The Restricted Hill's Problem (and the Power Series Method)

JMM 2024 in San Francisco

Roger Thelwell* & Stefano Colafranceschi

James Madison University & Eastern Mennonite University

January 6, 2024

Outline

Abstract

We present an approach to recast non-polynomial IVODEs into polynomial differential systems which exploit auxiliary variables, and provide several examples using the Power Series Method (PSM). We review an *a priori* error bound for the solution to IVODES of polynomial form. We apply this approach to Hill's Restricted Three Body Problem of celestial mechanics to demonstrate that PSM is effectively symplectic. If we have time, we will offer some novel conservation expressions for this system, and a general approach to generate others.

Act I:
Power series

$y' = \alpha y^2$

We start with a toy problem...

$$y' = \alpha y^2 \qquad y(0) = y_0$$

$y' = \alpha y^2$

We start with a toy problem...

$$y' = \alpha y^2 \qquad y(0) = y_0$$

An analytic solution to this one is easy:

$$y(t) = -\frac{y_0}{\alpha y_0 t - 1}$$

$y' = \alpha y^2$: series solution

What happens if we try (formal, for now) series?

$y' = \alpha y^2$: series solution

What happens if we try (formal, for now) series?
Let

$$y(t) = \sum_{k=0}^{\infty} y_k t^k$$

then

$$\sum_{k=0}^{\infty} (k+1) y_{k+1} t^k = \alpha \sum_{k=0}^{\infty} \left(\sum_{i,j \geq 0}^{i+j=k} y_i y_j \right) t^k$$

so we equate coefficients to get

$$y_{k+1} = \frac{\alpha}{k+1} \sum_{i+j=k} y_i y_j$$

$y' = \alpha y^2$: Maple

Or, with MAPLE:

```
> ODE1 := diff(y(t),t) = alpha*y(t)^2;
> IC := y(0) = y0;
> y1 := dsolve({ODE1,IC},y(t))
```

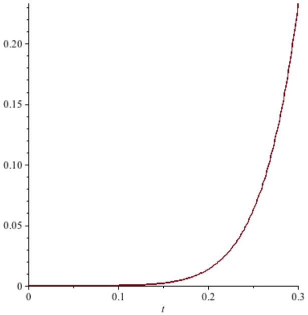
and

```
> Y1 := dsolve({ODE1,IC},y(t),series);
       Y1 := 1+2*alpha*t*y0+3*alpha^2*y0^2*t^2+...
```

$y' = \alpha y^2$: Error

What about the error?

```
> alpha := 1; y0 := 2; plot(abs(Y1-y1),t=0..0.5);
```

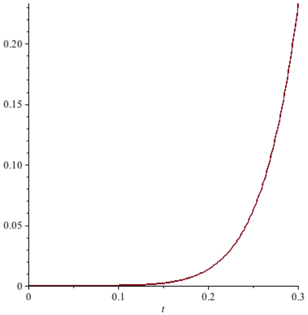


Can we quantify this?

$y' = \alpha y^2$: Error

What about the error?

```
> alpha := 1; y0 := 2; plot(abs(Y1-y1),t=0..0.5);
```



Can we quantify this?
Let's take a little diversion...

$y' = \alpha y^m$: important aside

The solution to the constant coefficient nonlinear IVODE

$$y' = \alpha y^m \qquad y(0) = y_0$$

is messy:

$$y(t) = \left((\alpha - \alpha m)t + y_0^{1-m}\right)^{-(m-1)^{-1}}.$$

$y' = \alpha y^m$: important aside

The solution to the constant coefficient nonlinear IVODE

$$y' = \alpha y^m \qquad y(0) = y_0$$

is messy:

$$y(t) = \left((\alpha - \alpha m)t + y_0^{1-m}\right)^{-(m-1)^{-1}}.$$

But the ratio $\frac{y'}{y}$ isn't!

$y' = \alpha y^m$: important aside

The solution to the constant coefficient nonlinear IVODE

$$y' = \alpha y^m \qquad y(0) = y_0$$

is messy:

$$y(t) = \left((\alpha - \alpha m)t + y_0^{1-m}\right)^{-(m-1)^{-1}}.$$

But the ratio $\frac{y'}{y}$ isn't!

$$\frac{y'}{y} = \frac{\alpha}{(\alpha - \alpha m)t + y_0^{1-m}}$$

and we see that

$$y'(t) = \underbrace{\frac{\alpha}{(\alpha - \alpha m)t + y_0^{1-m}}}_{K(t)} y(t),$$

is a non-constant coefficient LINEAR ode.

$y' = \alpha y^m$: important aside

So

$$y'(t) = \underbrace{\frac{\alpha}{(\alpha - \alpha t m)t + y_0^{1-m}}}_{K(t)} y(t),$$

has solution

$$y(t) = y_0 exp\left(\int_0^t K(\tau) d\tau\right)$$

or, via series,

$$Y_{k+1} = \frac{\alpha(1 + (m-1)k)}{y_0^{1-m}(k+1)} Y_k$$

$y' = \alpha y^m$: important aside

From

$$Y_{k+1} = \frac{\alpha(1 + (m-1)k)}{y_0^{1-m}(k+1)} Y_k$$

and for $m \geq 2$,

$$Y_{k+1} \leq \alpha(m-1)|y_0|^{m-1} Y_k \leq \alpha C_\infty Y_k.$$

This leads directly to a geometric series bounding $y(t)$:

$$y(t) \leq \frac{|y_0 \alpha|}{1 - C_\infty t} = |y_0 \alpha| \sum_{k=0} (C_\infty t)^k$$

Now for the bound...

