Conservation of First Integrals and Projection Methods

CASA

Fengnan Gao

Industrial and Applied Mathematics
Eindhoven University of Technology

13.10.2010

Outline

Intuition

- ODEs often conserve certain quantities
- Numerical solutions also should
- How to design such a numerical method?

Definition of First Integrals

- Consider differential equation

$$\dot{y} = f(y).$$

Definition of First Integrals

- Consider differential equation

$$\dot{y} = f(y).$$

- A non-constant function $I(y)$ is called a *first integral* if

$$I'(y) f(y) = 0 \qquad \text{for all } y.$$

Definition of First Integrals

- Consider differential equation

$$\dot{y} = f(y).$$

- A non-constant function $I(y)$ is called a *first integral* if

$$I'(y) f(y) = 0 \qquad \text{for all } y.$$

- This implies

$$I(y(t)) = I(y_0) = \textit{Const.}$$

  for *any* solution $y(t)$.

Example: Conservation of the Total Energy

- The Hamiltoninan systems of the form

$$\dot{p} = -H_q(p, q), \qquad \dot{q} = H_p(p, q),$$

- $H_q = \nabla_q H = \left(\frac{\partial H}{\partial q}\right)^T$ and $H_p = \nabla_p H = \left(\frac{\partial H}{\partial p}\right)^T$
- The Hamilton function $H(p, q)$ is a first integral

$$\frac{\partial H}{\partial p}\left(-\frac{\partial H}{\partial q}\right)^T + \frac{\partial H}{\partial q}\left(\frac{\partial H}{\partial p}\right)^T = 0.$$

Background: Explicit Runge-Kutta Methods

1. Given initial value problem $y' = f(t, y)$, $y(t_0) = y_0$
2. Given the current step y_n and t_n
3. Runge-Kutta methods are given by:

$$\begin{aligned}
k_1 &= f(t_n, y_n), \\
k_2 &= f(t_n + c_2 h, y_n + a_{21} h k_1), \\
k_3 &= f(t_n + c_3 h, y_n + a_{31} h k_1 + a_{32} h k_2), \\
&\ \ \dots \\
&\ \ \dots \\
k_s &= f(t_n + c_s h, y_n + a_{s1} h k_1 + a_{s2} h k_2 + \cdots + a_{s,s-1} h k_{s-1}), \\
y_{n+1} &= y_n + h \sum_{i=1}^{s} b_i k_i ,
\end{aligned}$$

4. y_{n+1} and t_{n+1} are the new step

Background: Runge-Kutta Method of Order 4

1. Given initial value problem $y' = f(t, y)$, $y(t_0) = y_0$
2. Given the current step y_n and t_n
3. Runge-Kutta method is given by:

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1) \\ k_3 &= f(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_2) \\ k_4 &= f(t_n + h, y_n + hk_3) \\ y_{n+1} &= y_n + \frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4) \end{aligned}$$

4. y_{n+1} and t_{n+1} are the new step

Background: Runge-Kutta Method of Order 2

1. Given initial value problem $y' = f(t, y)$, $y(t_0) = y_0$
2. Given the last step y_n and t_n
3. Runge-Kutta method is given by:

$$\begin{aligned} k_1 &= f(t_n, y_n) \\ k_2 &= f(t_n + h, y_n + hk_1) \\ y_{n+1} &= y_n + \frac{1}{2}h(k_1 + k_2) \end{aligned}$$

4. y_{n+1} and t_{n+1} are the new step

Conservation of Linear Invariants

Theorem

All explicit and implicit Runge-Kutta methods conserve linear invariants.

A Quadratic Invariant

Theorem

Consider ODE of the form

$$\dot{Y} = A(Y)Y,$$

if $A(Y)$ *is skew-symmetric for all* Y *(i.e.,* $A^T = -A$*), then the quadratic function* $I(Y) = Y^T Y$ *is an invariant.*

Quadratic Invariants

- Consider ODE $\dot{y} = f(y)$ and quadratic functions

$$Q(y) = y^T C y,$$

- $Q(y)$ is an invariant if $y^T C f(y) = 0$ for all y.

