\right) = f\left(x_i + kh\right) = f_i + k\nabla f_i + \frac{k\left(k+1\right)}{2!}\nabla^2 f_i + ... + \frac{k\left(k+1\right)\left(k+2\right)...\left(k+t-2\right)}{\left(t-1\right)!}\nabla^{t-1} f_i + T_t^p \tag{4}$$

Where, $k = \frac{x - x_i}{h}$ and $T_t^p = \frac{k\left(k+1\right)\left(k+2\right)...\left(k+t-1\right)}{t!} h^t f^{(t)}\left(\xi\right)$ is the error term, when ξ lies in some interval containing the points $x_i, x_{i-1}, ..., x_{i-t+1}$ and x. The limits of integration in equation (3) becomes: $x = x_i \Rightarrow k = 0$, $x = x_{i+1} \Rightarrow k = 1$ and $dx = hdk$.

Replacing $f(x, y)$ by $P_{t-1}\left(x\right)$ in equation (3) and using equation (4), we get:

$$y\left(x_{i+1}\right) = y\left(x_i\right) + \int_0^1 \left\{ f_i + k\nabla f_i + \frac{k\left(k+1\right)}{2!}\nabla^2 f_i + ... \right\} dk \tag{5}$$

By choosing different values for *t*, we get different methods. But for this particular study, we choose the value for $t = 8$ which is of order eighth method.

Now, on integrating term by term in equation (5) with respect to k, we obtain:

$$\int_0^1 kdk = \frac{1}{2}, \int_0^1 k\left(k+1\right)dk = \frac{5}{6}, \int_0^1 k\left(k+1\right)\left(k+2\right)dk = \frac{9}{4}, \int_0^1 k\left(k+1\right)\left(k+2\right)\left(k+3\right)dk = \frac{251}{30},$$

$$\int_0^1 k\left(k+1\right)\left(k+2\right)\left(k+3\right)\left(k+4\right)dk = \frac{475}{12}, \int_0^1 k\left(k+1\right)\left(k+2\right)\left(k+3\right)\left(k+4\right)\left(k+5\right)dk = \frac{19087}{84},$$

$\int_0^1 k(k+1)(k+2)(k+3)(k+4)(k+5)(x+6)\,dk = \frac{36799}{24}$, and

$\int_0^1 k(k+1)(k+2)(k+3)(k+4)(k+5)(x+6)(x+7)\,dk = \frac{1070017}{90}$.

Thus, we get:

$$
\begin{aligned}
y(x_{i+1}) &= y(x_i) + \int_0^1 \left\{ f_i + k\nabla f_i + \frac{k(k+1)}{2!}\nabla^2 f_i + \ldots \right\} dk \\
&= y(x_i) + h\Big[f_i + \frac{1}{2}\nabla f_i + \frac{5}{12}\nabla^2 f_i + \frac{3}{8}\nabla^3 f_i + \frac{251}{720}\nabla^4 f_i + \frac{475}{144}\nabla^5 f_i + \frac{19087}{60480}\nabla^6 f_i \\
&\quad + \frac{5257}{17280}\nabla^7 f_i \Big] + T_8 \\
&= y_i + h\Big[f_i + \frac{1}{2}(f_i - f_{i-1}) + \frac{5}{12}(f_i - 2f_{i-1} + f_{i-2}) + \frac{3}{8}(f_i - 3f_{i-1} + 3f_{i-2} - f_{i-3}) \\
&\quad + \frac{251}{720}(f_i - 4f_{i-1} + 6f_{i-2} - 4f_{i-3} + f_{i-4}) + \frac{475}{144}\begin{pmatrix} f_i - 5f_{i-1} + 10f_{i-2} - 10f_{i-3} \\ +5f_{i-4} - f_{i-5} \end{pmatrix} \\
&\quad + \frac{19087}{60480}(f_i - 6f_{i-1} + 15f_{i-2} - 20f_{i-3} + 15f_{i-4} - 6f_{i-5} + f_{i-6}) \\
&\quad + \frac{5257}{17280}(f_i - 7f_{i-1} + 21f_{i-2} - 35f_{i-3} + 35f_{i-4} - 21f_{i-5} + 7f_{i-6} - f_{i-7}) \Big] + T_8 \\
&= y_i + \frac{h}{120960}\big[434241 f_i - 1152169 f_{i-1} + 2183877 f_{i-2} - 2664477 f_{i-3} + 2102243 f_{i-4} \\
&\quad - 1041723 f_{i-5} + 295767 f_{i-6} - 36799 f_{i-7}\big] + T_8
\end{aligned} \tag{6}
$$

Where, $T_8 = \frac{1070017}{3628800} h^8 f^{(8)}(\xi)$ is the local truncation error. Hence, equation (6) is called eighth order predictor method.