$y' = \alpha y^2$: back to error

From

$$y(t) \le \frac{|y_0\alpha|}{1 - C_\infty t} = |y_0\alpha| \sum_{k=0} (C_\infty t)^k$$

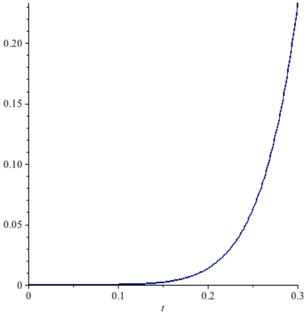
we see that the absolute error is

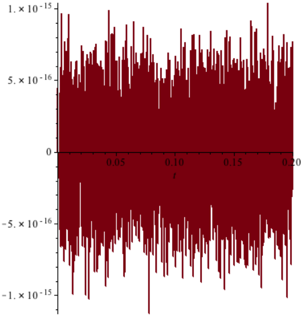
$$|Y1 - y1| \le |y_0\alpha| \sum_{k=n+1}^{\infty} C_\infty |t|^k \le \frac{|y_0\alpha| C_\infty^{n+1}}{1 - C_\infty |t|}$$

where $C_\infty = |y_0\alpha|$.
An ERROR bound!

$y' = \alpha y^2$: error plots

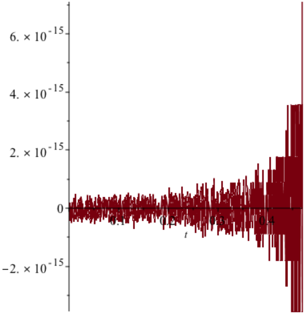
```
> ee :=  abs(Y1-y1);
> m := 2; Cinf := y0*alpha;
> EE := N -> abs(y0)*(Cinf*t)^(N+1)/( 1 - Cinf*abs(t));
> plot({ee,EE(5)},t=0..0.3);
> plot({ee-EE(5)},t=0..0.2);
```





$y' = 2y^2, \quad y(0) = 1$: radius of convergence?

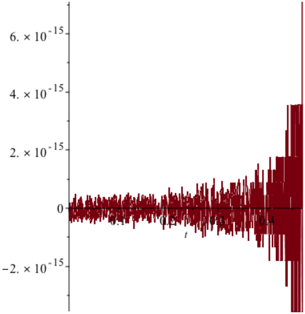
```
>plot({ee-EE(5)},t=0..0.48);
```



Hmmmm....

$y' = 2y^2, \quad y(0) = 1$: radius of convergence?

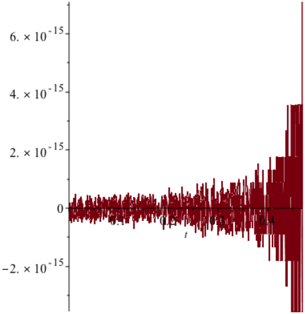
```
>plot({ee-EE(5)},t=0..0.48);
```



Hmmmm.... 💡

$y' = 2y^2, \quad y(0) = 1$: radius of convergence?

```
>plot({ee-EE(5)},t=0..0.48);
```



Hmmmm....

$$y(t) = -\frac{y_0}{\alpha y_0 t - 1} \qquad \text{with } \alpha = 2$$

$y' = \alpha y^2$: radius of convergence

But what if we only have this form?

```
> Y20 := dsolve({ODE1,IC},y(t),series);
       Y20 := 1+2*alpha*t*y0+3*alpha^2*y0^2*t^2+...
```

$y' = \alpha y^2$: radius of convergence

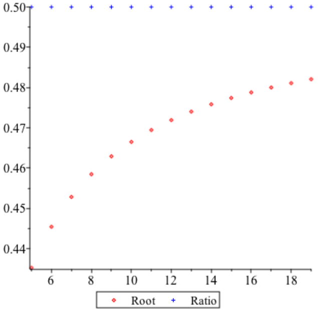
But what if we only have this form?

```
> Y20 := dsolve({ODE1,IC},y(t),series);
      Y20 := 1+2*alpha*t*y0+3*alpha^2*y0^2*t^2+...
```

Approximate using the root or ratio test. Or calculate the Padé approximant.

