MAX-PLANCK-INSTITUT FÜR PLASMAPHYSIK

GARCHING BEI MÜNCHEN

Symplectic Integrators for the ABC Flow

Michael K. Tippett
Max-Planck-Institut für Plasmaphysik
D-85748 Garching, Germany

September 6, 1995

Die nachstehende Arbeit wurde im Rahmen des Vertrages zwischen dem Max-Planck-Institut für Plasmaphysik und der Europäischen Atomgemeinschaft über die Zusammenarbeit auf dem Gebiete der Plasmaphysik durchgeführt.

Abstract

Explicit symplectic integration schemes for the Arnold-Beltrami-Childress flows are presented and compared to the fourth order Runge-Kutta method. For the calculation of stable orbits with moderate accuracy the symplectic schemes are more efficient. The structure of the Hamiltonian prevents the implementation of symplectic methods with constant time steps.

1. The ABC Flow

The Arnold-Beltrami-Childress (ABC) flows are 3D, incompressible flows with a variety of interesting features (c.f. [1]). The equations for the streamlines of a simplified case of the ABC flow are:

$$\dot{x} = \cos y + \epsilon \sin z \tag{1}$$

$$\dot{y} = \sin x + \epsilon \cos z \tag{2}$$

$$\dot{z} = \sin y + \cos x \tag{3}$$

where ϵ is a parameter. The equations may be rewritten in Hamiltonian form with z being the time-like variable [1]:

$$\frac{dx}{dz} = -\frac{\partial H}{\partial q} \tag{4}$$

$$\frac{dq}{dz} = \frac{\partial H}{\partial x} \tag{5}$$

with

$$H(x, q, z) = \sin y(x, q) + \cos x + \epsilon(y(x, q) \sin z - x \cos z) \tag{6}$$

and

$$q = \int_0^y (\sin y' + \cos x)\, dy' = 1 + y \cos x - \cos y. \tag{7}$$

For $\epsilon = 0$ the system is integrable. For $\epsilon \neq 0$, the system has both chaotic and stable quasi-periodic streamlines.

2. Symplectic Method

Symplectic integration schemes approximately preserve the Hamiltonian invariants of the system and are exact solutions of a "nearby" Hamiltonian system. So, rather than solving an exact Hamiltonian approximately, symplectic schemes solve an approximate Hamiltonian exactly. For this reason, symplectic schemes have been found to have good stability properties and allow longer time steps than non-symplectic methods (c.f. [2]).

Here, we use the approach of Yoshida to develop explicit symplectic integrators for the ABC flow [4]. The flow S associated with the Hamiltonian H is the mapping such that

$$S(\zeta) : (x(z_0), q(z_0)) \mapsto (x(z_0 + \zeta), q(z_0 + \zeta)). \tag{8}$$

The mapping S is symplectic. For a one degree of freedom Hamiltonian, symplecticness is equivalent to preservation of area in the (x, q) plane. Additionally, we define a flow T such that

$$T(\tau) : (x(t_0), y(t_0), z(t_0)) \mapsto (x(t_0 + \tau), y(t_0 + \tau), z(t_0 + \tau)) \tag{9}$$

where $x(t)$, $y(t)$, $z(t)$ solve equations (1, 2, 3). The flow T is more convenient to calculate than is S and as it can be expressed in terms of S, is also symplectic with respect to (x, q). We note additionally that since T is the flow of an incompressible velocity field, it preserves volume in (x, y, z) space.

The method of Yoshida requires that we be able to split the Hamiltonian into parts such that the Hamiltonian system associated with each part can be solved explicitly. We separate the Hamiltonian into two parts $H = H_1 + H_2$ with

$$H_1(x, q) = \sin y(x, q) + \cos x \tag{10}$$

and

$$H_2(x, q, z) = \epsilon(y(x, q) \sin z + x \cos z). \tag{11}$$

Let $T_1(\tau)$ and $T_2(\tau)$ be the flows associated with H_1 and H_2. Then the flow $T(\tau)$ is approximated by

$$T(\tau) = T_2(\tau/2)T_1(\tau)T_2(\tau/2) + \mathcal{O}(\tau^3). \tag{12}$$

As the maps T_1 and T_2 are symplectic, so is their product. A fourth order approximation is given by

$$\begin{aligned} T(\tau) = {} & T_2(\alpha\tau)T_1(\beta\tau)T_2(\gamma\tau)T_1(\delta\tau) \\ & \times T_2(\gamma\tau)T_1(\beta\tau)T_2(\alpha\tau) + \mathcal{O}(\tau^5). \end{aligned} \tag{13}$$

with

$$\alpha = \frac{2 + 2^{1/3} + 2^{-1/3}}{6} \tag{14}$$

$$\beta = \frac{1}{2 - 2^{1/3}} \tag{15}$$

$$\gamma = \frac{1 - 2^{1/3} - 2^{-1/3}}{6} \tag{16}$$

$$\beta = \frac{1}{1 - 2^{1/3}}. \tag{17}$$

A sixth order integrator can also be easily constructed from the fourth order one [4].

It is important to note that the subscripts 1 and 2 in equations (12) and (13) may be interchanged without affecting the order of the method. Since T_2 must be applied twice, it is desirable that T_2 be less computationally expensive than T_1. Secondly, the estimate of the error term suggests that the approximation is better if H_1 is smaller than H_2 [4].

