19pK-PS-51

AVX 命令を用いた Lennard-Jones ポテンシャルの力計算の SIMD 化

渡辺宙志 (東大物性研)

概要

古典 MD でよく用いられる Lennard-Jones ポテンシャルの力計算を AVX2 命令を使って加速した
・ナイーブな実装から、SIMD 化以外のチューニングで 65% の速度向上
・AVX2 命令を使った SIMD 化により、さらに 123% の速度向上
・実 MD コードにおいても SIMD 化による速度向上率 80%

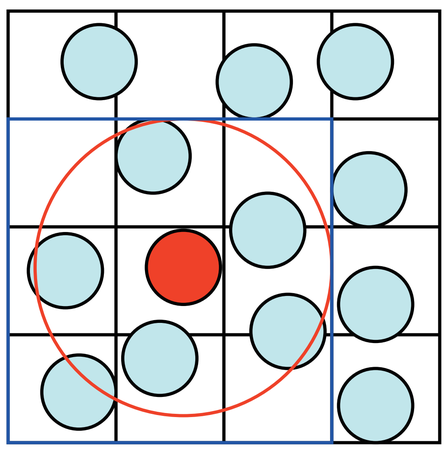
1. はじめに

Q. なぜ SIMD 化しないといけないの？
A. そうしないと性能が出ないから

AVX2 で 256bit、AVX-512 やポスト京で 512bit 幅
自明な場合を除き、コンパイラは SIMD 化できない
→ 自分で書くしか無い

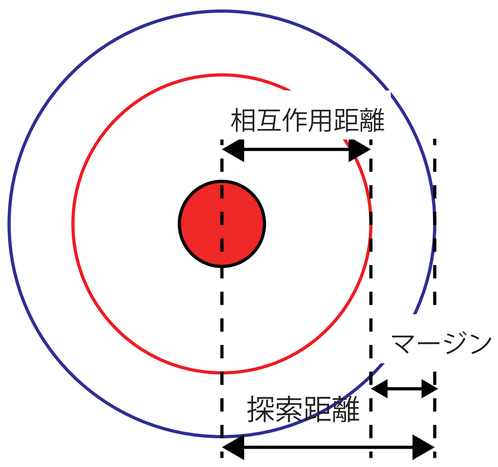
2. ペアリストと Bookkeeping 法

ペアリスト探索



カットオフ距離以内にある
相互作用ペアを探索する

Bookkeeping 法



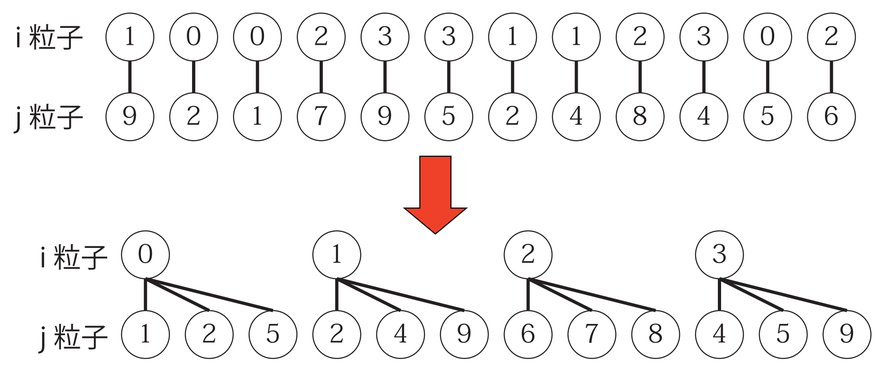
探索範囲を長めにしてペアリストを一定期間再利用する

・力の計算では相互作用ペアのリストが渡される
・相互作用距離外のペアも存在する (条件分岐)

