Soft Particle による月形成シミュレーションのエネルギー効率

牧野淳一郎 (神戸大学)、岩澤全規 (松江高専)、
細野七月 (神戸大学*)、野村昴太郎 (神戸大学*)

第 17 回アクセラレーション技術発表討論会 2023/9/28

* 現所属: Intel

発表概要

- デブリ円盤からの月形成過程の N 体シミュレーションについて、電力性能を測定した結果を報告する。
- 支配方程式は重力相互作用+物理的に接触した粒子の非弾性反発である。計算コードは並列粒子法計算コード開発フレームワーク FDPS を使ったもので、相互作用記述には PIKG を利用した。測定に使ったマシンの CPU は AMD Ryzen 9 7950X である。
- 演算速度は 527.76GF、消費電力は 83W、電力性能は単精度で 6.359 Gflops/W、倍精度換算では 3.179 Gflops/W となる。

発表構成

- シミュレーションの内容と手法、科学的インパクトについて
- 計算に使ったハードウエアとソフトウエア
- 達成した電力性能の値とそれを計算した手法
- まとめ

シミュレーションの内容と手法、科学的インパクトについて

- 月形成の「標準シナリオ」
- デブリ円盤からの月形成シミュレーション
- 最近の研究の例
- 未解明の問題

月形成の「標準シナリオ」

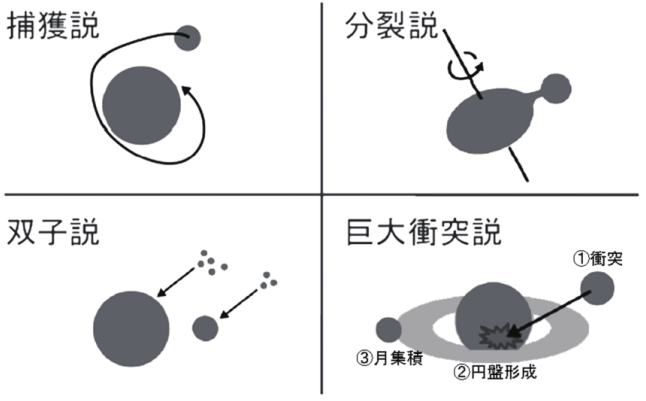


図1：分裂説・捕獲説・双子説・巨大衝突説のイメージ図

玄田 日本惑星科学会誌 2010 より
色々問題はあるが、巨大衝突説が現在の標準シナリオ。
単に他のは問題が多すぎるから。
巨大衝突説の主要な問題: 月の岩石の同位体比が地球と近過ぎる。シミュレーションではインパクターのほうが主体で月ができるのに、、、

デブリ円盤からの月形成シミュレーション

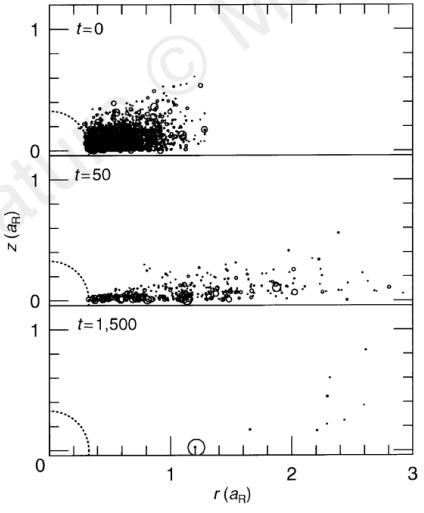


Figure 2 Snapshots of disk particles plotted in geocentric cylindrical coordinates (r,z). (Particles at negative z are plotted at $|z|$.) The units of length and time are the Roche limit radius, a_R, and Kepler time, T_{Kep} at a_R (~7 h). The solid and dotted circles are disk particles and the Earth, respectively. The sizes of the circles indicate physical sizes. The snapshots here are the result of run 4 in Table 1. The mean specific angular momentum, J_{disk}/M_{disk}, is initially $0.692\sqrt{GM_{\oplus}a_R}$. At $t = 1{,}500$ the moon has mass $0.40M_L$, semimajor axis $1.20a_R$, eccentricity 0.09 and inclination (radian) 0.02. The second body's mass is only $0.025M_L$. The masses ejected from the system (M_∞) and that hit the Earth are $0.026M_L$ and $1.95M_L$, respectively.

