

Schrödinger Web Service

RESTful web service for solving
multidimensional time-independent
Schrödinger equation using Hermite DVR
approach



Scientific Background



- Numerous nano-world phenomena are inherently quantum
- Their description and quantitative treatment requires usage of the apparatus of quantum mechanics.
- The Schrödinger equation (a “quantum analog” to the famous Newton’s equation in classical physics).

- For a stationary state of the system the Schrödinger equation has the form



$$\hat{H}\psi = E\psi$$

The Hamiltonian consists of the kinetic energy operator ($\hat{K}(q)$) and the potential energy operator ($\hat{V}(q)$):

$$\hat{H}(q) = \hat{K}(q) + \hat{V}(q)$$

The solution may be sought within a finite basis set of n functions ($\varphi_i(q)$), as in the variational approach:

$$\langle q|\psi\rangle = \sum_{i=1}^n c_i \cdot \langle q|\varphi_i\rangle$$

- Assuming an orthonormal set of basis functions

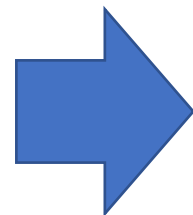
- The expansion coefficients c_i and the set of eigenenergies can be found by solving the matrix eigenvalue problem

$$\langle \varphi_i|\varphi_j\rangle = \int \varphi_i^*(q) \cdot \varphi_j(q) dq = \delta_{ij}$$



$$\mathbf{H}\mathbf{c} = E\mathbf{c}$$

\mathbf{c} is the column n -vector $[c_1, c_2, \dots, c_n]^T$, while \mathbf{H} is $n \times n$ square matrix containing the matrix elements of the Hamiltonian H_{ij} , given by

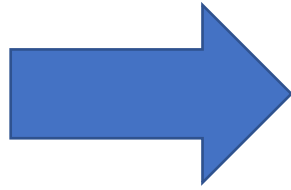


$$H_{ij} = \langle \varphi_i|\hat{H}|\varphi_j\rangle = \int \varphi_i^*(q)\hat{H}\varphi_j(q) dq$$

The finite basis representation (FBR)

- Considering a Hamiltonian of the form:

$$\hat{H}(q) = \hat{H}_0(q) + \hat{V}'(q)$$



- $\hat{H}_0(q)$: the harmonic oscillator “zeroth order” Hamiltonian
- $\hat{V}'(q)$: the “complicated” potential energy part

The most complicated part of the calculation involves computation of the matrix elements:

$$V'_{ij} = \langle \psi_i | \hat{V}' | \psi_j \rangle = \int \psi_i^*(q) \hat{V}' \psi_j(q) dq$$

Define an n -point quadrature such consisting of the abscissa set of values $\{x_1, x_2, \dots, x_n\}$ and the corresponding weights $\{w_1, w_2, \dots, w_n\}$. Such n -point quadrature leads to:

$$V_{ij} \approx V_{ij}^{FBR} = \sum_{k=1}^n w_k \psi_i^*(x_k) V(x_k) \psi_j(x_k)$$

- Apply unitary transformation such that the potential energy matrix is diagonal

$$\phi_k(x) = \sum_{i=1}^n A_{ki} \psi_i(x)$$

- Evaluate the DVR functions at the quadrature points

$$\phi_k(x_l) = \sum_{i=1}^n A_{ki} \psi_i(x_l) = \sum_{i=1}^n A_{ki} A_{li} w_l^{-\frac{1}{2}} = \delta_{kl} w_l^{-\frac{1}{2}}$$



Any operator which is multiplicative in coordinate representation is diagonal in this basis

$$\langle \phi_i | V | \phi_j \rangle = \sum_{k=1}^n w_k \phi_i(x_k) V(x_k) \phi_j(x_k) = \sum_{k=1}^n \delta_{ik} V(x_k) \delta_{jk} = V(x_k) \delta_{ij}$$

- The kinetic energy matrix is not diagonal in DVR (the kinetic energy operator is not multiplicative in coordinate representation)



In the case of one-dimensional problems, the advantage of DVR over FBR is questionable



However, in multidimensional problems, the advantage of DVR becomes immediately evident

Schrödinger API



- **RESTful web service for solving multidimensional time-independent Schrödinger equation using Hermite DVR approach**
 - solution of **one-dimensional, two-dimensional** and **three-dimensional** time-independent Schrödinger equation based on the Gauss-Hermite Discrete Variable Representation (DVR) approach

The solution of **1D** Schrödinger equation is illustrated in the case of following model potentials:

- [Morse potential](#)
- [Simple Harmonic Oscillator \(SHO\) potential](#)
- [Sombrero potential \(Mexican hat\)](#)
- [Woods-Saxon potential](#)

Solutions of **2D** and **3D** Schrödinger equations are illustrated for the following two model potentials:

