

op_tools ver 0.1.3 ユーザーマニュアル

土居 英男

Feb 2019

目次

これは何？	2
インストール	2
必要なデータ	2
すべてのオーダーパラメータで共通の設定	2
隣接粒子の設定	2
隣接半径を指定する場合	3
隣接粒子の数を指定する場合	3
隣接粒子で平均化を行う回数の設定	3
座標の取扱	4
各オーダーパラメータの説明	5
<i>S</i> : Onsager' s parameter	5
<i>T</i> : McMillan' s Sigma	5
<i>A</i> : neighborhood parameters	6
<i>B</i> : bond angle analysis (BAA)	7
<i>C</i> : centrosymmetry parameter analysis (CPA)	7
<i>D</i> : neighbor distance analysis (NDA)	8
<i>F</i> : angular Fourier series (AFS) like parameter	8
<i>H</i> : angle histogram analysis (AHA)	9
<i>I</i> : tetrahedron order parameter (TOP)	10
<i>Q</i> : Bond order parameter	10
<i>W</i> : Bond order parameter	11
<i>Q1, W1</i> : facet weighting Bond order parameter	13
<i>Q2, W2</i> : function weighting Bond order parameter	14
<i>LQ, LW</i> : local Bond order parameter	15
<i>Z</i> : user define order parameter	16
大量の解析を行う設定	16
出力のフォーマット	17
Reference	18

これは何？

分子シミュレーションや粒子のシミュレーションにおける結晶構造を解析するための python モジュールである。

個々の粒子や分子の周囲の環境を評価し、ある一つの数値に変換する。この数値はオーダーパラメータ (order parameter) と呼ぶ。似通った並び方の粒子は同じ数値になり、違った並び方をしている粒子は違う数値になる。例えば、体心立方格子 (body-centered cubic, BCC) や 面心立方格子 (face-centered cubic, FCC) は数値が知られており、どこが結晶で、どこがそうでないかを見分ける事が可能である。このような解析は、固体金属を融解させるようなシミュレーションや、結晶構造の粒界を調べるシミュレーションなどで使われる。オーダーパラメータの性能は、低い計算コスト、分類能力など評価される。

インストール

所定のサイトからパッケージをダウンロードし、次のコマンドを実行してください。

```
$ cd order_tools
$ pip3 install -e .
```

必要なデータ

1. coord : 各粒子の座標のリスト。次のような。[[0,0,0], [0,0,1], ...]
[[x1, y1, z1], [x2, y2, z2], ...] というような順番になっている。
2. direct : 各粒子の方向ベクトル。3次元の単位ベクトルでも、四元数でも良い。次のような。[[1,0,0], ...] か [[1,0,0,0], ...] 3次元の単位ベクトルの場合は、[x, y, z] の順番になっている。四元数の場合は、[w, x, y, z] の順番になっている。質点や球状粒子の場合、方向ベクトルが存在しない。その場合は、direct = [] として問題ない。
3. box_length : シミュレーションボックスの長さ。[3, 3, 3] xyz の順番になっている。
4. op_settings : 各 order parameter を計算するための設定

すべてのオーダーパラメータで共通の設定

隣接粒子の設定

各粒子のオーダーパラメータの数値は近くに存在する粒子の座標を使用して計算される。このある粒子 i の近くにある粒子のことを、ある粒子の隣接粒子と呼ぶ。隣接粒子の決める方法は2つある。

一つめの方法は、粒子 i からの距離がある距離 (隣接半径) 以内にある粒子を、粒子 i の隣接粒子とする方法である。この方法の利点は、実装が比較的簡単でユーザーにとってもわかりやすいと思われるところである。欠点としては、隣接粒子の個数が各粒子で違う場合がある事と、距離の単位によって隣接粒子数が変わる事である。例えば、隣接半径を 1 nm と 1 angstrom と指定するのでは結果が変わる事である。このことをスケール依存性という。もう一つの方法は、粒子 i から距離の近い n 個の粒子を隣接粒子とする方法である。利点としては、スケール依存性がなく、長さの単位が変わっても距離の大小関係が変わらない

め、隣接粒子が変化しない。欠点としては、実装が比較的難しい、計算コストが大きい（数割程度）、論文での使用実績が存在するが少ない、という点である。

この両方の方法を、`op_tools` では実装しており使用できる。他の方法も存在するが、まだ実装していない。各方法の使用法は、以下の通りである。

隣接半径を指定する場合

オーダーパラメータの計算条件を指定する `op_settings` で次のように指定を行う。

```
op_settings = {  
    'radius' : [1.5, 1.75] ,  
    ... ( 各オーダーパラメータの設定 ) }
```

list 形式で指定することが可能で、複数の条件を一度に指定する事も可能である。上の例では、距離 1.5 と 距離 1.75 での指定を行っている。

隣接粒子の数を指定する場合

オーダーパラメータの計算条件を指定する `op_settings` で次のように指定を行う。

```
op_settings = {  
    'neighbor' : [8, 12] ,  
    ... ( 各オーダーパラメータの設定 ) }
```

list 形式で指定することが可能で、複数の条件を一度に指定する事も可能である。上の例では、隣接粒子数 8 と 12 の指定を行っている。

隣接粒子で平均化を行う回数の設定

各粒子のオーダーパラメータの分類性能が低い場合は、以下の様な平均の計算を行う。粒子 i のオーダーパラメータと粒子 i の隣接粒子のオーダーパラメータで平均を計算する場合がある。粒子 i の N 個の隣接粒子の list を $N_b(i)$ 、 $N_b(i)$ に粒子 i 自身を追加した $N + 1$ 個の粒子の list を $\tilde{N}_b(i)$ とすると、粒子 i の平均化を行う前のオーダーパラメータを $X(i)$ 、平均化した後のオーダーパラメータを $Y(i)$ とすると、次の様な計算を行う。