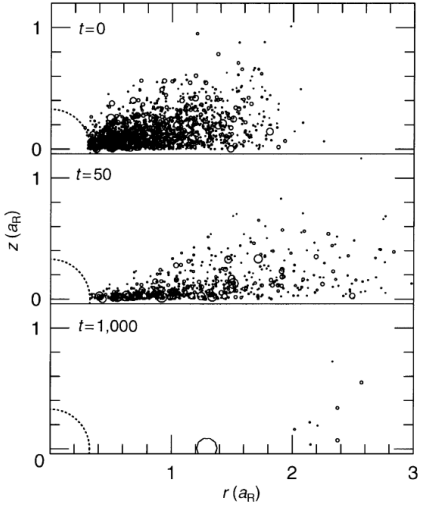
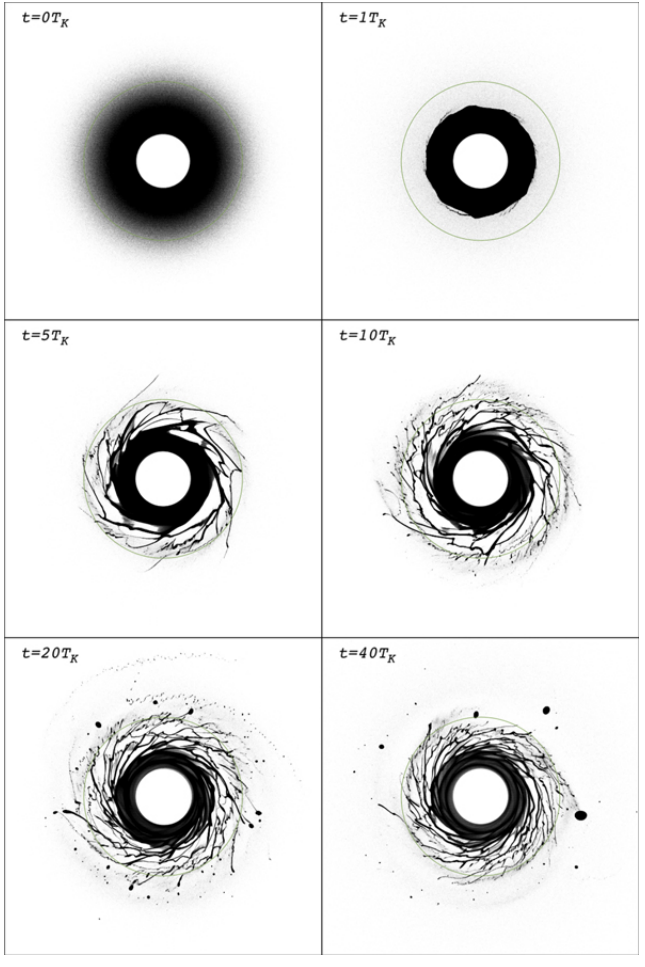


Figure 3 The same snapshots as in Fig. 2 but for run 9 of a more extended disk ($J_{disk}/M_{disk} = 0.813\sqrt{GM_{\oplus}a_R}$). At $t = 1{,}000$ the largest moon mass is $0.71M_L$.