- [2D Morse potential](#)
- [2D SHO potential](#)
- [3D Morse potential](#)

Returns one-dimensional Morse potential $V(x)$:

$$V(x) = D * (1 - \exp(-a * (x - x_0)))^2 - D$$

Request URL

```
https://schrodinger.chem-api.finki.ukim.mk/1dHermiteMorse?npts=5&D=3&a=0.5&x0=0&prec=8
```

Server response

Code	Details
------	---------

200	
-----	--

Response body

```
[-2.41671645 -1.39124794 -0.28535681 1.09633735 7.42311473]
```

Parameters:

npts - number of points (default value 10)

D - dissociation depth (default value 3.0)

a - inverse "width" of the potential (default value 0.5)

x0 - equilibrium bond distance (default value 0.0)

prec - precision (default value 6)

Returns one-dimensional version of the sombrero potential and requires $a < 0$ and $b > 0$:

$$V(x) = a * x^2 + b * x^4$$

Request URL

```
https://schrodinger.chem-api.finki.ukim.mk/1dHermiteSombbrero?npts=5&a=-5&b=1&prec=8
```

Server response

Code	Details
------	---------

200	
-----	--

Response body

```
[-3.31729755 -3.29056942 -1.70943058 -1.67240256 2.48970011]
```

Parameters:

npts - number of points (default value 10)

D - dissociation depth (default value 3.0)

a - inverse "width" of the potential (default value 0.5)

x0 - equilibrium bond distance (default value 0.0)

prec - precision (default value 6)

Returns a Woods-Saxon potential

$$V(r) = -V_0 / (1. + \exp((r - R) / z)) \text{ where } R = r_0 * A^{(1/3)}$$

Request URL

```
https://schrodinger.chem-api.finki.ukim.mk/1dHermitWoodSax?npts=5&V0=50&z=0.5&r0=1.2&A=16&prec=8
```

Server response

Code	Details
200	Response body

```
[-49.73342002 -49.02383 -47.92816698 -46.25839997 -42.70563227]
```

Parameters:

- npts - number of points (default value 5)
- V0 - potential depth (default value 50.0)
- z - surface thickness (default value 0.5)
- r0 - rms nuclear radius (default value 1.2)
- A - mass number (default value 16)
- prec - precision (default value 8)

```
Request URL
https://schrodinger.chem-api.finki.ukim.mk/2dHermiteMorse?npts=5&D1=3&a1=0.5&D2=3&a2=0.5&x0=0&y0=0&prec=8

Server response

Code    Details
200     Response body
[-4.83343289 -3.80796439 -3.80796439 -2.78249589 -2.70207325]
```

Parameters:

npts - number of points (default value 5)

D1 - dissociation depth for x (default value 3.0)

a1 - inverse "width" of the potential for x (default value 0.5)

x0 - equilibrium bond distance for x (default value 0.0)

D2 - dissociation depth for y (default value 3.0)

a2 - inverse "width" of the potential for y (default value 0.5)

y0 - equilibrium bond distance for y (default value 0.0)

prec - precision (default value 6)

3D Morse potential

Request URL	
<code>https://schrodinger.chem-api.finki.ukim.mk/3dHermiteMorse?npts=5&D1=3&a1=0.5&D2=3&a2=0.5&D3=3&a3=0.5&x0=0&y0=0&z0=0&prec=8</code>	
Server response	
Code	Details
200	Response body
<code>[-7.25014934 -6.22468084 -6.22468084 -6.22468084 -5.19921233]</code>	

Parameters:

npts - number of points (default value 5)

D1 - dissociation depth for x (default value 3.0), D2 - dissociation depth for y (default value 3.0), D3 - dissociation depth for z (default value 3.0)

a1 - inverse "width" of the potential for x (default value 0.5), a2 - inverse "width" of the potential for y (default value 0.5), a3 - inverse "width" of the potential for z (default value 0.5)

x0 - equilibrium bond distance for x (default value 0.0), y0 - equilibrium bond distance for y (default value 0.0), z0 - equilibrium bond distance for z (default value 0.0)

prec - precision (default value 6)

Jupyter notebook link:

<https://notebooks.finki.ukim.mk/user/user1/notebooks/SchrodingerAPI.ipynb>

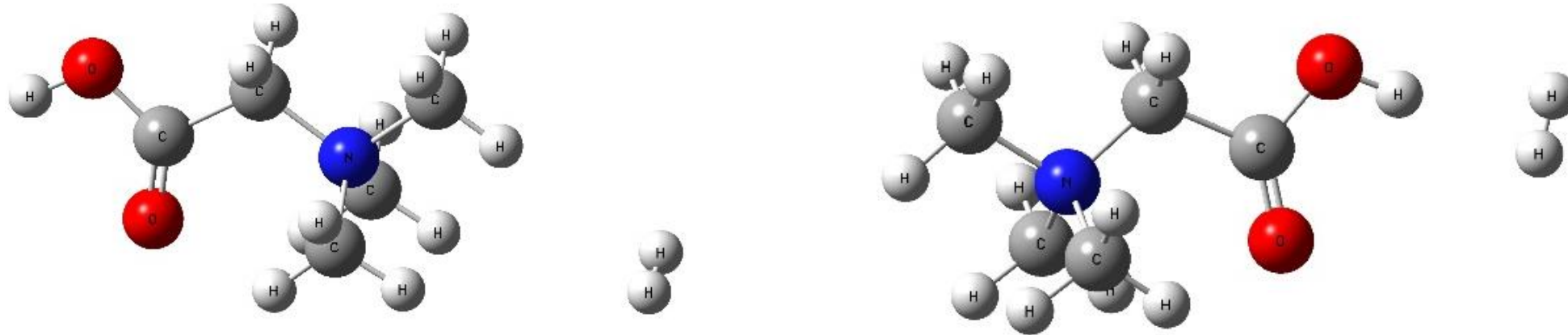
User: user1

Password: User1DEMO

Practical Example



- Understanding the H₂ – tagging of biomolecules by theoretical anharmonic vibrational spectroscopy with standard and long-range corrected exchange-correlation functionals



$$\vec{r}(O) = \frac{m_H}{m} \cdot \vec{r} \quad \vec{r}(H) = \frac{m_O}{m} \cdot \vec{r} \quad \vec{r} = \vec{r}(H) - \vec{r}(O) \quad V = f(r_{OH})$$

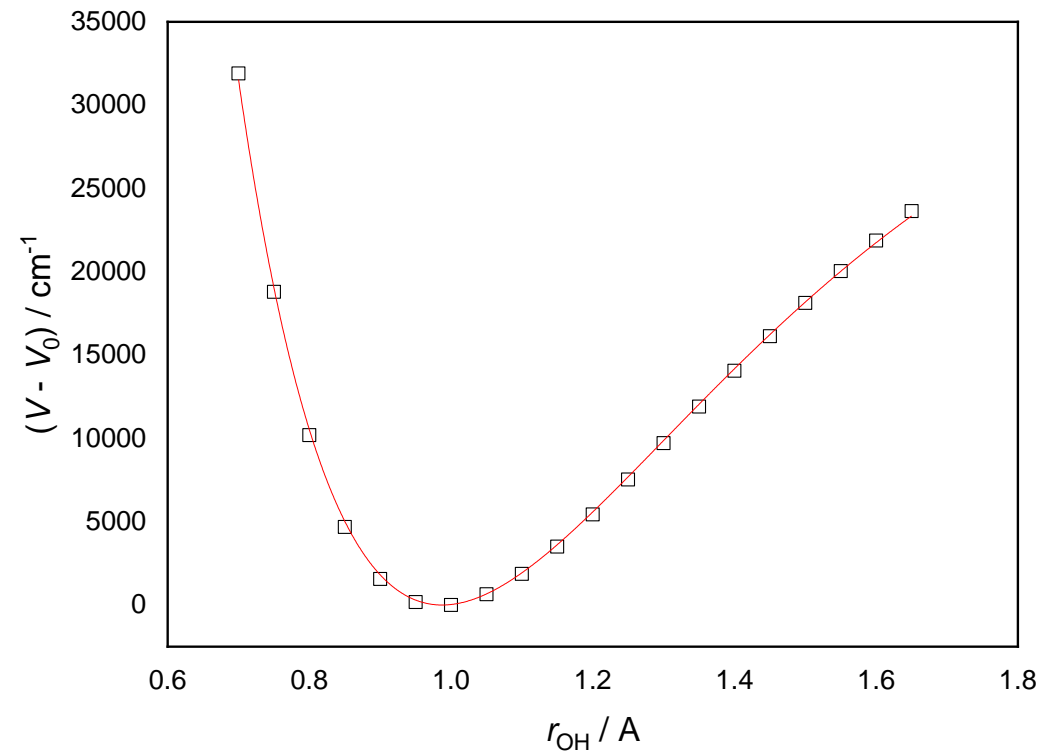
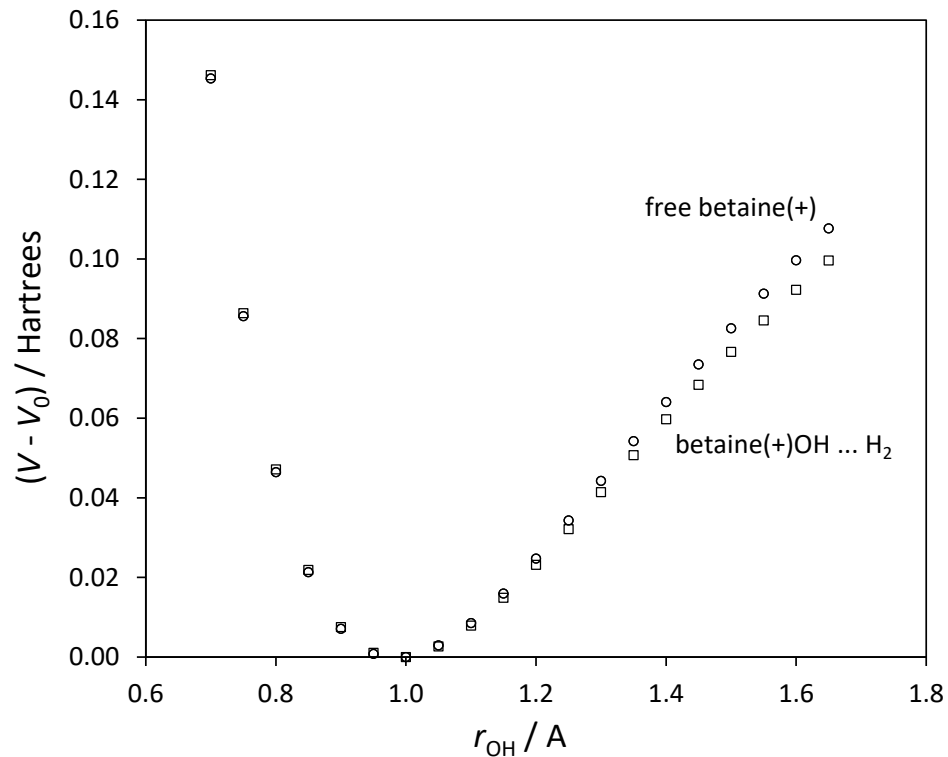
$$V(r_{OH}) = D \cdot \left\{ 1 - \exp[-a \cdot (r_{OH} - r_{OH,0})] \right\}^2 - D$$