$$Y(i) = \frac{\sum_{j \in \tilde{N}_b(i)} X(j)}{N + 1}$$

この平均化の操作は以下の利点と欠点を持っている。利点としては、平均を計算するだけであるため計算負荷が軽い事、同じオーダーパラメータの分類性能を上げる事が可能である事、何らかの数値であれば使用できる事、何回も平均化をする事が可能である事、である。一方、欠点としては、オーダーパラメータの計算に使用する情報が多くなり、オーダーパラメータの解像度が下がる事である。オーダーパラメータは隣接粒子の情報を使って計算され、さらに平均化の計算では隣接粒子のオーダーパラメータを使って計算される。平均化操作を 1 回行うごとに、隣接粒子の隣接粒子の情報を使用することになる。つまり、オーダーパラメータの計算に使用する粒子が多くなり、遠くの粒子の情報を使用することになる。これは、均一な系ではあまり問題にはならないが、不均一な系（界面など）では問題になる。不均一な系では、異常

が起こっている位置を知りたいであるとか、界面の位置を知りたいなどと言った目的で利用される場合が多いと思われるが、遠くの粒子の情報を使用すると、位置がはっきりとはわからなくなる。

この平均化を行う回数は、オーダーパラメータの計算条件を指定する `op_settings` で次のように指定を行う。なお、この値は、0 から始まり、0 の場合は、平均化を行わない。

```
op_settings = {
    'ave_times'      : 1,
    ... ( 各オーダーパラメータの設定 )
}
```

座標の取扱

このソフトウェアは、質点や球状粒子ではなく、方向の情報を持っている粒子（楕円体、棒、矢印、ディスクなど）を解析するために開発された。もちろん、質点や球状粒子の解析も可能である。質点や球状の粒子ではない場合、解析に使用したい座標が変わる可能性がある。例えば、棒状の粒子であれば、粒子の中心の座標で解析を行いたい場合や、先端の座標で解析を行いたい場合が考えられる。また、ある粒子 i の座標は中心を隣接粒子 j の座標は先端を使用したい場合なども考えられる。そのため、`op_tools` では粒子の座標のため次の式を使用する。

$$r^{(o_i)}(i) = r(i) + o_f o_i u(i)$$

ここで、ベクトル $r^{(o_i)}(i)$ は、解析に使用する粒子 i の座標ベクトル、ベクトル $r(i)$ は、python モジュールに引数として与えられた粒子 i の座標ベクトルである。変数 o_f は方向ベクトルの係数で、変数 o_i は粒子 i での位置を示す変数、ベクトル $u(i)$ は粒子 i の持つ方向ベクトルである。変数 o_f は解析全体で用いる方向ベクトルの強さを表現している。変数 o_i は、粒子 i の座標ベクトルの位置を表現するために用いている。通常、 $o_i \in \{1, 0, -1\}$ のようになっており、 $o_f = 1, o_i = 1$ の時は、ベクトル $r^{(o_i)}(i)$ は方向ベクトルの先端の座標になる。また、 $o_f = 1, o_i = -1$ の時は、ベクトル $r^{(o_i)}(i)$ は方向ベクトルの後端の座標になる。このように、方向ベクトルのどの座標を使うか切り替える事ができる。尚、質点や球状粒子の場合は、 $o_f = 0, o_i = 0$ とすれば良い。

楕円体の場合は、オーダーパラメータの計算条件を指定する `op_settings` で次のように指定を行う。

```
op_settings = {
    ... ( 各オーダーパラメータの設定 ) ...
    'o_factor'      : [1.0 ],
    'oi_oj'         : [1, 0, -1],
}
```

この例では、座標として粒子の中心座標を入力とした場合、楕円体の先端、中心、後端の3通りを座標として使用したい場合の設定である。

質点または球状粒子の場合は、オーダーパラメータの計算条件を指定する `op_settings` で次のように指定を行う。

```
op_settings = {
    ... ( 各オーダーパラメータの設定 ) ...
    'o_factor'      : [0],
    'oi_oj'         : [0],
}
```

}

また、粒子 i の座標ベクトル $r^{(o_i)}(i)$ から粒子 j の座標ベクトル $r^{(o_j)}(j)$ は以下のように計算する。

$$r^{(o_i, o_j)}(i, j) = r^{(o_j)}(j) - r^{(o_i)}(i)$$

各オーダーパラメータの説明

S : Onsager' s parameter

Onsager のオーダーパラメータ S は次の式で計算される。 [1][2]

$$S^{(a=0, n)}(i) = \frac{\sum_{j \in N_b(i)} P_n(\cos(\theta))}{N}, \text{neven.}$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。関数 P_n は Legendre polynomial である。変数 n は Degree of the polynomial である。変数 θ は、粒子 i の持つ方向ベクトル $u(i)$ と粒子 j の持つ方向ベクトル $u(j)$ との角度である。変数 $\cos(\theta)$ は通常、 $u(i) \cdot u(j)$ で計算される。

$n = 2, 4$ の時、オーダーパラメータ S はそれぞれ以下の式で計算される。

$$S^{(a=0, n=2)}(i) = \frac{\sum_{j \in N_b(i)} [3 \cos^2(\theta) - 1]/2}{N}$$

$$S^{(a=0, n=4)}(i) = \frac{\sum_{j \in N_b(i)} [35 \cos^4(\theta) - 30 \cos^2(\theta) + 3]/8}{N}$$

