Numerically Optimal Runge–Kutta Pairs with Interpolants

J.H. Verner *

January 12, 2010

Abstract. Explicit Runge–Kutta pairs are known to provide efficient solutions to initial value differential equations with inexpensive derivative evaluations. Two criteria for selection are proposed with a view to deriving pairs which minimize computation while achieving a user-specified accuracy. Coefficients of improved pairs, their stability regions and coefficents of appended optimal interpolatory Runge–Kutta formulas are included on the author's website (www.math.sfu.ca/∼jverner). This note reports results of tests on these pairs to illustrate their effectiveness in solving nonstiff initial value problems. These pairs and interpolants may be used for implementation or to provide comparison targets for other new types of methods.

Key words. explicit Runge–Kutta pairs, order conditions, local error estimation, continuous Runge–Kutta methods

AMS subject classifications. 65L05, 65L06, 65L20.

1 Introduction

Explicit Runge–Kutta pairs are known to be efficient algorithms for obtaining approximate solutions to initial value problems for nonstiff and mildly-

*Department of Mathematics and Statistics, PIMS, Simon Fraser University, Burnaby, Canada, V5A 1S6 (jverner@pims.math.ca) Phone: 01-778-782-6554 Fax: 01-778-782-6657.

stiff ordinary differential equations with relatively inexpensive derivative functions evaluations. Here, we shall write an initial value problem as

$$\begin{cases} y'(x) = f(x, y(x)), \quad x \in [x_0, x_{end}], \\ y(x_0) = y_0, \end{cases} \tag{1.1}$$

where the solution is vector valued: $y : R \to R^n$, and its derivative function $f : R^{n+1} \to R^n$ is assumed to be sufficiently smooth.

The author designed a strategy for constructing explicit Runge–Kutta pairs, and families of all orders up to order p, $p \leq 9$ [16, 17], which circumvented a problem with earlier pairs of orders greater than 5 derived by Fehlberg [7]. Moreover, a classification system proposed in [22], and a subset in [11] identify different structures from which to choose coefficients that satisfy the well-known order conditions [1]. Indeed, higher-order pairs conjectured by this classification were subsequently constructed in [15, 23, 24].

Since the initial design appeared, a variety of authors have contributed a spectrum of methods primarily focussed on reducing the 2-norm of the vector of local truncation error (LTE) coefficients to obtain efficient algorithms. Studies reported here indicates that this general strategy for selecting pairs yields effective procedures for solving nonstiff problems of type (1.1).

In particular, we focus on two types of pairs designed to be effective: i.e. for a given user-specified tolerance, the amount of computation required should be small on average for all nonstiff problems. Following the traditional criteria while restricting the size of the maximum coefficient used [11], we designate a pair to be *efficient* if the 2-norm of the vector of LTE coefficients is small, and the largest coefficient is not greater than about 200. Further, to restrict the possibility that large coefficients may lead to increased arithmetic truncation errors, we specify a pair to be *robust* if it is *efficient*, and all the weights of the propagating (higher order) formula of a pair are non-negative.

Using these two criteria, searches through known and recently derived pairs have lead to a selection of pairs which are recommended either for implementation in production software, or as models against which other types of new formula such as two-step Runge–Kutta or almost Runge-Kutta pairs might be compared. In tests to determine the relative effectiveness of these pairs, the higher-order approximation is propagated (extrapolation) so that the 2-norm of the error-per-step (XEPS) is bounded by a user-specified tolerance. In this mode, the global error is known to be approximately proportional to the requested tolerance [13].

For several reasons, coefficients of the selected pairs are provided on the author's website. First, this allows for an accurate and convenient transcription of the coefficients for use in codes implemented by interested users. Second, it allows for the inclusion of other information including several norms of the vectors of local truncation error coefficients, graphs of the stability regions of each pair, and coefficients for differentiable interpolatory Runge–Kutta methods of the same orders with arbitrary parameters selected to give (nearly) optimal interpolants. (When possible, each order p interpolant is selected so that the 2-norm of its LTE increases monotonically over the step, or is bounded by the endpoint 2-norm of the LTE of the discrete method.)

