C7790 Introduction to Molecular Modelling
TSM Modelling Molecular Structures

Lesson 15
Potential Energy Surface III

PS/2020 Distant Form of Teaching: Rev1

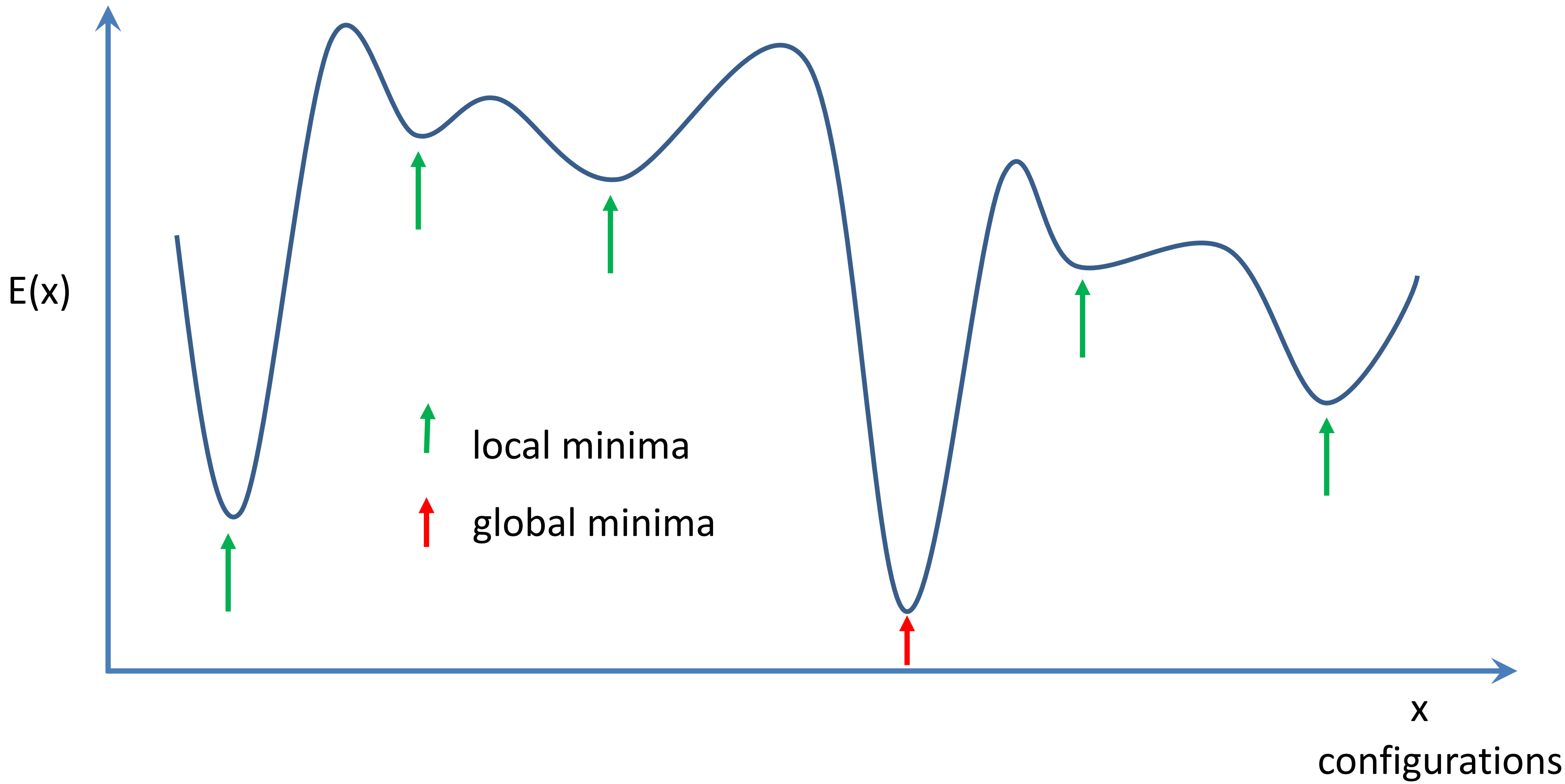
Petr Kulhánek

kulhanek@chemi.muni.cz

National Centre for Biomolecular Research, Faculty of Science
Masaryk University, Kamenice 5, CZ-62500 Brno

