

# 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

PPAM 2024, September 8-11th 2024, Ostrava, Czechia



UMEÅ UNIVERSITY



Universidad Zaragoza



# Introduction

This is not a talk about, say,

- ① computational speed
- ② parallelism
- ③ energy efficiency

This is a talk about

the relevance of the numbers returned by our simulations

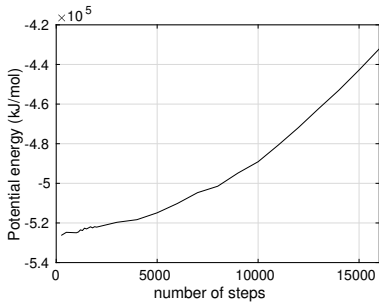
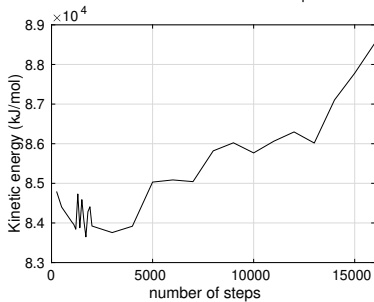
# A simple experiment

We did two sets of simulations of a molecule in water

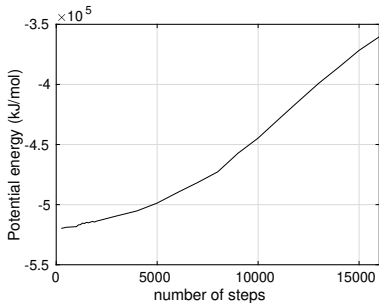
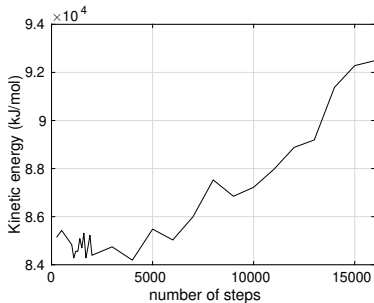
- 1 fixed real time
- 2 two different force-fields: `charmm36`, `oopsaa1`
- 3 identical initial conditions
- 4 ran the constraint solver to stagnation

and tracked the final energy as a function of the number of steps.

Force field = oplsaal constraint tolerance =  $1.00\text{e-}12$



Force field = charmm36 constraint tolerance =  $1.00\text{e-}12$



# Molecular dynamics with constraints

The system of differential algebraic equations

$$\mathbf{q}'(t) = \mathbf{v}(t), \quad (1)$$

$$\mathbf{M}\mathbf{v}'(t) = \mathbf{f}(\mathbf{q}(t)) - \mathbf{G}(\mathbf{q}(t))^T \boldsymbol{\lambda}(t), \quad (2)$$

$$\mathbf{g}(\mathbf{q}(t)) = \mathbf{0}. \quad (3)$$

is solved using the SHAKE algorithm

$$\mathbf{v}_{n+1/2} = \mathbf{v}_{n-1/2} + h\mathbf{M}^{-1} \left( \mathbf{f}(\mathbf{q}_n) - \mathbf{G}(\mathbf{q}_n)^T \boldsymbol{\lambda} \right), \quad (4)$$

$$\mathbf{q}_{n+1} = \mathbf{q}_n + h\mathbf{v}_{n+1/2}, \quad (5)$$

$$\mathbf{g}(\mathbf{q}_{n+1}) = \mathbf{0}. \quad (6)$$

## 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 may lose the ability to

- ① assert that rounding errors are irrelevant
- ② estimate the discretization error
- ③ compare the model against the physical reality

# 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

$$P - T \neq 0 \quad (7)$$

because our model is simpler than the real world.

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

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

because we cannot solve most differential equations exactly.

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

$$A_h - \hat{A}_h \neq 0 \quad (9)$$

due to rounding errors and truncation errors.

# Why should we care?

We validate our models by demonstrating that

$$P - T \approx 0 \quad (10)$$

is a good approximation, but

$$P - T \approx P - \hat{A}_h \quad (11)$$

is not necessarily a good approximation. We have

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

so we need to assert that

$$|T - A_h| \ll |P - \hat{A}_h| \quad (13)$$

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

# 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)

$$E_h = T - A_h = \alpha h^p + \beta h^q + O(h^r), \quad h \rightarrow 0_+ \quad (15)$$

where

$$0 < p < q < r \quad (16)$$

are not necessarily integers.