- A diatomic intramolecular oscillator may be conveniently described by a Morse potential with the following parameters:
 - $D = 0.176$ (a.u.)
 - $a = 1.02$ (a.u.)⁻¹
 - $x_0 = 1.4$ (a.u.)
- Find the ground state energy level of this oscillator, as well as the first two excited levels

```
import requests
response = \
requests.get('https://schrodinger.chem-api.finki.ukim.mk/ldHermiteMorse?npts=20&D=0.176&a=1.02&x0=1.4&prec=32')
if response.status_code == 200:
    print(response.content.decode('utf-8'))
else:
    print("None")
```

```
[3.8921199517049934e-02 3.9038990155621445e-01 9.7429027058233031e-01 1.7762049809339091e+00 2.7957732113286857e+00
4.0424446176786830e+00 5.5349419642707067e+00 7.3058068540176428e+00 9.4075531043067695e+00 1.1922958051944608e+01 1.
5257271309631083e+01 1.9669185870427675e+01 3.9962671614302174e+01 1.0407010971408560e+02 2.9596256113277428e+02 8.85
27537847856104e+02 2.7912331820222394e+03 9.4889099112009353e+03 3.6531903728941070e+04 1.8117696113249165e+05]
```

- Understanding the H₂ – tagging of biomolecules by theoretical anharmonic vibrational spectroscopy with standard and long-range corrected exchange-correlation functionals



- Understanding the H₂ – tagging of biomolecules by theoretical anharmonic vibrational spectroscopy with standard and long-range corrected exchange-correlation functionals

	Static approach			ADMP	Exp.	HCTH (static)
	B3LYP	CAM-B3LYP	DFTB(A)	DFTB(A)		
Betaine(+)	3534.6	3575.6	3486.7	3473.8	3554	3509.1
Betaine(+) OH...H ₂	3447.7	3483.0	3412.7	3405.2	3514	3466.0
Δv	-86.9	-92.6	-74.0	-68.6	-40	-43.1
Betaine(+) CH ₃ ...H ₂	3551.4	3593.1	3486.9	3475.9	3558	3525.8
Δv	16.8	17.5	0.2	2.1	4	16.7

Gaussian API



Scientific Background



The Density Functional Tight Binding (DFTB) Approach in Computational Chemistry and Physics

- The advent of density functional theory (DFT) – based methods has revolutionized the theoretical chemistry and physics community.
- DFT enabled us to significantly extend the size range of the studied molecular systems with exact quantum theoretical approaches.
- Yet another breakthrough in the field has been enabled with the development of the density functional tight binding (DFTB) approximation to DFT.

