

The need for accuracy and smoothness in numerical simulations

Carl Christian Kjelgaard Mikkelsen

Lorién López-Villellas

Department of Computer Science, Umeå University, Sweden,
spock@cs.umu.se

Department of Computer Science, Zaragoza University, Spain,
lorien.lopez@unizar.es

Visit to FSHMN
Pristina, Kosovo, September 12th 2024



UMEÅ UNIVERSITY



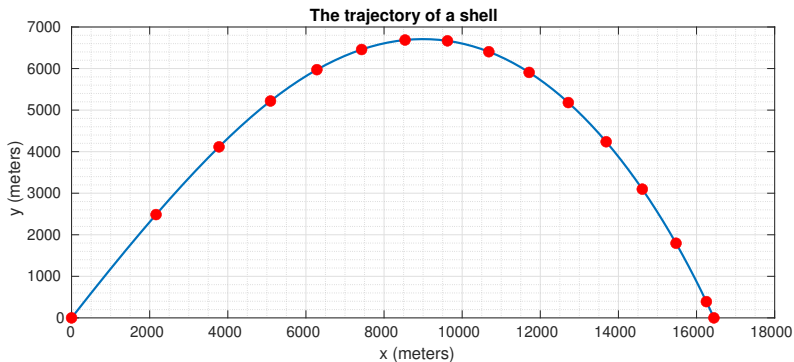
Universidad Zaragoza



Introduction

- Danish nationality
- Ass. Prof. of Comp. Sci. at Umeå University, Sweden
- Ph.D in mathematics from Purdue University, Indiana, USA
- MSc. in mathematics from Aarhus University, Denmark
- Research interests
 - Structured linear system
 - Nonsymmetric eigenvalue problems
 - Solution of nonlinear constraint equations
 - High performance scientific computing
- Software
 - Coauthor of StarNEIG
 - Coauthor of ILVES

External ballistics



The total force \mathbf{F} acting on the shell

$$\mathbf{F} = m\mathbf{g} - \frac{1}{2}\rho(y)AC_D(\nu)\|\mathbf{v} - \mathbf{w}\|_2(\mathbf{v} - \mathbf{w}) \quad (1)$$

is the combination of gravity and aerodynamic drag.

Key questions

Does it matter

- ① if the drag coefficient

$$\nu \rightarrow C_D(\nu)$$

is smooth or not?

- ② if the event equation

$$y(t) = 0$$

is solved accurately or not?

Molecular dynamics with constraints

The system of differential algebraic equations

$$\mathbf{q}'(t) = \mathbf{v}(t), \quad (2)$$

$$\mathbf{M}\mathbf{v}'(t) = \mathbf{f}(\mathbf{q}(t)) - \mathbf{G}(\mathbf{q}(t))^T \boldsymbol{\lambda}(t), \quad (3)$$

$$\mathbf{g}(\mathbf{q}(t)) = \mathbf{0}. \quad (4)$$

is solved using the SHAKE algorithm

$$\mathbf{v}_{n+1/2} = \mathbf{v}_{n-1/2} + h\mathbf{M}^{-1} \left(\mathbf{f}(\mathbf{q}_n) - \mathbf{G}(\mathbf{q}_n)^T \boldsymbol{\lambda} \right), \quad (5)$$

$$\mathbf{q}_{n+1} = \mathbf{q}_n + h\mathbf{v}_{n+1/2}, \quad (6)$$

$$\mathbf{g}(\mathbf{q}_{n+1}) = \mathbf{0}. \quad (7)$$

Key questions

Does it matter

- 1 if the force-field

$$\mathbf{q} \rightarrow \mathbf{f}(\mathbf{q})$$

is smooth or not?

- 2 if the nonlinear constraint equation

$$\mathbf{g}(\mathbf{q}_{n+1}(\boldsymbol{\lambda})) = \mathbf{0}$$

is solved accurately with respect to $\boldsymbol{\lambda}$ or not?