Local vs global minimum on PES

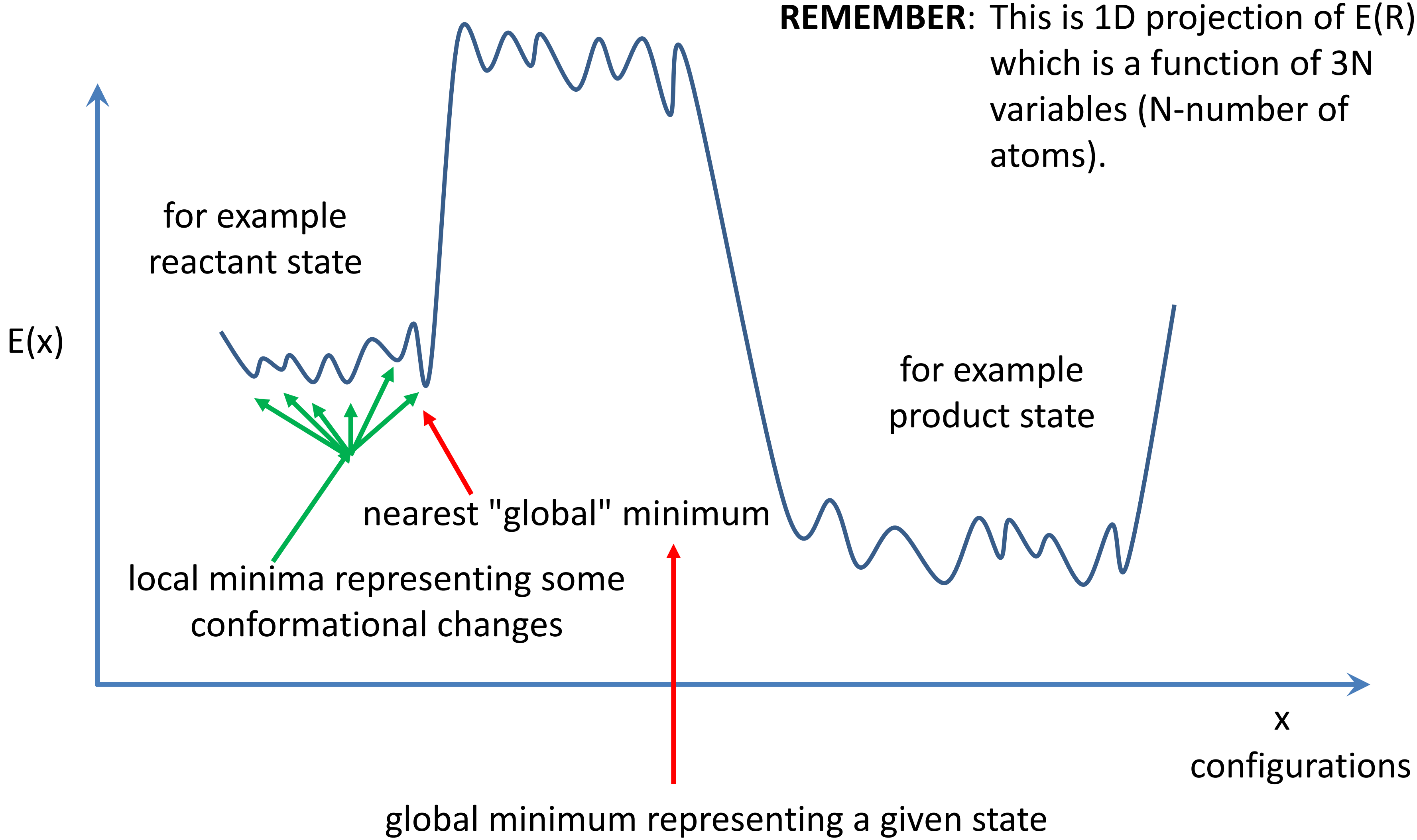
REMEMBER: This is 1D projection of E(R), which is a function of 3N variables (N-number of atoms).



To find a global minimum, it is necessary to find ALL local minima. Due to PES complexity, this is not computationally achievable even for small systems.

Local vs global minimum on PES

REMEMBER: This is 1D projection of E(R), which is a function of 3N variables (N-number of atoms).



Local vs global minimum on PES

Finding **local minimum**:

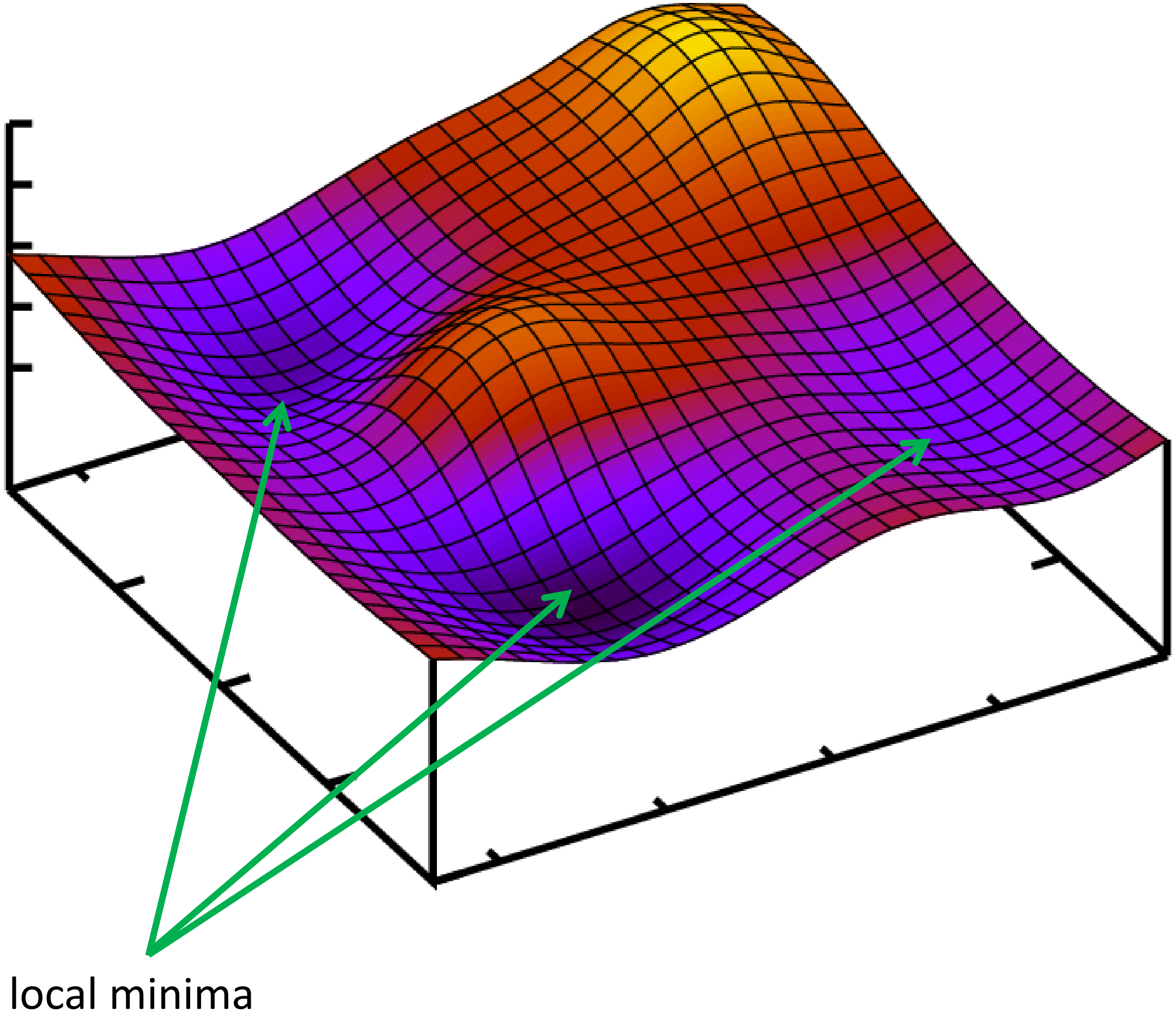
- it is rather **simple task**, which employs **local geometry optimizers**
- the success of finding of local minimum is almost always guaranteed (problematic might by shallow minima).