We note that in general the use of a constant time step τ for the flow T does not give a constant step size ζ for the flow S. Only in the case $\epsilon = 0$, is τ proportional to ζ. As can be seen from equation 3, there are points x_0, y_0, z_0 such that there is no step size τ that results in a given step size ζ. This inability to use a step size that is constant is important because it has been observed that many of the desirable features of symplectic integration schemes are lost when a variable step size is used (c.f. [6]). An intuitive reason for this effect is that when a constant step size is used, one is solving a single approximate Hamiltonian system. While if a variable step size is used, one is solving a different approximate Hamiltonain at each step.

2.1. Calculation of T_1

We calculate the flow T_1 by solving the system:

$$\dot{x} = \cos y \tag{18}$$

$$\dot{y} = \sin x \tag{19}$$

$$\dot{z} = \sin y + \cos x, \tag{20}$$

with initial conditions $x(0) = x_0$, $y(0) = y_0$, $z(0) = z_0$. Following [3] (with a few minor corrections) we note that $\dot{z} = const = v$, so that $z(t) = z_0 + vt$. We introduce new variables ϕ_1 and ϕ_2 with

$$\phi_1 = \frac{1}{2}(y + x - \frac{\pi}{2}) \tag{21}$$

$$\phi_2 = \frac{1}{2}(y - x - \frac{\pi}{2}). \tag{22}$$

Then both ϕ_1 and ϕ_2 satisfy the differential equation

$$\dot{\phi}^2 = k^2 - \sin^2 \phi. \tag{23}$$

where

$$k = \sqrt{1 - v^2/4}. \tag{24}$$

Separating variables in (23) gives

$$\frac{d\phi}{\sqrt{1 - 1/k^2 \sin^2 \phi}} = \pm k \, dt, \tag{25}$$

which can be integrated to obtain

$$F(\phi | 1/k^2) = \pm k(t - t_0) \tag{26}$$

where F is the elliptic integral of the first kind and t_0 depends on the initial conditions. Inverting (26) we have

$$\sin(\phi(t)) = \operatorname{sn}(\pm k(t - t_0) | 1/k^2), \tag{27}$$

or more conveniently,

$$\sin\phi(t) = k\ \mathrm{sn}(\pm(t-t_0)|k^2), \tag{28}$$

since $0 \le k \le 1$.

So ϕ_1 and ϕ_2 are given by

$$\sin\phi_1(t) = k\ \mathrm{sn}(\pm(t-t_{01})|k^2), \tag{29}$$

and

$$\sin\phi_2(t) = k\ \mathrm{sn}(\pm(t-t_{02})|k^2), \tag{30}$$

where t_{01} and t_{02} are chosen to satisfy the initial conditions for ϕ_1 and ϕ_2. The choice of sign in the definitions of ϕ_1 and ϕ_2 is given by the initial sign of $\dot{\phi}$. We note that

$$\dot{\phi}_1 = \frac{1}{2}(\sin x + \cos y) = -\cos\phi_1 \sin\phi_2, \tag{31}$$

$$\dot{\phi}_2 = \frac{1}{2}(\sin x - \cos y) = \cos\phi_2 \sin\phi_1. \tag{32}$$

Because the solution we have found is valid for $-\frac{\pi}{2} < \phi_{1,2} < \frac{\pi}{2}$, if $-\phi_2 > 0$ then

$$\sin\phi_1(t) = k\ \mathrm{sn}((t-t_{01})|k^2), \tag{33}$$

and if $\phi_1 > 0$

$$\sin\phi_2(t) = k\ \mathrm{sn}((t-t_{02})|k^2). \tag{34}$$

Otherwise we take the negative sign.

Additionally, there is a relation between t_{01} and t_{02}. A short calculation shows that

$$\cos\phi_1 \cos\phi_2 = \frac{v}{2}. \tag{35}$$

This can be rewritten as

$$\mathrm{dn}((t-t_{01})|k^2)\ \mathrm{dn}((t-t_{02})|k^2) = \sqrt{1-k^2}, \tag{36}$$

noting that $\mathrm{dn}(-u|k^2) = \mathrm{dn}(u|k^2)$. Using the following identity for elliptic functions

$$\sqrt{1-k^2} = \mathrm{dn}((t-t_{01})|k^2)\ \mathrm{dn}((t-t_{01} \pm K)|k^2) \tag{37}$$

where K is the usual quarter period, implies that t_{01} and t_{02} differ by the quarter period K.

The solution can be extended to the entire plane using periodicity and parity considerations. That is, we note that if ϕ is a solution then so are $\phi - \pi$ and $\phi \pm n2\pi$.

2.2. An Approximation to T_1

We can avoid the numerical cost of calculating T_1 by breaking H_1 itself into two parts,

$$H_1 = H_1^{(1)} + H_1^{(2)}, \tag{38}$$

with $H_1^{(1)} = \cos x$ and $H_1^{(2)} = \sin y$ and then calculating a symplectic approximation to T_1 using (12) or (13). The flow associated with $T_1^{(1)}$ is

$$x(t) = x_0 + t\cos y \tag{39}$$

$$y(t) = y_0. \tag{40}$$

The flow $T_1^{(2)}$ is

$$x(t) = x_0 \tag{41}$$

$$y(t) = y_0 + t\sin x. \tag{42}$$

We continue to use the relation $z(t) = z_0 + vt$.