2.1.2. Description of Corrector Method

Taking $t+1$ data values $(x_{i+1}, f_{i+1}), (x_i, f_i), (x_{i-1}, f_{i-1}), (x_{i-2}, f_{i-2}), \ldots, (x_{i-t+1}, f_{i-t+1})$, we fit the Newton's backward difference interpolating polynomial of degree t and we get:

$$
\begin{aligned}
P_t(x) = f(x_i + kh) &= f_{i+1} + (k-1)\nabla f_{i+1} + \frac{k(k-1)}{2!}\nabla^2 f_{i+1} + \ldots \\
&\quad + \frac{(k-1)k(k+1)(k+2)\ldots(k+t-1)}{(t)!}\nabla^t f_{i+1} + T_t^c
\end{aligned} \tag{7}
$$

Where, $k = \dfrac{x - x_i}{h}$, $x - x_{i+1} = (x - x_i) - (x_{i+1} - x_i) = kh - h = h(k-1)$ and $T_t^c = \dfrac{k(k+1)(k+2)...(k+t-1)}{(t+1)!} h^{t+1} f^{(t+1)}(\xi)$ is the error term, when ξ lies in some interval containing the points $x_{i+1}, x_i, x_{i-1}, ..., x_{i-t+1}$ and x. The limits of integration in equation (3) becomes: $x = x_i \Rightarrow k = 0$, $x = x_{i+1} \Rightarrow k = 1$ and $dx = hdk$.

Replacing $f(x, y)$ by $P_{t-1}(x)$ in equation (3) and using equation (7) we get:

$$y(x_{i+1}) = y(x_i) + \int_0^1 \left\{ f_{i+1} + (k-1)\nabla f_{i+1} + \frac{k(k-1)}{2!}\nabla^2 f_{i+1} + ... \right\} dk \tag{8}$$

By choosing different values for *t*, we get different methods. But for this particular study, we choose the value for $t = 7$ which is of order eighth method. Now, on integrating term by term in equation (8) with respect to k, we obtain:

$$\int_0^1 (k-1)\,dk = -\frac{1}{2},\ \int_0^1 k(k-1)\,dk = -\frac{1}{6},\ \int_0^1 (k-1)k(k+1)\,dk = -\frac{1}{4},$$

$$\int_0^1 (k-1)k(k+1)(k+2)\,dk = -\frac{19}{30},\ \int_0^1 (k-1)k(k+1)(k+2)(k+3)\,dk = -\frac{9}{4},$$

$$\int_0^1 (k-1)k(k+1)(k+2)(k+3)(k+4)\,dk = -\frac{863}{84},$$

$$\int_0^1 (k-1)k(k+1)(k+2)(k+3)(k+4)(k+5)\,dk = -\frac{1375}{24},$$

$$\int_0^1 (k-1)k(k+1)(k+2)(k+3)(k+4)(k+5)(k+6)\,dk = -\frac{33953}{90}$$

Thus, we get:

$$
\begin{aligned}
y(x_{i+1}) &= y(x_i) + h\Bigg[f_{i+1} - \frac{1}{2}\nabla f_{i+1} - \frac{1}{12}\nabla^2 f_{i+1} - \frac{1}{24}\nabla^3 f_{i+1} - \frac{19}{720}\nabla^4 f_{i+1} - \frac{3}{160}\nabla^5 f_{i+1} \\
&\quad - \frac{863}{60480}\nabla^6 f_{i+1} - \frac{275}{24192}\nabla^7 f_{i+1}\Bigg] + T_7 \\
&= y_i + h\Bigg[f_{i+1} - \frac{1}{2}(f_{i+1} - f_i) - \frac{1}{12}(f_{i+1} - 2f_i + f_{i+1}) - \frac{1}{24}\begin{pmatrix} f_{i+1} - 3f_i + 3f_{i-1} \\ -f_{i-2} \end{pmatrix} \\
&\quad - \frac{19}{720}(f_{i+1} - 4f_i + 6f_{i-1} - 4f_{i-2} + f_{i-3}) - \frac{3}{160}\begin{pmatrix} f_{i+1} - 5f_i + 10f_{i-1} - 10f_{i-2} \\ +5f_{i-3} - f_{i-4} \end{pmatrix} \\
&\quad - \frac{863}{60480}(f_{i+1} - 6f_i + 15f_{i-1} - 20f_{i-2} + 15f_{i-3} - 6f_{i-4} + f_{i-5}) \\
&\quad - \frac{275}{24192}(f_{i+1} - 7f_i + 21f_{i-1} - 35f_{i-2} + 35f_{i-3} - 21f_{i-4} + 7f_{i-5} - f_{i-6})\Bigg] + T_7 \\
&= y_i + \frac{h}{120960}\big[36799 f_{i+1} + 139849 f_i - 121797 f_{i-1} + 123133 f_{i-2} - 88547 f_{i-3} \\
&\quad + 41499 f_{i-4} - 11351 f_{i-5} + 1375 f_{i-6}\big] + T_7 .
\end{aligned} \tag{9}
$$

