CLASSICAL MECHANICS
DISCRETE MECHANICS
VARIATIONAL INTEGRATORS
STABILITY AND CONVERGENCE
FORCED MECHANICS
CODE ORGANIZATION
EXAMPLE

SPECTRAL VARIATIONAL INTEGRATORS

Roberto Porcù

Mattia Penati

CLASSICAL MECHANICS

Lagrangian point of view		Hamiltonian point of view
$\mathbf{q}(t) = \{q_i(t)\}_{i=1}^N$ $\dot{\mathbf{q}}(t) = \{\dot{q}(t)\}_{i=1}^N$	generalized variables	$\mathbf{q}(t) = \{q_i(t)\}_{i=1}^N$ $\mathbf{p}(t) = \frac{\partial L}{\partial \dot{\mathbf{q}}}(t)$
$L(\mathbf{q}, \dot{\mathbf{q}}, t)$	characteristic quantity	$H(\mathbf{q}, \mathbf{p}, t) = \mathbf{p} \cdot \dot{\mathbf{q}} - L(\mathbf{q}, \dot{\mathbf{q}}, t)$

Hamilton's principle

$$\mathcal{S}(\mathbf{q}) = \int_a^b L(\mathbf{q}, \dot{\mathbf{q}}, t)dt \qquad \delta\mathcal{S}(\mathbf{q}) = 0 \qquad \left.\begin{aligned} \mathbf{q}(a) &= \mathbf{q}_1 \\ \mathbf{q}(b) &= \mathbf{q}_2 \end{aligned}\right\} \text{ fixed}$$

equations of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} = 0$$

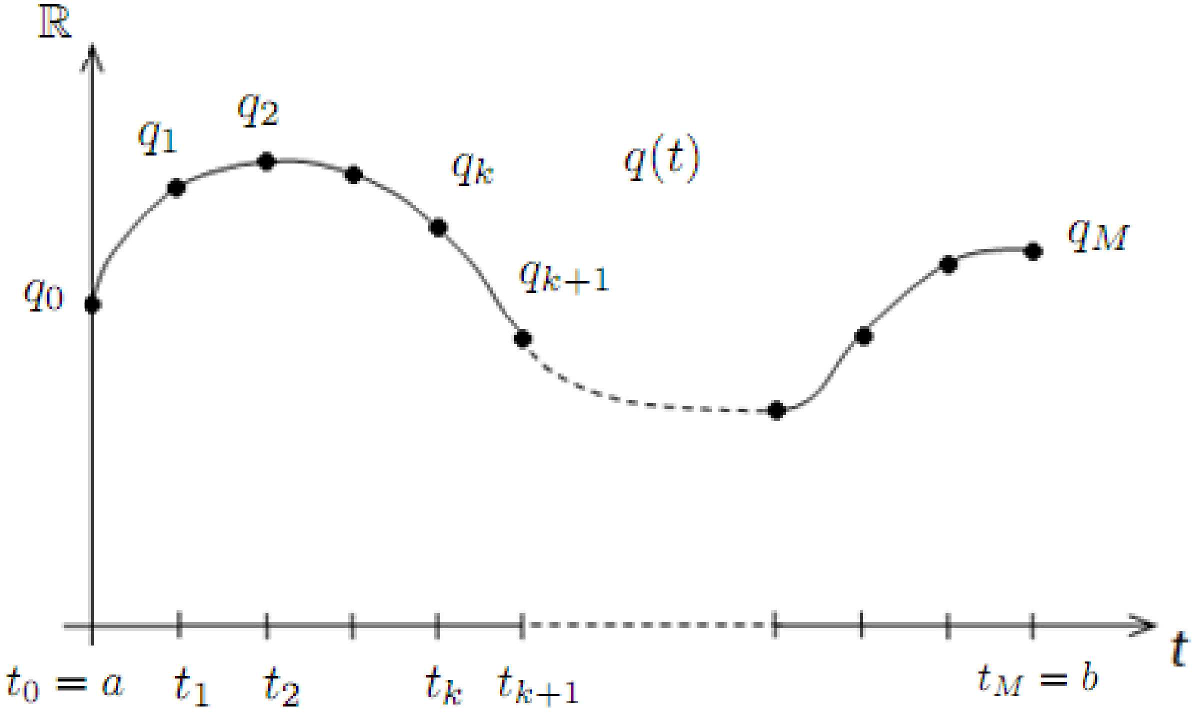
2nd order

$$\begin{cases} \dot{\mathbf{q}} = \dfrac{\partial H}{\partial \mathbf{p}} \\ \dot{\mathbf{p}} = -\dfrac{\partial H}{\partial \mathbf{q}} \end{cases}$$