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの設定として以下の例を示す。尚、このオーダーパラメータは計算のために方向ベクトルが必須であるため、質点や球状粒子の解析に使う事はできない。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'n_in_S' : [2],
    'analysis_type' : ['S']
}
```

T : McMillan' s Sigma

McMillan のパラメータ T は次の式で計算される。 [3]

$$T^{(a=0, n, o_i, o_j)} = \frac{\sum_{j \in N_b(i)} \cos(2\pi z^{(o_i, o_j)}(i, j)/d) P_n(\cos(\theta))}{N}$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。関数 P_n は Legendre polynomial である。変数 n は Degree of the polynomial である。変数 θ は、粒子 i の持つ方向ベクトル $u(i)$ と粒子 j の持つ方向ベクトル $u(j)$ との角度である。変数 $\cos(\theta)$ は通常、 $u(i) \cdot u(j)$ で計算される。変数 $z^{(o_i, o_j)}(i, j)$ は、ベクトル $r^{(o_i)}(i)$ を通過し、ベクトル $u(i)$ と垂直な平面 P から、ベクトル $r^{(o_j)}(j)$ までの距離である。変数 d は、層から層までの距離である。

$n = 2$ の時、オーダーパラメータ T は以下の式で計算される。

$$T^{(a=0, n=2, o_i, o_j)} = \frac{\sum_{j \in N_b(i)} \cos(2\pi z^{(o_i, o_j)}(i, j)/d) [3 \cos^2(\theta) - 1]/2}{N}$$

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。尚、このオーダーパラメータは計算のために方向ベクトルが必須であるため、質点や球状粒子の解析に使う事はできない。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'n_in_T': [2],
    'd_in_T': [1.0],
    'analysis_type' : ['T']
}
```

A : neighborhood parameters

common neighborhood parameter A [4]、predominant common neighborhood parameter P [5]、another predominant common neighborhood parameter N [5] は非常に似ている式で計算される。 P と N に関しては、 A の一種として実装されている。

$$A_i^{(a=0, type=A, m, o_i, o_j, o_k)} = A_i^{(a=0, m, o_i, o_j, o_k)} = \frac{1}{N} \sum_{j \in N_b(i)} \left| \sum_{k \in N_b(i, j)} \{r^{(o_i, o_k)}(i, k) + r^{(o_j, o_k)}(j, k)\} \right|^2$$

$$A_i^{(a=0, type=P, m, o_i, o_j, o_k)} = P_i^{(a=0, m, o_i, o_j, o_k)} = \frac{1}{N} \sum_{j \in N_b(i)} \left| \sum_{k \in N_b(i, j)} \{r^{(o_i, o_k)}(i, j) + r^{(o_j, o_k)}(k, j)\} \right|^2$$

$$A_i^{(a=0, type=N, m, o_i, o_j, o_k)} = N_i^{(a=0, m, o_i, o_j, o_k)} = \frac{1}{N} \left| \sum_{j \in N_b(i)} \sum_{k \in N_b(i, j)} \{r^{(o_i, o_k)}(i, j) + r^{(o_j, o_k)}(k, j)\} \right|^2$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $N_b(i, j)$ は粒子 i , 粒子 j から近い m 個の粒子のリストである。変数 $type$ で計算を行うオーダーパラメータの種類を指定する変数である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_param = {
    'neighbor': [12],
    'radius': [1.5],
    'ave_times': 1,
}
```

```

'oi_oj': [0],
'o_factor': [0.00],
'm_in_A': [2],
'types_in_A': ['A', 'P', 'N'],
'analysis_type': ['A']}

```

B : bond angle analysis (BAA)

bond angle analysis B は次の式で計算される。 [6]

$$B_i^{(a=0,m,n,\phi,o_i,o_j,o_k)} = \frac{1}{N(N-1)/2} \sum_{j>k \in N_b(i)} f^{(m,n,\phi,o_i,o_j,o_k)}(\theta^{(o_i,o_j,o_k)}(j,i,k))$$

$$f^{(m,n,\phi,o_i,o_j,o_k)}(\theta(j,i,k)) = \cos^n(m\theta^{(o_i,o_j,o_k)}(j,i,k) + \phi)$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\theta^{(o_j,o_i,o_k)}(j,i,k)$ はベクトル $r^{(o_i,o_j)}(i,j)$ とベクトル $r^{(o_i,o_k)}(i,k)$ との角度である。変数 m は、角度の係数である。変数 n は、 \cos 関数に使用する指数である。変数 ϕ は、関数の補正のために角度である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```

op_param = {
  'neighbor': [12],
  'radius': [1.5],
  'ave_times': 1,
  'oi_oj': [0],
  'o_factor': [0.00],
  'm_in_B': [2],
  'phi_in_B': [0],
  'n_in_B': [1, 2],
  'analysis_type': ['B']}

```

C : centrosymmetry parameter analysis (CPA)

centrosymmetry parameter C は次の式で計算される。 [7]

$$C_i^{(a=0,o_i,o_j,o_k)} = \sum_{j \in M_b(i)} |r^{(o_i,o_j)}(i,j) + r^{(o_i,o_k)}(i,k)|^2$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $M_b(i)$ は粒子 i の $N/2$ 個の隣接粒子のリストである。

