

隣接粒子数の数式解説 (compute_neighbor_atom.cpp)

Open DEM Japan

2025年6月30日

本計算アルゴリズムは、離散要素法において各粒子が持つ隣接粒子（近接粒子）の数

$$k_i(t)$$

を時刻 t ごとに評価し、シミュレーション全体で利用可能な配列として保持することを目的とする。以下では、空間的隣接性の判断と並列計算環境下での通信操作を数式で定式化する。

■(1) 隣接集合の定義 粒子 i の中心位置を $\mathbf{r}_i(t)$ とし、対となる粒子 j との中心間距離を

$$r_{ij}(t) = |\mathbf{r}_j(t) - \mathbf{r}_i(t)| \quad (1)$$

と定める。ペアポテンシャルが定義するカットオフ長さ r_c に基づき、粒子 i の隣接集合 $\mathcal{N}_i(t)$ は

$$\mathcal{N}_i(t) = \left\{ j \mid j \neq i, r_{ij}(t) < r_c \right\}. \quad (2)$$

ここで r_c は粒子半径や相互作用特性に応じて組み合わせ的に決まるが、アルゴリズム上は既に隣接リスト構築部で与えられているパラメータとして扱われる。

■(2) 隣接粒子数 粒子 i に対し、(??) で定義された集合の要素数を

$$k_i(t) = \sum_{j \in \mathcal{N}_i(t)} 1, \quad (3)$$

とする。これが本計算が返すスカラー値であり、隣接粒子数として出力される。

■(3) 並列分割とゴースト粒子 計算領域が複数の計算ノード（あるいは MPI プロセス）へ分割されている場合、各プロセス p は局所粒子集合 Ω_p と境界を跨ぐゴースト粒子集合 Γ_p を保持する。粒子 i が複数プロセスに複製される状況を考慮し、プロセス p における局所評価値を

$$k_i^{(p)}(t) = \sum_{j \in \mathcal{N}_i^{(p)}(t)} 1, \quad (4)$$

と表す。ここで $\mathcal{N}_i^{(p)}(t)$ はプロセス p 内で i が検知可能な隣接粒子集合である。

■(4) Newton-Pair 通信 (??) による $k_i^{(p)}$ はプロセスごとに部分的であるため、最終的な全体値は

$$k_i(t) = \sum_{p \in \mathcal{P}(i)} k_i^{(p)}(t), \quad (5)$$

によって得られる。 $\mathcal{P}(i)$ は粒子 i を保持する全プロセス集合であり、式 (??) は Newton_pair オプションが有効な場合に実装される逆通信（reverse communication）の数式的表現である。

■(5) 配列の伸縮操作 最大粒子数を N_{\max} とすると、本計算が確保するメモリ量は

$$M = N_{\max} \text{sizeof}(\text{double}) \quad [\text{byte}] \quad (6)$$

で与えられる。シミュレーション中に N_{\max} が更新されるとき、式(??)に従い動的再割当てが行われる。

■(6) 隣接リスト呼び出し頻度 時刻 $t_n = n\Delta t$ において隣接リストが「必要に応じて」再構築される。リスト再利用の判定は

$$\max_i [|\mathbf{r}_i(t_n) - \mathbf{r}_i(t_{n-1})|] < r_{\text{skin}} \quad (7)$$

で行われ、 r_{skin} はスキン厚みである。不等式が破れた場合にのみ新たな $\mathcal{N}_i(t_n)$ が形成される。

■(7) アルゴリズムの総括 以上の定式化を踏まえると、本計算の核心は

1. 最新の隣接リスト $\{\mathcal{N}_i(t)\}$ を取得する (式(??) - (??))。
2. 各プロセス上で局所隣接数 $k_i^{(p)}$ を求める (式(??))。
3. 必要に応じ逆通信により k_i を集約する (式(??))。
4. 出力配列に k_i を格納し、記憶域計算を式(??)に従い管理する。

これにより、各粒子の接触・近接状態を時系列で解析するための基礎情報が提供される。