Where, $T_7 = -\frac{33953}{3628800} h^9 f^{(8)}(\xi)$ is the local truncation error. Hence, equation (9) is called eighth order corrector method.

Remarks: The eighth order predictor corrector method uses $y_{k-7}, y_{k-6}, y_{k-5}, y_{k-4}, y_{k-3}, y_{k-2,} y_{k-1}$ and y_k in the calculation of y_{k+1}. This method is not self-starting; eight initial points, $(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6)$ and (x_7, y_7) must be given in advance in order to generate the points (x_k, y_k) for $k \geq 8$. For that reason, we applied an eighth order Runge-Kutta method to generate the first eight starting values.

3. ANALYSIS OF THE METHOD

3.1 Stability

Definition 1 Let $\lambda_1, \lambda_2, \ldots, \lambda_t$ are the (not necessarily distinct) roots of the characteristic equation given by:

$$P(\lambda) = \lambda^t - a_{k-1}\lambda^{t-1} - \ldots - a_1\lambda - a_0 = 0 \tag{10}$$

It is associated with the multistep difference method of equations (6) and (9) given as:

$$y_{i+1} = a_{k-1}y_i + a_{k-2}y_{i-1} + a_0 y_{i+1-k} + hF(x_i, y_{i+1}, y_i, \ldots, y_{i+1-1}) \tag{11}$$

$y_0 = \alpha,\ y_1 = \alpha_1, \ldots, y_{k-1} = \alpha_{k-1},$ for each $i = k-1, k, \ldots, N-1,$ where $a_0, a_1, \ldots, a_{k+1}$ are constants.

If $|\lambda_i| \le 1,$ for $i = 1, 2, \ldots, k$ and all roots with absolute value 1 are simple roots, then the difference method is said to satisfy the root condition.

Definition 2 (Stability)

i. Methods that satisfy the roots condition in which $|\lambda_i| = 1$ is the only root of the characteristic equation with magnitude one is called strongly stable.

ii. Methods that satisfy the root condition and have more than one distinct root with magnitude one is called weakly stable.

iii. Methods that do not satisfy the root condition are called unstable.

Theorem 1 The eighth-order predictor method in rquation (6) is strongly stable.

Proof: The eighth order predictor method in Eq. (6) can be expressed as:

$$y_{i+1} = y_i + hF\left(x_i, y_{i+1}, y_i, \ldots, y_{i-7}\right) \tag{12}$$

where

$$hF\left(x_i, y_{i+1}, y_i, \ldots, y_{i-7}\right) = \frac{h}{120960}\left[434241 f_i - 1152169 f_{i-1} + 2183877 f_{i-2} - 2664477 f_{i-3} + 2102243 f_{i-4} - 1041723 f_{i-5} + 295767 f_{i-6} - 36799 f_{i-7}\right].$$

In this case, we have: $k = 8,\ a_0 = a_1 = a_2 = a_3 = a_4 = a_5 = a_6 = 0$ and $a_7 = 1.$

The characteristic equation for the method becomes:

$$P(\lambda) = \lambda^8 - \lambda^7 = \lambda^7\left(\lambda - 1\right) = 0 \Rightarrow \lambda_1 = 1, \lambda_2 = \lambda_3 = \lambda_4 = \lambda_5 = \lambda_6 = \lambda_7 = \lambda_8 = 0$$

are the roots of the polynomial.

Therefore, it satisfies the root condition and is strongly stable by Definition 2 (i).

Theorem 2 The eight-order corrector method in Eq. (9) is also strongly stable.