粒子 k の計算の方法は 2 通り実装している。座標 $r^{(o_j)}(j)$ の、ベクトル $r^{(o_i)}(i)$ に関して反対側に移動した $r^{(o_j)}(j)$ から一番近い位置にある $N_b(i)$ リストにある粒子である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```

op_param = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'analysis_type': ['C'] }

```

D : neighbor distance analysis (NDA)

neighbor distance analysis D は次の式で計算される。 [8]

$$D_i^{(a=0, o_i, o_j, o_k, f_{ij}, f_{ik}, f_{jk})} = \frac{1}{N(N-1)/2} \sum_{j>k \in N_b(i)} f_{ij}(r^{(o_i, o_j)}(i, j)) f_{ik}(r^{(o_i, o_k)}(i, k)) f_{jk}(r^{(o_j, o_k)}(j, k))$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。関数 $f_{ij}(r(i, j)), f_{ik}(r(i, k)), f_{jk}(r(j, k))$ は、距離を引数とし、返り値として持つ何かの関数である。変数 $r(i, j), r(i, k), r(j, k)$ は、ベクトル $r^{(o_i, o_j)}(i, j), r^{(o_i, o_k)}(i, k), r^{(o_j, o_k)}(j, k)$ の距離である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。尚、下記の例は、関数 $\{f_{ij}(r), f_{ik}(r), f_{jk}(r)\} = r$ の設定での例である。

```

def f_1(r):
    return r

```

```

op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'function': [f_1],
    'analysis_type': ['D'] }

```

F : angular Fourier series (AFS) like parameter

angular Fourier series parameter F は次の式で計算される。 [9][10]

$$F_i^{(a=0, o_i, o_j, o_k, f_{ij}, f_{ik}, m)} = \frac{1}{N(N-1)/2} \sum_{j>k \in N_b(i)} f_{min}(min(r^{(o_i, o_j)}(i, j), r^{(o_i, o_k)}(i, k))) f_{max}(max(r^{(o_i, o_j)}(i, j), r^{(o_i, o_k)}(i, k))) \cos$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。関数 $f_{ij}(r), f_{ik}(r), f_{jk}(r)$ は、ベクトルを引数として値を返り値として持つ何かの関数である。変数 l は、角度の係数である。変数 $\theta^{(o_j, o_i, o_k)}(j, i, k)$ はベクトル $r^{(o_i, o_j)}(i, j)$ とベクトル $r^{(o_i, o_k)}(i, k)$ との角度である。関数

$f_{ij}(r(i,j)), f_{ik}(r(i,k))$ は、距離を引数とし、返り値として持つ何かの関数である。変数 $r(i,j), r(i,k)$ は、ベクトル $r^{(o_i, o_j)}(i,j), r^{(o_i, o_k)}(i,k)$ の距離である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。尚、下記の例は、関数 $\{f_{ij}(r), f_{ik}(r)\} = r$ の設定での例である。

```
def f_1(r):
    return r

op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_F': [1],
    'function': [f_1],
    'analysis_type': ['F']}
```

H : angle histogram analysis (AHA)

angle histogram analysis H は次の式で計算される。

$$h^{(b=1, bin, o_i, o_j, o_k)}(i) = \frac{1}{N(N-1)/2} \sum_{j > k \in N_b(i)} histogram^{(bin)}(\theta^{(o_i, o_j, o_k)}(j, i, k))$$

$$h^{(b, bin, o_i, o_j, o_k)}(i) = \frac{1}{N+1} \sum_{j \in \tilde{N}_b(i)} h^{(b-1, bin, o_i, o_j, o_k)}(j)$$

$$H_i^{(a=0, b, bin, o_i, o_j, o_k, \nu)} = FFT_{amplitude}(h^{(b, bin, o_i, o_j, o_k)}(i))\delta(X - \nu)$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\tilde{N}_b(i)$ は変数 $N_b(i)$ に粒子 i 自身を追加した $N+1$ 個の粒子の list である。変数 b は、各粒子のヒストグラムを隣接粒子のヒストグラムで平均化を行う回数である。変数 $\theta^{(o_j, o_i, o_k)}(j, i, k)$ はベクトル $r^{(o_i, o_j)}(i, j)$ とベクトル $r^{(o_i, o_k)}(i, k)$ との角度である。関数 $histogram$ はビンの数 bin のヒストグラムを作成する関数である。関数 $FFT_{amplitude}$ はヒストグラムをフーリエ変換し $amplitude$ を計算する関数である。関数 δ はディラックのデルタ関数である。変数 ν はフーリエ変換された後のヒストグラムの周波数成分である。周波数成分 X が ν と同じであった場合のみ、デルタ関数 δ は 1 になるためである。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'b_in_H': 1,
    'bin_in_H': [24],
```

```
'nu_in_H': [3],
'analysis_type': ['H'] }
```

I : tetrahedron order parameter (TOP)

Tetrahedron order parameter I は以下の式で計算される。[11][12]

$$I_i^{(a=0, o_i, o_j, o_k)} = 1 - \frac{3}{8} \sum_{j>k \in N_b(i)} \{\cos(\theta^{(o_i, o_j, o_k)}(j, i, k)) + 1/3\}^2$$

変数 a は、隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\theta^{(o_j, o_i, o_k)}(j, i, k)$ はベクトル $r^{(o_i, o_j)}(i, j)$ とベクトル $r^{(o_i, o_k)}(i, k)$ との角度である。

