IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 14, Issue 6 Ver. I (Nov - Dec 2018), PP 41-45
www.iosrjournals.org

Direct Integration of Two-Point Boundary Value Problem Using Symmetric Implicit Runge-Kutta Nyström Type Method

[1]Zamurat A. Adegboye, [2] Helen O.Edogbanya and [3] Ismail A.Kolawole
[1]*Department of Mathematical Sciences, Federal University Gusau, Zamfara State, Nigeria*
[2]*Department of Mathematical Sciences, Federal University Lokoja, Lokoja,Nigeria*
[3]*Department of Mathematics and Statistics, Kaduna Polytechnic, Kaduna,Nigeria*

Abstract: *Symmetric Methods are of much current interest due to their efficiency when solving stiff systems and the possibility to use them as basic methods for extrapolation processes. In this paper, we extend of the Symmetric Implicit Runge-Kutta Nyström Type Method (SIRKNTM) for the integration of first order ODEs to a SIRKNTM for Direct Integration of Two-Point Boundary Value Problem (BVPs).The theory of Nyström method was adopted in the formulation of the method. The method has an implicit structure for efficient implementation and produces simultaneously approximation of the solution of Two-Point Boundary Value Problem (BVPs). The proposed method was tested with Numerical experiment to illustrate its efficiency and the method can be extended to solve higher order differential equations.*

Date of Submission: 18-11-2018 Date of acceptance: 04-12-2018

I. Introduction

There are a several articles in literature addressing the numerical solution of the initial value problems (IVPs) of the form.

$$y'' = f(x, y) \qquad y(x_o) = y \qquad y'(x_o) = \beta \tag{1}$$

$$y'' = f(x, y, y') \qquad y(x_o) = y \qquad y'(x_o) = \beta \tag{2}$$

$$y''' = f(x, y) \qquad y(x_o) = y \qquad y'(x_o) = \beta \qquad y''(x_o) = \alpha \tag{3}$$

(see for example[1],[2],[3] and[4]) but not so much for the Two-Point Boundary Value Problem (BVPs) of the form

$$y''' = f(x, y, y', y'') \qquad y(x_o) = y \qquad y'(x_o) = \beta \qquad y'(\eta) = \alpha \qquad x_0 \le x \le \eta \tag{4}$$

(Different approaches appear in [5] and [6]).

Although it is possible to integrate a third order IVP by reducing it to first order system and apply one of the method available for such system it seem more natural to provide commercial method in order to integrate the problem directly. The advantage of these approaches lies in the fact that they are able to exploit special information about ODEs and this result in an increase in efficiency (that is, high accuracy at low cost). For instance, it is well known that Runge-Kutta Nyström method for (2) involve a real improvement as compared to standard Runge-Kutta method for a given number of stages [2].

In this work, we present a five stage Symmetric Implicit Super Runge-Kutta Nyström Method for Direct Integration of Two-Point Boundary Value Problem (BVPs) with the following advantage such as high order and stage order, low error constant and low implementation cost.

An s-stage Runge-kutta method for solution of first order ODEs is given by

$$y_{n+1} = y_n + h\sum_{i=1}^{s} a_{ij} k_i \tag{5}$$

for i = 1, 2 - - - - s,

$$k_i = f(x_i + \alpha_j h, y_n + h\sum_{j=1}^{s} a_{ij} k_j)$$

The real parameters α_j, k_i, a_{ij} define the method (5) and h is the step-size. In Butcher-array form as

$$\begin{array}{c|c} \alpha & \beta \\ \hline & W^T \end{array} \tag{6}$$

Where $\beta = a_{ij}$ matrix of the constant coefficients of method (see [7]).