2.3. Calculation of T_2

The calculation of the flow of H_2 is trivial:

$$x(t) = x_0 + \epsilon t \sin z \tag{43}$$

$$y(t) = y_0 + \epsilon t \cos z \tag{44}$$

and

$$z(t) = z_0 \tag{45}$$

3. Numerical Tests

3.1. Implementation

The numerical implementation of the symplectic scheme described above requires the numerical approximations of the elliptic integral of the first kind, $F(\phi|1/k^2)$ and of the Jacobi elliptic function $\mathrm{sn}(t|k^2)$. We use the IMSL functions `DELRF` and `DEJSN`. The function `DEJSN` calculates the Jacobi function. The function `DELRF` calculates the function R_F given by

$$R_F(x,y,z) = \frac{1}{2}\int_0^\infty \frac{dt}{\sqrt{(t+x)(t+y)(t+z)}}, \tag{46}$$

	order	T_1	time
h2	2	exact	12.34
h2r	2	exact	24.32
h2a	2	2nd order	0.90
h2ar	2	2nd order	1.13
h4	4	exact	36.57
h4r	4	exact	48.24
h4a	4	4th order	3.27
h4ar	4	4th order	3.89
rk4	4	-	2.08
h6	6	exact	109.89
h6r	6	exact	120.21
h6a	6	6th order	25.07
h6ar	6	6th order	27.16

Table 1: Integration Subroutine Times

where x, y and z are positive. This function is related to the elliptic integral of the first kind by

$$F(\phi|1/k^2) = \sin\phi R_F(q, r, 1), \tag{47}$$

with $q = \cos^2\phi$ and $r = 1 - 1/k^2 \sin^2\phi$. To avoid r being negative due to round-off errors, here we use

$$r = \frac{2(\cos y + \sin x)^2}{6 - 2\cos^2 x + 2\cos^2 y - 4\cos x \sin y}. \tag{48}$$

3.2. Timing

We implement several symplectic integrators. In particular, we implement schemes of order two, four, and six with both exact and approximate calculations of T_1. The r suffix means that we have reversed the subscripts 1 and 2 in equations (12) and (13). Additionally we compare a fourth order Runge-Kutta scheme, rk4. The time (in seconds) required for 100,000 time steps on an IBM RISC-60000, is shown in Table 1. The large numerical cost associated with the elliptic functions and integrals is clear.

3.3. Local Error

The quantity H satisfies the equation

$$\frac{dH}{dt} = \epsilon(\sin y + \cos x)(x \sin z + y \cos z). \tag{49}$$

We measure the local error of the scheme by comparing the numerical value of $\frac{dH}{dt}$ with its analytical one above. We use the relation

$$\frac{dH}{dt}(t) = \frac{1}{60}(H(t-3\Delta t) - H(t+3\Delta t)) - \frac{3}{20}(H(t-2\Delta t) - H(t+2\Delta t)) \\ + \frac{3}{4}(H(t-\Delta t) - H(t+\Delta t)) + \mathcal{O}(\Delta t^7), \tag{50}$$

and average over a time of length 10. Fig. 1 shows the local error for the second order scheme. Of the second order schemes, h2ar is the most efficient with regard to the local error. Fig. 2 shows the relative error of the fourth order schemes. The most efficient of the fourth order schemes is rk4, and of the symplectic schemes h4ar. The local error of the sixth order schemes is shown in Fig. 3. The cost in CPU time versus the local error plotted in Fig. 6 shows that with respect to local error the rk4 scheme is the most efficient.

3.4. Long Time Behavior

Here we present two examples to show the stability of the symplectic methods. We take $\epsilon = 0.25$, and $(x_0 = 3.14, y_0 = 2.77, z_0 = 0.0)$. The orbit beginning at this point is contained in a stable region phase space. We follow the orbit for a time 1×10^5 and examine the Poincaré section of the orbit in the xy plane $\pmod{2\pi}$. In Fig. 4 we see that the h2ar scheme gives stable orbit for time steps as large as $\tau = 0.5$. we note that Fig. 4 is typical of all the sympletic schemes. In Fig. 5 we see that for large time steps the rk4 method fails completely. To judge the relative merits of the schemes in a more quantitative manner, in Figure 7(a) we plot the quantity

$$e_i = 2\frac{|\mathbf{r}_i - \mathbf{r}_{i-1}|}{|\mathbf{r}_i + \mathbf{r}_{i-1}|} \tag{51}$$

where $\mathbf{r}_i$ is final point of the orbit calculated using $\Delta t = 2^{-i}$, and in Figure 7(b)

$$E_i = 2\frac{|\mathbf{r} - \mathbf{r}_c|}{|\mathbf{r} + \mathbf{r}_c|}, \tag{52}$$

where $\mathbf{r}_c$, the converged value is $(x_c = 5.01523, y_c = 4.88371 z_c = -63326.5054)$. The quantity e_i shows if the method is converging. For $e_i < 10^{-8}$ round-off errors begin to dominate. Because it uses fewer floating point operations, the rk4 method is slower to be effected by round off errors than are the four order symplectic schemes. In Fig 8 we see that the choice of the most economical method depends on the level of accuracy desired. For low accuracy h2ar is best. For greater levels of accuracy h4ar is better.