3. SIMD 化以前の高速化

i 粒子によるソート

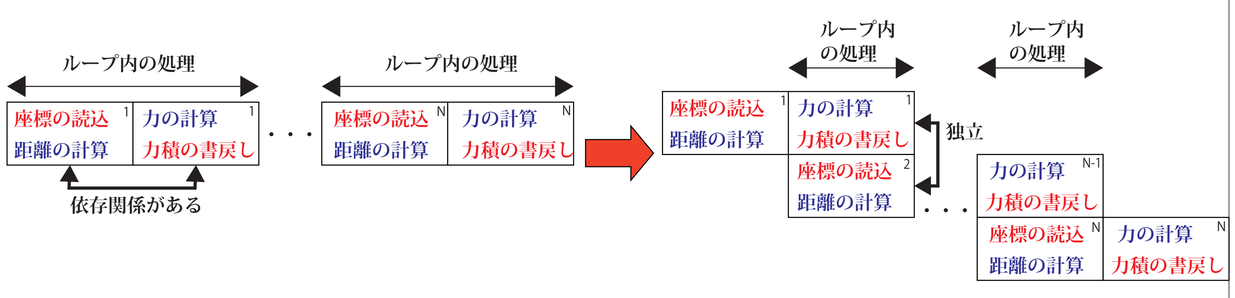
ペアごとに力を計算するとメモリアクセスが多すぎて性能が出ない
(i,j 粒子双方の座標、運動量の読み込み、運動量の書き戻しが発生)



i 粒子の情報がレジスタに乗るため、メモリアクセスが半分に

ソフトウェアパイプライニング

ループ内に依存関係があり、逐次的にしか処理できない



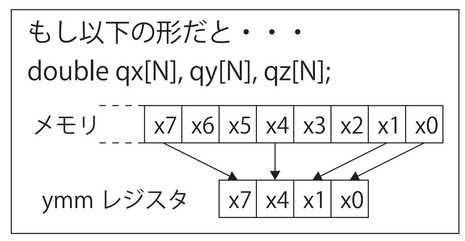
相互作用相手を先読みし、独立な計算とメモリアクセスを増やす
(IPC 向上を狙っている)

4. AVX2 命令を使った SIMD 化

データ構造の修正

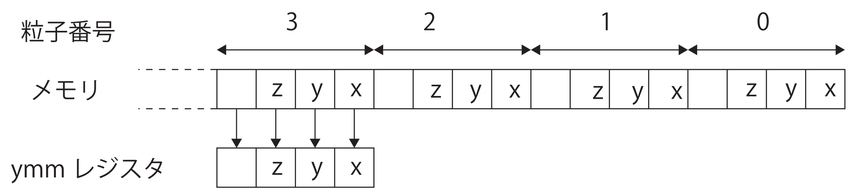
AVX2 の ymm レジスタは 4 つの倍精度実数を保持
ループを 4 倍展開し、4 つ独立な計算を行う
ただし j 粒子は不連続

もし以下の形だと・・・
double qx[N], qy[N], qz[N];



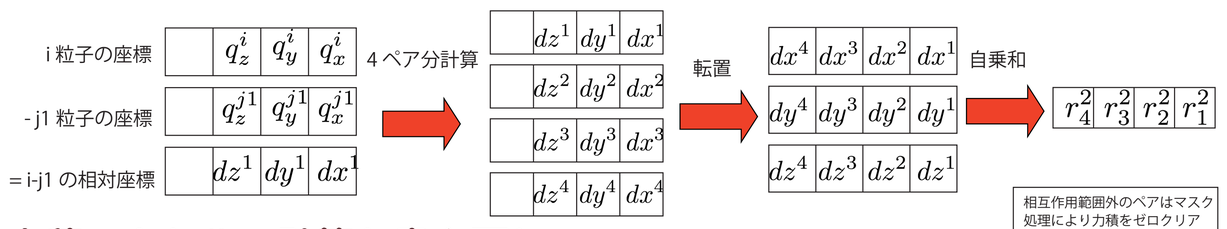
データを 4 次元で宣言
double q[N][4], p[N][4];

3 成分を一度にロード可能
(1 成分は無駄になる)

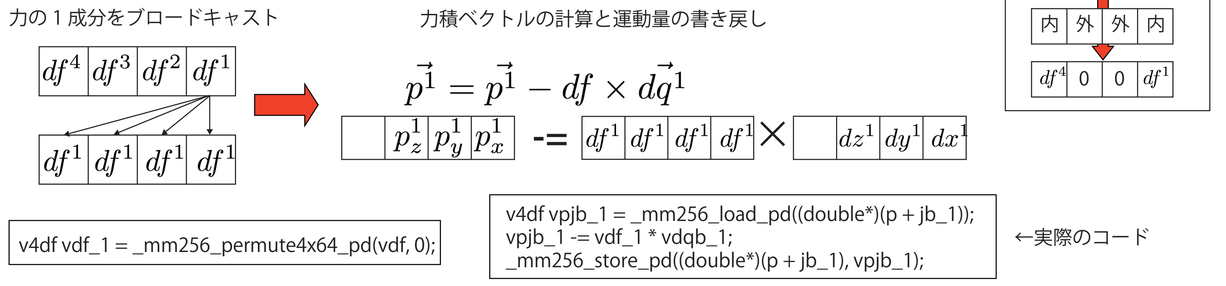


レジスタの転置と力の計算

$$df = \frac{(48 - 24 \times r^6) \times dt}{r^{14}}$$ ←Lennard-Jones 粒子の力として計算したい量 (相対距離の自乗の関数)