An s-stage Runge-Kutta Nystrom method for direct integration of general second order ODEs is given by

$$y_{n+1} = y_n + \alpha_i h y'_n + h^2 \sum_{j=1}^{i-1} a_{ij} k_j \qquad (7)$$

$$y'_{n+1} = y'_n + h \sum_{j=1}^{i-1} \bar{a}_{ij} k_j$$

Where for i = 1, 2 - - - - s.

$$K_i = f(x_i + \alpha_j h, y_n + \alpha_i y'_n + h^2 \sum_{j=1}^{i-1} a_{ij} k_j, y'_n + h \sum_{j=1}^{i-1} \bar{a}_{ij} k_j)$$

The real parameters $\alpha_j, k_i, a_{ij}, \bar{a}_{ij}$ define the method(7). In Butcher-array form is

$$\begin{array}{c|c|c} \alpha & \bar{A} & A \\ \hline & \bar{b}^T & b \end{array} \qquad (8)$$

A = a_{ij} = β^2 Ā = $\bar{a}_{ij}$ =β

β = βe $\bar{b}$ = W b = $W^T\beta$ matrix of the constant coefficients of method (see [8])
An s-stage Runge-Kutta-Type method for direct integration of general third order ODEs is given by

$$y_{n+1} = y_n + \alpha_i h y'_n + \frac{(\alpha_i h)^2}{2} y''_n + h^3 \sum_{j=1}^{i-1} a_{ij} k_j \qquad (9)$$

$$y'_{n+1} = y'_n + \alpha_i h y''_n + h^2 \sum_{j=1}^{i-1} \bar{a}_{ij} k_j$$

$$y''_{n+1} = y''_n + h \sum_{j=1}^{i-1} \bar{\bar{a}}_{ij} k_j$$

Where for i = 1, 2 - - - - s.

$$K_i = f(x_i + \alpha_j h, y_n + \alpha_i y'_n + \frac{(\alpha_i h)^2}{2} y'' + h^3 \sum_{j=1}^{i-1} a_{ij} k_j y'_n + \alpha_i{}^2 h y''_n + + h^2 \sum_{j=1}^{i-1} \bar{a}_{ij} k_j, y''_n + h \sum_{j=1}^{i-1} \bar{\bar{a}}_{ij} k_j)$$

The real parameters $\alpha_j, k_i, a_{ij}, \bar{a}_{ij}, \bar{\bar{a}}_{ij}$ define the method (9). In Butcher – array form is

$$\begin{array}{c|c|c|c} \alpha & \bar{\bar{A}} & \bar{A} & A \\ \hline & \bar{\bar{b}}^T & \bar{b}^T & b \end{array} \qquad (10)$$

A= a_{ij} = β^3 Ā = $\bar{a}_{ij}$ = β^2 $\bar{\bar{A}}$ = $\bar{\bar{a}}_{ij}$ =β

β = βe , $\bar{b}$ = W $\bar{b}^T$ = $W^T\beta$ $\bar{\bar{b}}^T$ = $W^T \beta^2$ matrix of the constant coefficients of method (see [4]).

II. Construction of The Present Method Methods

We particularly wish to emphasize the combination of a multi-step structure with the use of off-step points, we seek a method that are multistage and multi-value because it will be convenient to extend the general linear method formulation to the high order Runge – Kutta case(see[2]) by considering an approximate solution to first order initial value problem in the form of power series

$$y(x) = \sum_{j=o}^{t+m-1} a_j x^j$$

$$a; \in R, j = o(1)t + m - 1, Y \in C^m(a,b) \subset P(x) \qquad (11)$$

$$y'(x) = \sum_{j=o}^{t+m-1} j a_j x^{j-1} \qquad (12)$$

Where a_j 's are the parameters to be determine and m are points of interpolation and collocation . To form our matrix D we collocate (12) at x_{n+j}, $j=0,\ldots 10$ and interpolate (11) x_{n+j}, $j=0$ Specifically k=10,t=1 and m=11 yields the following system of equations

$$\sum_{j=o}^{t+m-1} a_j x^j = y_{n+j} \qquad j=0 \tag{13}$$

$$\sum_{j=o}^{t+m-1} j a_j x^{j-1} = f_{n+j} \qquad j=0,1,\ldots,10 \tag{14}$$

Following the multistep collocation of Yusuph and Onumanyi (2002). We invert once the matrix D which is of dimension (t+m)*(t+m) the proposed continuous formulation takes the form

$$y_n(x) = \sum_{j=0}^{k} \alpha_j(x)\, y_{n+j} + h \sum_{j=0}^{k} \beta_j(x)\; f_{n+j} \tag{15}$$

When using Maple mathematical software to invert (13) and (14), obtaining values for a_j, $j=0,1,\ldots,k+2$ and we obtained the continuous formulation which when evaluated at $\boldsymbol{x = x_{n+j}}$ $\boldsymbol{j = 1, 2, \ldots, 10}$ gives the following discrete schemes.

