

Fortran codes illustrating the methods presented in the paper “New families of symplectic splitting methods for numerical integration in dynamical astronomy”, by S. Blanes, F. Casas, A. Farrés, J. Laskar, J. Makazaga and A. Murua.

1 Introduction

The following notes have to be read jointly with Ref. [1]. The paper [2] can also be useful. The codes included here illustrate the methods by solving the two-dimensional perturbed Kepler problem, with two different perturbations.

1.1 Splitting methods with exact flows

We consider a generic differential equation of the form

$$x' = f^{[a]}(x) + \varepsilon f^{[b]}(x), \quad x(0) = x_0 \in \mathbb{R}^D, \quad (1)$$

where $|\varepsilon| \ll 1$ and each part

$$x' = f^{[a]}(x), \quad x' = \varepsilon f^{[b]}(x) \quad (2)$$

is exactly solvable (or can be numerically solved up to round off accuracy) with solutions

$$x(\tau) = \varphi_\tau^{[a]}(x_0), \quad x(\tau) = \varphi_\tau^{[b]}(x_0)$$

respectively, at $t = \tau$, the time step. We consider compositions like

$$\psi_\tau = \varphi_{a_{s+1}\tau}^{[a]} \circ \varphi_{b_s\tau}^{[b]} \circ \varphi_{a_s\tau}^{[a]} \circ \dots \circ \varphi_{b_1\tau}^{[b]} \circ \varphi_{a_1\tau}^{[a]} \quad (3)$$

for appropriately chosen coefficients a_i, b_i . We say that such a method is of (generalized) order (r_1, r_2, \dots, r_m) (where $r_1 \geq r_2 \geq \dots \geq r_m$) if the local error satisfies that

$$\psi_\tau(x) - \varphi_\tau(x) = \mathcal{O}(\varepsilon\tau^{r_1+1} + \varepsilon^2\tau^{r_2+1} + \dots + \varepsilon^m\tau^{r_m+1}),$$

where $\varphi_\tau(x_0)$ denotes the exact solution of (1).

1.2 Splitting methods with approximate flows

If $x' = \varepsilon f^{[b]}(x)$ is not exactly solvable but can be approximated by a symmetric second order method, we can consider a composition of the form

$$\tilde{\psi}_h = \varphi_{a_{s+1}\tau}^{[a]} \circ \tilde{\varphi}_{b_s\tau}^{[b]} \circ \varphi_{a_s\tau}^{[a]} \circ \cdots \circ \varphi_{a_2\tau}^{[a]} \circ \tilde{\varphi}_{b_1\tau}^{[b]} \circ \varphi_{a_1\tau}^{[a]}, \quad (4)$$

where $\tilde{\varphi}_\tau^{[b]}$ is a symmetric second order approximation to $\varphi_\tau^{[b]}$.

2 Two simple examples

2.1 The perturbation is exactly solvable

We consider the two-dimensional perturbed Kepler problem with Hamiltonian

$$H(q, p) = H_{Kep}(q, p) + \varepsilon H_b(q)$$

where

$$H_{Kep} = \frac{1}{2}(p_1^2 + p_2^2) - \frac{1}{r}, \quad H_b = -\frac{1}{2r^3} \left(1 - \frac{3q_1^2}{r^2} \right), \quad (5)$$

and $r = \sqrt{q_1^2 + q_2^2}$. This Hamiltonian is a first approximation used to describe the dynamics of a satellite moving into the gravitational field produced by a slightly oblate spherical planet and whose motion takes place in a plane containing the symmetry axis of the planet. The following schemes are used:

- ABA82: The 4-stage (8,2) method [4, 5].
- ABA84: The 5-stage (8,4) method [5].
- ABA104: The 7-stage (10,4) method [1, 2].
- ABA864: The 7-stage (8,6,4) method [1, 2].
- ABA1064: The 8-stage (10,6,4) method [1, 2].

