

# 分子の回転半径とジャイレージョンテンソルの数式解説 (compute\_gyration\_molecule.cpp)

Open DEM Japan

2025年6月30日

個々の分子に含まれる粒子  $i$  を考える。その質量を  $m_i$ 、空間位置を  $\mathbf{r}_i = (x_i, y_i, z_i)^\top$  とする。まず分子  $m$  全体の質量  $M_m$  は

$$M_m = \sum_{i \in m} m_i \quad (1)$$

で与えられる。

つぎに分子の重心（質量中心） $\mathbf{R}_m$  は

$$\mathbf{R}_m = \frac{1}{M_m} \sum_{i \in m} m_i \mathbf{r}_i \quad (2)$$

で定義される。周期境界条件下では各粒子をアンラップ (domain->unmap) して、物理空間で連続な座標  $\mathbf{r}_i$  を用いる。

粒子  $i$  の重心からの相対位置ベクトルを

$$\mathbf{s}_i = \mathbf{r}_i - \mathbf{R}_m \quad (3)$$

とおく。分子の二乗回転半径は

$$R_{g,m}^2 = \frac{1}{M_m} \sum_{i \in m} m_i \|\mathbf{s}_i\|^2 = \frac{1}{M_m} \sum_{i \in m} m_i (s_{ix}^2 + s_{iy}^2 + s_{iz}^2) \quad (4)$$

で定義される。ここから回転半径は

$$R_{g,m} = \sqrt{R_{g,m}^2} \quad (5)$$

となる。コードでは式 (4) の分子和を各プロセスで計算したのち MPI\_Allreduce により集約し、式 (5) を評価している。

より詳細な形状情報を得るために、ジャイレージョンテンソル（慣性テンソルに質量正規化を施した対称テンソル） $\mathbf{G}_m$  を

$$\mathbf{G}_m = \frac{1}{M_m} \sum_{i \in m} m_i \mathbf{s}_i \otimes \mathbf{s}_i = \frac{1}{M_m} \begin{pmatrix} \sum m_i s_{ix}^2 & \sum m_i s_{ix} s_{iy} & \sum m_i s_{ix} s_{iz} \\ \sum m_i s_{iy} s_{ix} & \sum m_i s_{iy}^2 & \sum m_i s_{iy} s_{iz} \\ \sum m_i s_{iz} s_{ix} & \sum m_i s_{iz} s_{iy} & \sum m_i s_{iz}^2 \end{pmatrix} \quad (6)$$

と定義する（和は  $i \in m$ ）。対称性から独立成分は6個だけであり、実装では

$$(G_{xx}, G_{yy}, G_{zz}, G_{xy}, G_{xz}, G_{yz})$$

の順で配列に格納している。式(6)の分子を各プロセスで計算後、`MPI_Allreduce`で総和を取り、 $M_m$ で割って正規化する。

ジャイレーションテンソルの固有値  $\lambda_{m1} \geq \lambda_{m2} \geq \lambda_{m3}$  を求めれば、分子の主回転半径  $\sqrt{\lambda_{mk}}$  やアスペクト比などの形状指標が得られる。しかし `compute_gyration_molecule.cpp` では固有値解析までは行わず、ユーザが後処理で評価できるようテンソルそのものを出力している。

まとめると、ファイル `compute_gyration_molecule.cpp` は

- 式(1), (2)に基づき分子質量と重心を算出し,
- 式(4), (5)から回転半径  $R_{g,m}$  を求め,
- 式(6)に従ってジャイレーションテンソル  $\mathbf{G}_m$  を評価する.

これらは分子のサイズや形状を統計的に特徴づける基本量であり、ポリマー鎖のコンフォメーション解析など幅広い応用を持つ。