Quadratic Conservation of Runge-Kutta Methods

Theorem

If the coefficients of a Runge-Kutta method satisfy

$$b_i a_{ij} + b_j a_{ji} = b_i b_j \quad \text{for all } i, j = 1, \ldots, s,$$

then it conserves quadratic invariants.

Invariants on a Sub-manifold

- Suppose an $(n-m)$-dimensional sub-manifold of $\mathbb{R}^n$,

$$\mathcal{M} = \{y;\, g(y) = 0\} \tag{1}$$

  ($g : \mathbb{R}^n \to \mathbb{R}^m$),
- And a differential equation $\dot{y} = f(y)$ with the property that

$$y_0 \in \mathcal{M} \quad \text{implies} \quad y(t) \in \mathcal{M} \quad \text{for all } t.$$

- It holds if $g'(y)f(y) = 0$ for $y \in \mathcal{M}$.
- We call $g(y)$ a *weak invariant.*

Algorithm of Projection Methods

Here the Standard Projection Methods are proposed as following:

Algorithm

Assume that $y_n \in \mathcal{M}$. *One step* $y_n \rightarrow y_{n+1}$ *is defined as follows:*

Algorithm of Projection Methods

Here the Standard Projection Methods are proposed as following:

Algorithm

Assume that $y_n \in \mathcal{M}$*. One step* $y_n \to y_{n+1}$ *is defined as follows:*

1. *Compute* $\tilde{y}_{n+1} = \Phi_h(y_n)$*, where* Φ_h *is an arbitrary one-step method applied to* $\dot{y} = f(y)$*;*

Algorithm of Projection Methods

Here the Standard Projection Methods are proposed as following:

Algorithm

Assume that $y_n \in \mathcal{M}$. One step $y_n \to y_{n+1}$ is defined as follows:

1. *Compute $\tilde{y}_{n+1} = \Phi_h(y_n)$, where Φ_h is an arbitrary one-step method applied to $\dot{y} = f(y)$;*
2. *project the value $\tilde{y}_{n+1}$ onto the manifold $\mathcal{M}$ to obtain $y_{n+1} \in \mathcal{M}$.*

Background: Newton Iterations

- Given a function $f(x)$ and its derivative $f'(x)$
- Want to solve an equation $f(x) = 0$
- Repeat

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

until sufficiently accurate

A Minimization Problem

In order to project $\tilde{y}_{n+1}$ onto $\mathcal{M}$ to obtain y_{n+1}, one needs to

- Solve the constrained minimization problem

$$||y_{n+1} - \tilde{y}_{n+1}||_{L^2} \to \text{min} \quad \text{subject to } g(y_{n+1}) = 0.$$

A Minimization Problem

In order to project $\tilde{y}_{n+1}$ onto $\mathcal{M}$ to obtain y_{n+1}, one needs to

- Solve the constrained minimization problem

$$||y_{n+1} - \tilde{y}_{n+1}||_{L^2} \to \text{min} \quad \text{subject to } g(y_{n+1}) = 0.$$

- Introduce Lagrange multipliers $\lambda = (\lambda_1, \ldots, \lambda_m)^T$, and the Langrange function

$$\mathcal{L}(y_{n+1}, \lambda) = ||y_{n+1} - \tilde{y}_{n+1}||^2/2 - g(y_{n+1})^T \lambda.$$

Langrange multiplier

- The necessary condition leads to the system

$$\begin{aligned} y_{n+1} &= \tilde{y}_{n+1} + g'(\tilde{y}_{n+1})^T \lambda \\ 0 &= g(y_{n+1}). \end{aligned}$$

- Inserting the first equation into the second gives

$$g\left(\tilde{y}_{n+1} + g'(\tilde{y}_{n+1})^T \lambda\right) = 0$$

- Solve the equation for λ with simplified Newton Method:

$$\Delta\lambda_i = -\left(g'(\tilde{y}_{n+1}) g'(\tilde{y}_{n+1})^T\right)^{-1} g\left(\tilde{y}_{n+1} + g'(\tilde{y}_{n+1})^T \lambda_i\right),$$

with $\lambda_{i+1} = \lambda_i + \Delta\lambda_i$.