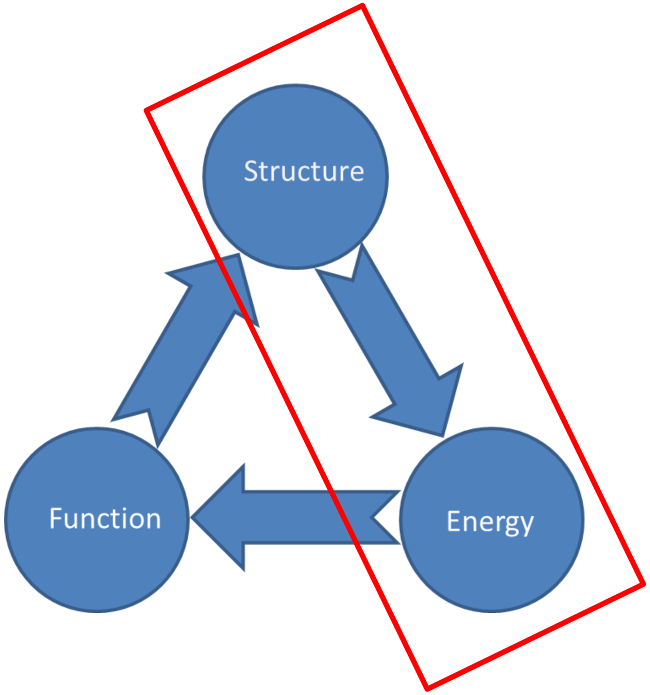
Finding **global minimum**:

- it is **VERY difficult task**, which can employ deterministic and/or stochastic methods such as
  - genetic algorithms
  - Monte-Carlo sampling algorithms
  - parallel tempering
  - others
- the success of finding of global minimum is not guaranteed

Finding the optimal geometry (local minima)

Optimal geometry





Optimization methods

Task: find R such E(R) is minimum (has zero gradient)

$$\boxed{E(\mathbf{R})} \longrightarrow \textbf{min!}$$

Geometry optimization methods

I. zero order (energy only)
   - downhill simplex method

II. first order (energy and gradient only)
   - steepest descent method
   - conjugate gradient method

III. second order (energy, gradient and Hessian)
   - Newton's method

IV. pseudo-second order (energy, gradient and approximate Hessian)
   - Broyden-Fletcher-Goldfarb-Shanno method (BFGS)

the most often used approach

Optimization methods

Geometry optimization methods

I. zero order (energy only)
   - downhill simplex method

II. first order (energy and gradient only)
   - steepest descent method
   - conjugate gradient method

III. second order (energy, gradient and Hessian)
   - Newton's method

IV. pseudo-second order (energy, gradient and approximate Hessian)
   - Broyden-Fletcher-Goldfarb-Shanno method (BFGS)

Above mentioned methods (algorithms) are general for any function:

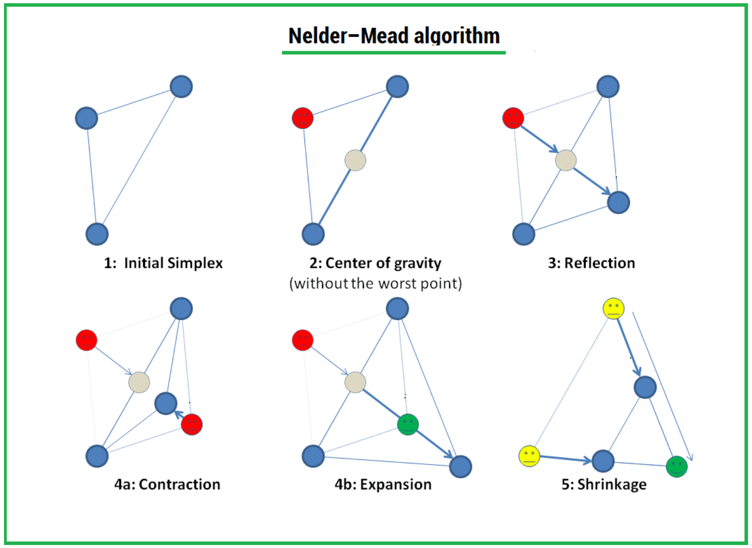
> The function f is called, variously, an objective function, a loss function or cost function (minimization), a utility function or fitness function (maximization), or, in certain fields, an energy function or energy functional. A feasible solution that minimizes (or maximizes, if that is the goal) the objective function is called an optimal solution.