# Elementary results

Define

$$R_h := \frac{A_h - A_{2h}}{2^p - 1}, \quad F_h := \frac{A_{2h} - A_{4h}}{A_h - A_{2h}} \quad (17)$$

Then

$$\frac{E_h - R_h}{h^q} \rightarrow \text{constant} \quad (18)$$

and

$$\frac{2^p - F_h}{h^{q-p}} \rightarrow \text{constant} \quad (19)$$

which implies

$$\log |2^p - F_h| \approx \log |\text{constant}| + (q - p) \log h \quad (20)$$

# An example from elementary numerical analysis

Our target value is

$$T = \int_0^1 f(x)dx, \quad f(x) = \exp(x) \quad (21)$$

Our approximation is the composite trapezoidal rule

$$A_h = \frac{h}{2} \sum_{j=0}^{n-1} (f(x_j) + f(x_{j+1})), \quad x_j = jh, \quad nh = 1 \quad (22)$$

We compute  $A_h$  for

$$h = h_k = 2^{-k} \quad (23)$$

# The evolution of $F_h$ and the quality of $R_h$

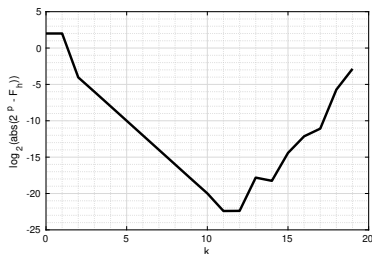


Figure: The evolution of  $F_h$

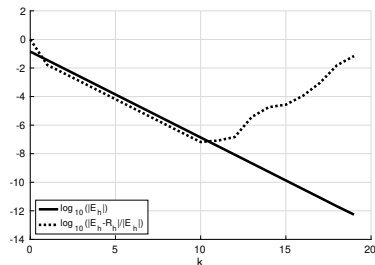


Figure: The accuracy of  $R_h$

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

$$A_h - \hat{A}_h \quad (24)$$

# 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

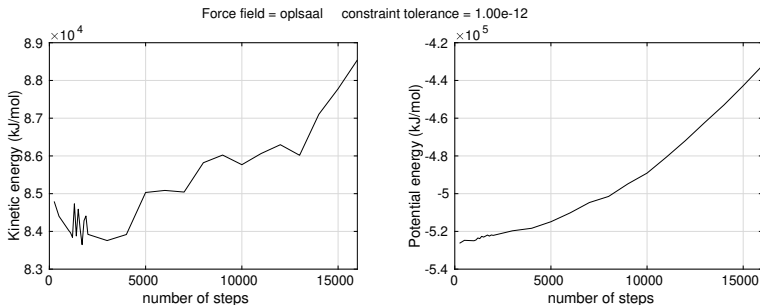
Temporal matters:

- 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?