Perturbed Kepler Problem

- Consider the perturbed Kepler problem

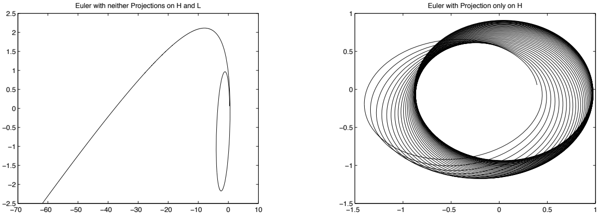
$$\dot{p} = -H_q(p,q), \quad \dot{q} = H_p(p,q),$$

with the Hamilton function

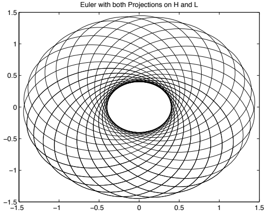
$$H(p,q) = \frac{1}{2}(p_1^2 + p_2^2) - \frac{1}{\sqrt{q_1^2 + q_2^2}} - \frac{0.005}{2\sqrt{(q_1^2 + q_2^2)^3}}$$

- Initial values: $q_1(0) = 1 - e$, $q_2(0) = 0$, $p_2(0) = \sqrt{(1+e)/(1-e)}$, $p_1(0) = 0$ ($e = 0.6$ as the eccentricity)
- Two known first integrals:
  - the Hamilton function $H(p,q)$
  - the angular momentum $L(p,q) = q_1 p_2 - q_2 p_1$

Explicit Euler



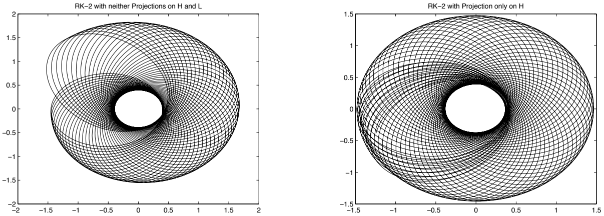
(a) Without Projection (b) With Projection onto H



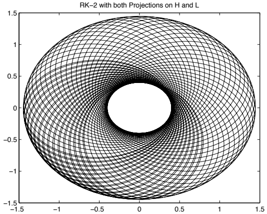
(c) With Projection onto H and L

Figure: Explicit euler, $h = 0.03$

Runge-Kutta of Order 2



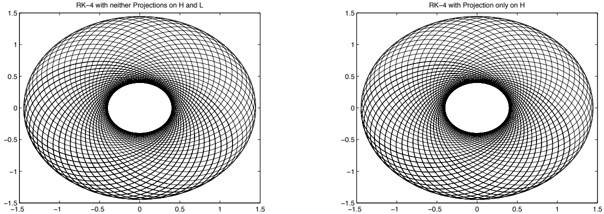
(a) Without Projection (b) With Projection onto H



(c) With Projection onto H and L

Figure: RK2, $h = 0.03$

Runge-Kutta of Order 4

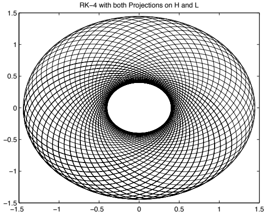


RK–4 with neither Projections on H and L

RK–4 with Projection only on H

(a) Without Projection

(b) With Projection onto H



RK–4 with both Projections on H and L

(c) With Projection onto H and L

Figure: RK4, $h = 0.03$

Concluding Remarks

- A lot of ODEs have the property to conserve the *First Integrals*
- Certain numerical methods conserve the first integrals automatically
- If not, apply projection methods to conserve it manually
- Projection methods improve the numerical results
- Projections are especially effective for low order methods

Questions?

Thanks for your attention.