Proof: The eight-order corrector method in Eq. (9) can be expressed as:

$$y_{i+1} = y_i + hF\left(x_i, y_{i+1}, y_i, \ldots, y_{i-6}\right)$$

where

$$hF\left(x_i, y_{i+1}, y_i, \ldots, y_{i-6}\right) = \frac{h}{120960}\left[36799 f_{i+1} + 139849 f_i - 121797 f_{i-1} + 123133 f_{i-2} - 88547 f_{i-3} + 41499 f_{i-4} - 11351 f_{i-5} + 1375 f_{i-6}\right]$$

In this case, we have: $k = 7,\ a_0 = a_1 = a_2 = a_3 = a_4 = a_5 = 0$ and $a_6 = 1.$

The characteristic equation for the method becomes:

$$P(\lambda)=\lambda^7-\lambda^6=\lambda^6(\lambda-1)=0$$
$$\Rightarrow \lambda_1=1, \lambda_2=\lambda_3=\lambda_4=\lambda_5=\lambda_6=\lambda_7=0$$

are the roots of the polynomial. Therefore, it satisfies the root condition and is strongly stable by Definition 2 (ii).

3.2. Consistency

Definition 3 (Consistency) The method is consistent, if the local truncation error $T_k(h)\to 0$ as $h\to 0$.

So, we have

$$T_8=\frac{1070017}{3628800}h^8 f^{(8)}(\xi) \text{ and } T_7=-\frac{33953}{3628800}h^7 f^{(7)}(\xi).$$

Thus $T_k(h)\to 0$ as $h\to 0$ for $k=7,8$.

Therefore, the methods in Eq. (6) and (9) are consistent by Definition 3.

Definition 4. Consistency and zero stability are the necessary and sufficient conditions for the convergence of any linear multistep methods.

Hence, according to Definition 4 our methods are convergent since they are both consistent and stable.

4. NUMERICAL EXAMPLES

In order to test the validity of the proposed method, two quadratic Riccati differential equations have been considered. Since all predictor corrector methods are not a self-starter, we take the eight order Runge-Kutta method for the first seven nodal points. For each N, the point wise absolute errors are approximated by the formula, $\|E\|=|y(x_i)-y_i|$, for $i=0,1,2,\ldots N$ and where, $y(x_i)$ and y_i are the exact and computed approximate solution of the given problem respectively, at the nodal point x_i.

Example 1.

Consider the following quadratic Riccati differential equation.

$$\frac{dy}{dx}=-\frac{1}{1+x}+y(x)-y^2(x), y(0)=1, 0\le x\le 1.$$

The exact solution is $y(x)=\dfrac{1}{1+x}$.

Table 1. Comparison of maximum absolute errors for Example 1.

x		$N=10$	$N=40$	$N=70$	$N=100$	$N=200$	$N=400$
Present Method							
0.1		7.2927e-09	1.1063e-10	2.0551e-11	5.1508e-12	3.4039e-13	2.1982e-14
0.2		1.1505e-08	1.5498e-10	1.8855e-11	4.7724e-12	3.1597e-13	2.0539e-14
0.3		1.3882e-08	1.3455e-10	1.7661e-11	4.4893e-12	2.9743e-13	1.9318e-14
0.4		1.5170e-08	1.2330e-10	1.6788e-11	4.2758e-12	2.8333e-13	1.8541e-14
0.5		1.5819e-08	1.1644e-10	1.6142e-11	4.1155e-12	2.7267e-13	1.8097e-14
0.6		1.6102e-08	1.1201e-10	1.5670e-11	3.9970e-12	2.6490e-13	1.6542e-14
0.7		1.6187e-08	1.0909e-10	1.5335e-11	3.9126e-12	2.5924e-13	1.5432e-14
0.8		7.5180e-08	1.0724e-10	1.5114e-11	3.8570e-12	2.5546e-13	1.4433e-14
0.9		1.1630e-07	1.0621e-10	1.4990e-11	3.8256e-12	2.5357e-13	1.3767e-14
1		1.3724e-07	1.0584e-10	1.4950e-11	3.8157e-12	2.5291e-13	1.3101e-14
Gemechis and Tesfaye (2016)							
0.1		3.8296e-07	1.2712e-09	1.3226e-10	3.1445e-11	1.9426e-12	1.2057e-13
0.2		5.7951e-07	1.9396e-09	2.0206e-10	4.8062e-11	2.9710e-12	1.8452e-13
0.3		6.8133e-07	2.2939e-09	2.3918e-10	5.6914e-11	3.5196e-12	2.1860e-13
0.4		7.3394e-07	2.4816e-09	2.5893e-10	6.1630e-11	3.8125e-12	1.8452e-13
0.5		7.6091e-07	2.5808e-09	2.6941e-10	6.4137e-11	3.9686e-12	2.4647e-13
0.6		7.7483e-07	2.6340e-09	2.7506e-10	6.5490e-11	4.0530e-12	2.5280e-13
0.7		7.8257e-07	2.6648e-09	2.7834e-10	6.6278e-11	4.1022e-12	2.5668e-13
0.8		7.8799e-07	2.6865e-09	2.8066e-10	6.6837e-11	4.1374e-12	2.5946e-13
0.9		7.9326e-07	2.7069e-09	2.8284e-10	6.7358e-11	4.1697e-12	2.6190e-13
1		7.9961e-07	2.7304e-09	2.8533e-10	6.7954e-11	4.2070e-12	2.6240e-13