2 Selected Runge–Kutta Pairs

A single step of a Runge–Kutta pair using s stages $F_j = f(x_n + c_j h, Y_j)$ is specified by

$$\begin{aligned} Y_i &= y_n + h \sum_{j<i} a_{ij} F_j \\ y_{n+1} &= y_n + h \sum_{i=1}^{s} b_i F_i \\ \widehat{y}_{n+1} &= y_n + h \sum_{i=1}^{s*} \widehat{b}_i F_i \end{aligned} \tag{2.1}$$

(s*=s+1 for FSAL pairs), its coefficients are displayed in a Butcher tableau:

$$\begin{array}{c|c} c & A \\ \hline & b^T \\ & \widehat{b}^T \end{array} = \begin{array}{c|cccc} c_1 & & & & \\ c_2 & a_{21} & & & \\ \cdot & & \ddots & & \\ c_s & a_{s1} & \cdots & a_{s,s-1} & \\ \hline 1 & b_1 & \cdots & & b_s \\ 1 & \widehat{b}_1 & \cdots & & \widehat{b}_s \end{array}$$

Table 1: Butcher tableau of a Runge–Kutta pair

This allows a brief description of the classification scheme in [22]. For each problem in which f of (1.1) depends on y, the accuracy of the endpoint solution depends upon *stage-order* (or "sub-quadrature") conditions

$$\sum_{k=1}^{j-1} a_{j,k} c_k^{l-1} = \frac{{c_j}^l}{l}, \quad l = 1, .., q_j, \tag{2.2}$$

where q_j defines the stage-order of stage j. In [23], we define the *dominant stage order* to be

$$\text{DSO} = \min\{q_j, |b_j \neq 0, \quad j = 1, .., s\} \tag{2.3}$$

Stages	DSO=p-4	DSO=p-3
8-stage propagation	IIa1.1 [16] IIa1.2 [12] IIa1.3 [11] IIa2 [23]	IIb [19, 15]
FSAL propagation	IIIXa1 [2] IIIXa2 [23]	IIIXb [5, 19, 15]

Table 2: Partition of RK6(5) Pairs by structure

Experience has shown that pairs with DSO=p-4 or DSO=p-3 (with subsequent names designated as Type a or Type b respectively) are most efficient for computation. (Other pairs such as Hairer's formula [8] with DSO=p-5, do not appear to be effective for use as general purpose algorithms.)

A second partitioning of pairs is determined by their structure: we specify a pair of Type II to be one for which s basic approximate derivatives are used to compute two approximations of successive orders. In a pair of Type III,

0									
$\frac{9}{50}$	$\frac{9}{50}$								
$\frac{1}{6}$	$\frac{29}{324}$	$\frac{25}{324}$							
$\frac{1}{4}$	$\frac{1}{16}$	0	$\frac{3}{16}$						
$\frac{53}{100}$	$\frac{79129}{250000}$	0	$-\frac{261237}{250000}$	$\frac{19663}{15625}$					
$\frac{3}{5}$	$\frac{1336883}{4909125}$	0	$-\frac{25476}{30875}$	$\frac{194159}{185250}$	$\frac{8225}{78546}$				
$\frac{4}{5}$	$-\frac{2459386}{14727375}$	0	$\frac{19504}{30875}$	$\frac{2377474}{13615875}$	$-\frac{6157250}{5773131}$	$\frac{902}{735}$			
1	$\frac{2699}{7410}$	0	$-\frac{252}{1235}$	$-\frac{1393253}{3993990}$	$\frac{236875}{72618}$	$-\frac{135}{49}$	$\frac{15}{22}$		
1	$\frac{11}{144}$	0	0	$\frac{256}{693}$	0	$\frac{125}{504}$	$\frac{125}{528}$	$\frac{5}{72}$	
1	$\frac{28}{477}$	0	0	$\frac{212}{441}$	$-\frac{312500}{366177}$	$\frac{2125}{1764}$	0	$-\frac{2105}{35532}$	$\frac{2995}{17766}$

Table 3: IIIXb+6(5), a robust nine-stage FSAL pair of orders 6 and 5

the derivative evaluation for the first approximation is used in computing the second approximation; if the first approximation is that of higher order, Dormand and Prince [3] use the term FSAL to indicate that the first derivative of the new step is the same as the last derivative of the previous step, and we designate this as Type IIIX. Other types are possible, but do not seem to yield effective algorithms. Table 2 partitions some 6(5) pairs of known types.