The primary point of this talk

If one of the following is true:

- ① the central functions are not smooth enough
- ② the central equations are not solved accurately enough

then we will almost certainly lose the ability to

- ① assert that rounding errors are irrelevant
- ② estimate the discretization error
- ③ estimate the modelling error

The key terms of this talk

- ① P : physical quantity that can be measured
- ② T : approximation of P predicted by our model
- ③ A_h : approximation of T returned by our algorithm
- ④ \hat{A}_h : the computed value of A_h .

The three different error terms

- ① $P - T$ is the modelling error and

$$P - T \neq 0 \quad (8)$$

because our model is simpler than the real world.

- ② $T - A_h$ is the discretization error and

$$T - A_h \neq 0 \quad (9)$$

because we cannot solve most differential equations exactly.

- ③ $A_h - \hat{A}_h$ is the computational error and

$$A_h - \hat{A}_h \neq 0 \quad (10)$$

due to rounding errors and truncation errors.

Why should we care?

We validate our models by demonstrating that

$$P - T \approx 0 \quad (11)$$

is a good approximation, but

$$P - T \approx P - \hat{A}_h \quad (12)$$

is not necessarily a good approximation. We have

$$\underbrace{P - T}_{\text{modelling error}} = (P - \hat{A}_h) - \underbrace{(T - A_h)}_{\text{discretization error}} - \underbrace{(A_h - \hat{A}_h)}_{\text{computational error}} \quad (13)$$

so we need to assert that

$$|T - A_h| \ll |P - \hat{A}_h| \quad (14)$$

$$|A_h - \hat{A}_h| \ll |P - \hat{A}_h| \quad (15)$$

Practical error estimation

It is frequently possible to simultaneously

- 1 assert that the computational error

$$A_h - \hat{A}_h$$

is irrelevant, and

- 2 estimate the discretization error

$$T_h - A_h$$

accurately

The key is to have an asymptotic error expansion (AEX)

$$E_h = T - A_h = \alpha h^p + \beta h^q + O(h^r), \quad h \rightarrow 0_+ \quad (16)$$

where

$$0 < p < q < r \quad (17)$$

are not necessarily integers and α, β are independent of h .

We define Richardson's error estimate by

$$R_h := \frac{A_h - A_{2h}}{2^p - 1} \quad (18)$$

and Richardson's fraction by

$$F_h := \frac{A_{2h} - A_{4h}}{A_h - A_{2h}} \quad (19)$$

Elementary results

If

$$E_h = T - A_h = \alpha h^p + \beta h^q + O(h^r), \quad h \rightarrow 0_+ \quad (20)$$

then

$$\frac{E_h - R_h}{h^q} \rightarrow \text{constant} \quad (21)$$

and

$$\frac{2^p - F_h}{h^{q-p}} \rightarrow \text{constant} \quad (22)$$

- 1 We can determine p from

$$F_h \rightarrow 2^p \quad (23)$$

- 2 We can determine q from

$$\log |2^p - F_h| \approx \log |\text{constant}| + (q - p) \log h \quad (24)$$

- 3 We can then compute R_h and estimate

$$E_h \approx R_h \quad (25)$$

when h is *sufficiently* small.

Our target value is

$$T = \int_0^1 f(x) dx, \quad f(x) = \exp(x) \quad (26)$$

Our approximation is the composite trapezoidal rule

$$A_h = \frac{h}{2} \sum_{j=0}^{n-1} (f(x_j) + f(x_{j+1})), \quad x_j = jh, \quad nh = 1 \quad (27)$$

