Using Neural Networks for the derivation of Runge–Kutta–Nyström pairs.

Ch. Tsitouras

TEI of Chalkis, Dept. of Applied Sciences, GR34400, Psahna, Greece.

Abstract. A Runge–Kutta–Nyström (RKN) pair of orders 4(3) is presented in this paper. A test orbit from the Kepler problem is chosen to be integrated for a specific tolerance. Then the two free parameters of the above RKN4(3) family are trained to perform best. Thus a neural network approach is formed and its objective function is minimized using a differential evolution optimization technique. Finally we observe that the produced pair outperforms standard pairs from the literature for the Kepler orbits over a wide range of eccentricities and tolerances.

Keywords: Neural Networks, Runge–Kutta, Kepler problem, Differential Evolution
PACS: 02.60D.Lj; 07.05.Mh; 02.60.Pn

INTRODUCTION

Explicit Runge–Kutta–Nyström pairs are widely used for the numerical solution of the initial value problem

$$y'' = f(x,y), \quad y(x_0) = y_0 \in \Re^m, \quad y'(x_0) = y_0' \in \Re^m, \quad x \in [x_0, x_e]$$

where $f : \Re \times \Re^m \mapsto \Re^m$. These pairs are characterized by the extended Butcher tableau [1]:

$$\begin{array}{c|c} c & A \\ \hline & b,\, b' \\ & \hat{b},\, \hat{b}' \end{array}$$

with $b^T, \hat{b}^T, b'^T, \hat{b}'^T, c \in \Re^s$ and $A \in \Re^{s\times s}$ is strictly lower triangular. The procedure that advances the solution from (x_n, y_n, y_n') to $x_{n+1} = x_n + h_n$ computes at each step the approximations $y_{n+1}, \hat{y}_{n+1}$ to $y(x_{n+1})$ of orders p and $p-1$ respectively, given by

$$y_{n+1} = y_n + hy_n' + h_n^2 \sum_{i=1}^{s} b_i f_{ni}$$

and

$$\hat{y}_{n+1} = y_n + hy_n' + h_n^2 \sum_{i=1}^{s} \hat{b}_i f_{ni}.$$

It also produces another two approximations $y_{n+1}', \hat{y}_{n+1}'$ to $y'(x_{n+1})$ of orders p and $p-1$, given by

$$y_{n+1}' = y_n' + h \sum_{i=1}^{s} b_i' f_{ni}$$

and

$$\hat{y}_{n+1}' = y_n' + h \sum_{i=1}^{s} \hat{b}_i' f_{ni}$$

with $f_{ni} = f(x_n + c_i h_n, y_n + h_n \sum_{j=1}^{i-1} a_{ij} f_{nj}) \in \Re^m$ for $i = 1, 2, .., s \geq p$. From this embedded form (called RKN$p(p-1)$) we can obtain an estimate $u_{n+1} = \max(\|y_{n+1} - \hat{y}_{n+1}\|_\infty, \|y_{n+1}' - \hat{y}_{n+1}'\|_\infty)$ of the local truncation error of the $p-1$ order formula. So the step-size control algorithm

$$h_{n+1} = 0.9 h_n \cdot \left(\frac{\mathrm{TOL}}{u_{n+1}}\right)^{1/p},$$

is in common use, with TOL being the requested tolerance. The above formula is used even if TOL is exceeded by u_{n+1}, but then h_{n+1} is simply the recomputed current step. See [6] for more details on the implementation of these type of step size policies.

DERIVATION OF RK PAIRS OF ORDERS 4(3)

The derivation of better RKN pairs is of continued interest the last $30-40$ years, see [5] and references therein. The main framework for the construction of RKN pairs is matching Taylor series expansions of $y(x+h)-y_{n+1}$ and $y'(x+h)-y'_{n+1}$ after we have expanded various f_{ni}'s.

It is common knowledge that a pair of orders four and three that we are interested has to satisfy the following equations of condition:

$$b'e=1,\, b'c=\frac{1}{2},\, b'c^2=\frac{1}{3},\, b'c^3=\frac{1}{4} \text{ and } b'Ac=\frac{1}{24}$$

since we set $Ae=c^2/2$ and $b=b'(e-c)$ with $e=[11\cdots 1]^T\in\Re^s$.