Using MAPLE codes designed for deriving different types of pairs, we searched for optimal *efficient* and *robust* pairs. A review of known pairs provided starting values for arbitrary parameters, and searches to meet the criteria of each type of pairs motivated modification of these parameters to find nearly or *numerically* optimal pairs. In [11], the authors discuss differences of some 6(5) pairs partitioned by type in Table 2. The three variations indicated for pairs of type IIa1 arise because of restrictions in choices for node c_{s-1} and weight b_s. The numerically optimal robust pair IIIXb+6(5) displayed in Table 3 has $b_5 = 0$, a possible additional advantage for special types of problems such as delay differential equations.

Pair	p	s	$A_{p+1,2}$	$\widehat{B}_{p+1,2}$	$\widehat{C}_{p+1,2}$	$\widehat{A}_{p,2}$	D_∞	SI_p	SI_{p-1}
IIIXb+6(5)	6	9*	1.01(-4)	1.80	1.83	5.49(-3)	3.26	-4.32	-4.05
IIIXb-6(5)	6	9*	1.44(-6)	1.72	1.72	2.25(-3)	207.9	-4.86	-4.39
IIb-6(5)[16]	6	8	5.17(-5)	1.31	1.32	1.48(-3)	26.3	-4.36	-3.62
IIa1+7(6)	7	10	2.70(-5)	1.57	1.57	5.23(-4)	80.50	-4.64	-4.00
IIa1-7(6)	7	10	1.68(-5)	1.73	1.74	3.71(-4)	187.3	-4.64	-4.00
IIa1+8(7)	8	13	7.54(-6)	3.06	3.01	2.11(-5)	5.92	-4.81	-5.00
IIa1-8(7)	8	13	2.82(-7)	1.97	1.98	8.35(-6)	123.4	-5.86	-5.70
DVERK-8(7)	8	13	8.35(-7)	2.02	2.03	9.32(-6)	16.95	-5.78	-5.53
IIa1+9(8)	9	16	3.51(-7)	3.42	3.41	4.15(-5)	23.80	-4.52	-3.90
IIa1-9(8)	9	16	3.49(-7)	3.54	3.41	3.56(-6)	3.26	-4.47	-4.75

Table 4: Characteristic Properties of selected RK Pairs

Characteristic properties of explicit Runge–Kutta pairs have been used as a strategy to identify their effectiveness. For an s-stage pair, $A_{p+1,2}$ and $\widehat{A}_{p,2}$ are 2-norms of the LTE coefficients, $\widehat{B}_{p+1,2}$ and $\widehat{C}_{p+1,2}$ assess the quality of the error estimate [4], D_∞ is the magnitude of the largest coefficient, and SI_q give the left ends of the real stability intervals. For some pairs of moderate orders, tables of such properties appear in [12, 4, 19, 23]. Table 4 provides corresponding properties for robust and efficient pairs studied here. These include pairs on the author's webpage, and as well, that of Table 5b in [15], and the pair used in *dsolve[dverk78]* of the MAPLE subroutine library.