$y' = 2y^2, y(0) = 1$: radius of convergence?

```
> Order := 20: alpha := 2; y0 := 1;
> Y20 := rhs(dsolve({ODE1,IC},y(t),series)):
> Ycoeff := [seq(coeff(convert(Y20,polynom),t,i),i=1..20)];
> RatioT := i -> abs(a[i+1]/a[i]): RootT := i -> abs(a[i])^(1/
> RootEst := [seq(1/RootT(i),i=1..Order)]: RatioEst := [seq(1/
> plot ....
```



Act II
Hill's Lunar 3 body

Timeline

The Sun, the Earth, and the Moon

- 1000 B.C.E Babylonians
- 200 B.C.E Greeks
- (1543-1609) Copernicus to Kepler

Timeline

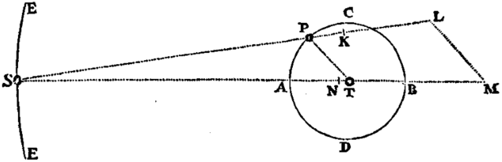
The Sun, the Earth, and the Moon

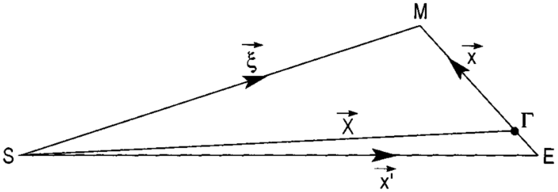
- 1000 B.C.E Babylonians
- 200 B.C.E Greeks
- (1543-1609) Copernicus to Kepler
- 1687 Newton (I.66)

Timeline

The Sun, the Earth, and the Moon

- 1000 B.C.E Babylonians
- 200 B.C.E Greeks
- (1543-1609) Copernicus to Kepler
- 1687 Newton (I.66)





Timeline

The Sun, the Earth, and the Moon

- 1000 B.C.E Babylonians
- 200 B.C.E Greeks
- (1543-1609) Copernicus to Kepler
- 1687 Newton (I.66)
- 1740-55 Clairaut, Euler, and d'Alembert
- 1767 Euler
- 1860-67 Delaunay
- 1877-1878 Hill

Hill's Lunar equations

Hill's 1877 "On the Part of the Motion of the Lunar Perigee which is a Function of the Mean Motions of the Sun and the Moon"

Key features:

- earth-centric coordinate frame
- sun massive, infinitely far away.
- moon as point mass in rotating coordinate, one axis towards sun.
- looked for a periodic orbit of a perturbed system
- not a systematic perturbation theory, but thoughtful expansion of variables

For much more detail, see survey by:

Martin C. Gutzwiller, Moon-Earth-Sun: The oldest three-body problem, Rev. Mod. Phys. 1998 [4]

Hill's Lunar equations

From [Waldvogel, 1997] in geocentric cartesian (x, y):

$$\ddot{x} - 2\dot{y} = 3x - xr^{-3} \tag{1}$$

$$\ddot{y} + 2\dot{x} = \quad + yr^{-3} \tag{2}$$

Or, in conjugate momenta (q, p), with
$q_1 = x, q_2 = y, p_1 = \dot{q}_1 - q_2,$ and $p_2 = \dot{q}_2 + q_1,$

$$\dot{q}_1 = p_1 + q_2 \tag{3}$$

$$\dot{q}_2 = p_2 - q_1 \tag{4}$$

$$\dot{p}_1 = \ p_2 + 2q_1 - q_1 r^{-3} \tag{5}$$

$$\dot{p}_2 = -p_1 - q_2 - q_2 r^{-3}, \tag{6}$$

Hill's Lunar equations

From [Waldvogel, 1997] in geocentric cartesian (x, y):

$$\ddot{x} - 2\dot{y} = 3x - xr^{-3} \quad (1)$$
$$\ddot{y} + 2\dot{x} = \quad + yr^{-3} \quad (2)$$

Or, in conjugate momenta (q, p), with
$q_1 = x, q_2 = y, p_1 = \dot{q}_1 - q_2,$ and $p_2 = \dot{q}_2 + q_1,$

$$\dot{q}_1 = p_1 + q_2 \quad (3)$$
$$\dot{q}_2 = p_2 - q_1 \quad (4)$$
$$\dot{p}_1 = \ p_2 + 2q_1 - q_1 r^{-3} \quad (5)$$
$$\dot{p}_2 = -p_1 - q_2 - q_2 r^{-3}, \quad (6)$$

with a conserved quantity $h = \frac{1}{2}\left(\dot{x}^2 + \dot{y}^2\right) - \frac{3}{2}x^2 - \frac{1}{r}, \quad r = \sqrt{x^2 + y^2}$ or
$H(q, p) = \frac{1}{2}(p_1^2 + p_2^2) + p_1 q_2 - p_2 q_1 - q_1^2 + \frac{1}{2}q_2^2 - \frac{1}{r}, \quad r = \sqrt{q_1^2 + q_2^2}.$

Symplectic integrator?

Powerful symplectic tools are available, such as:

- Martin Hairer's `GNI` package (link here)
- `SymInt` package (link here)
- Velocity Verlet (one link here)

Symplectic integrator?

Powerful symplectic tools are available, such as:

- Martin Hairer's `GNI` package (link here)
- `SymInt` package (link here)
- Velocity Verlet (one link here)

Unfortunately, these general integrators are for canonical second order system. They require that

$$\vec{U}'' = F(t, \vec{U}),$$

or

$$\vec{U}' = \vec{V} \tag{7}$$

$$\vec{V}' = F(t, \vec{U}), \tag{8}$$

where U is position and V is velocity.

Symplectic integrator?

