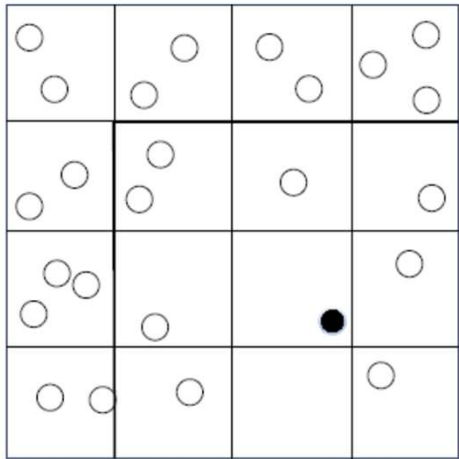


Particle Mesh Multipole Method (PM³)

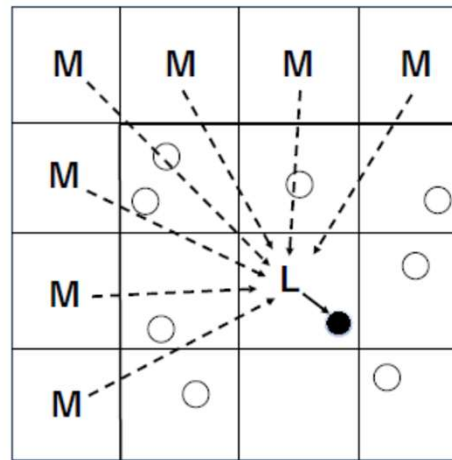
- 主に周期境界条件でのシミュレーションを目的として開発されたスキーム (Nitadori 2014)。
- 高速多重極展開法(FMM)と同じように、多重極展開から局所展開を求めて遠方セルからの寄与を計算。近傍は直接計算。
 - FMMとの大きな違いの一つは局所展開の計算方法

PM³のイメージ

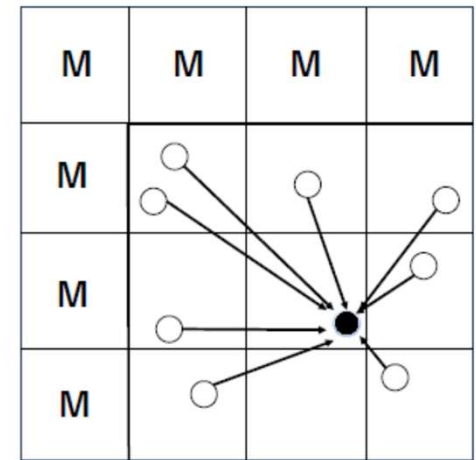
黒丸への力を計算する事を考える



計算領域を規則格子に分割



遠方の格子からの寄与は
P2M- $\rightarrow$ M2L- $\rightarrow$ L2P



近傍の粒子からの寄与は
P2P

遠方セルからの寄与の計算手順

- P2Mは以下の式

$$M_l^m|_{\mathbf{r}_M} = \sum_i q_i R_l^m(\mathbf{r}_M - \mathbf{r}_i).$$

- M2Lは以下の式

$$L_l^m|_{\mathbf{r}_L} = \sum_M \sum_{\lambda=0}^p \sum_{\mu=-\lambda}^{\lambda} M_{\lambda}^{\mu}|_{\mathbf{r}_M} S_{l+\lambda}^{-(m+\mu)}(\mathbf{r}_L - \mathbf{r}_M)$$

これは畳み込み演算なので、FFTを用いて計算

- L2Pは以下の式

$$\Phi(\mathbf{r}) = \sum_{l=0}^{\infty} \sum_{m=-l}^l L_l^m|_{\mathbf{r}_L} R_l^m(\mathbf{r} - \mathbf{r}_L).$$

近傍粒子からの寄与

- オリジナルの実装では粒子同士の直接計算
- FDPSではここをBarnes-Hut Tree法で計算できるようにした。

FDPSを用いたPM3の計算

- 近傍粒子からの寄与はFDPSのTreeForForceクラスで計算
 - 新たにサーチモードを追加
- 遠方セルからの寄与はPM3モジュールで計算
- 領域分割、粒子交換は通常通り

FDPSを用いたPM3の計算

```
PS::TreeForForce<PS::SEARCH_MODE_LONG_PARTICLE_MESH_MULTIPOLE, Force, EP, EP,  
PS::MomentQuadrupoleGeometricCenter, PS::MomentQuadrupoleGeometricCenter,  
PS::SPJQuadrupoleGeometricCenter> tree_pm3; // treeオブジェクト生成
```

```
tree_pm3.initialize(n_ptcl_loc, params.theta);
```

```
tree_pm3.setParamPMMM(params.n_cell_1d, params.i_cut); // PMMMの格子数などを設定
```

```
PS::PMMM::ParticleMeshMultipole<Force, EP> pm3; // PM3オブジェクトを生成
```

```
pm3.initialize(p); // 展開次数を設定
```

```
tree_pm3.calcForceAllAndWriteBack(calc_p0<EP, EP, Force>, calc_p2<EP, PS::SPJQuadrupoleGeometricCenter,  
Force>, ptcl, domain_info); // 近接粒子からの寄与を計算
```

```
pm3.calcForceAllAndWriteBack(tree_pm3, domain_info, ptcl); // 遠方セルからの寄与を計算
```

詳しくはチュートリアルを参照

PM3の利点

- TreePM/PMEに比べて、小規模のFFTでよいので、ネットワークに優しい。高精度化も容易。
- FMMに比べて、計算コストが低い。