3 Numerical Experiments

To test the order of the derived methods and the quality of local error estimation, we have applied the RK pairs listed in Section 2 to problems of the DETEST set [10]. Each pair was implemented over a range of tolerances with "error-per-step" and the solution was propagated by extrapolation using the higher order formula of the pair (XEPS). This implementation makes the global error approximation proportional to the tolerance, even though the local truncation error is that of the lower order formula (see [13]). We begin by tabulation of results on the interval [0,20] for two particular problems. Problem B5,

$$\begin{cases} y_1' = y_2 y_3, & y_1(0) = 0, \\ y_2' = -y_1 y_3, & y_2(0) = 1, \\ y_3' = -0.51 y_1 y_2, & y_3(0) = 1, \end{cases}$$

gives Euler equations of motion of a rigid body without external forces. Problem E3,

$$\begin{cases} y_1' = y_2, & y_1(0) = 0, \\ y_2' = y_1^3/6 - y_1 + 2\sin(2.78535x), & y_2(0) = 0, \end{cases}$$

is derived from Duffing's equation $y'' + y - y^3/6 = 2\sin(2.78535x)$. Later, we include results from the Arenstorf orbit problem over one period of approximately [0,17.065], equations for which appear in [9] (page 129).

Tables 5 and 6 report results of XEPS implementation on the first two problems for two *efficient* pairs over a range of Tolerances. The number of steps (NS), number of rejected steps (NR), number of function evaluations (NFE) and the maximum global error (GE) are tabulated. These tables illustrate reduction of computation required with higher order pairs even for relaxed requirements of accuracy. Moreover, the process for estimating the order p achieved in a variable step implementation (reviewed below), illustrates that the effective order can be higher than expected. In other computations done with these two problems, the application of the higher-order

Tol	Problem B5 (Euler)					Problem E3 (Duffing)				
	NS	NR	NFE	GE	p	NS	NR	NFE	GE	p
10^{-6}	70	2	650	3.77e-7		123	0	1109	1.13e-7	
10^{-7}	101	0	911	3.43e-8	7.10	174	0	1568	7.36e-9	7.88
10^{-8}	148	0	1334	2.66e-9	6.70	251	0	2261	4.92e-10	7.39
10^{-9}	216	0	1946	1.90e-10	6.99	363	0	3269	4.52e-11	6.47
10^{-10}	316	0	2846	1.45e-11	6.77	528	0	4754	7.52e-12	4.79

Table 5: Maximal global error for IIIX-65, the efficient RK6(5) pair with $||LTE||_{72} = .000001446174055$ and DSO = 3.

Tol	Problem B5 (Euler)					Problem E3 (Duffing)				
	NS	NR	NFE	GE	p	NS	NR	NFE	GE	p
10^{-6}	31	5	470	3.62e-7		53	1	704	5.22e-7	
10^{-7}	39	7	600	6.11e-8	7.29	66	0	860	4.33e-8	12.44
10^{-8}	49	7	730	1.11e-8	8.70	82	1	1081	4.59e-9	9.81
10^{-9}	63	7	912	8.52e-10	11.53	105	0	1367	1.82e-10	13.75
10^{-10}	82	5	1133	9.20e-11	10.26	135	0	1757	2.53e-11	7.86

Table 6: Maximal global error for IIIX-8(7), an efficient RK8(7) pair with $||LTE||_{92} = .000000282$ and DSO = 4.

method of each pair with a *fixed stepsize* selected to use the same amount of computation as required in the corresponding variable step simulation, the global errors are slightly smaller than those displayed in the tables. This suggests that these two problems selected may not adequately challenge the variable step format.

In contrast, the problem characterizing Arenstorf orbits in [9], shows a substantial advantage for *variable* step over *fixed* step Runge–Kutta implementations. For each of higher-order methods of the pairs studied, solutions to this model system with a fixed stepsize could only be achieved with a very large number of steps. Some results for this problem appear in Section 4.

For convenience, we review an accepted strategy for estimating the achieved order in a variable-step implementation of a Runge–Kutta pair. By maintaining the 2-norm of the local truncation error

$$LTE_n \equiv y_n - \widetilde{y}_n \approx Ch^p \tag{3.1}$$

less than but almost equal to a fixed fraction of the tolerance, the tolerance proportionality of the global error leads to

$$GE(h) \approx KCh^p \tag{3.2}$$

for a fixed stepsize h. In a variable stepsize implementation, we may estimate an average stepsize by the reciprocal of the number of steps, or conveniently as the multiple (M/NFE) of the reciprocal of the number of function evaluations required over an interval [a,b]. In turn, we can estimate

$$p \approx \frac{log_{10}|KC|M^p - log_{10}(|GE(h)|)}{log_{10}(NFE)} \tag{3.3}$$

Thus, a plot of $log_{10}(|GE(h)|)$ against a logarithmically scaled number of function evaluations will be approximately a straight line with (negative) slope p. Such a graph characterizes the efficiency of a pair: the lower or more to the left is the line indicates improved efficiency.