Powerful symplectic tools are available, such as:

- Martin Hairer's `GNI` package (link here)
- `SymInt` package (link here)
- Velocity Verlet (one link here)

Unfortunately, these general integrators are for canonical second order system. They require that

$$\vec{U}'' = F(t, \vec{U}),$$

or

$$\vec{U}' = \vec{V} \tag{7}$$

$$\vec{V}' = F(t, \vec{U}), \tag{8}$$

where U is position and V is velocity.

We have a velocity-dependent force.

Symplectic integrator

What to do? Several authors have constructed special formulations to handle non-symmetry in Hills three body, which include:

- [Waldvogel, 1997] ‘Symplectic Integrator’s for Hill’s Lunar Problem‘
- [Quinn, 2010] ‘A Symplectic Integrator for Hill’s Equations’

Both of these require gymnastics: finding a canonical transform to build new Hamiltonian $K(u, v) = K_1(u, v) + K_2(u)$, such that K_1 and K_2 are both integrable.

Hill's Lunar equations: Symplectic integrator?

Power series?

Hill's Lunar equations: Symplectic integrator?

Power series?

Power series methods may provide *effectively symplectic* integration of conservative systems through a (numerically) faithful algorithm.

Hill's Lunar equations: Symplectic integrator?

Power series?

Power series methods may provide *effectively symplectic* integration of conservative systems through a (numerically) faithful algorithm.

```
function dYdt = fhill3_xy(t,Y)

recipr3 = 1/(Y(1)^2 + Y(3)^2)^(3/2);

dYdt = [ Y(2);
    2*Y(4) + 3*Y(1) - Y(1)*recipr3;
    Y(4);
    -2*Y(2) - Y(3)*recipr3];
end
```

Hill's Lunar equations: PSM (x,y)

```
>> analyze(fhill3_xy,0,[1;0;0;1])
    'Definition          Series recur *=CP                     '
    'u1 = y1 * y1        u1 = y1 * y1                          '
    'u2 = y3 * y3        u2 = y3 * y3                          '
    'u3 = u1 + u2                                              '
    'u4 = u3^1.5         u3 * u4' = 1.5 u4 * u3' solve for u4(k)'
    'u5 = 1 / u4         1 = u5 * u4 solve for u5(k)           '
    'u6 = 2 * y4                                               '
    'u7 = 3 * y1                                               '
    'u8 = u6 + u7                                              '
    'u9 = y1 * u5        u9 = y1 * u5                          '
    'u10 = u8 - u9                                             '
    'u11 = -2 * y2                                             '
    'u12 = y3 * u5       u12 = y3 * u5                         '
    'u13 = u11 - u12                                           '
    '                    y1' = y2                              '
    '                    y2' = u10                             '
    '                    y3' = y4                              '
    '                    y4' = u13                             '
```

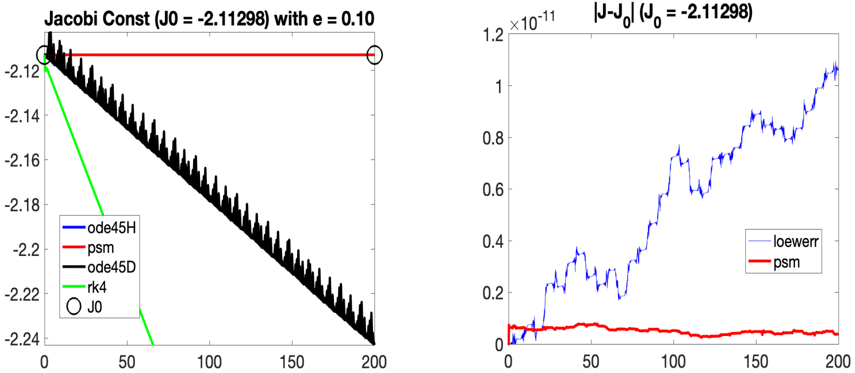
Hill's Lunar equations: PSM (q,p)

```
>> analyze(fhill3_pq,0,[1;0;0;1])

    'Definition          Series recur *=CP                    '
    'u1 = y1 * y1        u1 = y1 * y1                         '
    'u2 = y2 * y2        u2 = y2 * y2                         '
    'u3 = u1 + u2                                             '
    'u4 = u3^1.5         u3 * u4' = 1.5 u4 * u3' solve for u4(k)'
    'u5 = 1 / u4         1 = u5 * u4 solve for u5(k)          '
    'u6 = y3 + y2                                             '
    'u7 = y4 - y1                                             '
    'u8 = 2 * y1                                              '
    'u9 = y4 + u8                                             '
    'u10 = y1 * u5       u10 = y1 * u5                        '
    'u11 = u9 - u10                                           '
    'u12 = -2 * y4                                            '
    'u13 = u12 - y2                                           '
    'u14 = y2 * u5       u14 = y2 * u5                        '
    'u15 = u13 - u14                                          '
    '                    y1' = u6                             '
    '                    y2' = u7                             '
    '                    y3' = u11                            '
    '                    y4' = u15                            '
```