隣接粒子数 4、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_settings = {
  'neighbor': [4],
  'radius': [2.0],
  'ave_times': 1,
  'oi_oj': [0],
  'o_factor': [0.00],
  'analysis_type': ['I'] }
```

Q : Bond order parameter

Bond order parameter Q は以下の式で計算される。[13][14]

$$Q_i^{(l, a=0, b, o_i, o_j, p)} = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^l |q_{lm}^{(l, a=0, b, o_i, o_j, p)}(i)|^2}$$

$$q_{lm}^{(l, a=0, b, o_i, o_j, p)}(i) = \frac{1}{N+1} \sum_{j \in \tilde{N}_b(i)} q_{lm}^{(l, a=0, b-1, o_i, o_j, p)}(j)$$

$$q_{lm}^{(l, a=0, b=1, o_i, o_j, p)}(i) = \frac{1}{N+c} \left[\sum_{j \in N_b(i)} Y_{lm}(\theta^{(o_i, o_j)}(i, j), \phi^{(o_i, o_j)}(i, j)) + p \sum_{v \in O} Y_{lm}(\theta^{(o_i, v)}(i, i), \phi^{(o_i, v)}(i, i)) \right]$$

変数 a は、オーダーパラメータの値を隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\tilde{N}_b(i)$ は $N_b(i)$ に粒子 i 自身を追加した $N+1$ 個の粒子のリストである。変数 b は、球面調和関数を隣接粒子で平均化を行う回数である。関数 Y は、球面調和関数である。変数 $\theta^{(o_i, o_j)}(i, j), \phi^{(o_i, o_j)}(i, j)$ はベクトル $r^{(o_i, o_j)}(i, j)$ の球面座標系での表現における角度で、 θ は z 軸からの角度、 ϕ は x 軸からの角度である。変数 o は、粒子の位置を表現する数値の集合 O かつ、 $o_i \neq O$ であるような b 個の数値の集合である。例えば、 $O = \{1, 0, -1\}$ かつ $o_i = 0$ であった、 $o = \{1, -1\}$ となる。変数 p は、粒子 i 自身の他の位置ベクトル重さである。変数 c は、規格化のための係数で、 $c = bp$ となるような数値である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q': [4],
    'b_in_Q': 1,
    'p_in_Q': [0],
    'analysis_type': ['Q'] }
```

尚、変数 `p_in_Q` では、変数として隣接粒子数 N を使用することが可能であり、次のように指定する事も可能である。

```
op_settings = {
    'p_in_Q' : [0, 'N', 'N/2', '2*N'],
    ... ( 各オーダーパラメータのパラメータ ) }
```

Steinhardt のオーダーパラメータ [13] は以下のオプションで計算できる。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 0,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q' : [4,6],
    'b_in_Q' : 0,
    'p_in_Q' : [0],
    'analysis_type' : ['Q'] }
```

Lechner のオーダーパラメータ [14] は以下のオプションで計算できる。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 0,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q' : [4,6],
    'b_in_Q' : 1,
    'p_in_Q' : [0],
    'analysis_type' : ['Q'] }
```

W : Bond order parameter

Bond order parameter W は以下の式で計算される。 [13][14]

$$W_i^{(l,a=0,b,o_i,o_j,p)} = \frac{\sum_{m_1+m_2+m_3=0} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} q_{lm_1}^{(l,a=0,b,o_i,o_j,p)}(i) q_{lm_2}^{(l,a=0,b,o_i,o_j,p)}(i) q_{lm_3}^{(l,a=0,b,o_i,o_j,p)}(i)}{\left(\sum_{m=-l}^l |q_{lm}^{(l,a=0,b,o_i,o_j,p)}(i)|^2 \right)^{3/2}}$$

$$q_{lm}^{(l,a=0,b,o_i,o_j,p)}(i) = \frac{1}{N+1} \sum_{j \in \tilde{N}_b(i)} q_{lm}^{(l,a=0,b-1,o_i,o_j,p)}(j)$$

$$q_{lm}^{(l,a=0,b=0,o_i,o_j,p)}(i) = \frac{1}{N+c} \left[\sum_{j \in N_b(i)} Y_{lm}(\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j)) + p \sum_{v \in O} Y_{lm}(\theta^{(o_i,v)}(i,i), \phi^{(o_i,v)}(i,i)) \right]$$

変数 a は、オーダーパラメータの値を隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\tilde{N}_b(i)$ は $N_b(i)$ に粒子 i 自身を追加した $N+1$ 個の粒子のリストである。変数 b は、球面調和関数を隣接粒子で平均化を行う回数。関数 Y は、球面調和関数である。変数 $\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j)$ はベクトル $r^{(o_i,o_j)}(i,j)$ の球面座標系での表現における角度で、 θ は z 軸からの角度、 ϕ は x 軸からの角度である。変数 o は、粒子の位置を表現する数値の集合 O かつ、 $o_i \neq O$ であるような b 個の数値の集合である。例えば、 $O = \{1, 0, -1\}$ かつ $o_i = 0$ であった、 $o = \{1, -1\}$ となる。変数 p は、粒子 i 自身の他の位置ベクトル重さである。変数 c は、規格化のための係数で、 $c = bp$ となるような数値である。変数 m_1, m_2, m_3 は $-l$ から l までの値をとるが、 $m_1 + m_2 + m_3 = 0$ の時だけ計算される。行列 $\begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix}$ は Wigner 3- j symbol である。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q': [4],
    'b_in_Q': 1,
    'p_in_Q': [0],
    'analysis_type': ['W'] }
```

尚、変数 p_in_Q では、変数として隣接粒子数 N を使用することが可能であり、次のように指定する事も可能である。

```
op_settings = {
    'p_in_Q' : [0, 'N', 'N/2', '2*N'],
    ... ( 各オーダーパラメータのパラメータ ) }
```

Steinhardt のオーダーパラメータ [13] は以下のオプションで計算できる。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 0,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q': [4, 6],
```

```

'b_in_Q': 0,
'p_in_Q': [0],
'analysis_type': ['W'] }

```

Lechner のオーダーパラメータ [14] は以下のオプションで計算できる。

```

op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 0,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q': [4,6],
    'b_in_Q': 1,
    'p_in_Q': [0],
    'analysis_type': ['W'] }