4 Efficiency Graphs for Three Problems

For each of the three problems selected in Section 3, we have plotted efficiency graphs for the ten pairs identified in Table 4, in sets partitioned by their orders ranging from 6(5) to 9(8). For each problem, in the subgraph arising from order 6(5) pairs, we include lines of all slopes -6,-7,-8,-9, to indicate the expected slopes of each subplot, and the relative differences expected among the four subplots.

In general, these graphs manifest expected and unexpected results. We observe that as the order increases, the computation required to achieve a particular global error is reduced. Further, for the higher order pairs, the downward curve of the graph shows that the achieved order is higher than expected. In general, the new selected *efficient* pair of each order is best for most of these comparisons. For the pairs of order 6, only the results for Duffing's problem show a clear advantage for the *efficient* pair. It is perhaps unexpected that the *robust* pairs seem to be uniformly less effective than the efficient pairs.

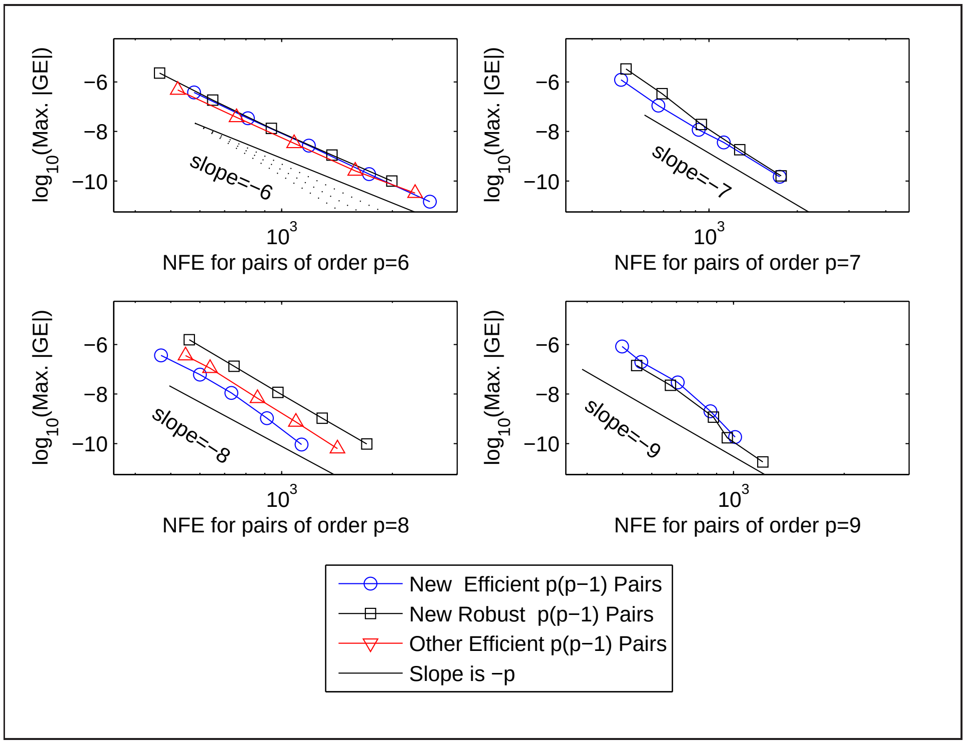

Figure 1: Maximum Global Error for the Euler System

From this restricted comparison, we can observe that some slight improvement on currently implemented algorithms might be possible. The closeness of these graphs indicate that improvements by further reduction of the 2-norms of the LTE coefficients are likely to be marginal at best.