$$y_1 := y_n + h\left(\frac{26842253}{95800320} f_n + \frac{164046413}{119750400} f_1 - \frac{296725183}{159667200} f_2 + \frac{12051709}{3991680} f_3 - \frac{33765029}{8870400} f_4 + \frac{2227571}{623700} f_5 - \frac{21677723}{8870400} f_6 + \frac{23643791}{19958400} f_7 - \frac{12318413}{31933440} f_8 + \frac{9071219}{119750400} f_9 - \frac{3250433}{479001600} f_{10}\right)$$

$$y_2 := y_n + h\left(\frac{2046263}{7484400} f_n + \frac{645431}{374220} f_1 - \frac{2149811}{2494800} f_2 + \frac{355583}{155925} f_3 - \frac{1258463}{415800} f_4 + \frac{904403}{311850} f_5 - \frac{166931}{83160} f_6 + \frac{21833}{22275} f_7 - \frac{800243}{2494800} f_8 + \frac{118291}{1871100} f_9 - \frac{8501}{1496880} f_{10}\right)$$

$$y_3 := y_n + h\left(\frac{108223}{394240} f_n + \frac{840607}{492800} f_1 - \frac{879183}{1971200} f_2 + \frac{762497}{246400} f_3 - \frac{670233}{197120} f_4 + \frac{24399}{7700} f_5 - \frac{2136259}{985600} f_6 + \frac{259071}{246400} f_7 - \frac{675441}{1971200} f_8 + \frac{6637}{98560} f_9 - \frac{11899}{1971200} f_{10}\right)$$

$$y_4 := y_n + h\left(\frac{128242}{467775} f_n + \frac{800276}{467775} f_1 - \frac{2159}{4455} f_2 + \frac{79504}{22275} f_3 - \frac{46846}{17325} f_4 + \frac{461576}{155925} f_5 - \frac{35758}{17325} f_6 + \frac{4496}{4455} f_7 - \frac{51508}{155925} f_8 + \frac{30452}{467775} f_9 - \frac{547}{93555} f_{10}\right)$$

$$y_5 := y_n + h\left(\frac{5256425}{19160064} f_n + \frac{8183125}{4790016} f_1 - \frac{2996375}{6386688} f_2 + \frac{2793625}{798336} f_3 - \frac{2306375}{1064448} f_4 + \frac{89035}{24948} f_5 - \frac{2325625}{1064448} f_6 + \frac{838375}{798336} f_7 - \frac{311375}{912384} f_8 + \frac{320875}{4790016} f_9 - \frac{114985}{19160064} f_{10}\right)$$

$$y_6 := y_n + h\left(\frac{1689}{6160} f_n + \frac{13169}{7700} f_1 - \frac{14787}{30800} f_2 + \frac{1363}{385} f_3 - \frac{35229}{15400} f_4 + \frac{16083}{3850} f_5 - \frac{25373}{15400} f_6 + \frac{1887}{1925} f_7 - \frac{2007}{6160} f_8 + \frac{71}{1100} f_9 - \frac{179}{30800} f_{10}\right)$$

$$y_7 := y_n + h\left(\frac{18775351}{68428800} f_n + \frac{5843887}{3421440} f_1 - \frac{10669897}{22809600} f_2 + \frac{9973607}{2851200} f_3 - \frac{2767667}{1267200} f_4 + \frac{353633}{89100} f_5 - \frac{241129}{253440} f_6 + \frac{4148249}{2851200} f_7 - \frac{8312311}{22809600} f_8 + \frac{1190357}{17107200} f_9 - \frac{84427}{13685760} f_{10}\right)$$

$$y_8 := y_n + h\left(\frac{25636}{93555} f_n + \frac{800896}{467775} f_1 - \frac{76352}{155925} f_2 + \frac{556544}{155925} f_3 - \frac{24368}{10395} f_4 + \frac{660736}{155925} f_5 - \frac{68864}{51975} f_6 + \frac{353792}{155925} f_7 + \frac{7996}{155925} f_8 + \frac{4736}{93555} f_9 - \frac{2368}{467775} f_{10}\right)$$