We calculate another example this time with $(x_0 = 5.325, y_0 = 4.7, z_0 = 0.0)$. Again we follow the orbit for a time 10^5. In Fig. 9 we see that the h4ar scheme produce stable orbits for time steps as large as $\Delta t = 0.5$. In Fig 10 we see that the rk4 method gives a stable orbit only for $\Delta t = 0.0625$. Large time steps produce error like those seen in Fig. 5 where the structure of Hamiltonian is lost as well error that cause initially stable orbit to become enter the chaotic region of phase space. We plot the convergence and error at the endpoints in Fig. 11. In Fig 12 we see that for sufficiently good accuracy `rk4` is best, though for moderate accuracy `h4ar` is more efficient.

The study of orbits in the unstable region of phase space is considerably more difficult. By definition, the Lyaponov exponents are positive and errors grow exponentially. For times and time steps like those used above for the stable orbits, none of the calculations converge. One would expect that many of the good properties of the symplectic schemes would also be found in the unstable case. For example, one would expect that the sympletic methods to be less likely than the Runge-Kutta methods to "drift" from the unstable region into the stable region. However, without converged long time orbits, such comments are only speculation. Since many studies are interested in statistical properties of the orbits rather then exact solutions, a direction for future investigation would be to examine the statistical properties such as diffusion coefficients, Poincaré recurence times and exit times for the approximate solutions [7, 8].

4. Conclusions

Here we have constructed symplectic integrators for the ABC flow and have made some preliminary test calculations. For moderate accuracy the second order symplectic schemes are more efficient than fourth order Runge-Kutta. The symplectic methods are stable even with very large time steps. We suspect that because of a variable "time" step these symplectic schemes lack the usual benefits associated with symplectic methods. It is not clear if it is possible to implement a method with constant step size. The same methods could easily be applied to so called Q-flows [1].

References

[1] G. M. Zaslavsky, R. Z. Sagdeev, D. A. Usikov, and A. A. Chernikov, *Weak chaos and quasi-regular patterns*, (Cambridge University Press, Cambridge, 1991)

[2] J. R. Cary and I. Doxax *J. Comp. Phys.*, **107**. 98 (1993)

[3] T. Dombre *et al.*, *J. Fluid Mech.* **167**, 353 (1986)

[4] Haruo Yoshida, *Physics Letters A*, **150**, 262 (1990)

[5] J. Candy and W. Rozmus *J. Comp. Phys.*, **92**. 130 (1991)

[6] J. M. Sanz-Serna and M. P. Calvo, *Numerical Hamiltonian Problems*, (Chapman & Hall, London, 1994)

[7] G. M. Zaslavsky and M. K. Tippett *Phys. Rev. Lett.* **67**, 3251 (1991)

[8] S. Benkadda, Y. Elskins and B. Ragot *Phys. Rev. Lett.* **72**, 2859 (1994)