5 Results from DETEST

To illustrate the general effectiveness of the pairs considered, we have executed the DETEST test set of problems with each of the pairs. Each pair is implemented using different fractions of EST $= ||y_n - \widehat{y}_n||_2$ as the Error Estimate in order to approximately equalize the total amount of computation required for execution over the range of Tolerances $10^{-k},\ k = 3, \ldots, 9$. These results may be contrasted with others that appear in [4, 18, 19, 20, 15]. From Table 5, we observe that as the order of the pair increases, there is a slight improvement in efficiency. As well, for a given level of global error,

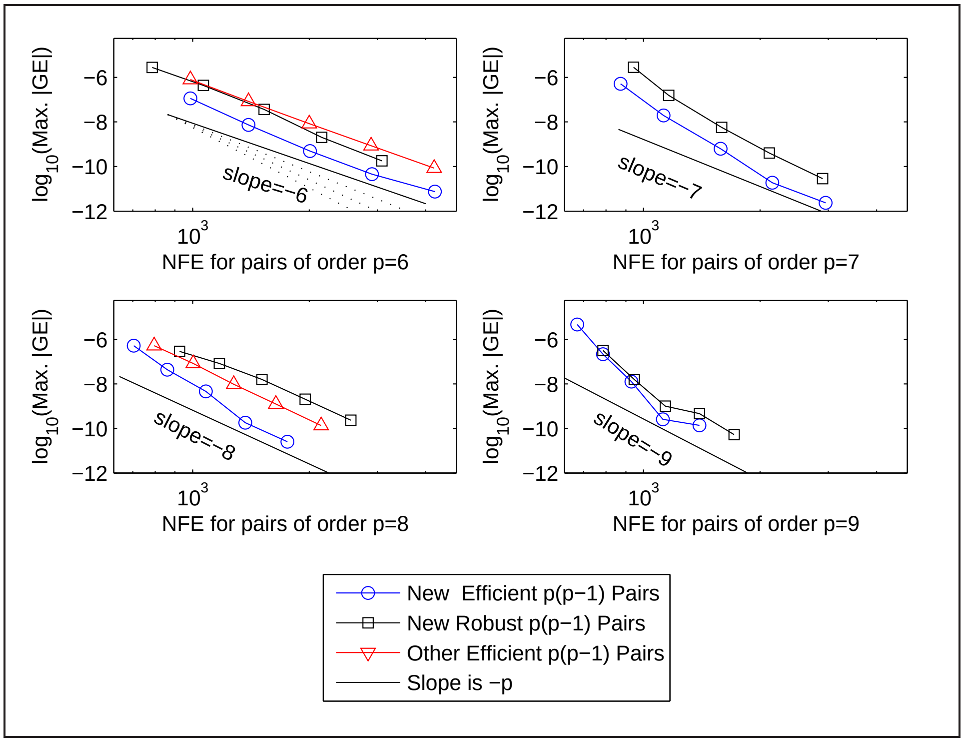

Figure 2: Maximum Global Error for Duffing's system

the amount of computation required is slightly lower for *efficient* pairs. Over the range of tolerances for the DETEST results, the *efficient* 9(8) pair appears to be best among those examined, and is comparable to the algorithm implementated as dsolve[dverk78] in MAPLE.

6 Conclusions

The tests reported here indicate that to meet a required accuracy with a minimum of computation on nonstiff problems, *efficient* pairs are more effective than *robust* pairs, and the pairs of higher orders tend to be more effective. In addition, the IIb pair in Table 5 of [15] and the pair implemented as dsolve[dverk78] in MAPLE are good choices. Pairs displayed on the author's website have coefficients for appended differentiable approximations; as well, dverk78 is implemented with options for selecting continuous solutions.