Here we consider the family of Dormand et. al. [2] that needs four stages per step ($s=4$). This family uses FSAL device so it effectively needs only three stages per step. FSAL demands $c_4=1$ and $a_{4i}=b_i$, $i=1,2,3$. Thus the parameters available for fulfilling the above mentioned five equations of condition are: c_2, c_3, b'_1, b'_2, b'_3, b'_4 and a_{32}. Two of them are set free, namely c_2, and c_3. All the other coefficients are defined by the simplifying assumptions.

Similarly we produce the coefficients for the lower order formulas. Then

$$\hat{b}'e=1,\ \hat{b}'c=\frac{1}{2},\ \hat{b}'c^2=\frac{1}{3}$$

are to be solved for $\hat{b}'_1$, $\hat{b}'_2$, $\hat{b}'_3$ and $\hat{b}'_4$. So we set $\hat{b}'_4=-1/3$ and compute the rest coefficients. Finally,

$$\hat{b}.e=1/2 \text{ and } \hat{b}.c=1/6$$

are to be solved for $\hat{b}_i$, $i=1,2,3,4$. We set $\hat{b}_3=0.15$ and $\hat{b}_4=-1/20$ and evaluate $\hat{b}_1$ and $\hat{b}_2$. The fixed coefficients for the lower order formulas affect mainly the step size. For example smaller values may produce smaller estimations for the error and in consequence this is similar of using more lax tolerances. So for reasons of comparison we use the ones chosen in [4].

After solving all the equations we conclude to the following expressions with respect to c_2 and c_3:
$a_{21}=\frac{c_2^2}{2}$, $a_{31}=\frac{(c_3(c_2^2(1-12c_3)+6c_2^3c_3-c_3^2+3c_2c_3(1+c_3)))}{(6c_2(1-3c_2+2c_2^2))}$, $a_{32}=\frac{((c_2-c_3)c_3(-c_3+c_2(-1+3c_3)))}{(6c_2(1-3c_2+2c_2^2))}$, $a_{41}=\frac{(1-2c_2-2c_3+6c_2c_3)}{(12c_2c_3)}$,
$a_{42}=\frac{(1-2c_3)}{(12c_2^2-12c_2c_3)}$, $a_{43}=\frac{(-1+2c_2)}{(12(c_2-c_3)c_3)}$, $b'_1=\frac{(1-2c_2-2c_3+6c_2c_3)}{(12c_2c_3)}$, $b'_2=\frac{(-1+2c_3)}{(12c_2(c_2^2+c_3-c_2(1+c_3)))}$, $b'_3=\frac{(1-2c_2)}{(12(c_2-c_3)(-1+c_3)c_3)}$,
$b'_4=\frac{(3-4c_3+c_2(-4+6c_3))}{(12(-1+c_2)(-1+c_3))}$, $\hat{b}_1=\frac{-13+24c_2+9c_3}{60c_2}$, $\hat{b}_2=\frac{13-9c_3}{60c_2}$, $\hat{b}'_1=\frac{4-5c_2-5c_3+8c_2c_3}{6c_2c_3}$, $\hat{b}'_2=\frac{4-5c_3}{6c_2^2-6c_2c_3}$, $\hat{b}'_3=-\frac{4-5c_2}{6c_2c_3-6c_3^2}$.

The main question raising now is how to select c_2 and c_3? Traditionally the norm of the fifth order truncation error is minimized. This technique does not consider the nature of the problems. Thus many authors considered many other approaches utilizing various properties of the problems. For example periodic problems have been studied extensively and very promising methods have been produced for them.

For a p-order RKN method, the minimization of the $p+1$ order term in the truncation error expansion seems the best choice in a case of a general problem. Although a lot of speculation is raised for problems where it is believed that their properties can be handled. Such problems are Hamiltonians, orbits, periodic, Schrödinger and many others.

Unfortunately in most cases analytical consideration of test problems produces complicated algebra and enforces us to proceed with oversimplifications. In other cases we deal with some side properties such as symplectiness.

Our purpose here is to produce a RKN4(3) pair that is best for the two body problem. It is very difficult to derive simple algebraic formulas for the coefficients that may produce better pairs for this problem.

An interesting alternative could be the consideration of Runge-Kutta type neural networks, where the various new families pairs are tested on some model problems to give good predictions for their coefficients.