1st order

DISCRETE MECHANICS

$$\{t_k = kh\}_{k=0}^{M}$$



DISCRETE MECHANICS

Lagrangian point of view		Hamiltonian point of view
$\mathbf{q}_d = \{\mathbf{q}_k\}_{k=0}^{M}$	generalized variables	$\mathbf{p}_k^- = -D_1 L_d(\mathbf{q}_k, \mathbf{q}_{k+1})$ $\mathbf{p}_k^+ = D_2 L_d(\mathbf{q}_k, \mathbf{q}_{k+1})$
$L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) \approx \int_{t_k}^{t_{k+1}} L(\mathbf{q}, \dot{\mathbf{q}}; t)\, dt$	characteristic quantity	$H_d(\mathbf{p}_k^-, \mathbf{p}_k^+) = \mathbf{p}_k^+ \mathbf{q}_{k+1} - \mathbf{p}_k^- \mathbf{q}_k - L_d(\mathbf{q}_k, \mathbf{q}_{k+1})$

Hamilton's principle

$$S_d(\{\mathbf{q}_k\}_{k=0}^{M}) = \sum_{k=0}^{M-1} L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) \qquad \delta S_d(\{\mathbf{q}_k\}_{k=0}^{M}) = 0 \qquad \left.\begin{array}{l} \mathbf{q}(a) = \mathbf{q}_1 \\ \mathbf{q}(b) = \mathbf{q}_2 \end{array}\right\} \text{fixed}$$

equations of motion

$$D_2 L_d(\mathbf{q}_{k-1}, \mathbf{q}_k) + D_1 L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) = 0$$

2 steps

$$\begin{cases} \mathbf{p}_k = -D_1 L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) \\ \mathbf{p}_{k+1} = D_2 L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) \end{cases}$$

1 step

Discrete Liouville's Theorem

The Hamiltonian map $(q_k, p_k) \mapsto (q_{k+1}, p_{k+1})$ defined by discrete Hamilton's equations preserves volume in discrete phase space (simplecticity).

Discrete Noether's Theorem

If the discrete Lagrangian is invariant under the action of a group G, then the corresponding discrete Lagrangian momentum map is a conserved quantity.

translation → linear momentum

rotation → angular momentum

Variational error analysis

If L_d is a discrete Lagrangian of order *p* then the Hamiltonian map has the same order.

trapezoids → stormer-verlet

second order

VARIATIONAL INTEGRATORS

Variational integrators differ from each other for the quadrature rule used to approximate the action.

the order of the method is equal to the quadrature order

1) Symplectic Euler

$$L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) = hL\left(\mathbf{q}_k, \frac{\mathbf{q}_{k+1} - \mathbf{q}_k}{h}\right)$$

2) Midpoint Rule

$$L_d(\mathbf{q}_k, \mathbf{q}_{k+1}, h) = hL\left(\frac{\mathbf{q}_k + \mathbf{q}_{k+1}}{2}, \frac{\mathbf{q}_{k+1} - \mathbf{q}_k}{h}\right)$$

3) Stormer-Verlet

$$L_d(\mathbf{q}_k, \mathbf{q}_{k+1}) = \frac{1}{2}hL\left(\mathbf{q}_k, \frac{\mathbf{q}_{k+1} - \mathbf{q}_k}{h}\right) + \frac{1}{2}hL\left(\mathbf{q}_{k+1}, \frac{\mathbf{q}_{k+1} - \mathbf{q}_k}{h}\right)$$

CLASSICAL MECHANICS
DISCRETE MECHANICS
VARIATIONAL INTEGRATORS
STABILITY AND CONVERGENCE
FORCED MECHANICS
CODE ORGANIZATION
EXAMPLE

4) Spectral Variational Integrators

For simplicity let $q(t) \in \mathbb{R}$ with $t \in [t_k, t_{k+1}]$, $t_{k+1} - t_k = h$.

Rescaled problem: $q = q(z(t))$ $\quad z(t) = \frac{2}{h}t - 1$ $\quad z \in [-1, 1]$

Spatial discretization $\quad q_n(z(t)) = \sum_{i=0}^{n-1} q_k^i l_i(z(t))$ $\quad \dot{q}_n(z(t)) = \sum_{i=0}^{n-1} q_k^i \dot{l}_i(z(t)) \frac{dz}{dt}$

Gauss quadrature rule

$$\int_{t_k}^{t_{k+1}} L(q(t), \dot{q}(t))dt = \int_{-1}^{1} L(q(z(t)), \dot{q}(z(t)))\frac{h}{2}dz \approx \frac{h}{2}\sum_{j=0}^{m-1} \omega_j L(q(t_j), \dot{q}(t_j))$$