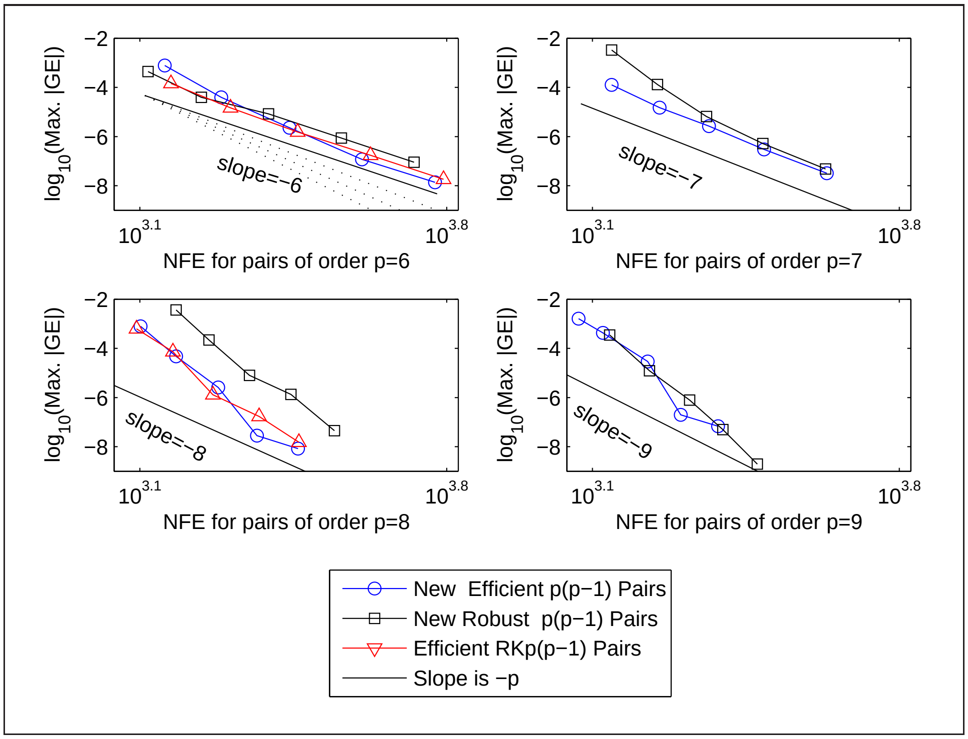

Figure 3: Maximum Global Error for Arenstorf Orbits

Pair	EE	$T_{p+1,2}$	FCN calls	No. of Steps	Max. Error	Frac. Decv	Frac. Bad Decv
IIIXb+6(5)	EST	.00010102	104247	11441	1.5	0.000	0.000
IIIXb-6(5)	EST/3	.00000144	103271	11269	1.7	0.001	0.000
IIb-6(5)	EST/3	.0000517	98960	10942	1.6	0.001	0.000
IIa1+7(6)	EST	.00002701	103072	8752	7.4	0.002	0.000
IIa1-7(6)	EST	.00001676	102036	8715	0.5	0.000	0.000
IIa1+8(7)	EST	.000007547	93626	5918	4.2	0.000	0.000
IIa1-8(7)	EST*7	.000000282	101220	6348	1.6	0.000	0.000
DVERK-8(7)	EST*5	.000000835	96309	6105	1.1	0.000	0.000
IIa1+9(8)	EST	.000000351	99670	5125	1.7	0.000	0.000
IIa1-9(8)	EST	.000000349	100067	5147	0.4	0.000	0.000

Table 7: Results from DETEST using XEPS

7 Acknowledgements

The work of this author was supported by the National Science and Engineering Council of Canada under grant A8147.

References

[1] J.C.Butcher, *The Numerical Analysis of Ordinary Differential Equations* John Wiley & Sons, (1987), pp. 297-303.

[2] M. Calvo, J.I. Montijano, and L. Randez, *A new embedded pair of Runge–Kutta formulas of orders 5 and 6*, Computers Math. Applic. **20**, Vol. 1 (1990), pp. 15-24.

[3] J. R. Dormand and P.J. Prince, *A family of embedded embedded Runge–Kutta formulae*, J. Comput Appl. Math . **6** (1980), pp. 19–26.