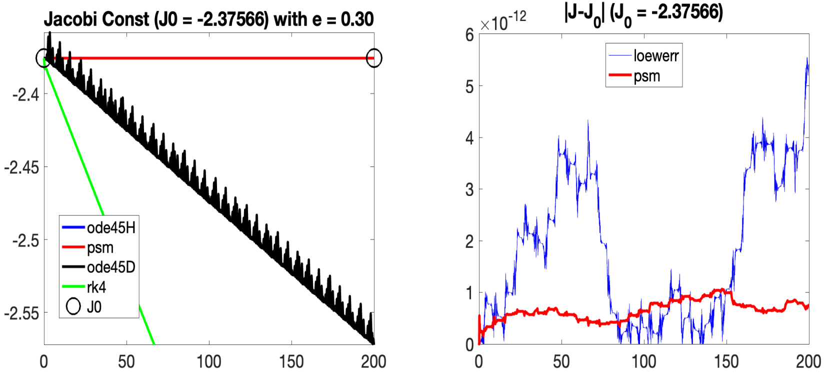
Hill's Lunar equations: 3 body

$e = 0.1$



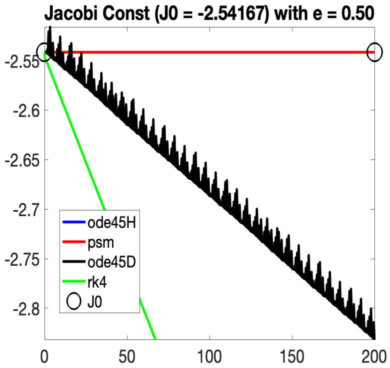
Hill's Lunar equations: 3 body

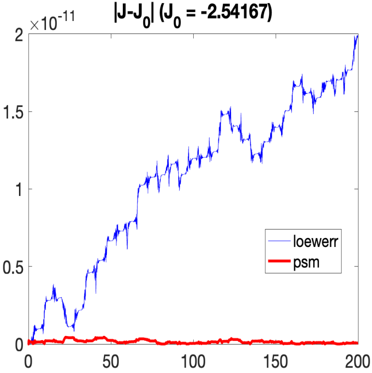
$e = 0.3$



Hill's Lunar equations: 3 body

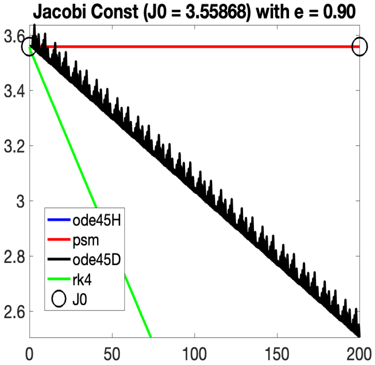
$e = 0.5$

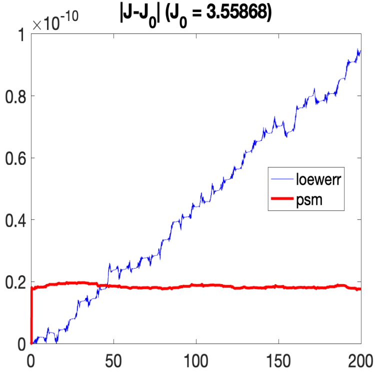




Hill's Lunar equations: 3 body

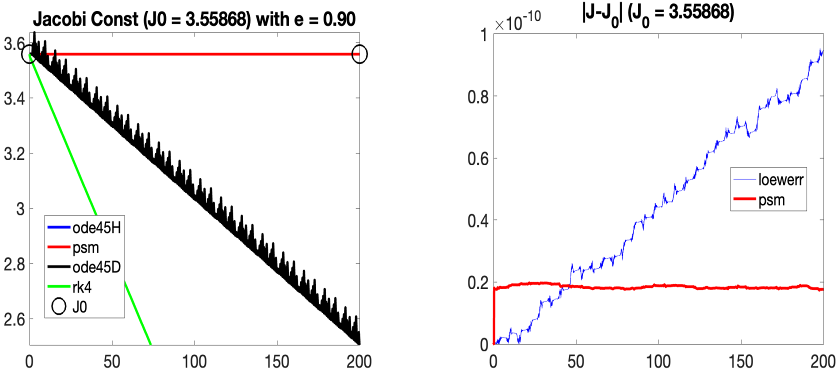
$e = 0.9$





Hill's Lunar equations: 3 body

$e = 0.9$

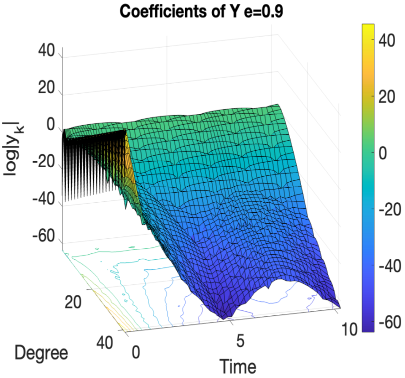


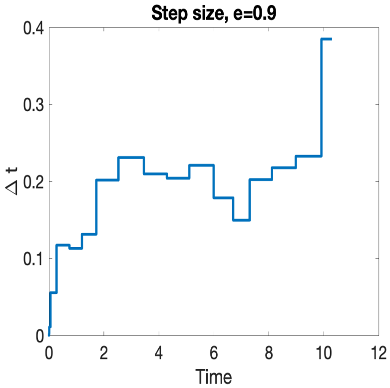
Hmmmm..... Can we do better?