```

$Q1, W1$: facet weighting Bond order parameter

Mickel の Bond order parameter $Q1$ は以下のような式で計算される。 [15]

$$\begin{aligned}
Q1_i^{(l,a=0,b)} &= \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^l |q_{lm}^{(l,a=0,b)}(i)|^2} \\
q_{lm}^{(l,a=0,b)}(i) &= \frac{1}{N+1} \sum_{j \in \tilde{N}_b(i)} q_{lm}^{(l,a=0,b-1,o_i,o_j)}(j) \\
q_{lm}^{(l,a=0,b=0,o_i,o_j)}(i) &= \sum_{j \in N_b(i)} F(i,j) Y_{lm}(\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j))
\end{aligned}$$

Mickel の方法では、Voronoi 分割した後の facet の面積 A を使う。粒子 i と粒子 j の間の面積を $A(i,j)$ とすると、

$$F(i,j) = \frac{A(i,j)}{\sum_{k \in N_b(i)} A(i,k)}$$

変数 a は、オーダーパラメータの値を隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\tilde{N}_b(i)$ は $N_b(i)$ に粒子 i 自身を追加した $N+1$ 個の粒子のリストである。変数 b は、球面調和関数を隣接粒子で平均化を行う回数である。関数 Y は、球面調和関数である。変数 $\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j)$ はベクトル $r^{(o_i,o_j)}(i,j)$ の球面座標系での表現における角度で、 θ は z 軸からの角度、 ϕ は x 軸からの角度である。

また、オーダーパラメータ $W1$ に関しては以下のように、計算する。

$$W1_i^{(l,a=0,b,o_i,o_j)} = \frac{\sum_{m_1+m_2+m_3=0} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} q_{lm_1}^{(l,a=0,b,o_i,o_j,p)}(i) q_{lm_2}^{(l,a=0,b,o_i,o_j,p)}(i) q_{lm_3}^{(l,a=0,b,o_i,o_j,p)}(i)}{\left(\sum_{m=-l}^l |q_{lm}^{(l,a=0,b,o_i,o_j,p)}(i)|^2 \right)^{3/2}}$$

この order parameter は隣接粒子を Delaunay 分割で計算することが必要であるため、隣接条件は Delaunay のみで使用する。

隣接条件 Delaunay 分割、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_settings = {
    'Delaunay': ['standard'],
    'radius': [2.0],
    'ave_times': 1,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q': [4],
    'analysis_type': ['Q1', 'W1'] }
```

$Q2, W2$: function weighting Bond order parameter

$$Q2_i^{(l,a=0,b)} = \sqrt{\frac{4\pi}{2l+1} \sum_{m=-l}^l |q_{lm}^{(l,a=0,b)}(i)|^2}$$

$$q_{lm}^{(l,a=0,b)}(i) = \frac{1}{N+1} \sum_{j \in \tilde{N}_b(i)} q_{lm}^{(l,a=0,b-1)}(j)$$

$$q_{lm}^{(l,a=0,b=0)}(i) = \sum_{j \in N_b(i)} F(i,j) Y_{lm} \{ \theta(i,j), \phi(i,j) \}$$

$$q_{lm}^{(l,a=0,b=0,o_i,o_j)}(i) = \sum_{j \in N_b(i)} F(i,j) Y_{lm} (\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j))$$

粒子 i と粒子 j の間の重み付けを設定する関数 $F(i,j)$ を使用することができる。

変数 a は、オーダーパラメータの値を隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\tilde{N}_b(i)$ は $N_b(i)$ に粒子 i 自身を追加した $N+1$ 個の粒子のリストである。変数 b は、球面調和関数を隣接粒子で平均化を行う回数である。関数 Y は、球面調和関数である。変数 $\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j)$ はベクトル $r^{(o_i,o_j)}(i,j)$ の球面座標系での表現における角度で、 θ は z 軸からの角度、 ϕ は x 軸からの角度である。

また、オーダーパラメータ $W2$ に関しては以下のように、計算する。

$$W2_i^{(l,a=0,b,o_i,o_j)} = \frac{\sum_{m_1+m_2+m_3=0} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} q_{lm_1}^{(l,a=0,b,o_i,o_j)}(i) q_{lm_2}^{(l,a=0,b,o_i,o_j)}(i) q_{lm_3}^{(l,a=0,b,o_i,o_j)}(i)}{\left(\sum_{m=-l}^l |q_{lm}^{(l,a=0,b,o_i,o_j)}(i)|^2 \right)^{3/2}}$$

隣接条件 Delaunay 分割、隣接半径 2.0 での Mckel のオーダーパラメータの計算条件として以下の例を示す。

```
def f1(j, voronoi_area_list, distance_list):
    weight = voronoi_area_list[j] / np.sum(voronoi_area_list)
    return weight
```

```
op_settings = {
```

```

'Delaunay': ['standard'],
'radius': [2.0],
'ave_times': 1,
'oi_oj': [0],
'o_factor': [0.00],
'l_in_Q': [4],
'function_in_Q2': [f1],
'analysis_type': ['Q2', 'W2'] }

```

他にも、距離に応じた重み付けなどができる。以下の例は、粒子 i と粒子 j 間の距離が近い場合はより重く、遠い場合はより軽くするような設定である。

```

def f1(j, voronoi_area_list, distance_list):
    weight = distance_list[j] / np.sum(distance_list)
    return weight

```