力積ベクトルの計算と書き戻し




```
v4df vdf_1 = _mm256_permute4x64_pd(vdf, 0);
```

```
v4df vpjb_1 = _mm256_load_pd((double*)(p + jb_1));
vpjb_1 -= vdf_1 * vdqb_1;
_mm256_store_pd((double*)(p + jb_1), vpjb_1);
```
←実際のコード

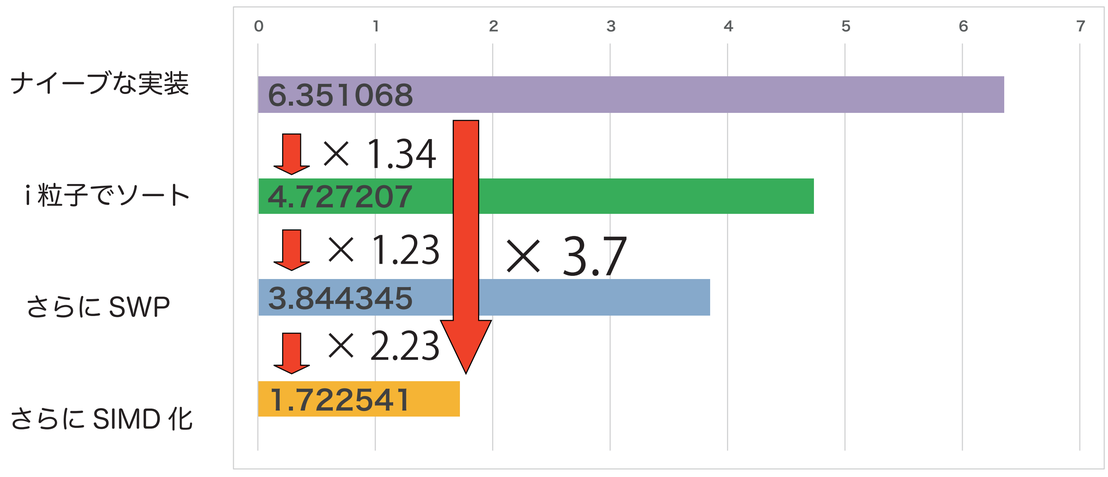
5. 結果

力の計算のみ

計算条件
・12万粒子、密度1.0、カットオフ 3.0
・力計算(力積の書き戻しまで)を100回行うのにかかった時間

計算環境
・3.3GHz Intel Core i5 (Skylake)
・g++ 6.3.0
・-O3 -mavx2 -march=native

実行時間 [s]

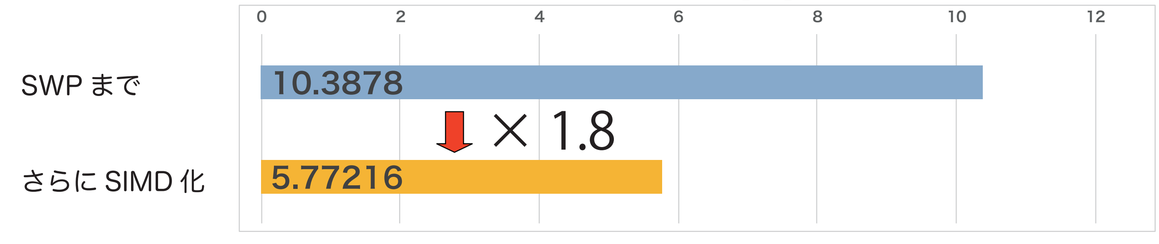


サンプルコード
https://github.com/kaityo256/lj_simdstep

時間発展

・密度0.712、カットオフ3.0、時間刻み0.001
・二次のシンプレクティック積分
・77000粒子を1000ステップ時間積分するのにかかった時間

実行時間 [s]



謝辞 本研究は科研費による助成を受けて実施されました (課題番号 23740287)。また、AVX2 を使った SIMD 化について中川恒さん (物性研) より助言をいただきました。

THE UNIVERSITY OF TOKYO
The Institute for Solid State Physics