Hamilton's principle

$$\operatorname*{ext}_{q_n \in V([t_k, t_{k+1}];\mathbb{R})} \frac{h}{2}\sum_{j=0}^{m-1} \omega_j L\left(\sum_{i=0}^{n-1} q_k^i l_i(t_j), \frac{2}{h}\sum_{i=0}^{n-1} q_k^i \dot{l}_i(t_j)\right)$$

with constraints: $\quad q_k = \sum_{i=0}^{n-1} q_k^i l_i(-1)$ $\quad q_{k+1} = \sum_{i=0}^{n-1} q_k^i l_i(1)$

impose constraints with Lagrangian multipliers

$$L_d^\lambda(q_k^0,\dots,q_k^{n-1},\lambda^0,\lambda^h) = \frac{h}{2}\sum_{j=0}^{m-1}\omega_j L\left(\sum_{i=0}^{n-1} q_k^i l_i(t_j), \frac{2}{h}\sum_{i=0}^{n-1} q_k^i \dot{l}_i(t_j)\right) + \dots$$

$$\dots + \lambda^0\left(\sum_{i=0}^{n-1} q_k^i l_i(-1) - q_k\right) + \lambda^h\left(\sum_{i=0}^{n-1} q_k^i l_i(1) - q_{k+1}\right)$$

$$\Rightarrow \begin{cases} 0 = \dfrac{\partial L_d^\lambda}{\partial q_k^s} \\ 0 = \dfrac{\partial L_d^\lambda}{\partial \lambda^0} \\ 0 = \dfrac{\partial L_d^\lambda}{\partial \lambda^h} \end{cases}$$

+

$$\dot{q} = \frac{\partial H}{\partial p}(q,p) \Rightarrow \frac{\partial H}{\partial p}\left(\sum_i q_k^i l_i(t_j), p_j\right) = \frac{2}{h}\sum_i q_k^i \dot{l}_i(t_j)$$

finally we obtain the following nonlinear system:

$$\begin{cases} \displaystyle\sum_{j=0}^{m-1}\omega_j\left[p_j \dot{l}_s(t_j) - \frac{h}{2} l_s(t_j)\frac{\partial H}{\partial q}\left(\sum_{i=0}^{n-1} q_k^i l_i(t_j), p_j\right)\right] + l_s(-1)p_k - l_s(1)p_{k+1} = 0 & \forall s = 0,\dots,n-1 \\ \displaystyle\frac{\partial H}{\partial p}\left(\sum_{i=0}^{n-1} q_k^i l_i(t_j), p_j\right) - \frac{2}{h}\sum_{i=0}^{n-1} q_k^i \dot{l}_i(t_j) = 0 & \forall j = 0,\dots,m-1 \\ \displaystyle\sum_{i=0}^{n-1} q_k^i l_i(-1) - q_k = 0 \\ \displaystyle\sum_{i=0}^{n-1} q_k^i l_i(1) - q_{k+1} = 0 \end{cases}$$

n+m+2 unknowns

Exixtence, uniqueness and convergence :

see Melvin Leok or Marsden and West

STABILITY and CONVERGENCE ANALYSIS

ARMONIC OSCILLATOR

equation $M\ddot{q}(t) + \omega^2 q(t) = 0$

exact solution $q(t) = \cos\left(\frac{\omega}{\sqrt{M}} t + \phi\right)$

Hamiltonian $H(q,p) = \frac{1}{2M}(p(t))^2 + \frac{1}{2}(\omega q(t))^2$

1) *STORMER-VERLET*

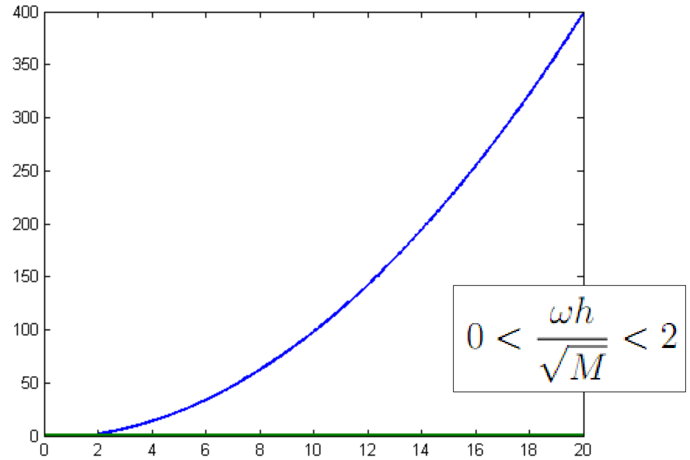
$$\begin{cases} p_k = M\frac{q_{k+1} - q_k}{h} + \frac{1}{2}h\omega^2 q_k \\ p_{k+1} = M\frac{q_{k+1} - q_k}{h} - \frac{1}{2}h^2\omega^2 q_{k+1} \end{cases}$$

$$\begin{bmatrix} q_{k+1} \\ p_{k+1} \end{bmatrix} = \underbrace{\begin{bmatrix} (1 - \frac{\omega^2 h^2}{2M}) & \frac{h}{M} \\ h\omega^2\left(\frac{h^2\omega^2}{4M} - 1\right) & (1 - \frac{h^2\omega^2}{2M}) \end{bmatrix}}_{\Omega} \begin{bmatrix} q_k \\ p_k \end{bmatrix}$$