The Density Functional Tight Binding (DFTB) Approach in Computational Chemistry and Physics

- Starting from the Harris-Foulkes energy expression for the total energy in the framework of DFT, given by the equation (1) below, one can smoothly introduce the DFTB approximation thereof.

$$E[\rho] = \sum_i f_i \epsilon_i - \frac{1}{2} \int V_H[\rho] \rho(\vec{r}) d\vec{r} - \int V_{xc}[\rho] \rho(\vec{r}) d\vec{r} + E_{xc}[\rho] + E_{NN} \quad (1)$$

- In (1), ρ denotes the electron density, f_i the orbital occupation numbers, V_H and V_{xc} are the Hartree and the exchange-correlation potentials, while E_{NN} is the nucleus-nucleus repulsive energy component.

The Density Functional Tight Binding (DFTB) Approach in Computational Chemistry and Physics

$$E[\rho] = \sum_i f_i \epsilon_i - \frac{1}{2} \int V_H[\rho] \rho(\vec{r}) d\vec{r} - \int V_{xc}[\rho] \rho(\vec{r}) d\vec{r} + E_{xc}[\rho] + E_{NN} \quad (1)$$

- The first term in (1), corresponding to the sum of single-particle energies is often referred to as the band-structure energy (E_{BS}).
- While the Harris-Foulkes energy expression becomes identical to the more common Kohn-Sham formula if one sets ρ as the ground state density (ρ_{GS}), in case when an approximate reference density is used ($\rho_0 \approx \rho_{GS}$), the two formalisms are different.

The Density Functional Tight Binding (DFTB) Approach in Computational Chemistry and Physics

$$E[\rho] = \sum_i f_i \epsilon_i - \frac{1}{2} \int V_H[\rho] \rho(\vec{r}) d\vec{r} - \int V_{xc}[\rho] \rho(\vec{r}) d\vec{r} + E_{xc}[\rho] + E_{NN} \quad (1)$$

- If ρ_0 is constructed as a superposition of isolated atomic densities subject to a confinement potential then the first term in (1) can be computed using a nonempirical determination of all matrix elements in the tight-binding approximation.
- The repulsive part of the potential is, however, still empirical. Such scheme (known as the DFTB0 in the current parlance) has shown great success for most of the covalently bonded systems. In polar or partially ionic systems, on the other hand, the approximation completely breaks down.

The Density Functional Tight Binding (DFTB) Approach in Computational Chemistry and Physics

- This is due to the fact that, in the later systems, it is not possible to construct ρ_0 from isolated neutral atomic densities.
- To overcome this deficiency, the self-consistent charge (SCC) approach has been introduced, which includes the density fluctuations effects ($\delta\rho$) to the second or third order – the two approaches are nowadays known under the acronyms DFTB2 and DFTB3.
- In these two approaches, the total energy is expressed in the form:

$$E_{DFTB}[\rho] = E_{BS}[\rho_0, \delta\rho] + E_{Coul}[\delta\rho] + E_{rep}[\rho_0] \quad (2)$$

The Density Functional Tight Binding (DFTB) Approach in Computational Chemistry and Physics

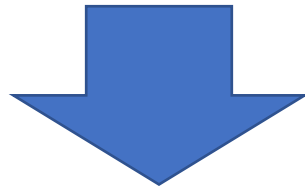
$$E_{DFTB}[\rho] = E_{BS}[\rho_0, \delta\rho] + E_{Coul}[\delta\rho] + E_{rep}[\rho_0] \quad (2)$$

- In (2), $E_{Coul}[\delta\rho]$ is the screened Coulomb term (between partial atomic charges) plus a Hubbard-like energy term for partially charged atoms.
- $E_{rep}[\rho_0]$, on the other hand, is the repulsive energy.
- It is the key component on which we will further on focus our attention in the present lecture.

The Repulsive Potential in DFTB

- Starting from the equation (1), the repulsive potential can be derived by removal of the first term in parallel to including the reference density:

$$E[\rho] = \sum_i f_i \epsilon_i - \frac{1}{2} \int V_H[\rho] \rho(\vec{r}) d\vec{r} - \int V_{xc}[\rho] \rho(\vec{r}) d\vec{r} + E_{xc}[\rho] + E_{NN} \quad (1)$$



$$E_{rep.}[\rho_0] = E_{NN} - \frac{1}{2} \int \int \frac{\rho_0(\vec{r}) \rho_0(\vec{r}')}{|\vec{r} - \vec{r}'|} d\vec{r} d\vec{r}' - \int V_{xc}[\rho_0] \rho_0(\vec{r}) d\vec{r} + E_{xc}[\rho_0] \quad (3)$$

- The second term in (3) is the Hartree energy of ρ_0 .