https://en.wikipedia.org/wiki/Mathematical_optimization

Zero-order methods

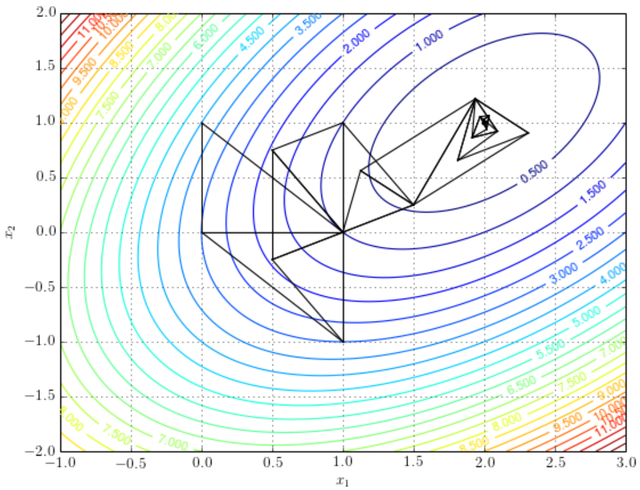
Downhill simplex method
(Nelder-Mead algorithm)

- iterative method
- only energy is required (no gradient or Hessian)
- can escape local minima and can find nearest "global" minimum



Other zero-order methods:

- BOBYQA
- COBYLA
- majority of global optimizers
- etc.



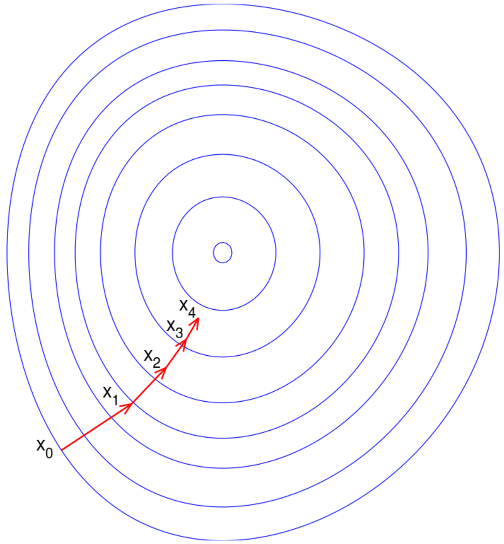
Further details:
https://en.wikipedia.org/wiki/Derivative-free_optimization

https://sudonull.com/post/69185-Nelder-Mead-optimization-method-Python-implementation-example
https://codesachin.wordpress.com/2016/01/16/nelder-mead-optimization/

First-order methods

Steepest Descent Method

- iterative method
- only gradient is required (energy is only required for monitoring)
- it can find a local minimum



https://en.wikipedia.org/wiki/Gradient_descent

$$\boldsymbol{R}_n = \boldsymbol{R}_{n-1} - \gamma \frac{\partial E(\boldsymbol{R}_{n-1})}{\partial \boldsymbol{R}}$$

"step" size

energy gradient

step size can be a constant, varying, or different for geometry domains (bonds, angles, ...)

Other first-order methods:

- conjugate-gradients
- etc.

- Rarely used for QM as these methods require more iterations to reach a minimum than psedo-second order methods.
- Quite often used for MM.

Second-order methods

https://en.wikipedia.org/wiki/Newton%27s_method_in_optimization

Newton's Method

- iterative method
- gradient and Hessian are required (energy is only required for monitoring)
- it can find a local minimum
- the method converges significantly faster (in smaller number of steps) than zero-order or first-order methods
- **it is EXTREMELY computationally expensive due to Hessian calculations**

$$\boldsymbol{R}_n = \boldsymbol{R}_{n-1} - \gamma \left[\frac{\partial^2 E(\boldsymbol{R}_{n-1})}{\partial \boldsymbol{R}^2}\right]^{-1} \frac{\partial E(\boldsymbol{R}_{n-1})}{\partial \boldsymbol{R}}$$

"step" size

Hessian

energy gradient

step size can be a constant, varying, or different for geometry domains (bonds, angles, …)

Solution: pseudo-second order methods (such as BFGS)

- initial Hessian is approximated (empirical approaches, or unit matrix)
- in next iterations, Hessian is updated using gradients
- Hessian is thus improving during optimization, which results in faster convergence in final steps

OPTIMAL CHOICE

Cartesian vs Internal Coordinates

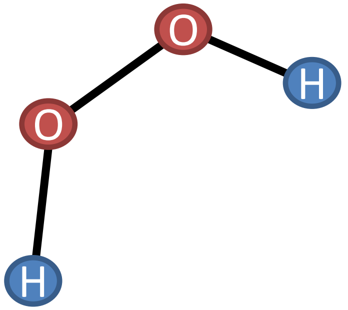
Cartesian coordinates

```
O -0.180077 -0.046023 -0.062789
H  0.196208 -0.747659  0.498793
O  0.006537  1.047922  0.877207
H -0.931885  1.299156  0.951390
```

x y z

Number of degrees of freedom:

3N



Internal coordinates (Z-matrix)

```
O
H 1 0.974298
O 1 1.454349 2 96.868054
H 3 0.974298 1 96.868054 2 239.552651
```

bond length bond angle torsion angle

Number of degrees of freedom:

3N-6

3N-5 (linear diatomic molecule)

Internal vs Cartesian coordinates

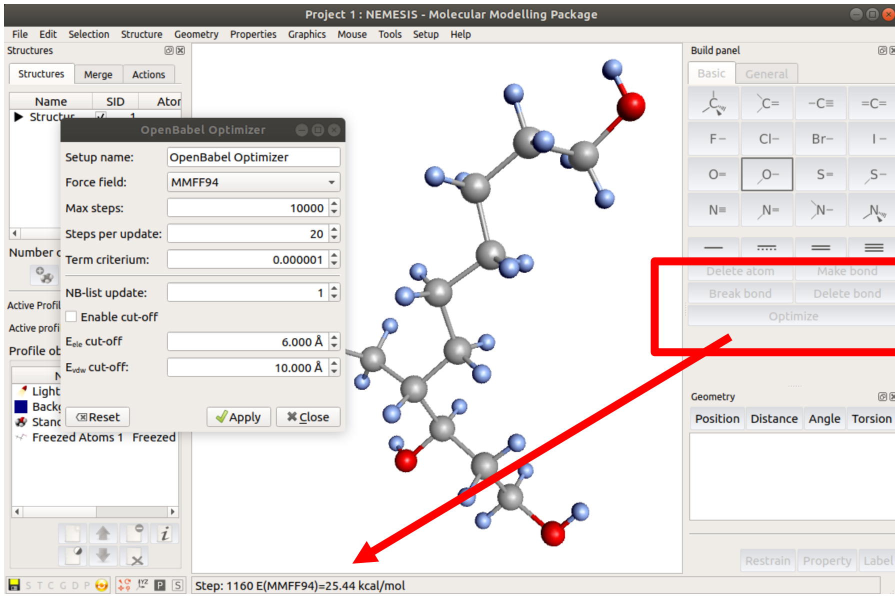
Optimization in internal coordinates converges fasters than in Cartesian coordinates:

- Hessian in internal coordinates can naturally provide difference between force constants of different geometry parameters (bonds, angles, torsions)
- This property of internal coordinates allows to use different step sizes for different geometry parameters (bonds, angles, torsions).

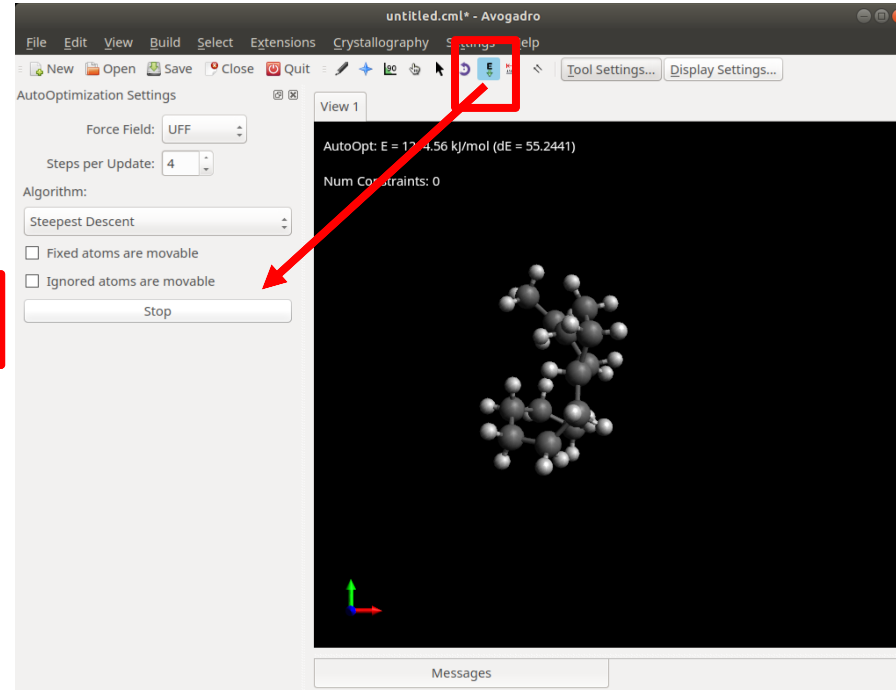
In some rare cases, optimization in internal coordinates can fail (oscillation, etc.)
Potential solutions:

- use different optimization algorithm
- switch to Cartesian coordinates

Practical realizations



Nemesis employing MM potential



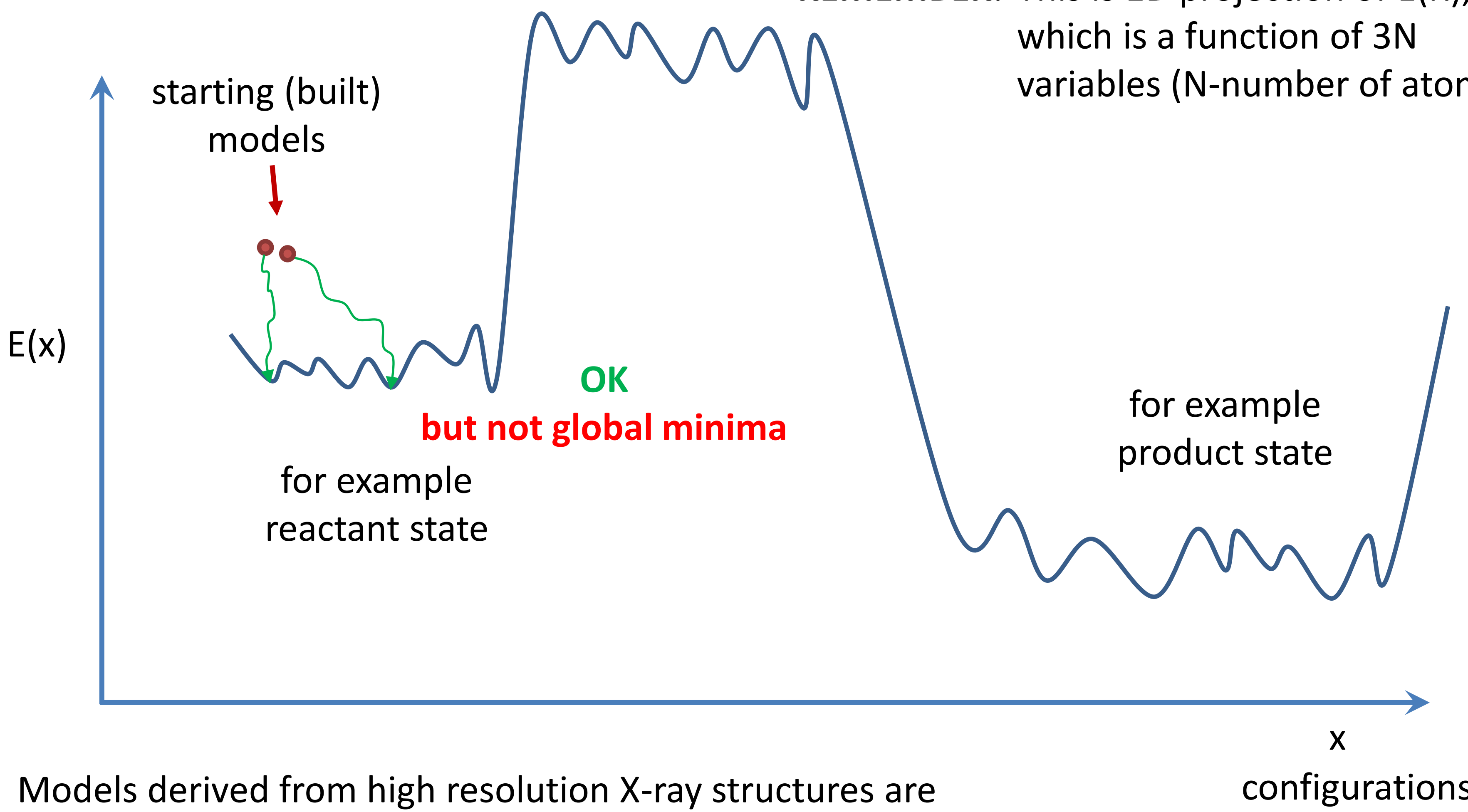
Avogadro employing MM potential

Gaussian employing QM potential

RHF/cc-pVDZ Opt NoSymm

Optimization of initial model

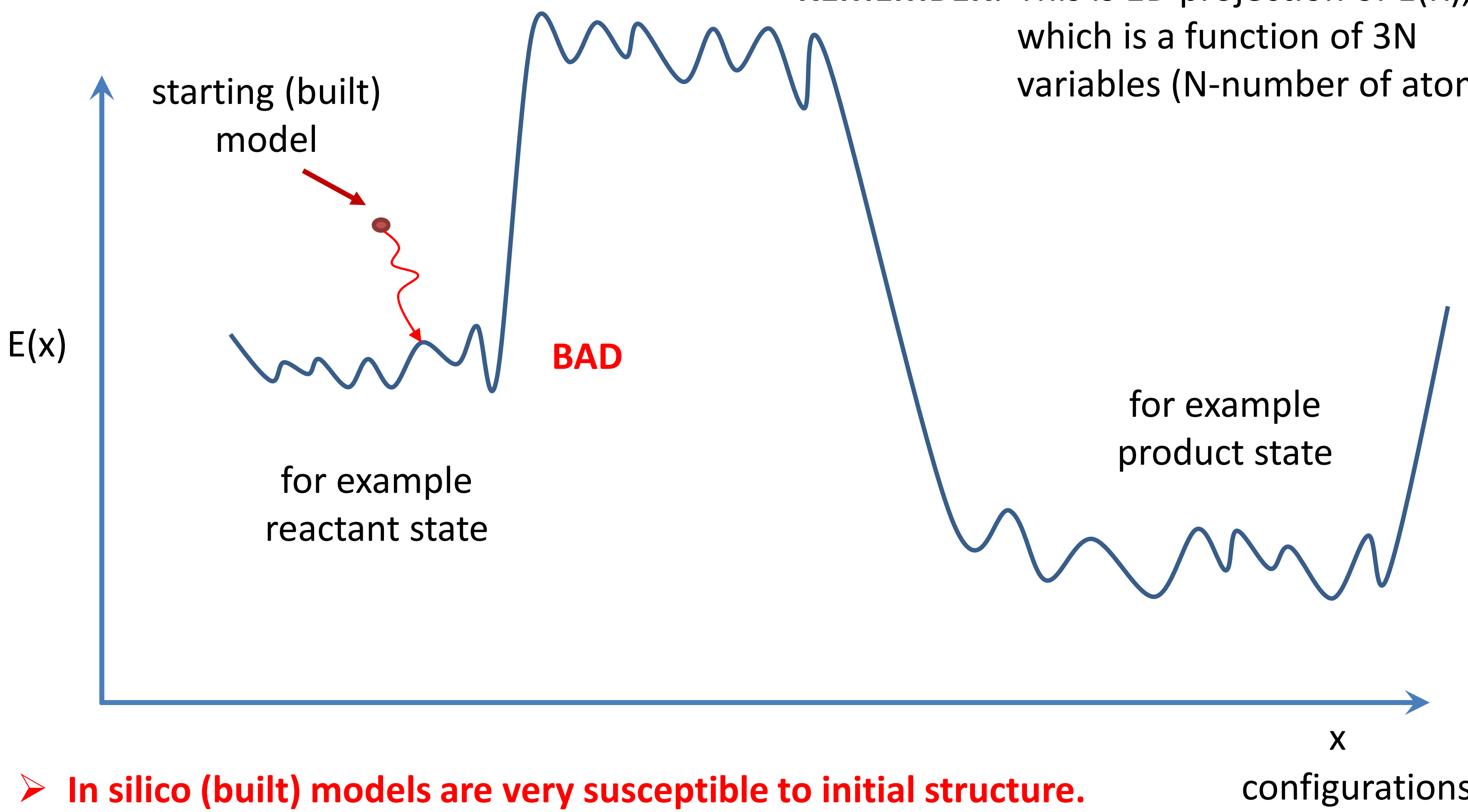
REMEMBER: This is 1D projection of E(R), which is a function of 3N variables (N-number of atoms).