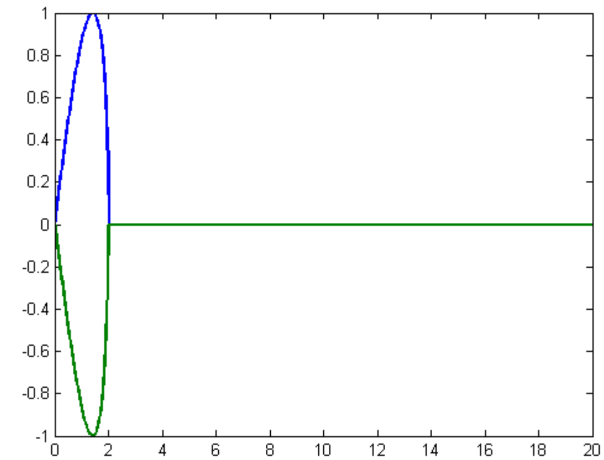
$$det(\Omega) = 1$$

$$tr(\Omega) = 2 - \frac{\omega^2 h^2}{M}$$

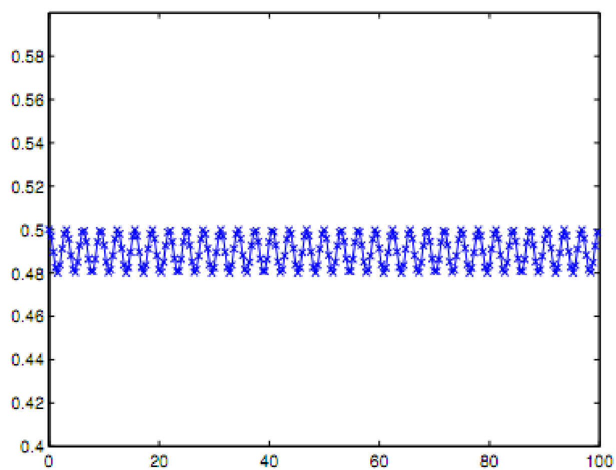
eigenvalues moduli



eigenvalues imaginary parts



energy



h	$\|\mathbf{e}\|_{l^\infty}$	order
0.8	$0.22 \cdot 10^{0}$	
0.4	$0.54 \cdot 10^{-1}$	2.05
0.2	$0.13 \cdot 10^{-1}$	2.01
0.1	$0.33 \cdot 10^{-2}$	2.00
0.05	$0.82 \cdot 10^{-3}$	2.00
0.025	$0.21 \cdot 10^{-3}$	2.00

2) *SPECTRAL VARIATIONAL INTEGRATORS*

$$S_d^\lambda(\{q_k^i\}_{i=0}^{n-1}) = \sum_{j=1}^{m}\left[\frac{M\alpha_j}{h}\left(\sum_i q_k^i \dot{l}_i(z_j)\right)^2 - \frac{h\omega^2\alpha_j}{4}\left(\sum_i q_k^i l_i(z_j)\right)^2\right] + \dots$$

$$\dots + \lambda^0\left(q_k - \sum_i q_k^i l_i(-1)\right) + \lambda^h\left(q_{k+1} - \sum_i q_k^i l_i(1)\right)$$

$$\mathbf{q} = [q_k^0, \dots, q_k^{n-1}]^T$$

$$\boldsymbol{\lambda} = [\lambda^0, \lambda^h]^T$$

$$\mathbf{g} = \begin{bmatrix} q_k \\ q_{k+1} \end{bmatrix}$$

$$\begin{bmatrix} A & B \\ B^T & 0 \end{bmatrix}\begin{bmatrix} \mathbf{q} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{g} \end{bmatrix}$$

$$A_{ij} = \sum_{s=1}^{m}\left[\frac{2M\alpha_s}{h}\dot{l}_i(z_s)\dot{l}_j(z_s) - \frac{h\omega^2\alpha_s}{2}l_i(z_s)l_j(z_s)\right]$$