The Repulsive Potential in DFTB

- Equation (3), however, is far from practical and at the same time not so accurate, as the exchange-correlation and Hartree potentials are seldom explicitly calculated.
- It is therefore much more convenient to approximate $E_{rep.}$ with short-range atom pairwise contributions, so that it acquires the following form:

$$E_{rep.} = \frac{1}{2} \sum_I \sum_{J(\neq I)} V_{rep.}(R_{IJ}) \quad (4)$$

- The summation in (4) is carried out over all atom pairs, while R_{IJ} is the interatomic distance within the pair.

The Repulsive Potential in DFTB

- In this manner, the short-range potential:
 - a) depends only on interatomic distances;
 - b) has to be defined for each possible combination of elements.
- To enforce the short-range character of the potential defined by (4), a cutoff radius is defined, so that the following condition is fulfilled:

$$V_{rep.}(R > R_{cut.}) = 0 \quad (5)$$

The Repulsive Potential from Gaussian Process Regression (GPR) – [1]

- In reference [1], the $V_{\text{rep.}}$ formulation within the Bayesian machine learning (ML) framework of GPR has been presented.
- The derivation has been essentially based on the GAP approach [2].

[1]. C. Panosetti, A. Engelmann, L. Nemeč, K. Reuter, J. T. Margraf, J. Chem. Theory Comput. 2020, 16, 2181-2191.

[2]. A. P. Bartok, M. C. Payne, R. Kondor, G. Csanyi, Phys. Rev. Lett. 2010, 104, 136403.

The Repulsive Potential from Gaussian Process Regression (GPR)

- Within the GPR framework, $V_{rep.}$ can be modelled as a linear combination of kernel (covariance) functions:

$$V_{rep.}(R) = \sum_{I,J \in \{X\}} \alpha_{IJ} k(R, R_{IJ}) \quad (6)$$

- The sum in (6) spans over all pairs within the set of reference structures $\{X\}$.
- α_{IJ} denote the regression coefficients, while $k(R, R_{IJ})$ are the kernel functions.

The Repulsive Potential from Gaussian Process Regression (GPR)

- Now, $V_{rep.}$ is fitted so that the differences between the DFTB model (repulsion-less one) and DFT energies and forces (appropriately normalized). Ideally:

$$E_{rep.} = \frac{1}{2} \sum_I \sum_{J(\neq I)} V_{rep.}(R_{IJ}) = E_{DFT} - E_{DFTB} \quad (7)$$

- But what are the advantages of using GPR as compared to “conventional” potential fitting?
- There exist a linear algebraic expression for the coefficients α_{ij} in a closed form.

The Repulsive Potential from Gaussian Process Regression (GPR)

- This expression has been derived so that it minimizes the loss function:

$$L = \sum_{I,J \in \{X\}} \left(t_{IJ} - V_{rep.}(R_{IJ}) \right)^2 + \sigma_n \|\alpha\|^2 \quad (8)$$

- In (8), t_{IJ} are the reference values for the repulsive potential (*i.e.* the target ones), while the role of the parameter σ_n is to function as a reference data uncertainty measure, and at the same time, as a means of overfitting avoidance.

The Repulsive Potential from Gaussian Process Regression (GPR)

- The expression for calculation of the coefficients is:

$$\alpha = (K + \sigma_n^2 \mathbf{1})^{-1} y \quad (9)$$

- In (9), K is the covariance matrix with elements $k(R_{IJ}, R_{KL})$ for all pairs of atoms I, J and K, L in $\{X\}$; y , on the other hand, is the vector containing the target values t_{IJ} (energies, or, in certain cases, forces).
- A notable problem in application of the GPR procedure to $V_{\text{rep.}}$ is that one can only calculate the reference values for the overall $E_{\text{rep.}}$, not for the individual components $V_{\text{rep.}}(R_{IJ})$.

The Repulsive Potential from Gaussian Process Regression (GPR)

- An elegant solution to the aforementioned issue is the GAP procedure, more specifically using a sparse formulation of GPR.
- The total repulsive energy $E_{\text{rep.}}$ is first represented as a linear combination of pairwise potentials $V_{\text{rep.}}(R_{ij})$ which are unknown.
- A set of N_s sparse training points is defined ($N_s < N_{\text{pairs}}$) as a representative sample of the training set.
- A new covariance matrix is constructed projecting the full data set onto the sparse points – the fitting thus now requires inversion of only $N_s \times N_s$ matrix.

The GAP machinery

- The kernel function within the GAP procedure – $k(R, R')$ is defined as a similarity measure between two inputs. The most common choice is the squared exponential kernel of the form:

$$k^{SE}(R, R') = \exp\left(\frac{-|R-R'|^2}{\theta^2}\right) \quad (10)$$

- In (10), θ is a length-scale parameter. It is essentially a measure of the “generosity” of the kernel in the course of comparing two inputs.
- To ensure a smooth decay of $V_{\text{rep.}}(R)$ to zero at $R = R_{\text{cut}}$, the SE kernel is multiplied by a damping function.