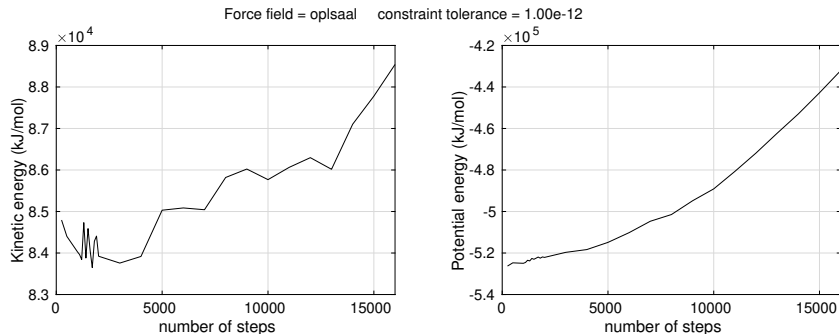


# A spectacular failure



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

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

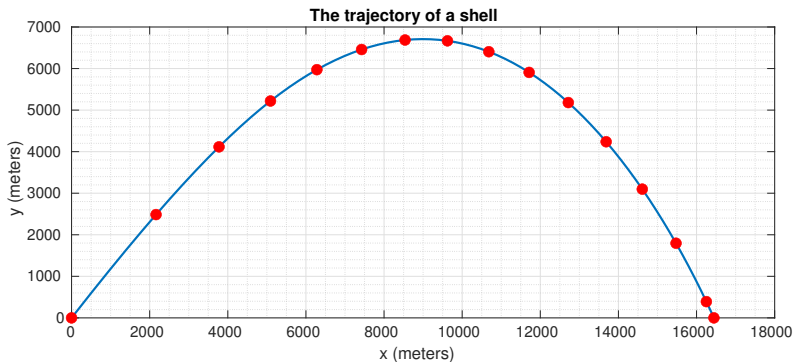
is not a good approximation, then

$\hat{A}_h$  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

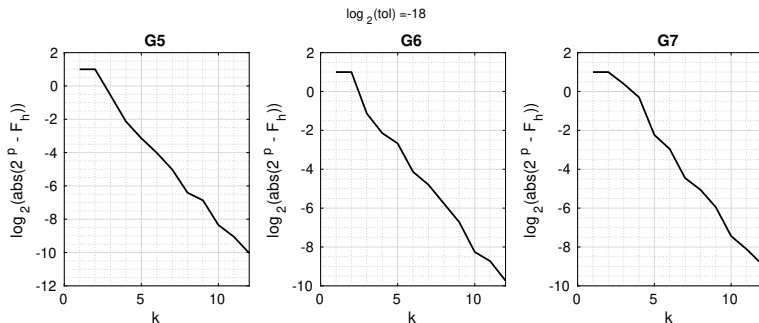


The total force  $\mathbf{F}$  acting on the shell

$$\mathbf{F} = m\mathbf{g} - \frac{1}{2}\rho(y)AC_D(\nu, y)\|\mathbf{v} - \mathbf{w}\|_2(\mathbf{v} - \mathbf{w}) \quad (25)$$

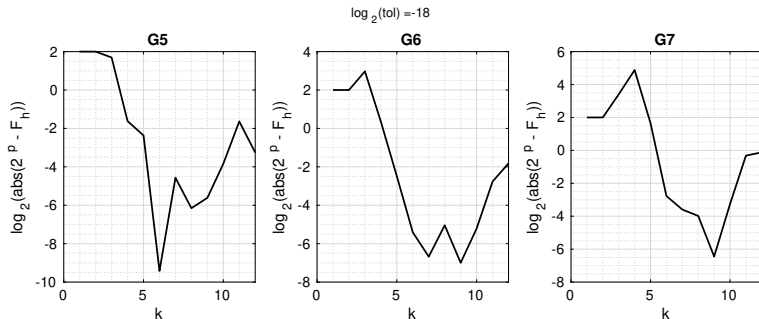
is the combination of gravity and aerodynamic drag.

# 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.

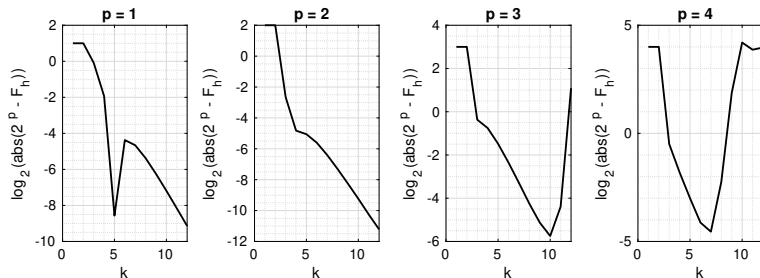
# Modelling ions: Setup

The total force on an ion is

$$\mathbf{F}(\mathbf{r}_i) = -\alpha \sum_{j \neq i} \frac{1}{\|\mathbf{r}_i - \mathbf{r}_j\|^3} (\mathbf{r}_i - \mathbf{r}_j) - \beta \mathbf{r}_i - \gamma \mathbf{v}_i \quad (26)$$

We measure the total kinetic energy  $A_h$  at a fixed time as function of the stepsize  $h_k = 2^{-k} h_0$ .

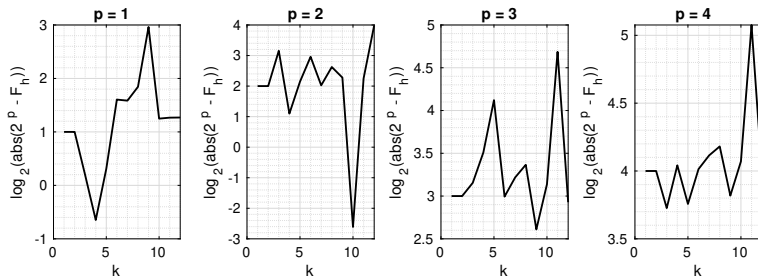
# Modelling ions: Success



**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.

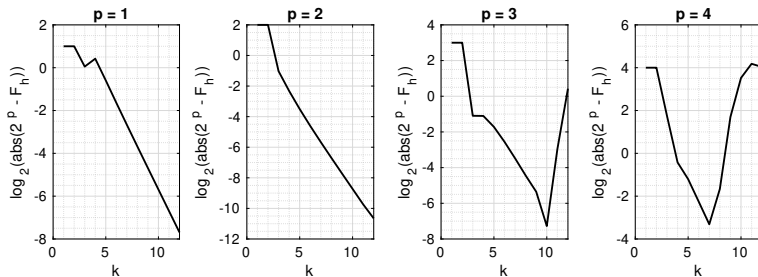


# Modelling ions: Failure



**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 PPAM community?

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?**