Models derived from high resolution X-ray structures are not problematic.

Optimization of initial model

REMEMBER: This is 1D projection of E(R), which is a function of 3N variables (N-number of atoms).



- **In silico (built) models are very susceptible to initial structure.**
- **Therefore, frequency (vibration, Hessian) analysis is a MUST to check the nature of optimized stationary point.**

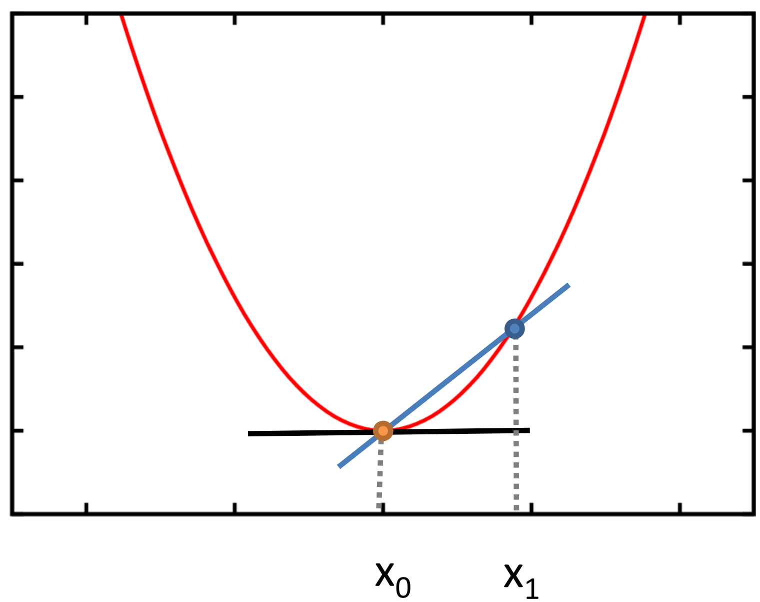
Summary

- It is relatively easy to find a local minimum.
- All geometry optimizers stops at a stationary point (a point with zero gradient), which dos not necessarily need to represent a local minimum.
- Due to complexity of PES, it **is important to verify a nature of found stationary point** because the found geometry can represent a transition state or a higher order saddle point.
- This is especially important for *in silico* models, which usually starts a far away from optimal geometry.

Homework

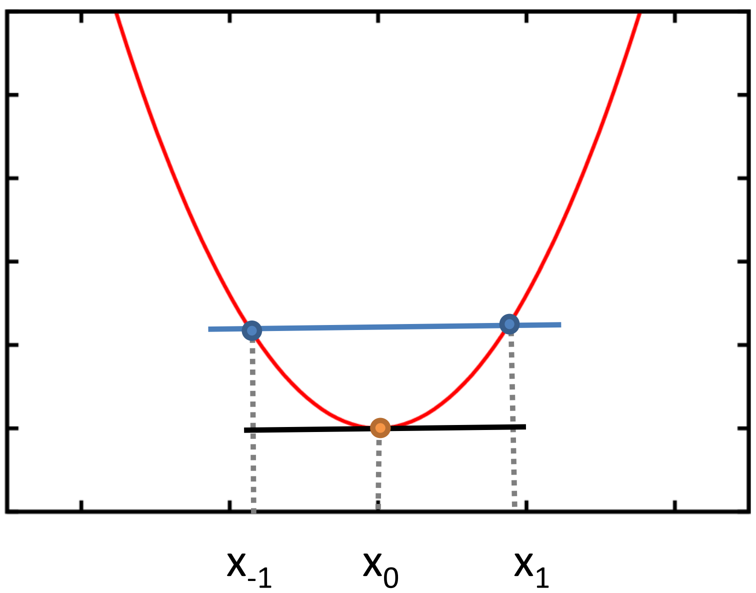
Numerical gradient calculation

Forward differences



$$\left.\frac{\partial E(\mathbf{R})}{\partial x}\right|_{x_0} = \frac{E(x_1) - E(x_0)}{x_1 - x_0} = \frac{E(x_0 + h) - E(x_0)}{h}$$