Hill's Lunar equations: 3 body

What do the coefficients look like?

$$e = 0.9$$

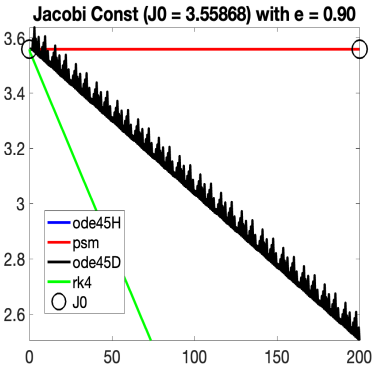


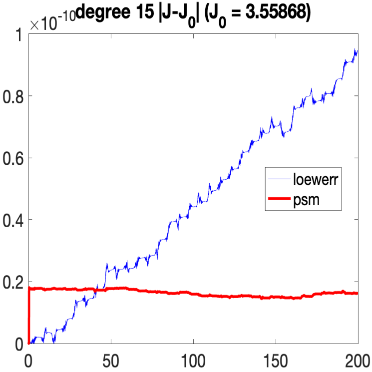


Hill's Lunar equations: 3 body

Refine step?

Back to $e = 0.9$ with refined step.

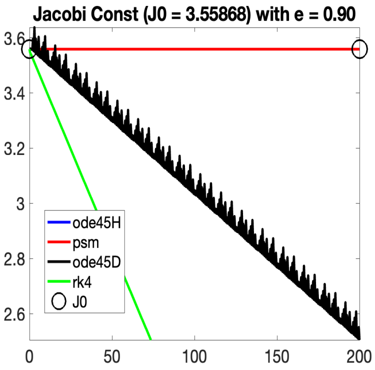


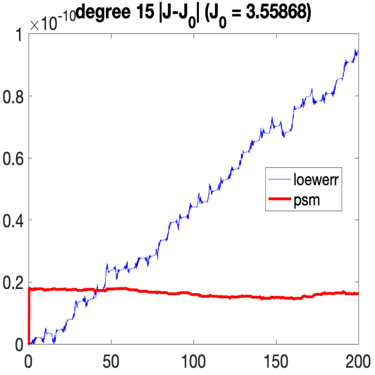


Hill's Lunar equations: 3 body

Refine step?

Back to $e = 0.9$ with refined step.



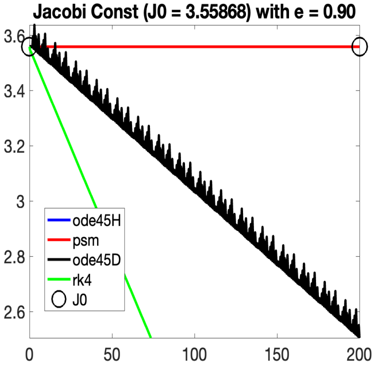


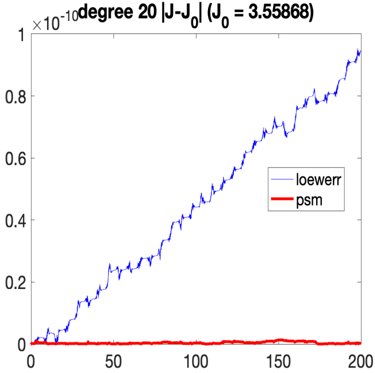
Hmmmmm.......

Hill's Lunar equations: 3 body

What if we raise degree?

Back to $e = 0.9$ with higher degree.

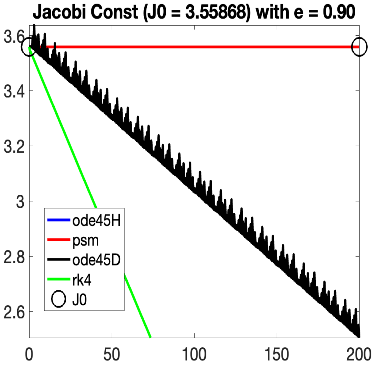


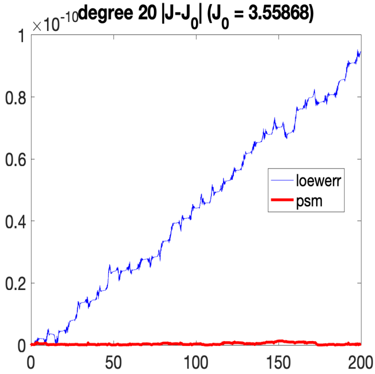


Hill's Lunar equations: 3 body

What if we raise degree?

Back to $e = 0.9$ with higher degree.





Conclusions

Our PSM methods rely on the use of auxiliary variables to build a system of polynomial IVODEs. Once the system is polynomial, series methods allow remarkably direct analysis. We made a small study of Hill's 3 body lunar problem and demonstrated:

- Effectively symplectic
- Easy to apply
- Easy error BOUND
- Transparent information

These techniques should apply to a broad range of highly nonlinear ODE. And something we didn't talk about - PSM techniques can be used to identify numerically conserved quantities for validation.

Thank you

Thanks!

Questions?

thelwerj@jmu.edu

Collaborators: J.S. Sochacki, G.E. Parker, D.C. Carothers, S.K. Lucas, J.D. Rudmin, A. Tongen, D.A. and P.G. Warne, R.D. Neidinger

References: Hill, and history

George William Hill.
On the part of the motion of the lunar perigee which is a function of the mean motions of the sun and moon.
1886.