We compute A_h for

$$h = h_k = 2^{-k} \quad (28)$$

The evolution of F_h and the quality of R_h

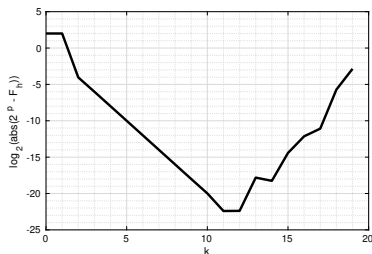


Figure: The evolution of F_h

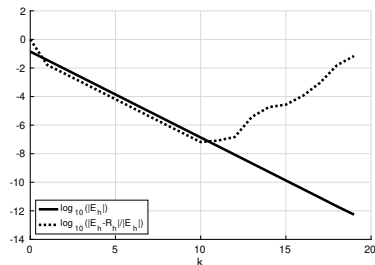


Figure: The accuracy of R_h

Strictly speaking we are not observing F_h and R_h

we are observing \hat{F}_h and \hat{R}_h

The difference is controlled by the computational error

$$A_h - \hat{A}_h \quad (29)$$

Our target value is

$$T = \int_0^1 f(x)dx, \quad f(x) = \sqrt{x}. \quad (30)$$

Our approximation is the composite trapezoidal rule

$$A_h = \frac{h}{2} \sum_{j=0}^{n-1} (f(x_j) + f(x_{j+1})), \quad x_j = jh, \quad nh = 1 \quad (31)$$

We compute A_h for

$$h = h_k = 2^{-k} \quad (32)$$

The evolution of F_h and the quality of R_h

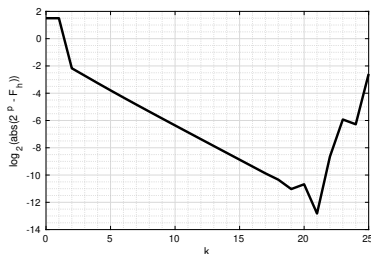


Figure: The evolution of F_h

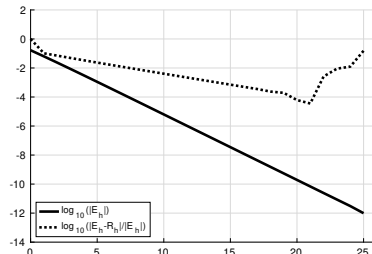


Figure: The accuracy of R_h

We observe that

- 1 the asymptotic range is much wider (good!)
- 2 the error estimate is less accurate (mostly harmless)

A spectacular failure

GROMACS simulation of hen egg white lysozyme in water:

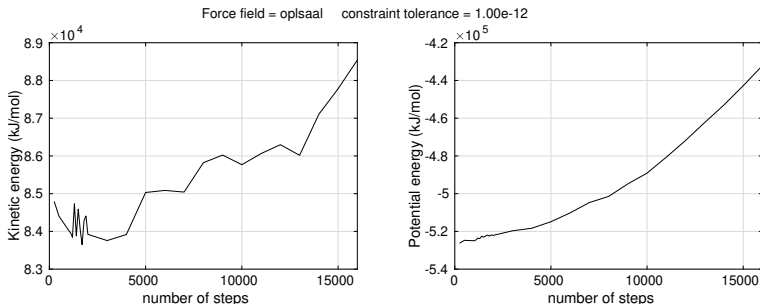
- ① T : total energy of the simulation at the end
- ② A_h : the approximation of T computed using SHAKE
- ③ \hat{A}_h : the value of A_h returned by the computer

Temporal matters:

- ① The length of the simulations was 10^{-12} s (1 ps)
- ② A common step size in MD is 10^{-15} s (1 fs) or $n = 1000$ steps
- ③ $n \in \{250, 500, 1000 : 100 : 2000, 3000 : 1000 : 16000\}$

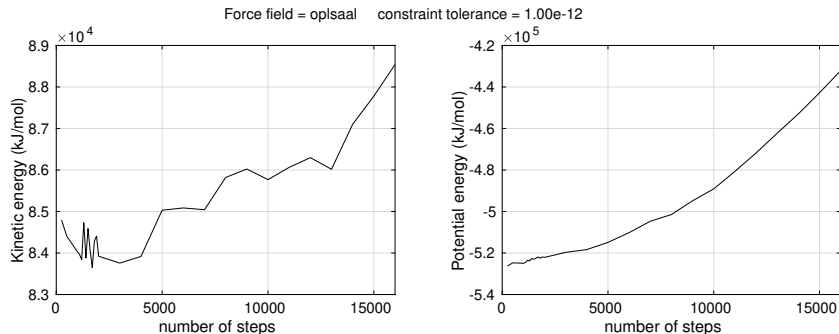
What could possibly go wrong?