TABLE 1. Coefficients of NEW4(3)

0				
$\frac{63}{83}$	$\frac{3969}{13448}$			
$\frac{62}{119}$	$\frac{811706573}{7006891122}$	$\frac{139302871}{7006891122}$		
1	$\frac{1004}{5859}$	$-\frac{8405}{456057}$	$\frac{155771}{448818}$	
b	$\frac{1004}{5859}$	$-\frac{8405}{456057}$	$\frac{155771}{448818}$	
b'	$\frac{1004}{5859}$	$-\frac{689210}{8665083}$	$\frac{18536749}{25582626}$	$\frac{1193}{6498}$
$\widehat{b}$	$\frac{9883}{44982}$	$\frac{40549}{224910}$	$\frac{3}{20}$	$-\frac{1}{20}$
$\widehat{b}'$	$\frac{7375}{23436}$	$\frac{558092}{456057}$	$-\frac{184093}{897636}$	$-\frac{1}{3}$

THE NEW RUNGE–KUTTA–NYSTRÖM PAIR

We consider the well known Kepler problem

$$y_1'' = -\frac{y_1}{\sqrt{y_1^2+y_2^2}^3}$$
$$y_2'' = -\frac{y_2}{\sqrt{y_1^2+y_2^2}^3}$$

$x \geq 0$, $y(0) = [1-\varepsilon, 0]^T$, $y'(0) = [0, \sqrt{\frac{1+\varepsilon}{1-\varepsilon}}]^T$ with ε the eccentricity of the orbit.

We construct a Neural Network (NN) similar to the one given in [4] for Runge–Kutta methods. In the input we give the eccentricity ε, the tolerance TOL, the endpoint x_e and the two parameters c_2 and c_3. Then the corresponding problem is integrated and we record the endpoint global error ge and the number of the function evaluations N. The output is a measure of the efficiency:

$$\text{eff} = N \cdot (\frac{ge}{\text{TOL}})^{0.25}.$$

We tested DEP4(3) pair for $c_2 = 0.25, c_3 = 0.7$, $\varepsilon = 0.5$, TOL$= 10^{-4}$, $x_e = 20\pi$ and found $\text{eff}_{\text{DEP}} = 5140.26$. Then we trained the coefficients in this NN and we got $\text{eff}_{\text{NEW}} = 1348.60$ for $c_2 = 63/82$ and $c_3 = 62/119$. The neural networks are actually nonlinear optimizers and a differential evolution (DE) technique was used for this purpose here. DE is a population based method which seems to perform better here where the output comes after a hole run of the Initial Value Problem. `DeMat` software for `Matlab` was used as DE method, see [3].

Finally we tested the new pair for a wide range of tolerances and eccentricities. Actually for TOL=10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6} and $\varepsilon = 0.05, 0.10, , 0.15, \cdots, 0.95$ we recorded the values $\text{eff}_{DEP}^{\text{TOL},\varepsilon}$ for DEP4(3) pair and $\text{eff}_{NEW}^{\text{TOL},\varepsilon}$ for the new pair. The average of the corresponding quotients $\text{eff}_{DEP}^{\text{TOL},\varepsilon}/\text{eff}_{NEW}^{\text{TOL},\varepsilon}$ is 1.445 which means that the new pair is about 45% more efficient in the family of Kepler problems.

REFERENCES

1. J. C. Butcher, The Numerical Analysis of Ordinary Differential Equations, John Wiley & Sons Inc., 1987, New York.
2. J. R. Dormand, M. E. El-Mikkawy and P. J. Prince, Families of Runge-Kutta-Nyström formulae, IMA J. Numer. Anal., 7 (1987) 235-250.
3. K. V. Price, R. M. Storn and J. A. Lampinen, Differential Evolution: A practical approach to global optimization, Springer, Berlin, 2005.
4. Ch. Tsitouras, Neural Networks with multidimensional transfer functions, IEEE T. Neural Networks, 20 (2002) 222-228.
5. Ch. Tsitouras and I. Th. Famelis, Symbolic derivation of Runge-Kutta-Nyström order conditions. J. Comput. Appl. Math., to appear.
6. Ch. Tsitouras and S. N. Papakostas, Cheap Error Estimation for Runge-Kutta pairs, SIAM J. Sci. Comput., 20 (1999) 2067-2088.