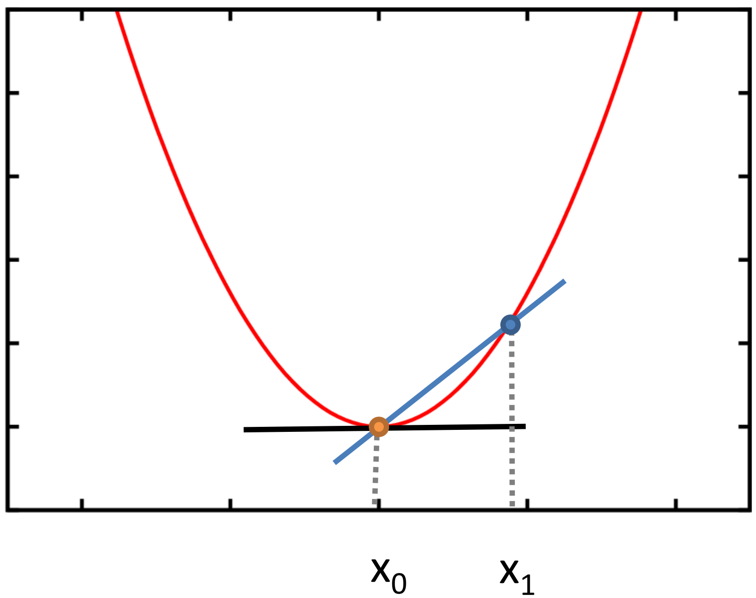
Central differences



$$\left.\frac{\partial E(\mathbf{R})}{\partial x}\right|_{x_0} = \frac{E(x_1) - E(x_{-1})}{x_1 - x_{-1}} = \frac{E(x_0 + h) - E(x_0 - h)}{2h}$$

Numerical gradient calculation

Forward differences



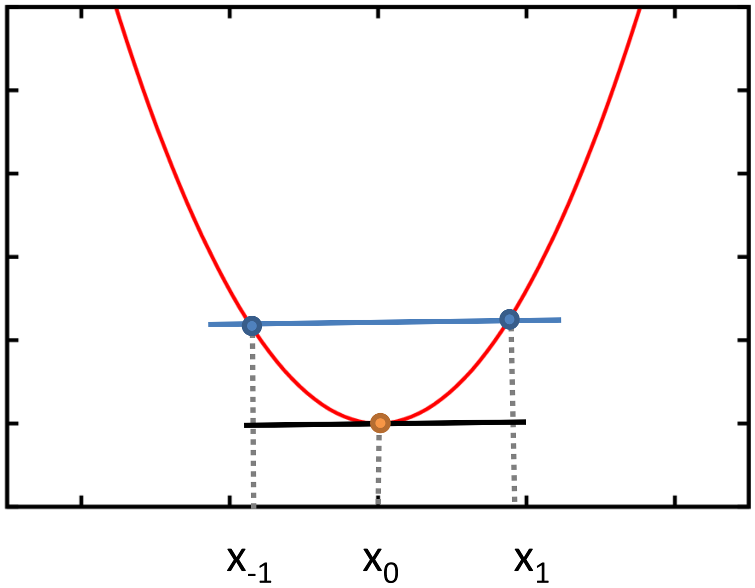
$$\left.\frac{\partial E(\mathbf{R})}{\partial x}\right|_{x_0} = \frac{E(x_1)-E(x_0)}{x_1-x_0} = \frac{E(x_0+h)-E(x_0)}{h}$$

it is calculated for each gradient component

it is calculated once

a total of 3N + 1 energy calculations

Central differences



$$\left.\frac{\partial E(\mathbf{R})}{\partial x}\right|_{x_0} = \frac{E(x_1)-E(x_{-1})}{x_1-x_{-1}} = \frac{E(x_0+h)-E(x_0-h)}{2h}$$

they are calculated for each gradient component

a total of 6N energy calculations

Tasks I

1. Express the gradient of the function E(**R**) according to the Cartesian coordinates of both atoms.

$$E(\mathbf{R}) = \frac{1}{2}K(r - r_0)^2$$

2. The system contains 300 atoms. The calculation of its energy by the quantum-chemical method takes 15 minutes. Calculation of energy and analytical gradient then 20 minutes.

    1. Determine the calculation time of the numerical gradient and compare it with the calculation time of the analytical gradient.
    2. Determine the calculation time of numerical Hessian, which is calculated a) from energies and b) from analytical gradients.
    3. Suggest a way to speed up the calculation of the numerical gradient a Hessian.

Tasks II

1) For the function below, determine the character of the points with values:

$$E(x) = 15x^2 + 30x + 3$$

   a) x = 1
   b) x = 0
   c) x = -1

2) In what situation can the second derivative of a function be zero?

3) What is the relationship between the extent of the reaction ξ and reaction coordinate r_C (also referred to as ξ)?

Tasks III

1. Study the mentioned local geometry optimization methods. Focus on their principle, advantages and disadvantages compared to other optimization methods.

Literature:

(1) Leach, A.R. Molecular Modeling: Principles and Applications, 2nd ed .; Prentice Hall: Harlow, England; New York, 2001.

(2) Jensen, F. Introduction to Computational Chemistry, 2nd ed .; John Wiley & Sons:Chichester, England; Hoboken, NJ, 2007.