$$B_{i1} = -l_i(-1) \qquad B_{i2} = -l_i(1)$$

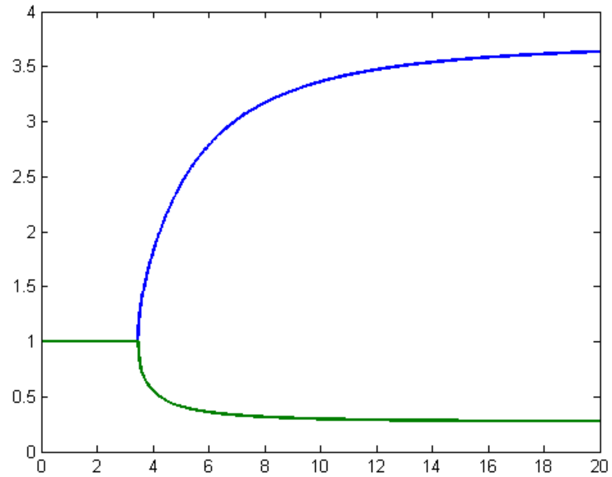
$$C = (B^T A^{-1} B)^{-1}$$

$$\begin{bmatrix} q_{k+1} \\ p_{k+1} \end{bmatrix} = \underbrace{-\frac{1}{C_{12}}\begin{bmatrix} C_{11} & 1 \\ det(C) & C_{22} \end{bmatrix}}_{\Omega}\begin{bmatrix} q_k \\ p_k \end{bmatrix}$$

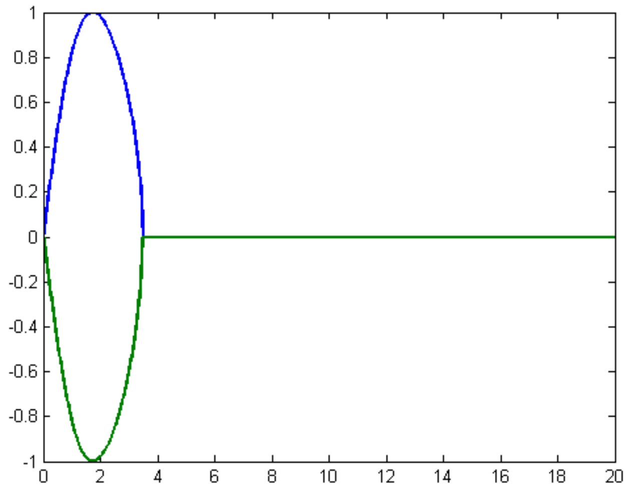
n = maximum degree of basis polynomials
m = number of quadrature nodes

In case of *n=1, m=2*: $0 < \frac{h\omega}{\sqrt{M}} < 2\sqrt{3}$.

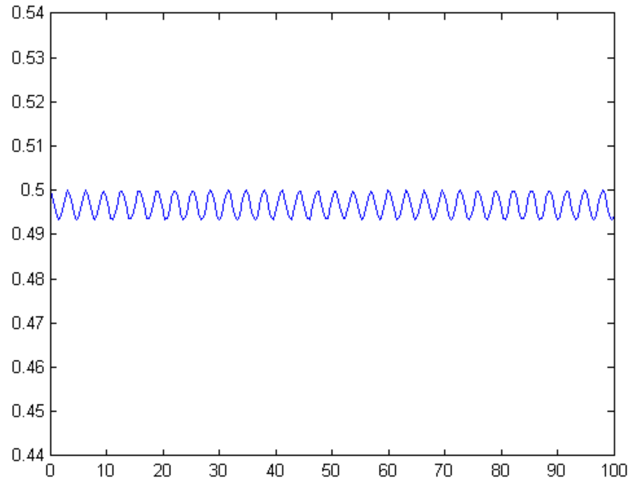
eigenvalues moduli



eigenvalues imaginary parts

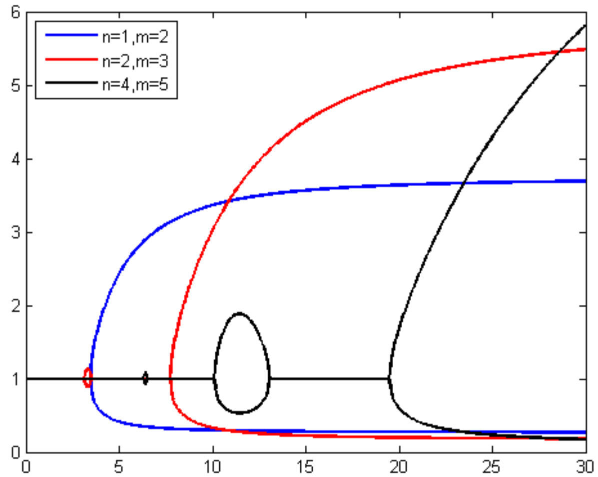


energy

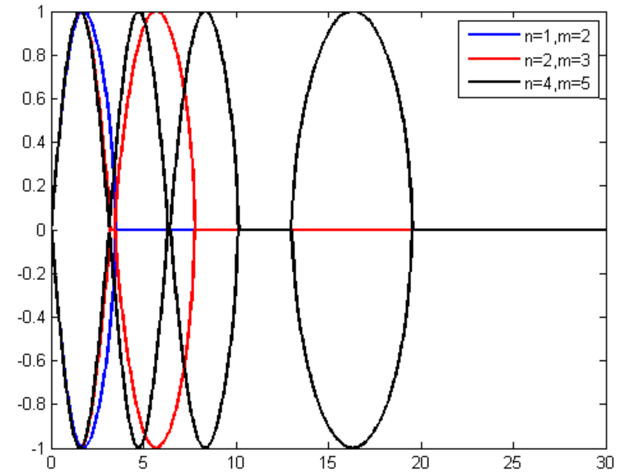


Comparison between *(n=1, m=2), (n=2,m=3), (n=4,m=5)*:

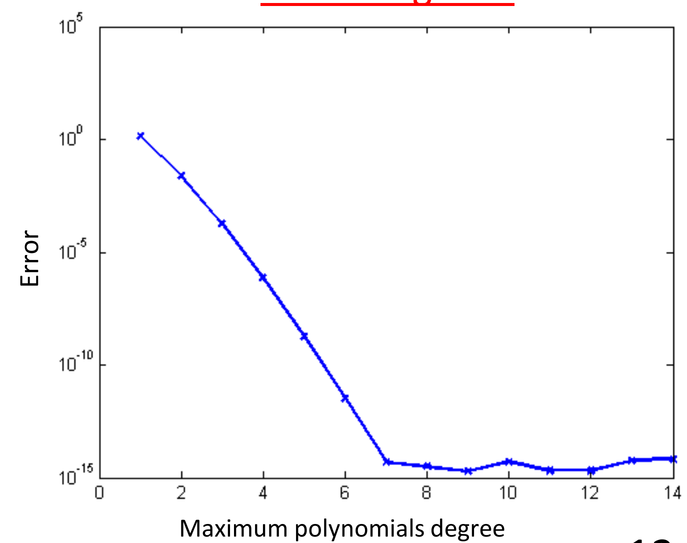
eigenvalues moduli



eigenvalues imaginary parts



n-convergence

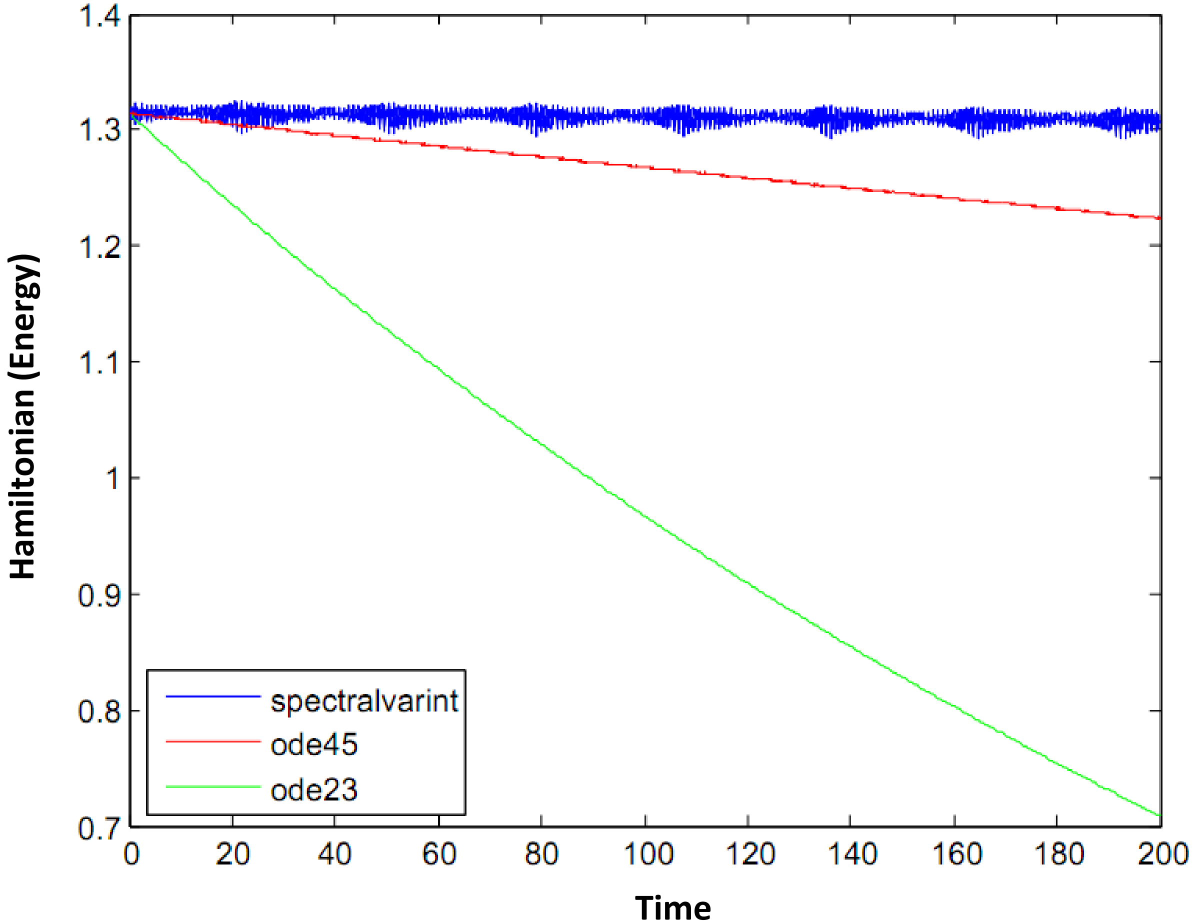


convergence orders

Degree:1 Nodes:2			Degree:2 Nodes:3			Degree:4 Nodes:5		
h	error	order	h	error	order	h	error	order
0.8	$0.43{\cdot}10^{0}$		1.6	$0.20{\cdot}10^{-1}$		3.2	$0.13{\cdot}10^{-4}$	
0.4	$0.12{\cdot}10^{0}$	1.90	0.8	$0.15{\cdot}10^{-2}$	3.70	1.6	$0.54{\cdot}10^{-7}$	7.93
0.2	$0.30{\cdot}10^{-1}$	1.94	0.4	$0.97{\cdot}10^{-4}$	3.98	0.8	$0.23 \cdot 10^{-9}$	7.86
0.1	$0.72{\cdot}10^{-2}$	2.06	0.2	$0.62{\cdot}10^{-5}$	3.95	0.4	$0.95{\cdot}10^{-12}$	7.96

ARMONIC OSCILLATOR

Spectral Variational Integrators do not artificially dissipate energy



FORCED MECHANICS

Lagrangian point of view

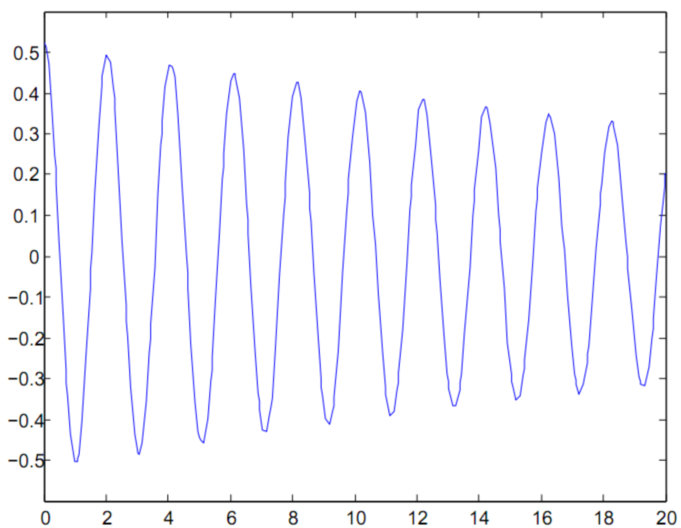
Hamiltonian point of view

Lagrange-d'Alembert principle

$$\delta \int_0^T L(\mathbf{q}, \dot{\mathbf{q}})dt \; + \int_0^T f_L(\mathbf{q}, \dot{\mathbf{q}}) \cdot \delta \mathbf{q}(t) dt = 0$$

$$\frac{\partial L}{\partial \mathbf{q}}(\mathbf{q}, \dot{\mathbf{q}}) - \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}(\mathbf{q}, \dot{\mathbf{q}})\right) + f_L(\mathbf{q}, \dot{\mathbf{q}}) = 0$$