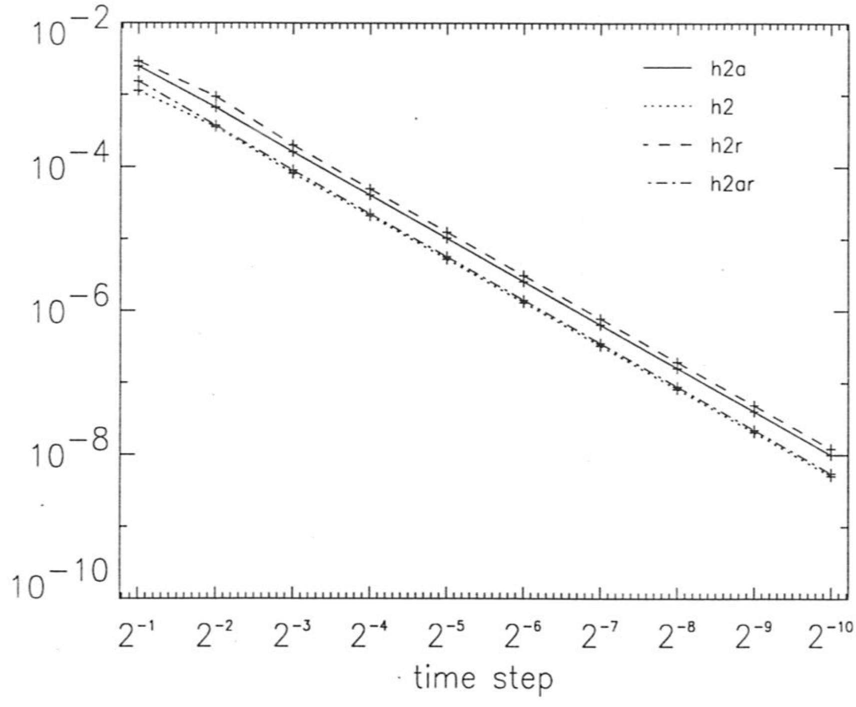

Figure 1: Local error for second order symplectic schemes

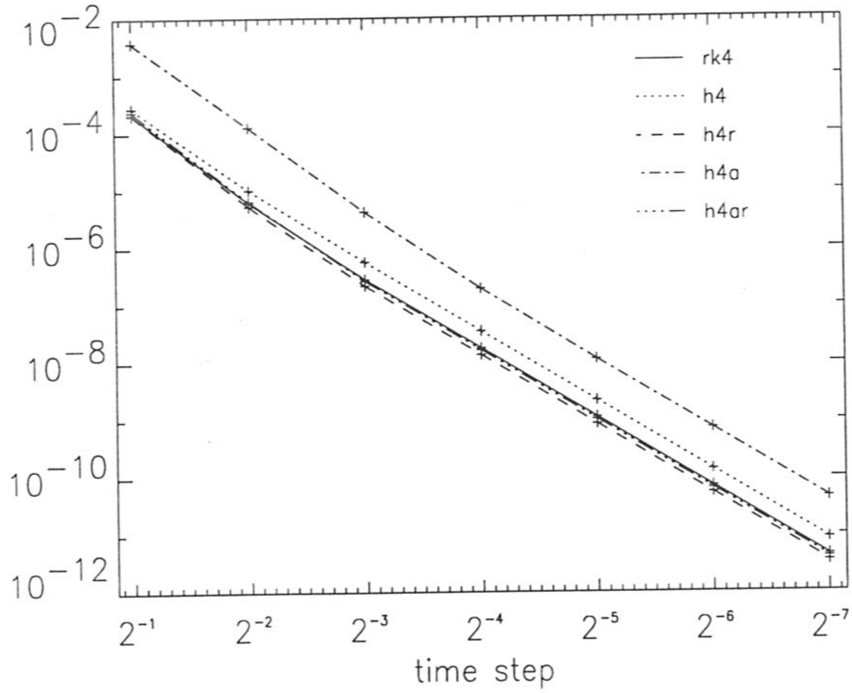

Figure 2: Local error for fourth order schemes

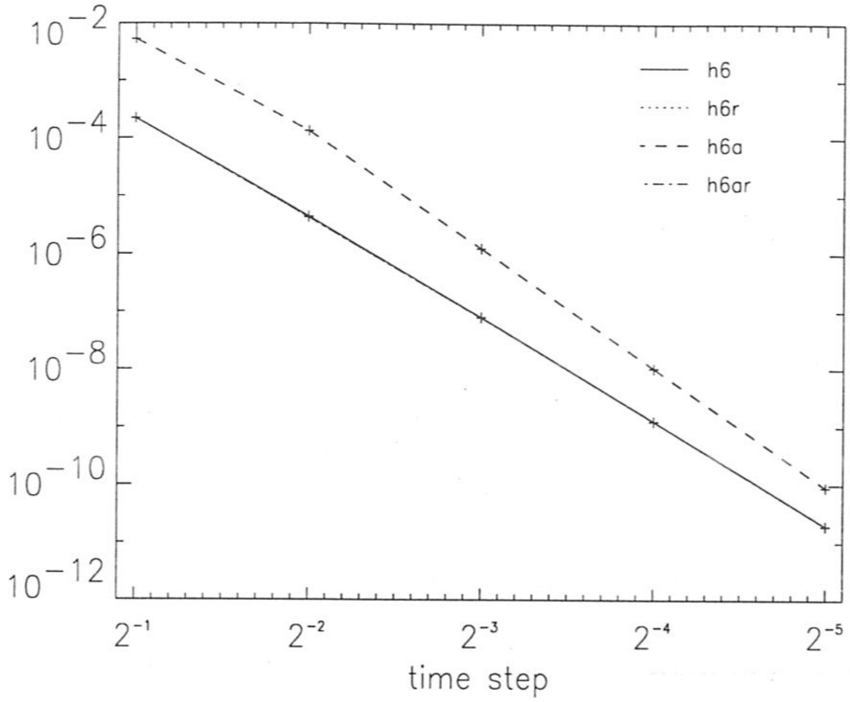

Figure 3: Local error for sixth order symplectic schemes

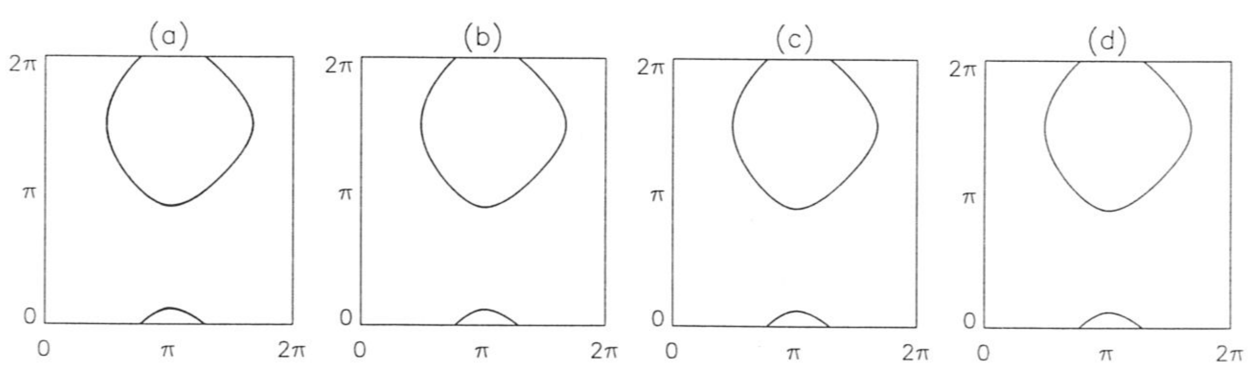

Figure 4: h4ar (a) $\Delta t = 0.5$ (b) $\Delta t = 0.25$ (c) $\Delta t = 0.125$(d) $\Delta t = 0.0625$