A spectacular failure



The rapid growth of the energy for large value of n
can likely be cured using compensated summation.

A spectacular failure



- The wiggles near $n = 1000$ steps are a great concern.
- If there is an AEX, then 1 fs is not inside the asymp. range.
- We cannot assert that rounding errors are irrelevant
- We cannot estimate the discretisation error

When do we have an AEX?

- 1 Every AEX refers to the exact value of the A_h . If

$$\hat{A}_h \approx A_h$$

is not a good approximation, then

\hat{A}_h will not behave nicely

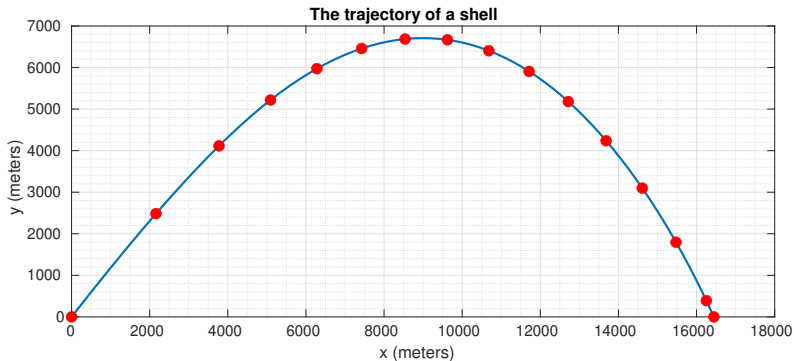
and we cannot identify the asymptotic range.

- 2 Deriving an AEX is an exercise in Taylor expansions. If

our functions are not many times differentiable

then the foundation crumbles.

External ballistics



The total force \mathbf{F} acting on the shell

$$\mathbf{F} = m\mathbf{g} - \frac{1}{2}\rho(y)AC_D(\nu, y)\|\mathbf{v} - \mathbf{w}\|_2(\mathbf{v} - \mathbf{w}) \quad (33)$$

is the combination of gravity and aerodynamic drag.

Computing the optimal range of the a howitzer: Success

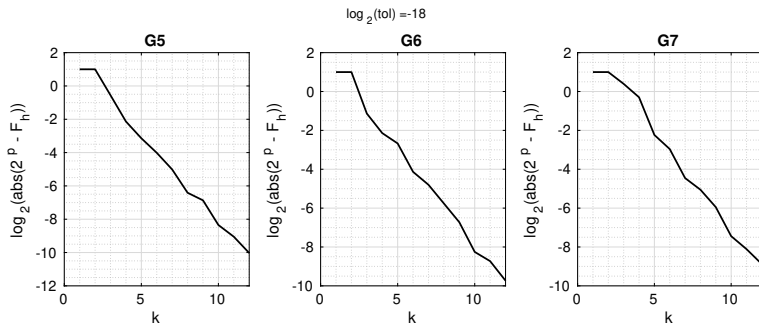


Figure: The evolution of F_h for 3 different drag coefficients, 1st order Runge-Kutta and sufficiently accurate event location.

Computing the optimal range of the a howitzer: Failure

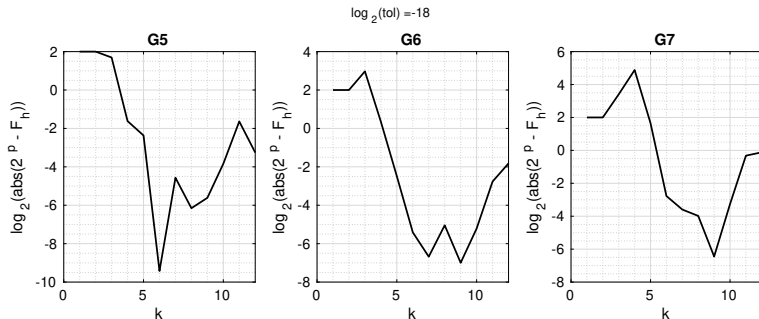


Figure: The evolution of F_h for 3 different drag coefficients, 2nd order Runge-Kutta and inaccurate event location.