$$\begin{cases} \dot{\mathbf{q}} = \dfrac{\partial H}{\partial \mathbf{p}}(\mathbf{q}, \mathbf{p}) \\ \dot{\mathbf{p}} = -\dfrac{\partial H}{\partial \mathbf{q}}(\mathbf{q}, \mathbf{p}) + f_H(\mathbf{q}, \mathbf{p}) \end{cases}$$



Example
Simple pendulum dumped with friction-type forcing.

CODE ORGANIZATION

function signature recalls the same style of ode45 function

Integrates the system with Hamilton's equations given by *odefun* from time *tspan(1)* to *tspan(end)*, with initial condition *y0.* To obtain solutions at specific times use *tspan* = [t_0, t_1, ..., t_f].

default parameters (options, options_B)

[T, Y] = spectralvarint (odefun, tspan, y0, options, options_B)

options_B:

- polynomial degree
- quadrature order
- nonlinear solutor parameters

CODE ORGANIZATION

- spectralvarint
  - initial_step — *calculates the initial step on the basis of the function and of the initial condition.*
  - **class** @spectral_integrator
    - spectral_integrator — *class constructor; It sets polynomial order, quadrature order and computes quadrature nodes and weights.*
    - set_problem — *assigns the function with Hamiltonian's equations to the class.*
    - step — *calculates the solution of the system from time t_k to time t_{k+1}.*