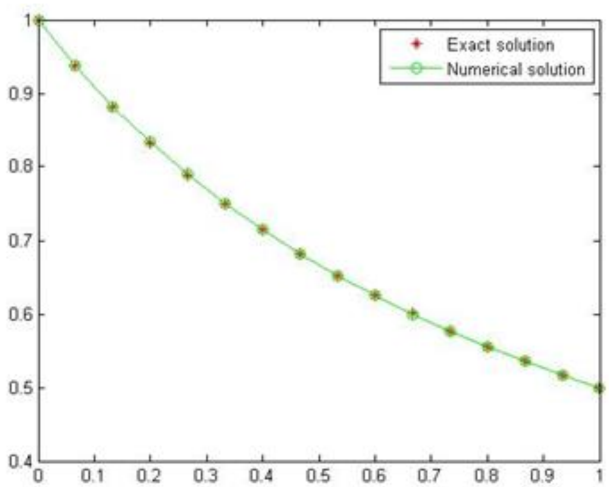

Figure 1. The graph of numerical and exact solution of example 1 for N=15.

Example 2.

Consider the following quadratic Riccati differential equation.

$\frac{dy}{dx}=\left(\frac{1}{2(1+x)}-\sqrt{x+1}\right)y(x)+y^2(x), y(0)=1, 0\le x\le 1.$ The exact solution is $y(x)=\sqrt{x+1}.$

Table 2. Comparison of maximum absolute errors for Example 2.

x	$N=10$	$N=40$	$N=70$	$N=100$	$N=200$	$N=400$
Present Method						
0.1	1.4651e-08	2.4095e-10	4.5286e-11	1.1260e-11	7.2786e-13	4.6407e-14
0.2	3.0951e-08	4.5548e-10	5.2663e-11	1.3088e-11	8.4577e-13	5.3957e-14
0.3	4.9486e-08	5.2983e-10	6.1297e-11	1.5234e-11	9.8410e-13	6.2839e-14
0.4	7.0879e-08	6.1749e-10	7.1448e-11	1.7756e-11	1.1471e-12	7.4163e-14
0.5	9.5829e-08	7.2093e-10	8.3418e-11	2.0731e-11	1.3392e-12	8.6819e-14
0.6	1.2515e-07	8.4327e-10	9.7576e-11	2.4250e-11	1.5667e-12	1.0081e-13
0.7	1.5979e-07	9.8835e-10	1.1436e-10	2.8422e-11	1.8361e-12	1.1680e-13
0.8	1.9052e-07	1.1608e-09	1.3432e-10	3.3383e-11	2.1563e-12	1.3656e-13
0.9	2.2509e-07	1.3664e-09	1.5811e-10	3.9295e-11	2.5380e-12	1.6076e-13
1	2.6702e-07	1.6120e-09	1.8653e-10	4.6357e-11	3.0183e-12	1.9007e-13

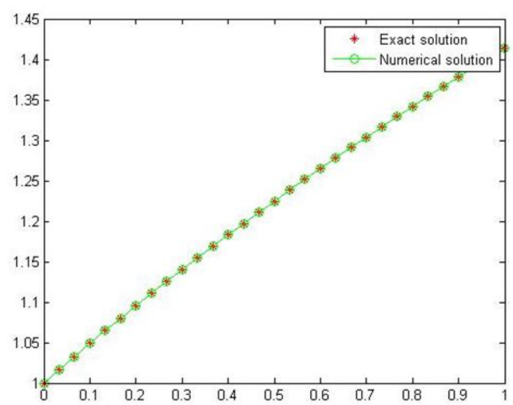

Figure 2. The graph of numerical and exact solution of example 2 for N=15.