Modelling ions: Setup

The total force on an ion is given

$$\mathbf{F}(\mathbf{r}_i) = -\alpha \sum_{j \neq i} \frac{1}{\|\mathbf{r}_i - \mathbf{r}_j\|^3} (\mathbf{r}_i - \mathbf{r}_j) - \beta \mathbf{r}_i - \gamma \mathbf{v}_i \quad (34)$$

where \mathbf{r}_k is the position of the k th ion and \mathbf{v}_k is its velocity.

- We wish to know the total kinetic energy T at a fixed time.
- We compute approximation A_{h_k} for $h_k = 2^{-k} h_0$.

Modelling ions: Success

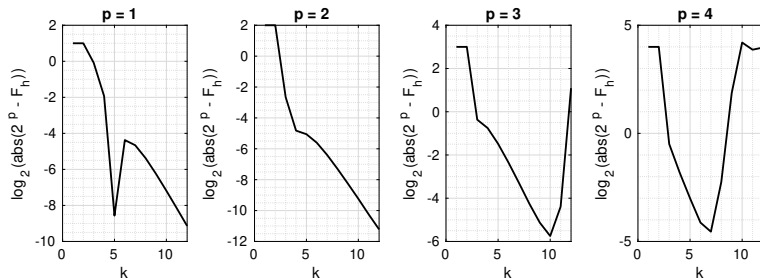


Figure: The evolution of F_h for Runge-Kutta methods of order $p \in \{1, 2, 3, 4\}$ and smooth force-fields with infinite range.

Modelling ions: Failure

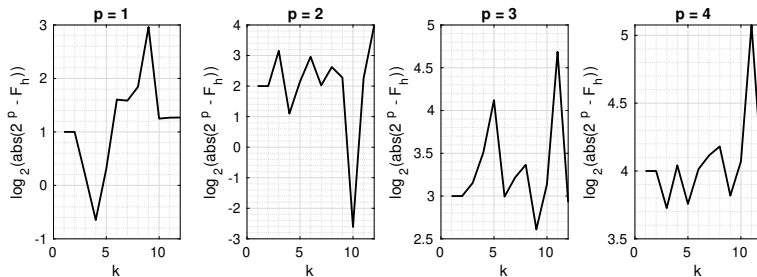


Figure: The evolution of F_h for Runge-Kutta methods of order $p \in \{1, 2, 3, 4\}$ and truncated force-fields with jump discontinuities.

Modelling ions: Success

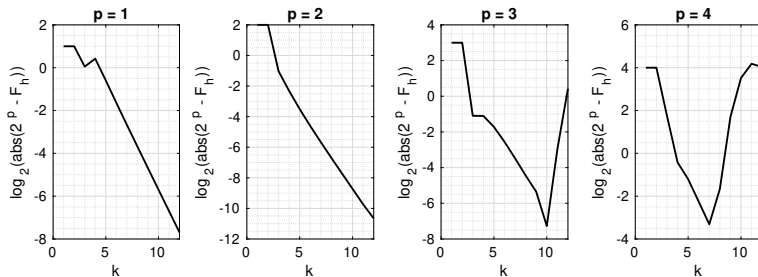


Figure: The evolution of F_h for Runge-Kutta methods of order $p \in \{1, 2, 3, 4\}$ and smoothly truncated force-fields.

Why should this concern the computational scientist?

We use every trick in the book to

- ① reduce time-to-solution
- ② increase the parallel efficiency
- ③ reduce energy-to-solution

We should not forget to ask the question:

Can we still validate our models against the real world?



Thank your for your attention