```

op_settings = {
'Delaunay': ['standard'],
'radius': [2.0],
'ave_times': 1,
'oi_oj': [0],
'o_factor': [0.00],
'l_in_Q': [4],
'function_in_Q2': [f1],
'analysis_type': ['Q2'] }

```

LQ, LW : local Bond order parameter

このオーダーパラメータは、[16][17][18] を模して実装したものである。

$$\begin{aligned}
LQ_i^{(l,a=0,b,o_i,o_j)} &= \frac{1}{N} \sum_{j \in N_b(i)} \frac{\text{Re}(q_{lm}^{(l,a,b,o_i,o_j)}(i,j))}{|q_{lm}^{(l,a,b,o_i,o_j)}(i)| |q_{lm}^{(l,a,b,o_i,o_j)}(j)|} \\
q_{lm}^{(l,a,b,o_i,o_j)}(i,j) &= \sum_{m=-l}^l q_{lm}^{(l,a,b,o_i,o_j)}(i) q_{lm}^{*(l,a,b,o_i,o_j)}(j) \\
q_{lm}^{(l,a=0,b,o_i,o_j)}(i) &= \frac{1}{N+1} \sum_{j \in \tilde{N}_b(i)} q_{lm}^{(l,a=0,b-1,o_i,o_j)}(j) \\
q_{lm}^{(l,a=0,b=1,o_i,o_j)}(i) &= \frac{1}{N} \sum_{j \in N_b(i)} Y_{lm}(\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j))
\end{aligned}$$

変数 a は、オーダーパラメータの値を隣接粒子で平均化を行う回数。変数 $N_b(i)$ は粒子 i の N 個の隣接粒子のリストである。変数 $\tilde{N}_b(i)$ は $N_b(i)$ に粒子 i 自身を追加した $N+1$ 個の粒子のリストである。変数 b は、球面調和関数を隣接粒子で平均化を行う回数である。関数 Y は、球面調和関数である。変数 $\theta^{(o_i,o_j)}(i,j), \phi^{(o_i,o_j)}(i,j)$ はベクトル $r^{(o_i,o_j)}(i,j)$ の球面座標系での表現における角度で、 θ は z 軸からの角度、 ϕ は x 軸からの角度である。変数 o は、粒子の位置を表現する数値の集合 O かつ、 $o_i \neq O$ であるような b 個の数値の集合である。例えば、 $O = \{1, 0, -1\}$ かつ $o_i = 0$ であった、 $o = \{1, -1\}$ となる。変数 p

は、粒子 i 自身の他の位置ベクトル重さである。変数 c は、規格化のための係数で、 $c = bp$ となるような数値である。

$$LW_i^{(l,a=0,b,o_i,o_j)} = \frac{\sum_{m_1+m_2+m_3=0} \begin{pmatrix} l & l & l \\ m_1 & m_2 & m_3 \end{pmatrix} lq_{lm_1}^{(l,a=0,b,o_i,o_j)}(i) lq_{lm_2}^{(l,a=0,b,o_i,o_j)}(i) lq_{lm_3}^{(l,a=0,b,o_i,o_j)}(i)}{\left(\sum_{m=-l}^l |lq_{lm}^{(l,a=0,b,o_i,o_j)}(i)|^2 \right)^{3/2}}$$

$$lq_{lm}^{(l,a,b,o_i,o_j)}(i) = \frac{1}{N} \sum_{j \in N_b(i)} \frac{q_{lm}^{(l,a,b,o_i,o_j)}(i) q_{lm}^{*(l,a,b,o_i,o_j)}(j)}{|q_{lm}^{(l,a,b,o_i,o_j)}(i)| |q_{lm}^{(l,a,b,o_i,o_j)}(j)|}$$

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。[16] と同じようなオーダーパラメータは以下のオプションで計算できる。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'ave_times': 0,
    'oi_oj': [0],
    'o_factor': [0.00],
    'l_in_Q': [4,6],
    'b_in_Q': 0,
    'p_in_Q': [0],
    'analysis_type': ['LQ', 'LW'] }
```

Z : user define order parameter

ユーザーが定義するためのオーダーパラメータ Z を用意している。

隣接粒子数 12、隣接半径 2.0 でのオーダーパラメータの計算条件として以下の例を示す。

```
op_settings = {
    'neighbor': [12],
    'radius': [2.0],
    'analysis_type': ['Z'] }
```

オーダーパラメータの計算部分としては、`op_tools/op_z_user_define.py` である。このファイルを編集し、ユーザーの考えたオーダーパラメータを実装することが可能である。

大量の解析を行う設定

現実的に実行するためには、非常に長い計算が必要だろう。

```
def f_1(r):
    return r
op_settings = {
    # 隣接粒子半径の設定
```



```

'neighbor'      : [8],          # 隣接粒子数
'radius'        : [1.5],        # 隣接半径
'ave_times'     : 1,            # あるオーダーパラメータを周囲の粒子で平均を計算
する回数
'oi_oj'         : [1,0,-1],     # 粒子の座標をどの点にするかという設定 粒
子 => [0], 直線 => [1,-1] (先端、後端), 楕円体 => [0, 1, -1] (中心、先端、後端)
'o_factor'      : [0.5, 1.0, 1.5], # 方向ベクトルの長さの設定
# A
'op_types' : ['A','P','N'],      # オーダーパラメータ A での解析の種類の指定
'm_in_A'      : [2, 4],         # 粒子 i, 粒子 j からの距離の近い粒子 j の粒子数
# B
'm_in_B' : [2],                 # 角度の係数
'n_in_B': [1, 2],               # cosine 関数の指数
'phi_in_B': [0],                # 角度の offset
# C
# D
'function': [f_1]               # 関数の種類
# F
'l_in_F' : [1],                 # 角度の係数
# H
'b_in_H' : 1,                   # 角度のヒストグラムを平均化する回数
'bin_in_H' : [24],              # ヒストグラムのビンの数
'nu_in_H' : [3],                # 抜き出す角度の周波数成分。この例では pi/3 の周
波数の成分の指定になっている。
# I
# Q
'b_in_Q'       : 1,              # 球面調和関数を平均化する回数
'l_in_Q'       : [2, 4, 6],      # 球面調和関数の次数を指定するパラメータ 1
'p_in_Q'       : [0],            # その粒子の方向ベクトルに関する重み
# S
'n_in_S'       : [2],            # Degree of Legendre_polynomials
# T
'n_in_T'       : [2],            # Degree of Legendre_polynomials
'd_in_T': [1.0],
'analysis_type': ['A', 'B', 'C', 'D', 'F', 'H', 'I', 'Q', 'W', 'S', 'T']] # 解析す
るオーダーパラメータの種類