5. DISCUSSION

In this paper, eight order predictor-corrector method is presented for solving quadratic Riccati differential equations. The stability and convergence of the method have been investigated. The study is implemented on two model examples with exact solutions by taking different values for N, and the computational results are presented in the Tables. The results obtained by the present method are compared with the results of Gemechis and Tesfaye (2016). Furthermore, from the Tables it is significant that all of the absolute errors decrease rapidly as h decreases which in turn shows the convergence of the computed solution. This shows that the small step size provides the better approximation. Briefly, the present method is stable, more accurate and effective method for solving quadratic Riccati differential equations.

6. ACKNOWLEDGEMENTS

The authors would like to thank the College of Natural Sciences, Jimma University for funding this research work and the anonymous reviewers for their constructive comments which helped us to improve the status of the paper.

7. CONFLICT OF INTERESTS

There are no conflicts of interest.

8. REFERENCE

Al Bastami, A., Belić, M. R & Petrović, N. Z. 2010. Special solutions of the Riccati equation with applications to the Gross-Pitaevskii nonlinear PDE. *Electronic Journal of Differential Equations*, **2010 (66)**: 1-10.

Batiha, B. 2015. A new efficient method for solving quadratic Riccati differential equation. *International Journal of Applied Mathematics Research*, **4(1)**: 24.

Biazar, J & Eslami, M. 2010. Differential transform method for quadratic Riccati differential equation. *International Journal of Nonlinear Science*, **9**(**4**): 444-447.

Butcher, J. C. 1996. A history of Runge-Kutta methods. *Applied numerical mathematics*, **20**(**3**): 247-260.

Butcher, J. C. 2016. *Numerical methods for ordinary differential equations*. John Wiley & Sons, ISBN: 978-1-119-12150-3, 544p.

File, G & Aga, T. 2016. Numerical solution of quadratic Riccati differential equations. *Egyptian Journal of Basic and Applied sciences*, **3**(**4**):392-397.

Ghomanjani, F & Khorram, E. 2017. Approximate solution for quadratic Riccati differential equation. *Journal of Taibah University for Science*, **11**(**2**): 246-250.

Ghomanjani, F & Shateyi, S. 2020. Solving a Quadratic Riccati Differential Equation, Multi-Pantograph Delay Differential Equations, and Optimal Control Systems with Pantograph Delays. *Axioms*, **9**(**3**): 82-94.

Ince, E. L. 1956. *Ordinary differential equations*. Dover Publications Inc., ISBN-10: 0486603490, 576p.

Khalid, M., Sultana, M., Zaidi, F & Arshad, U. 2015. An effective perturbation iteration algorithm for solving Riccati differential equations. *International Journal of Computer Applications*, **111**(**10**): 1-5.

Kiltu, G. G., Roba, G & Hailu, K. 2017. Fifth Order Predictor-Corrector Method for Solving Quadratic Riccati Differential Equations. *International Journal of Engineering and Applied Sciences*, **9**(**4**): 51-64.

Mathews, J. H & Fink, K. D. 2004. *Numerical methods using MATLAB* (Vol. 4). Pearson prentice hall, New Jersey. Pearson Education, Inc. Upper Saddle River, New Jersey 07458, ISBN 0-13-065248-2, 680p.

Mishra, V., Rani, D., Srivastava, T., Rani, S & Kakkar, S. 2016. Newton-Raphson based modified Laplace Adomian decomposition method for solving quadratic Riccati differential equations. *MATEC Web of Conferences,* EDP Sciences, 57: 05001, DOI: 10.1051/matecconf/20165705001.

Mukherjee, S & Roy, B. 2012. Solution of Riccati equation with variable co-efficient by Differential transform method. *International journal of nonlinear science*, **14**(**2**):251-256.

Opanuga, A. A., Edeki, S., Okagbue, H. I & Akinlabi, G. 2015. A novel approach for solving quadratic Riccati differential equations. *International Journal of Applied Engineering Research*, **10**(**11**): 29121-29126.

Reid, W. T (Ed.). 1972. *Riccati differential equations*. Volume 86, 1st Edition, Academic Press, eBook ISBN: 9780080955957, 215p.

Tan, Y & Abbasbandy, S. 2008. Homotopy analysis method for quadratic Riccati differential equation. *Communications in Nonlinear Science and Numerical Simulation*, **13**(**3**): 539-546.