We have some fortran programs to solve this problem and can be easily adapted to solve any other problem of this class. The main program is:

```
PerturbedKepJacobiEnergy.f
```

and the coefficients of the methods are collected in

```
coefficientsJacobi(a,b,c,imet,iord)
```

where `iord` is the number to label each method and `imet` is the number of stages of each method (the cost per step). In the main file we can choose the method to be used. The following choices are possible:

```
iord=1: ABA (8,2) - imet=4 stages
iord=2: ABA-McL (8,4) - imet=5 stages
iord=3: ABA (10,4) - imet=7 stages
iord=4: ABA (8,6,4) - imet=7 stages
iord=5: ABA (10,6,4) - imet=8 stages
```

One step is computed as follows

```

do 40 j=1,imet+1
    bh=b(j)*h;    ch=c(j)*h**3;    ah=a(j)*h
    call keplerh(x,vx,ah)
    call perturbation(bh,ch,eps,x,vx)
40    continue

```

where h is the time step and $b(j)$, $a(j)$ are the coefficients of the method. The code allows to use a corrector (or modified potential with coefficients $c(j)$). Here `keplerh(x,vx,ah)` calls to a subroutine to advance the coordinates x and the momenta vx from some initial conditions to their final value after a time step ah through a pure Kepler problem (in two dimensions). It integrates the Kepler problem in Cartesian coordinates using the f and g Gauss functions [3]. In a similar way, `perturbation(bh,ch,eps,x,vx)` advances the system for the perturbation (it advances only the momenta).

The FSAL property is not applied for simplicity of the presentation (it only requires to rewrite a few lines and to indicate when the output is desired).

For the numerical experiments the code is written with initial conditions $q_1 = 1 - e$, $q_2 = 0$, $p_1 = 0$, $p_2 = \sqrt{(1+e)/(1-e)}$, with $e = 1/4$, which would correspond to the eccentricity of the unperturbed Kepler problem, and $\varepsilon = 10^{-3}$. It is integrated along the interval $t \in [0, 10000]$ and compute the averaged error in energy. This numerical test is repeated several times for each method using different time steps (changing the computational cost for the numerical integration). Finally, it shows the average errors versus the time step scaled by the number of stages per step, i.e. τ/s , in logarithmic scale (the cost is inversely proportional to τ/s , and the best methods should provide a given accuracy with the largest value of τ/s).

2.2 The perturbation is NOT exactly solvable

We now consider the two-dimensional perturbed Kepler problem with Hamiltonian

$$H(q, p) = H_{Kep}(q, p) + \varepsilon(H_a(p) + H_b(q))$$

where

$$H_a = \frac{1}{2}(p_1^2 + p_2^2) \tag{6}$$

and H_{Kep}, H_b are the same as in the previous example. The following schemes are used:

- ABA82: The 4-stage (8,2) method.
- ABAH844: The 6-stage (8,4) method [1, 2].
- ABAH864: The 8-stage (8,6,4) method [1, 2].
- ABAH1064: The 9-stage (10,6,4) method [1, 2].

We have some fortran programs for which contain the coefficients for all methods to solve this problem and can be easily adapted to solve any other problem of this class. The main program is:

`PerturbedKepHeliocentricEnergy.f`

and the coefficients for the method are collected in

`coefficientsHeliocentric(a,b,imet,iord)`

where now

```
iord=1: ABA (8,2) - imet=4 stages
iord=2: ABA (8,4,4) - imet=6 stages
iord=3: ABA (8,6,4) - imet=8 stages
iord=4: ABA (10,6,4) - imet=9 stages
```

One step is just a simple modification of the previous case

```
do 40 j=1,imet+1
  bh=b(j)*h;    ah=a(j)*h
  call keplerh(x,vx,ah)

  x=x+eps*(bh/2.d0)*vx
  call perturbation(bh,ch,eps,x,vx)
  x=x+eps*(bh/2.d0)*vx
```

```
40      continue
```

Observe that the difference with the previous case is that the exact solution for the perturbation is replaced by the leapfrog method for the fractional time step `bh`

```
x=x+eps*(bh/2.d0)*vx
call perturbation(bh,ch,eps,x,vx)
x=x+eps*(bh/2.d0)*vx
```

References

- [1] S. Blanes, F. Casas, A. Farrés, J. Laskar, J. Makazaga, and A. Murua. New families of symplectic splitting methods for numerical integration in dynamical astronomy. 2012. (arXiv:1208.0689)
- [2] A. Farrés, J. Laskar, S. Blanes, F. Casas, J. Makazaga, and A. Murua. High precision Symplectic Integrators for the Solar System. 2012. (arXiv:1208.0716)
- [3] J.M.A. Danby. *Fundamentals of Celestial Mechanics*. Willmann-Bell, (1992).
- [4] J. Laskar and P. Robutel. High order symplectic integrators for perturbed Hamiltonian systems. *Celest. Mech. and Dyn. Astro.*, **80** (2001), 39-62.

- [5] R. I. McLachlan. Composition methods in the presence of small parameters. *BIT*, **35** (1995), 258-268.