The GAP machinery

- The overall damped kernel function thus takes the form:

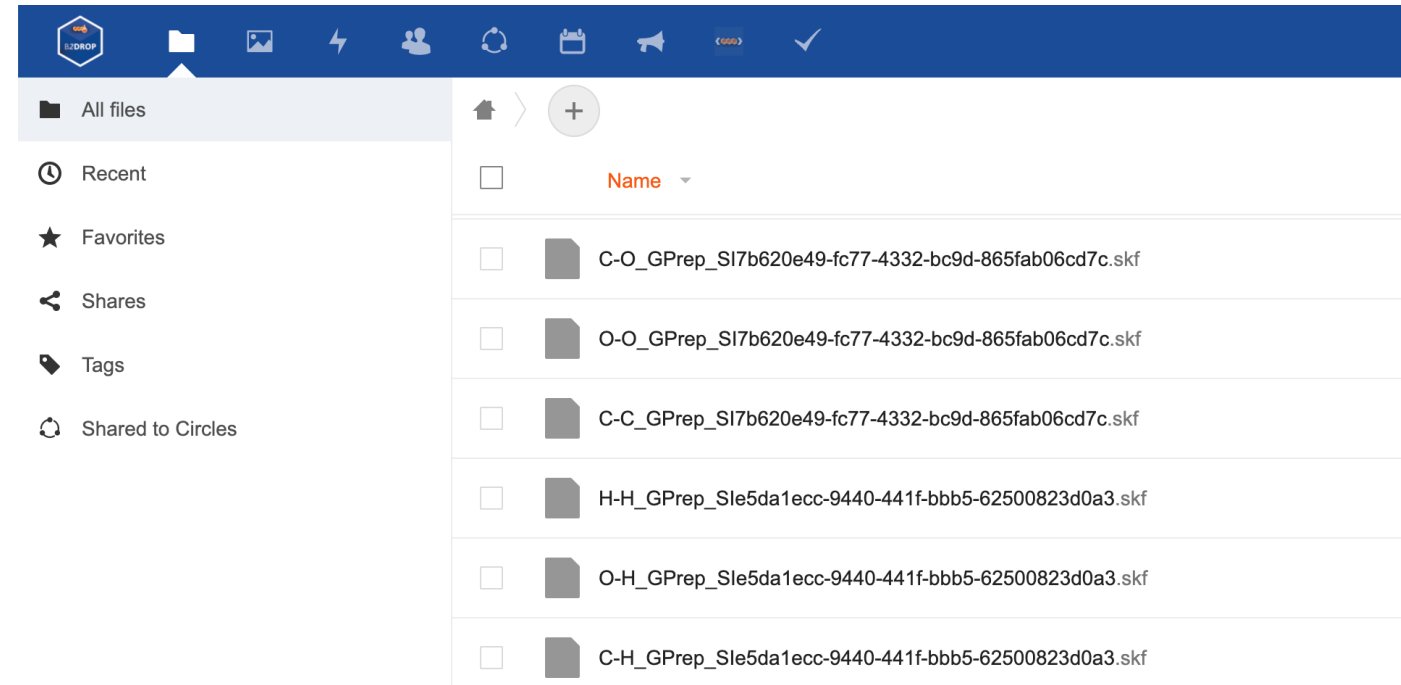
$$k^{damp}(R, R') = e^{-\beta R} f_{cut}(R) k^{SE}(R, R') \quad (11)$$

- where:

$$f_{cut}(R) = \begin{cases} 1 & R \leq R_{cut} - d \\ \frac{\cos\left(\pi \frac{R - R_{cut} + d}{d}\right) + 1}{2} & R_{cut} - d < R \leq R_{cut} \\ 0 & R > R_{cut} \end{cases} \quad (12)$$

- It provides two methods:
 - GPrep (POST method) and
 - GPrepRemote (GET method).
- In case of **GPrep**, the user should provide an input file by browsing the file system on the local device, while in case of **GPrepRemote** the user should provide a public URL where the input file can be accessed.
 - The provided URL should be a direct link to a public file (https://gaussian.chem-api.finki.ukim.mk/static/reference_data.xyz) or public **Dropbox** link (https://www.dropbox.com/s/qnk7r3ey6pkfzb9/reference_dataB.xyz?dl=0) or public **B2DROP** link (<https://b2drop.eudat.eu/s/QWPRFGwYHEno99P>).

