

分子平均二乗変位 (MSD) の数式解説 (compute_msd_molecule.cpp)

Open DEM Japan

2025年6月30日

離散要素法 (DEM) における「分子平均二乗変位 (Mean Square Displacement, MSD)」は、各分子 (あるいは剛体集合) の重心が初期位置からどれだけ拡散したかを測る統計量である。ここでは LIGGGHTS/LAMMPS の `compute_msd_molecule.cpp` が実装するアルゴリズムを数学的に整理する。

まず、シミュレーションに含まれる分子の総数を N_{mol} とする。各分子 α は粒子集合 \mathcal{P}_α で表され、その全質量を

$$M_\alpha = \sum_{i \in \mathcal{P}_\alpha} m_i \quad (1)$$

で定義する。ここで m_i は粒子 i の質量である。

■(1) 重心座標の算出 周期境界条件下での連続的な軌跡を保つために、粒子座標 \mathbf{r}_i は“アンラップ (unmap)”操作により箱を跨いだ累積変位を包含するベクトル $\tilde{\mathbf{r}}_i$ に変換される。これを用いて分子 α の時刻 t における重心 (center of mass, COM) は

$$\mathbf{R}_\alpha(t) = \frac{1}{M_\alpha} \sum_{i \in \mathcal{P}_\alpha} m_i \tilde{\mathbf{r}}_i(t) \quad (2)$$

で与えられる。

MPI 並列計算では、式 (2) の分子・分母項それぞれを各プロセスで計算した後に `MPI_Allreduce` により和を取ることで、全体 COM が得られる。

■(2) 初期 COM の記録 シミュレーション開始時刻 t_0 において、各分子の COM を

$$\mathbf{R}_\alpha^{(0)} = \mathbf{R}_\alpha(t_0) \quad (3)$$

としてメモリに保存する。この値は固定であり、後の MSD 評価で基準点となる。

■(3) 平均二乗変位の定義 時刻 t における分子 α の変位ベクトルを

$$\Delta \mathbf{R}_\alpha(t) = \mathbf{R}_\alpha(t) - \mathbf{R}_\alpha^{(0)} \quad (4)$$

とすると、その各成分の二乗および全二乗和は

$$\text{MSD}_x^\alpha(t) = [\Delta R_{\alpha,x}(t)]^2, \quad (5)$$

$$\text{MSD}_y^\alpha(t) = [\Delta R_{\alpha,y}(t)]^2, \quad (6)$$

$$\text{MSD}_z^\alpha(t) = [\Delta R_{\alpha,z}(t)]^2, \quad (7)$$

$$\text{MSD}_{\text{tot}}^\alpha(t) = |\Delta \mathbf{R}_\alpha(t)|^2 = \text{MSD}_x^\alpha(t) + \text{MSD}_y^\alpha(t) + \text{MSD}_z^\alpha(t). \quad (8)$$

プログラムは式 (??)-(??) を各分子について配列に格納し，出力・後処理による拡散係数算出などに供する。

■(4) メモリ使用量概算 実装では，分子数 N_{mol} に比例した配列を確保する．主要なバッファサイズは

$$2N_{\text{mol}} (\text{double}) + 2N_{\text{mol}} \times 3 (\text{double}) + N_{\text{mol}} \times 3 (\text{double}) + N_{\text{mol}} \times 4 (\text{double})$$

であり，総メモリ使用量 (byte 単位) は

$$\mathcal{M} = 2N_{\text{mol}} \text{sizeof}(\text{double}) + 2N_{\text{mol}} \times 3 \text{sizeof}(\text{double}) + N_{\text{mol}} \times 3 \text{sizeof}(\text{double}) + N_{\text{mol}} \times 4 \text{sizeof}(\text{double}) \quad (9)$$

となる.*¹

以上により，`compute_msd_molecule.cpp` の主要数値処理は式 (??)-(??) に要約される．プログラム上の実装はこれらの量を反復的に評価し，タイムステップごとの分子拡散情報を提供する．

*¹ 分子 ID 変換用の `molmap` が有効な場合は追加配列 ($id_{\text{hi}} - id_{\text{lo}} + 1$) 個の `int` が加わる．