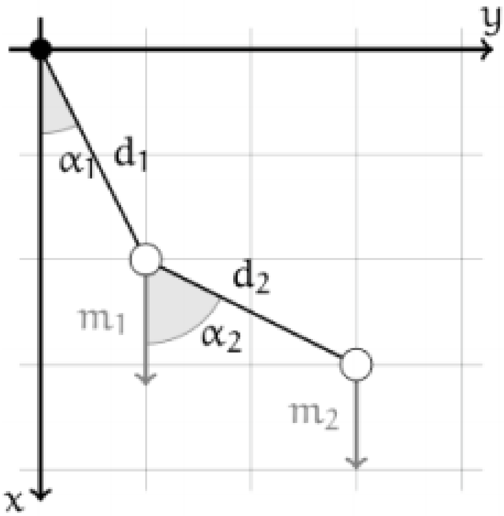
MAIN PROGRAM

object oriented programming → few rows in main program

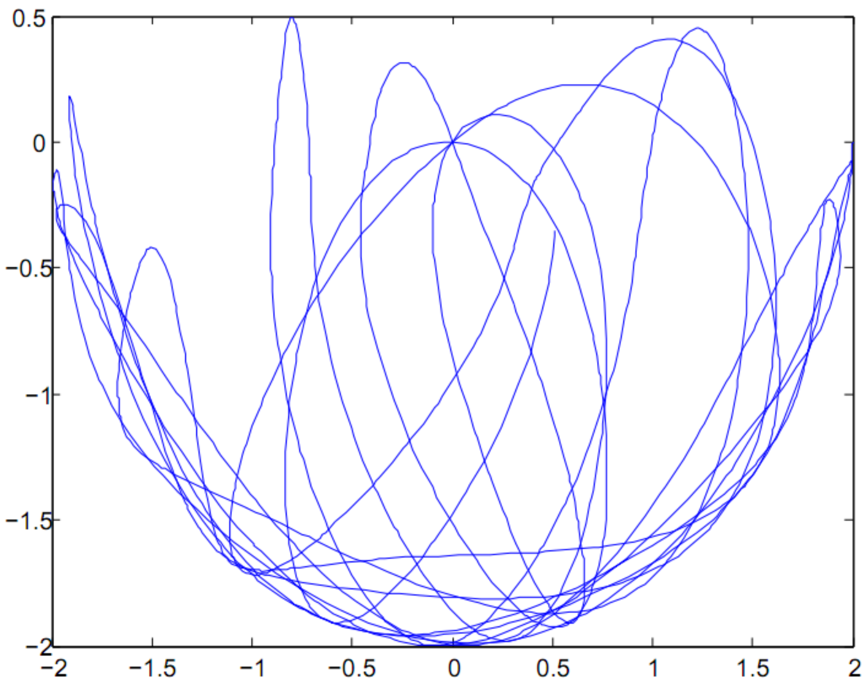
```
% solving on h=2*dt into two steps with h=dt
[X1,t1,Y1] = step(mysolver,time,time+dt,y0);
[X2,t1,Y1] = step(mysolver,time+dt,time+2*dt,X1);

% solving on h=2*dt into one step
[W,t1,Y1] = step(mysolver,time,time+2*dt,y0);

% calculus of h_optimal
hopt =  dt*(2.0*(2^order+1.0)*toll/norm(W-X2))^(1/(order+1));
dt = hopt;

dt_extra = (time+hopt)-tspan(iter);
if(dt_extra > 0)
  % adjusting hopt for this exception
  hopt = tspan(iter)-time;
end

% solving with h = h_optimal
[y0,tt,yy] = step(mysolver,time,time+hopt,y0);
```

EXAMPLE

DOUBLE PENDULUM



$$\begin{cases} T = \dfrac{m_1 + m_2}{2} d_1^2 \dot{\alpha}_1^2 + \dfrac{m_2}{2} d_2^2 \dot{\alpha}_2^2 + m_2 d_1 d_2 \dot{\alpha}_1 \dot{\alpha}_2 \cos(\alpha_1 - \alpha_2) \\ U = -(m_1 + m_2) g d_1 \cos(\alpha_1) - m_2 g d_2 \cos(\alpha_2) \end{cases}$$



TODO

- Add the possibility to have a quadrature nodes number indipendent from maximum polynomials degree;
- Add the possibility to use the Jacobian in the solution of the nonlinear system;
- Add the possibility to do polynomials degree adaptivity;
- Optimize the code; especially It would be very interesting to implement in C++ the expensive parts of the code (now is all implemented in Octave).

CLASSICAL MECHANICS
DISCRETE MECHANICS
VARIATIONAL INTEGRATORS
STABILITY AND CONVERGENCE
FORCED MECHANICS
CODE ORGANIZATION
EXAMPLE

THANKS