[4] J. R. Dormand and P.J. Prince, *A reconsideration of some embedded embedded Runge–Kutta formulae*, J. Comput Appl. Math . **15** (1986), pp. 203–211.

[5] J. R. Dormand, M.A. Lockyer, N.E. McGorrigan and P.J. Prince, *Global error estimation with Runge–Kutta triples*, Comput Math. Applic. **18** No. 9 (1989), pp. 835–846.

[6] W.H. Enright, *The relative efficiency of alternate defect control schemes for high-order continuous Runge–Kutta formulas*, SIAM J. Numer. Anal. **30** (1993), pp. 1419–1445.

[7] E. Fehlberg, *Klassische runge–Kutta-Formeln fünfter und siebenter Ordnung mit Schrittweiten-Kontrolle*, Computing **4** (1969), pp. 93–106.

[8] E. Hairer, *A Runge–Kutta method of order 10*, J. Inst. Math. Appl. **21** (1978), pp. 47–59.

[9] E. Hairer, S.P. Nørsett and G. Wanner, *Solving Ordinary Differential Equations I, Nonstiff Problems*, Springer-Verlag, Berlin, Heidelberg, New York 1991.

[10] T.E. Hull, W.H. Enright, B.M. Fellen and A.E. Sedgwick, *Comparing numerical methods for ordinary differential equations*, SIAM J. Numer. Anal. 9 (1972), pp. 603–637.

[11] S.N. Papakostas; Ch. Tsitouras; G. Papageorgiou, *A general family of explicit Runge–Kutta pairs of Orders 6(5)*, SIAM J. NA **33**, (1996), pp. 917–936.

[12] P.J. Prince and J. R. Dormand, *High order embedded Runge–Kutta formulae*, J. Comput Appl. Math . **7** (1981), pp. 67–75.

[13] L.F. Shampine, *The Numerical Solution of Ordinary Differential Equations*, Chapman & Hall, New York, (1994).

[14] P.W. Sharp and E. Smart, *RKSS56: A Runge–Kutta pair of orders 5 and 6*, Private Communication. (See CVSS6 in [6].)

[15] P.W. Sharp and J.H. Verner, *Completely Imbedded Runge–Kutta pairs*, SIAM J. Numer. Anal. **31** (1994), pp. 1169–1190.

[16] J.H. Verner, *Explicit Runge–Kutta methods with estimates of the local truncation error*, SIAM J. Numer. Anal. **15** (1979), pp. 772–790.

[17] J.H. Verner, *Families of imbedded Runge–Kutta methods*, SIAM J. Numer. Anal. **15** (1979), pp. 857–875.

[18] J.H. Verner, *A contrast of some Runge–Kutta formula pairs*, SIAM J. Numer. Anal. **27** (1990), pp. 1332–1344.

[19] J.H. Verner, *Some Runge–Kutta formula pairs*, SIAM J. Numer. Anal. **28** (1991), pp. 496–511.

[20] J.H. Verner, *The comparison of some Runge–Kutta pairs using DE-TEST*, Computational Ordinary Differential Equations: Proc. of the Thrid and Fourth International Conferences, Benin City, Nigeria, Ed. S.O. Fatunla, Caxton Press, Benin City, Nigeria (1992), pp. 273-284.

[21] J.H. Verner, *Differentiable Interpolants for high-order Runge–Kutta methods*, SIAM J. Numer. Anal. **30** (1993), pp. 1446-1466.

[22] J.H. Verner, *A classification scheme for studying explicit Runge–Kutta pairs*, Scientific Computing: Proc. of the Fifth International Conference, Benin City, Nigeria. Ed. S.O. Fatunla, Ada Jane Press, Benin City, Nigeria (1994), pp. 201–225.

[23] J.H. Verner, *Strategies for deriving new explicit Runge–Kutta pairs*, Annals of Numerical Mathematics 1, (1994), pp. 225–244.

[24] J.H. Verner, *High order explicit Runge–Kutta pairs with low stage order*, Applied Numerical Mathematics, 20 (1996), 1–13.