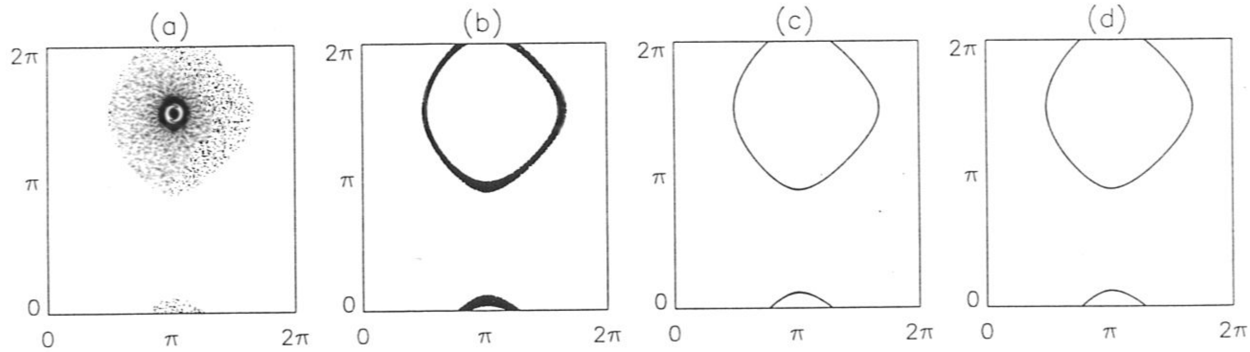

Figure 5: rk4 (a) $\Delta t = 0.5$ (b) $\Delta t = 0.25$ (c) $\Delta t = 0.125$ (d) $\Delta t = 0.0625$