```

出力のフォーマット

次のように使用する。

```

import order_tools
order_param_data = \
    op_analyze(coord, direct, box_length, op_settings)

```

出力として各粒子のオーダーパラメータが計算される。各粒子のオーダーパラメータにアクセスするため

には、

```
order_param_data['Q_N6_a=1_b=1_of_1.00_oi=0_oj=0']
```

- Q : オーダーパラメータの種類
- N6 : 隣接粒子の条件
- 'a=1_b=1_of_1.00_oi=0_oj=0' : オーダーパラメータの計算に使用したパラメータ一覧 でアクセスできる。

Reference

- [1] L. Onsager, THE EFFECTS OF SHAPE ON THE INTERACTION OF COLLOIDAL PARTICLES, *Annals of the New York Academy of Sciences*. 51 (1949) 627-659. doi:10.1111/j.1749-6632.1949.tb27296.x.
- [2] C. Zannoni, Distribution function and order parameters, in: G.R. Luchhurst, G.W. Gray (Eds.), *The Molecular Physics of Liquid Crystals*, Academic Press Inc. (London) Ltd., 1979: p. 51.
- [3] W. McMillan, Simple Molecular Model for the Smectic A Phase of Liquid Crystals, *Physical Review A*. 4 (1971) 1238-1246. doi:10.1103/PhysRevA.4.1238.
- [4] J.D. Honeycutt, H.C. Andersen, Molecular dynamics study of melting and freezing of small Lennard-Jones clusters, *The Journal of Physical Chemistry*. 91 (1987) 4950-4963. doi:10.1021/j100303a014.
- [5] A. Radhi, K. Behdian, Identification of crystal structures in atomistic simulation by predominant common neighborhood analysis, *Computational Materials Science*. 126 (2017) 182-190. doi:10.1016/j.commatsci.2016.09.035.
- [6] G.J. Ackland, A.P. Jones, Applications of local crystal structure measures in experiment and simulation, *Physical Review B - Condensed Matter and Materials Physics*. 73 (2006) 1-7. doi:10.1103/PhysRevB.73.054104.
- [7] C.L. Kelchner, S.J. Plimpton, J.C. Hamilton, Dislocation nucleation and defect structure during surface indentation, *Physical Review B*. 58 (1998) 11085-11088. doi:10.1103/PhysRevB.58.11085.
- [8] A. Stukowski, Structure identification methods for atomistic simulations of crystalline materials, *Modelling and Simulation in Materials Science and Engineering*. 20 (2012) 045021. doi:10.1088/0965-0393/20/4/045021.
- [9] A.P. Bartók, R. Kondor, G. Csányi, On representing chemical environments, *Physical Review B - Condensed Matter and Materials Physics*. 87 (2013) 1-16. doi:10.1103/PhysRevB.87.184115.
- [10] A. Seko, A. Togo, I. Tanaka, Descriptors for Machine Learning of Materials Data, in: *Nanoinformatics*, Springer Singapore, Singapore, 2018: pp. 3-23. doi:10.1007/978-981-10-7617-6_1.
- [11] P.-L. CHAU, A.J. HARDWICK, A new order parameter for tetrahedral configurations, *Molecular Physics*. 93 (1998) 511-518. doi:10.1080/002689798169195.
- [12] E. Duboué-Dijon, D. Laage, Characterization of the Local Structure in Liquid Water by Various Order Parameters, *The Journal of Physical Chemistry B*. 119 (2015) 8406-8418. doi:10.1021/acs.jpcb.5b02936.
- [13] P.J. Steinhardt, D.R. Nelson, M. Ronchetti, Bond-orientational order in liquids and glasses, *Physical Review B*. 28 (1983) 784-805. doi:10.1103/PhysRevB.28.784.
- [14] W. Lechner, C. Dellago, Accurate determination of crystal structures based on averaged local bond order parameters, *Journal of Chemical Physics*. 129 (2008) 114707. doi:10.1063/1.2977970.

- [15] W. Mickel, S.C. Kapfer, G.E. Schröder-Turk, K. Mecke, Shortcomings of the bond orientational order parameters for the analysis of disordered particulate matter, *Journal of Chemical Physics*. 138 (2013) 044501. doi:10.1063/1.4774084.
- [16] E.B. Moore, E. De La Llave, K. Welke, D.A. Scherlis, V. Molinero, Freezing, melting and structure of ice in a hydrophilic nanopore, *Physical Chemistry Chemical Physics*. 12 (2010) 4124-4134. doi:10.1039/b919724a.
- [17] M. Fitzner, P. Pedevilla, A. Michaelides, Predicting heterogeneous ice nucleation with a data-driven approach, *Nature Communications*. 11 (2020) 1-9. doi:10.1038/s41467-020-18605-3.
- [18] G.A. Tribello, F. Giberti, G.C. Sosso, M. Salvalaglio, M. Parrinello, Analyzing and Driving Cluster Formation in Atomistic Simulations, *Journal of Chemical Theory and Computation*. 13 (2017) 1317-1327. doi:10.1021/acs.jctc.6b01073.