$$y_9 := y_n + \mathrm{h}\Big(\frac{542331}{1971200}f_n + \frac{837567}{492800}f_1 - \frac{167427}{394240}f_2 + \frac{829089}{246400}f_3 - \frac{1880253}{985600}f_4 + \frac{27459}{7700}f_5 - \frac{537219}{985600}f_6 + \frac{10773}{7040}f_7 + \frac{2065743}{1971200}f_8 + \frac{199809}{492800}f_9 - \frac{4671}{394240}f_{10}\Big)$$

$$y_{10} := y_n + \mathrm{h}\Big(\frac{80335}{299376}f_n + \frac{132875}{74844}f_1 - \frac{80875}{99792}f_2 + \frac{28375}{6237}f_3 - \frac{24125}{5544}f_4 + \frac{89035}{12474}f_5 - \frac{24125}{5544}f_6 + \frac{28375}{6237}f_7 - \frac{80875}{99792}f_8 + \frac{132875}{74844}f_9 + \frac{80335}{299376}f_{10}\Big) \qquad (15)$$

III. Implementation Strategies/ Numerical Experiment

To complete the required initial conditions for implementation, we implement the method first at h=h_∞ (h_∞ is the largest value of h for which the method is stable),this gives the missing initial condition with the right end boundary condition this allow direct application of the numerical method to the two-point boundary value problem via initial value problem.

To show the efficiency of method we present numerical solution of the Falkner Skan Equation (a two-point boundary value problem

$$y'''(x) + y(x)y''(x) + \beta\{1-[y'(x)]^2\} = 0$$

$$y(0) = y'(0) = 0, \qquad y'(\infty) = 1 \qquad 0 \le x \le \infty$$

The results obtained by Summiya Parveen (2014) and SIRKNTM (15) are given in table 1 below. Where for $\eta = x$, $y(\eta), y'(\eta)$ and $y''(\eta)$ are the values of $y(\eta), y'(\eta)$ and $y''(\eta)$ obtained by Summiya Parveen (2014) and $Y(\eta), Y'(\eta)$ and $Y''(\eta)$ are the values of $y(\eta), y'(\eta)$ and $y''(\eta)$ obtained by using SIRKNTM (14).

Table 1: Numerical values of $y(\eta), y'(\eta)$ and $y''(\eta)$ **for** $\boldsymbol{\beta}$ = **0** and

η	$y(\eta)$	$Y(\eta)$	$y'(\eta)$	$Y'(\eta)$	$y''(\eta)$	$Y''(\eta)$
0.0	0	0	0	0	0.4696	0.4690
0.2	0.009391	0.009379	0.093905	0.09379	0.469306	0.468711
0.4	0.037549	0.037502	0.187605	0.187367	0.467254	0.466664
0.6	0.084386	0.084279	0.280575	0.280221	0.461734	0.461158
0.8	0.149674	0.149485	0.371963	0.371496	0.451190	0.450640
1.0	0.232990	0.232901	0.460633	0.460657	0.434379	0.434461
1.2	0.333657	0.333239	0.545246	0.544577	0.410565	0.410114
1.4	0.450724	0.450163	0.624386	0.623632	0.379692	0.379311
1.6	0.582956	0.582238	0.696700	0.695878	0.342487	0.342187
1.8	0.728872	0.727932	0.761057	0.760134	0.300445	0.300220
2.0	0.886797	0.886783	0.816695	0.816820	0.255669	0.255742
2.2	1.054947	1.053604	0.863304	0.862370	0.210580	0.2105553
2.4	1.231528	1.230769	0.901065	0.900685	0.167560	0.167613
2.6	1.414824	1.413202	0.930601	0.929672	0.128613	0.128650
2.8	1.603284	1.601475	0.952875	0.951955	0.095113	0.095175
3.0	1.795568	1.795708	0.969055	0.969226	0.067710	0.067721
3.2	1.990581	1.988382	0.980365	0.979461	0.046370	0.046445
3.4	2.187467	2.185048	0.987970	0.987067	0.030535	0.030605
3.6	2.385590	2.382922	0.992888	0.991978	0.019329	0.019384
3.8	2.584499	2.581560	0.995944	0.995025	0.011759	0.011784
4.0	2.783886	2.784203	0.997770	0.997944	0.006874	0.006854
4.2	2.983555	2.980207	0.998818	0.997926	0.003861	0.003743
4.4	3.183383	3.180128	0.999397	0.998588	0.002084	0.001851
4.6	3.383296	3.380582	0.999703	0.999066	0.001081	0.000724
4.8	3.583254	3.581748	0.999859	0.999509	0.000538	0.000086
5.0	3.783235	3.7837826	0.999936	1.000000	0.000258	0.000210