George William Hill.
Researches in the lunar theory.
American journal of Mathematics, 1(1):5–26, 1878.

Siegfried Bodenmann.
The 18th-century battle over lunar motion.
Physics Today, 63(1):27–32, 2010.

Martin C Gutzwiller.
Moon-earth-sun: The oldest three-body problem.
Reviews of Modern Physics, 70(2):589, 1998.

References: Symplectic methods

J Waldvogel.
Symplectic integrators for hill's lunar problem.
In The Dynamical Behaviour of our Planetary System: Proceedings of the Fourth Alexander von Humboldt Colloquium on Celestial Mechanics, pages 291–305. Springer, 1997.

Thomas Quinn, Randall P Perrine, Derek C Richardson, and Rory Barnes.
A symplectic integrator for hill's equations.
The Astronomical Journal, 139(2):803, 2010.

Robert I McLachlan and G Reinout W Quispel.
Geometric integrators for odes.
Journal of Physics A: Mathematical and General, 39(19):5251, 2006.

Brett Gladman, Martin Duncan, and Jeff Candy.
Symplectic integrators for long-term integrations in celestial mechanics.
Celestial Mechanics and Dynamical Astronomy, 52:221–240, 1991.

References: PSM

E. Fehlberg
Numerical integration of differential equations by power series expansions, illustrated by physical examples.
Technical Report NASA-TN-D-2356, NASA, 1964.

David Carothers et. al.
Some properties of solutions to polynomial systems of differential equations.
Electronic Journal of Differential Equations, 2005:1–18, 2005.

P. G. Warne et. al.
Explicit a-priori error bounds and adaptive error control for approximation of nonlinear initial value differential systems.
Comput. Math. Appl., 52(12):1695–1710, 2006.

Conserved quantities in a polynomial system

Start with

$$\begin{aligned} x_1' &= x_2 \\ x_2' &= 2x_4 + 3x_1 - x_1 x_5^3 \\ x_3' &= x_4 \\ x_4' &= -2x_2 - x_3 x_5^3 \\ x_5' &= -x_1 x_2 x_5^3 - x_3 x_4 x_5^3 \\ r' &= x_1 x_2 x_5 + x_3 x_4 x_5 \end{aligned}$$

Then

$$x_2 x_2' = 2x_2 x_4 + 3x_1 x_2 - x_1 x_2 x_5^3 \text{ and } x_4 x_4' = -2x_2 x_4 - x_3 x_4 x_5^3,$$

so

$$x_5' - x_2 x_2' - x_4 x_4' = -3x_1 x_2 = -3x_1 x_1'.$$

And we have a conserved (Jacobi) constant:

$$2h = 2x_5 - x_2^2 - x_4^2 + 3x_1^2.$$

Act III
The 4 Body Problem

Hill's Lunar 4 body equations

THE RESTRICTED HILL FOUR-BODY PROBLEM WITH APPLICATIONS TO THE EARTH–MOON–SUN SYSTEM

D. J. SCHEERES

Department of Aerospace Engineering & Engineering Mechanics, Iowa State University, Ames, IA 50011-3231, U.S.A (e-mail: scheeres@iastate.edu)

(Received: 14 October 1997; accepted: 13 May 1998)

Point-masses, with space-craft orbiting binary asteroids in mutual orbit about a massive sun.

Hill's Lunar 4 body equations

[Scheeres, 1998]

$$\begin{aligned} x'' - 2(1+m)y' &= V_x, \\ y'' + 2(1+m)x' &= V_y, \\ z'' &= V_z, \end{aligned} \tag{55}$$

where

$$\begin{aligned} V(x, y, z, \tau; \nu, m) \;=\; & \tfrac{1}{2}\left(1 + 2m + \tfrac{3}{2}m^2\right)(x^2 + y^2) - \tfrac{1}{2}m^2 z^2 \\ & + \tfrac{3}{4}m^2((x^2 - y^2)\cos 2\tau - 2xy \sin 2\tau) \\ & + \frac{m^2}{a_0^3}\left[\frac{1-\nu}{R_{1-\nu}} + \frac{\nu}{R_\nu}\right] \end{aligned} \tag{56}$$

and where

$$R_{1-\nu} = \sqrt{[x + \nu(1 + \bar{\xi})]^2 + [y + \nu\bar{\eta}]^2 + z^2}, \tag{57}$$

$$R_\nu = \sqrt{[x - (1-\nu)(1 + \bar{\xi})]^2 + [y - (1-\nu)\bar{\eta}]^2 + z^2}. \tag{58}$$

These equations are periodic in τ with period π. They contain the two parameters ν and m. If $m \to 0$ then the equations of motion for the restricted three-body problem are recovered. Thus Equations (55) are a generalization of both the restricted three-body problem and the Hill equations of motion.

Hetrogenenous system:

Combining different types of processors, accelerators, and specialized hardware to work together.

- CPU
- GPU
- FPGA

Hetrogeneous system: Why?

Performance Boost:
- Leveraging specialized processors for specific tasks leads to enhanced overall performance.