- Secure and trusted data exchange service for researchers and scientists to keep their research data synchronized and up-to-date and to exchange with other researchers.
- The output Slater-Koster files (.skf) with potentials will be uploaded to the user **B2DROP** account
<https://b2drop.eudat.eu/apps/files/>
- The user should log in to **B2DROP** (preferably by using her/his institutional account) and to generate username and password.



- **file** - reference data file from which the relevant forces and pair distances are extracted
 - **GPrep** method: user should upload the file (*an example file can be found [here](#)*)
 - **GPrepRemote** method: user should provide public URL of the file (*default value: https://gaussian.chem-api.finki.ukim.mk/static/reference_data.xyz*)
- **sigma** - data noise standard deviation (*default value 0.05*)
- **beta** - exponential damping factor (*default value 3.0*)
- **theta** - latent function length scale (*default value 1.0*)
- **delta** - latent function standard deviation (*default value 1.0*)
- **d** - cutoff transition width (*default value 1.0*)
- **c** - cutoff (*default value = 5.0*)
- **N** - number of data points (*default value = 100*)
- **b2dropUsername** - **B2DROP** generated username – to upload the result files
- **b2dropPassword** - **B2DROP** generated password - to upload the result files

GPrep (POST method)

- Upload the input file directly on the Gaussian API home page (*an example file can be found [here](#)*)

Request body

file
string(\$binary) reference_dataB.xyz
 Send empty value

Execute

Server response

Code	Details
200	<p>Response body</p> <pre>Files were successfully uploaded to b2drop.</pre>

POST /GPrep

Parameters

Name	Description
sigma number(\$double) (query)	Default value : 0.05 <input type="text" value="0.05"/>
beta number(\$double) (query)	Default value : 3 <input type="text" value="3"/>
theta number(\$double) (query)	Default value : 1 <input type="text" value="1"/>
delta number(\$double) (query)	Default value : 1 <input type="text" value="1"/>
d number(\$double) (query)	Default value : 1 <input type="text" value="1"/>
c number(\$double) (query)	Default value : 5 <input type="text" value="5"/>
N integer(\$int32) (query)	Default value : 100 <input type="text" value="100"/>
b2dropUsername string (query)	Default value : username <input type="text" value="username"/>
b2dropPassword string (query)	Default value : password <input type="text" value="password"/>

GPrepRemote (GET method)

- B2DROP username and password must be entered in the b2dropUsername and b2dropPassword textboxes in order to receive the output files (to be uploaded on the user's B2DROP account).

GET /GPrepRemote

Parameters

Name	Description
file string <small>(query)</small>	<input type="text" value="https://gaussian.chem-api.finki.ukim.mk/static"/>
sigma number(\$double) <small>(query)</small>	<input type="text" value="0.05"/>
beta number(\$double) <small>(query)</small>	<input type="text" value="3"/>
theta number(\$double) <small>(query)</small>	<input type="text" value="1"/>
delta number(\$double) <small>(query)</small>	<input type="text" value="1"/>
d number(\$double) <small>(query)</small>	<input type="text" value="1"/>
c number(\$double) <small>(query)</small>	<input type="text" value="5"/>
N integer(\$int32) <small>(query)</small>	<input type="text" value="100"/>
b2dropUsername string <small>(query)</small>	<input type="text" value="username"/>
b2dropPassword string <small>(query)</small>	<input type="text" value="password"/>

- Other way to use this REST API method is to access it directly from the browser address bar.

https://gaussian.chem-api.finki.ukim.mk/GPrepRemote?b2dropUsername=YOUR_B2DROPPASSWORD&b2dropPassword=YOUR_B2DROPPASSWORD&file=YOUR_FILE_LOCATION

- *If user prefers to change other parameters, they can be added as **&PARAMETER=VALUE***
- Another option is to consume this method in a program source code (Python)

```
import requests
response = requests.get('https://gaussian.chem-api.finki.ukim.mk/GPrepRemote?b2dropUsername=YOUR_B2DROPPASSWORD\
&b2dropPassword=YOUR_B2DROPPASSWORD&file=https://gaussian.chem-api.finki.ukim.mk/static/reference_data.xyz\
&sigma=0.05&beta=3.0&theta=1.0&delta=1.0&d=1.0&c=5.0&N=100')
if response.status_code == 200:
    print(response.content.decode('utf-8'))
else:
    print("None")
```

- Gaussian Swagger link:

<https://gaussian.chem-api.finki.ukim.mk/>

- Jupyter notebook link:

<https://notebooks.finki.ukim.mk/user/user1/notebooks/GaussianAPI.ipynb>

- **User:** user1
- **Password:** User1DEMO

The service demo videos can be seen on the next slides

- Bojana Koteska
 - bojana.koteska@finki.ukim.mk
- Ljupco Pejov
 - ljupcop@pmf.ukim.mk
- Anastas Mishev
 - anastas.mishev@finki.ukim.mk