Table 2: Numerical values of $y(\eta), y'(\eta)$ and $y''(\eta)$ using SIRKNTM (15) **for** $\boldsymbol{\beta}$ = 2

η	$y(\eta)$	$y'(\eta)$	$y''(\eta)$	η	$y(\eta)$	$y'(\eta)$	$y''(\eta)$
0.0	0	0	1.688181	2.6	2.103000	0.998780	0.004201
0.2	0.031117	0.298137	1.297874	2.8	2.302824	0.999394	0.002152
0.4	0.114309	0.522298	0.953773	3.0	2.502737	0.999703	0.001062

0.6	0.235872	0.684048	0.675100	3.2	2.702694	0.999853	0.000503
0.8	0.384662	0.796773	0.462661	3.4	2.902673	0.999923	0.000224
1.0	0.551707	0.871731	0.306113	3.6	3.102661	0.999952	0.000087
1.2	0.731385	0.921429	0.197526	3.8	3.302652	0.999962	0.000022
1.4	0.919081	0.953083	0.123980	4.0	3.502643	0.999964	-0.000013
1.6	1.111819	0.972702	0.075715	4.2	3.702636	0.999964	-0.000014
1.8	1.307645	0.984533	0.044985	4.4	3.902628	0.999951	-0.000028
2.0	1.505302	0.991456	0.025961	4.6	4.102619	0.999953	-0.000037
2.2	1.704026	0.995411	0.014576	4.8	4.302609	0.999945	-0.000044
2.4	1.903341	0.997602	0.007945	5.0	4.502537	1.000000	-0.000129

IV. Discussion

The results in table 1are in excellent agreement with those mentioned in the literature by using shooting method and results in table 2 present the numerical solution of Falkner-Skan equation for $\boldsymbol{\beta} = 2$.

V. Conclusion

Through the approach presented in this paper, the SIRKNTM can be extended to solve higher order differential equations. The method requires less work with little cost (when compared with shooting technique method), possesses a gain in efficiency with no overlapping of results and larger h_∞ for $\boldsymbol{\beta} = 2$.

References

[1]. Agam S.A and Badmus A.M (2010), Improved Runge-Kutta Nyström Method for Second Order Ordinary Differential Equations J. of Research in Physical Science, 6, No 1, 71-76.
[2]. Butcher J.C and Hojjat G. (2005) Second Derivative Methods with Runge-Kutta Stability, Num alg, 40, 415-429.
[3]. Yusuph Y. and Onumanyi P (2002) New Multiple Finite Difference Method through Multi –Step Collocation for y”= f (x,y), Abacus, 29; 92-96.
[4]. Z .A. Adegboye and R.Kehinde (2016) Super Runge-Kutta Nyström method for direct Integration of third order ordinary differential equations. The Journal of Mathematical Scences of Nigeria,NMC Abuja Nigeria,Vol 4,847-856.
[5]. **Zamurat A. Adegboye**, Okeowo G.F and Isiyaku Aliyu(2014) Explicit Runge-Kutta Type Method For Direct Solution Of Linear Third Order Boundary Value Problems. The Nigerian Association of Mathematical Physics, Vol.28(2),121-128.
[6]. Summiya Parveen(2014) Numerical solution of the Falkner Skan Equation by using shooting techniques. IOSR Journal of Mathematics Volume 10, Issue 6, 78-83.
[7]. Butcher J.C and Wright W.M (2003) The Construction of Practical General Linear Methods, BIT, 43, 695-721.
[8]. Hairer E. and Warner G. (1976). A Theory for Nystrom Method, Numerical Math, 25, 383-400.

Zamurat A. Adegboye ." Direct Integration of Two-Point Boundary Value Problem Using Symmetric Implicit Runge-Kutta NystrÖm Type Method." IOSR Journal of Mathematics (IOSR-JM) 14.6 (2018): 41-45.