Hetrogeneous system: Why?

Performance Boost:
- Leveraging specialized processors for specific tasks leads to enhanced overall performance.

Energy Efficiency:
- Optimal utilization of resources, reducing power consumption.

Hetrogeneous system: Why?

Performance Boost:
- Leveraging specialized processors for specific tasks leads to enhanced overall performance.

Energy Efficiency:
- Optimal utilization of resources, reducing power consumption.

Parallel Processing
- Simultaneous execution of tasks, accelerating complex computations.

Heterogenous system: considerations

Programming complexity usually is a barrier, decision about which code would run where (calculating overheads and data "movements" across diverse hardware)

Heterogenous system: considerations

Programming complexity usually is a barrier, decision about which code would run where (calculating overheads and data ”movements” across diverse hardware)

Specialized hardware (fpga/gpu) is more costly than general purpose (cpu), benchmarking for specific application drives the development

Heterogenous system: considerations

Programming complexity usually is a barrier, decision about which code would run where (calculating overheads and data "movements" across diverse hardware)

Specialized hardware (fpga/gpu) is more costly than general purpose (cpu), benchmarking for specific application drives the development

But....

Heterogenous system: considerations

Programming complexity usually is a barrier, decision about which code would run where (calculating overheads and data ”movements” across diverse hardware)

Specialized hardware (fpga/gpu) is more costly than general purpose (cpu), benchmarking for specific application drives the development

But....

Heterogeneous computing “done right” is a powerful paradigm for improving performance and efficiency in a wide range of applications.

Heterogenous system: so far

Different parts of PSM were rewritten in VHDL to be synthetized in hardware to study the behavior and the performance of the new dedicated hardware

1. Intel Cyclone V for preliminary testing
   - CPU/FPGA on the same silicon, handy memory sharing but limited FPGA logic elements)
   - Acceptable number of logic elements but somewhat complex memory sharing via PCIe
   - Limited number of hardware wires to submit the large number of inputs/arrays required by an n-body problem

Heterogenous system: so far

Different parts of PSM were rewritten in VHDL to be synthetized in hardware to study the behavior and the performance of the new dedicated hardware

1. Intel Cyclone V for preliminary testing
2. Exploiting the heterogeneous framework oneAPI (by Intel) to bridge CPU/FPGA (and also GPU)
   - No need to struggle with memory mapping
   - No need to port to VHDL, C++ code

Heterogenous system: so far

Different parts of PSM were rewritten in VHDL to be synthetized in hardware to study the behavior and the performance of the new dedicated hardware

1. Intel Cyclone V for preliminary testing
2. Exploiting the heterogeneous framework oneAPI (by Intel) to bridge CPU/FPGA (and also GPU)
3. The compiler takes care of most of the complexity

Heterogenous system: so far

Different parts of PSM were rewritten in VHDL to be synthetized in hardware to study the behavior and the performance of the new dedicated hardware

1. Intel Cyclone V for preliminary testing
2. Exploiting the heterogeneous framework oneAPI (by Intel) to bridge CPU/FPGA (and also GPU)
3. The compiler takes care of most of the complexity

but.....

Heterogenous system: challenges (so far)

- oneAPI support (getting better and better) was/is somewhat limited for different hardware (Cyclone family)
- Machine setup/libraries/dependencies quite complex (not fully supported yet by Linux)
- Several errors/difficult debugging due to limited documentation and knowledge base

Heterogenous system: Looking ahead

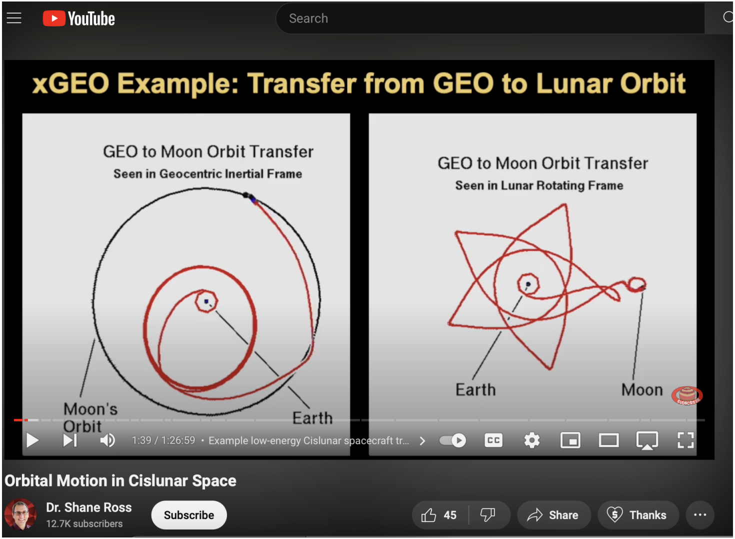
- Finalizing oneAPI approach for PSM/n-body problems so that it can seamlessly/easily compile on a diverse hardware; FPGA+CPU to start with (GPU in the future)/
- online n-body webapp that is hardware accelerated with a dedicated performance benchmark.

Heterogenous system: Goal

Orbital Motion!

Heterogenous system: Goal

Orbital Motion!



from https://www.youtube.com/watch?v=gpXAACF5eOI