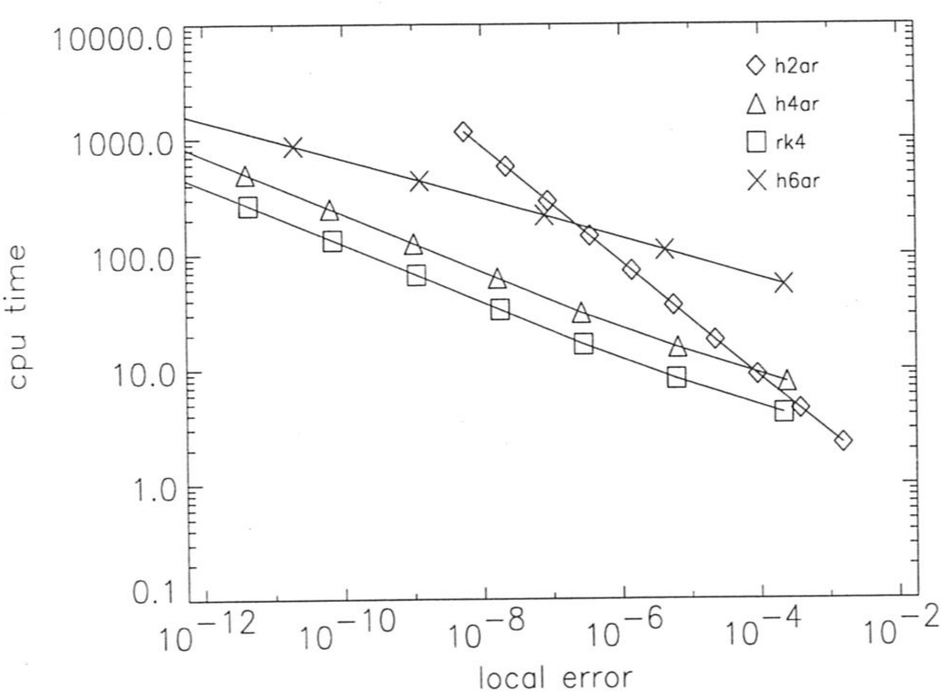

Figure 6: Cost in cpu time versus local error

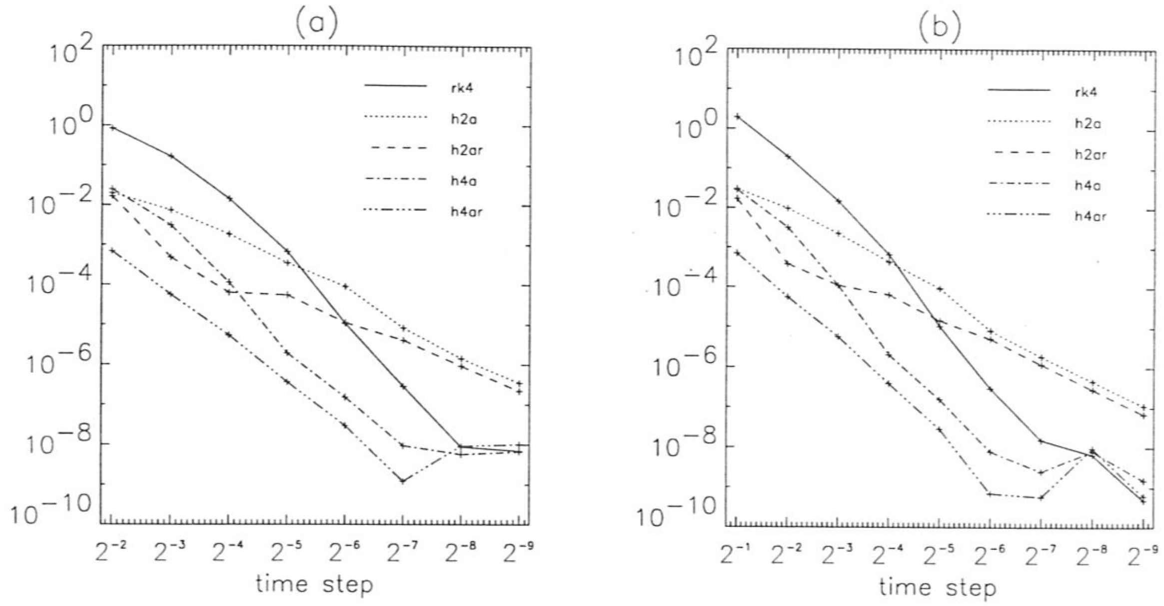

Figure 7: Error at end points (a) e_i (b) E_i

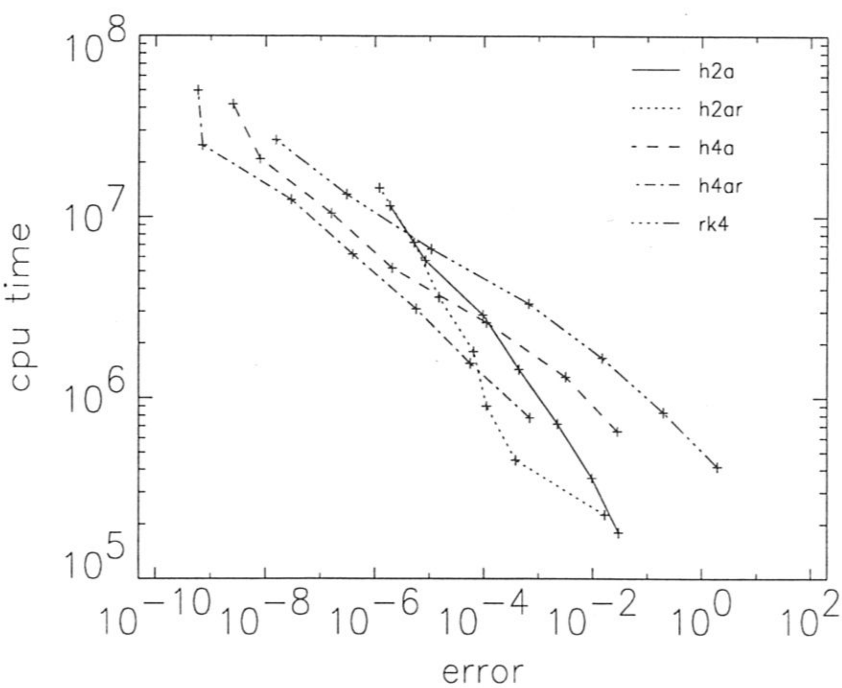

Figure 8: CPU time versus error at end points

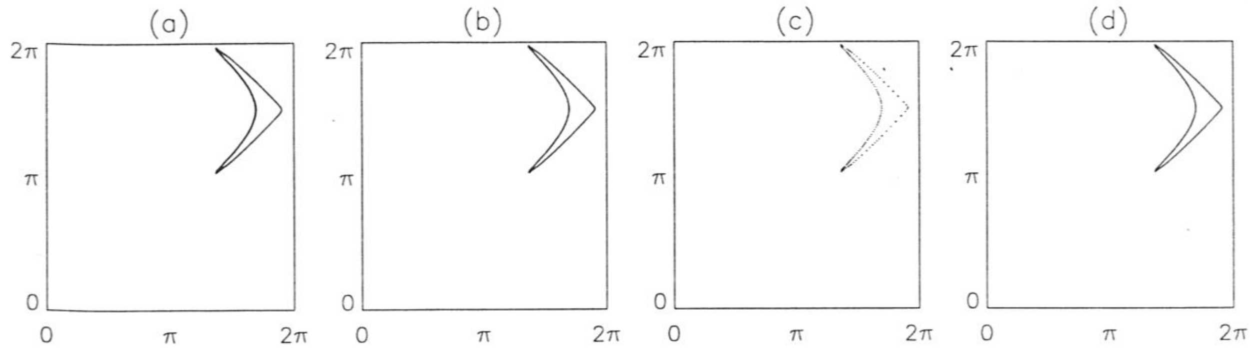

Figure 9: `h4ar` (a) $\Delta t = 0.5$ (b) $\Delta t = 0.25$ (c) $\Delta t = 0.125$(d) $\Delta t = 0.0625$

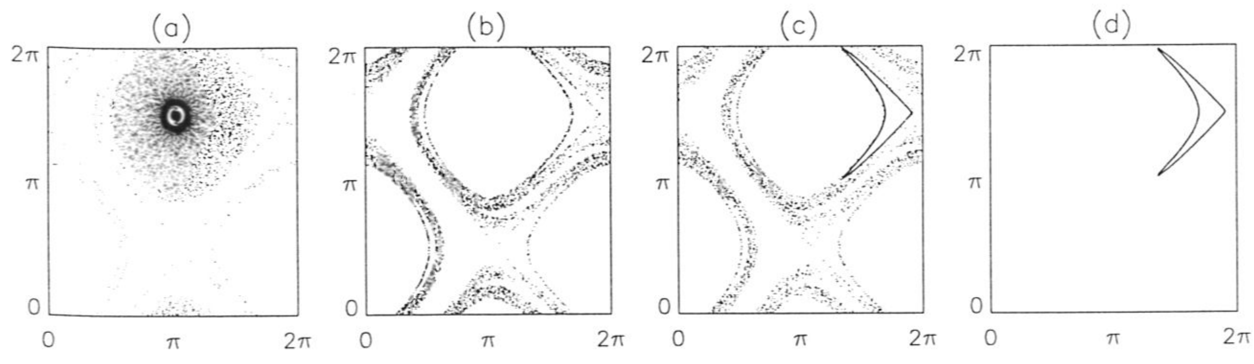

Figure 10: `rk4` (a) $\Delta t = 0.5$ (b) $\Delta t = 0.25$ (c) $\Delta t = 0.125$ (d) $\Delta t = 0.0625$

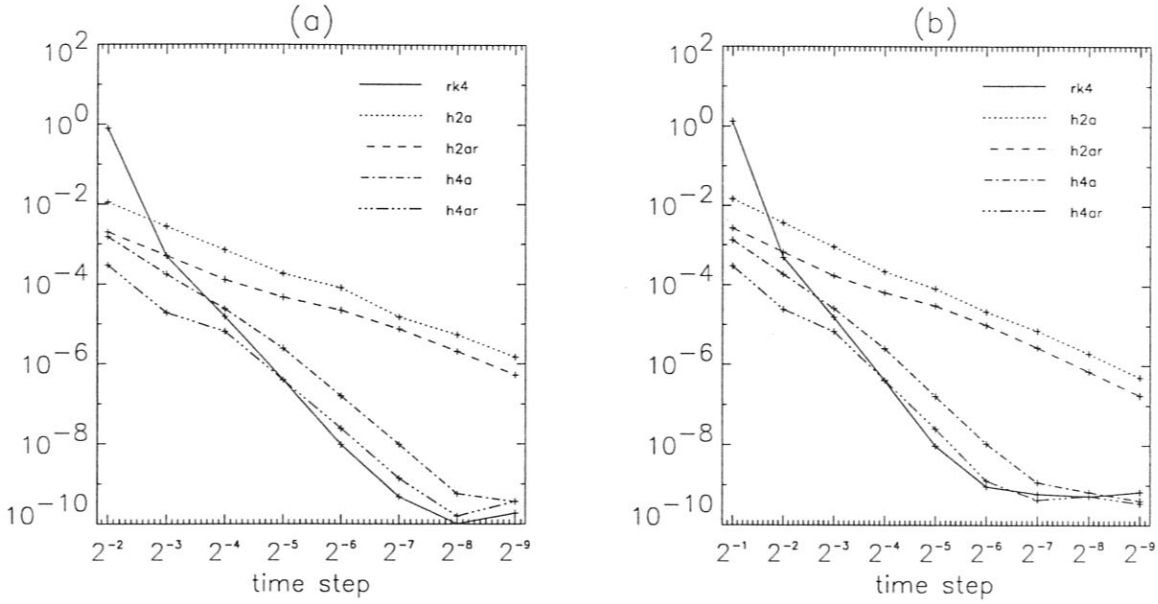

Figure 11: Error at end points (a) e_i (b) E_i

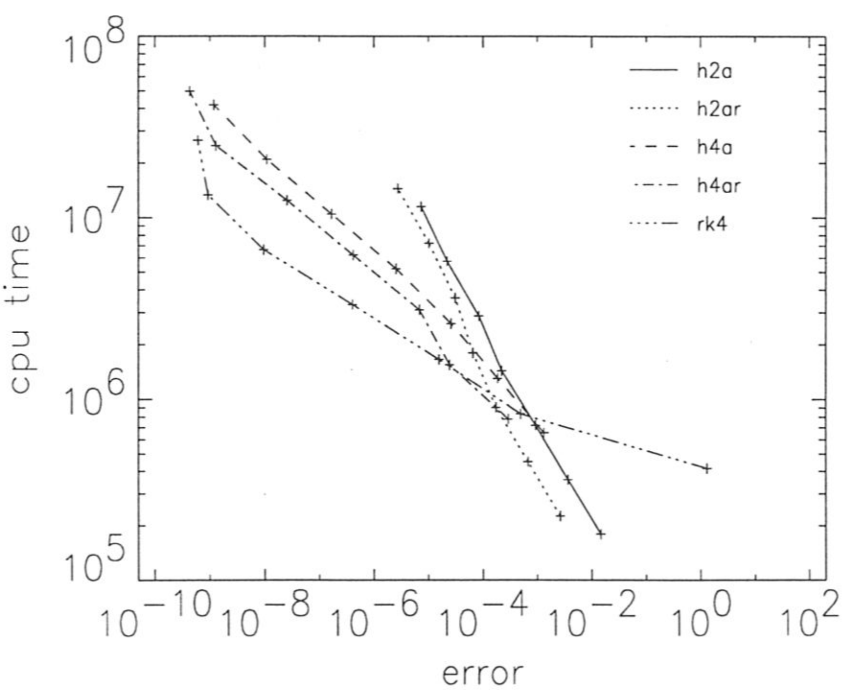

Figure 12: CPU time versus error at end points