Ida et al. 1997, Nature, 389, 353–357
3000粒子。非等質量。
アニメーション

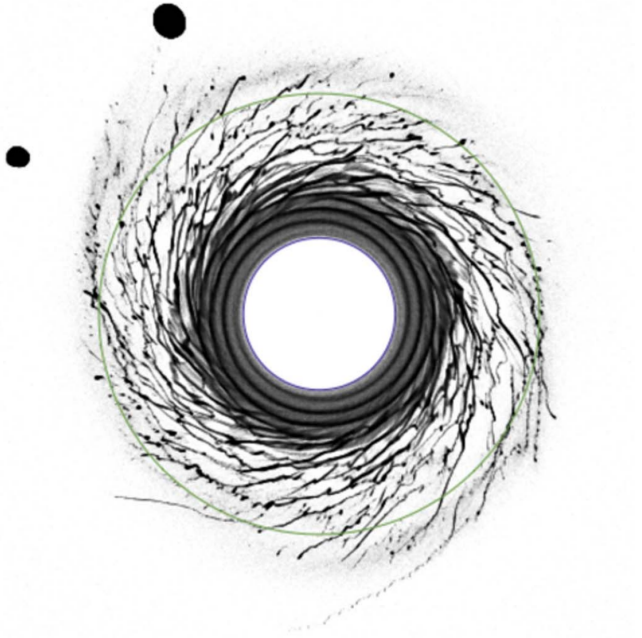
最近の研究の例



Sasaki and Hosono 2018 ApJ 856 175
PEZY-SC + FDPS で最大 1000 万粒子まで計算
できる月質量はあまり粒子数によらない？
角運動量輸送のタイムスケールはかなり粒子数による — スパイラルパターンが違うせいと推測

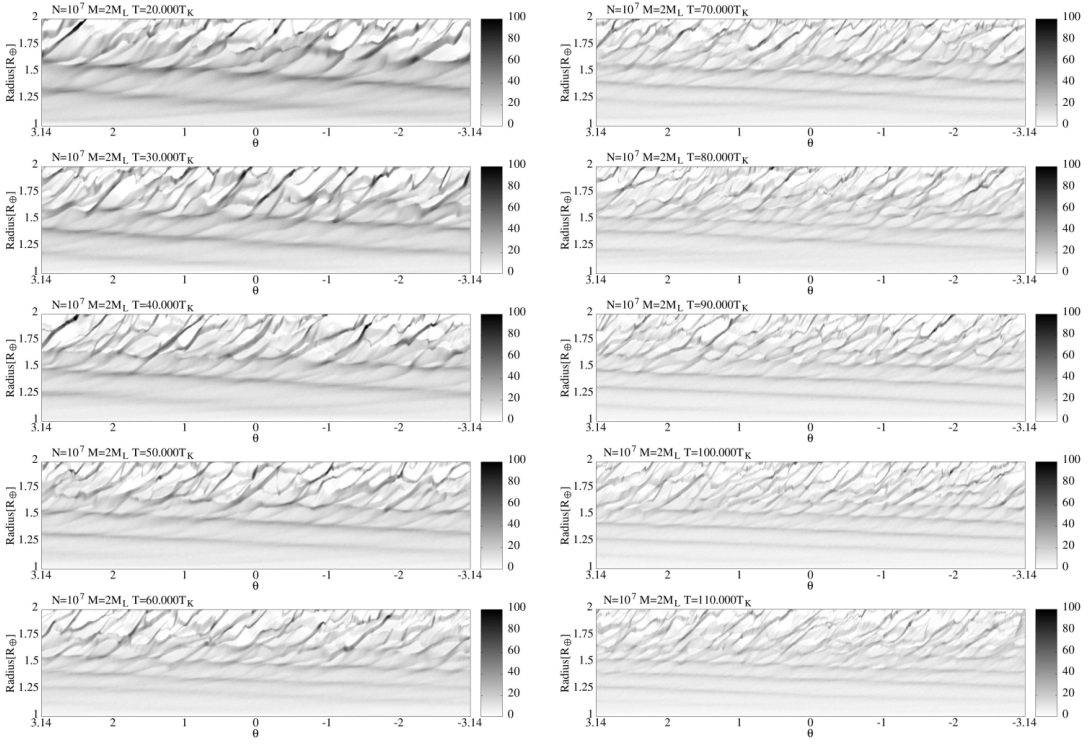
よく見ると、、、

$t=110T_K$



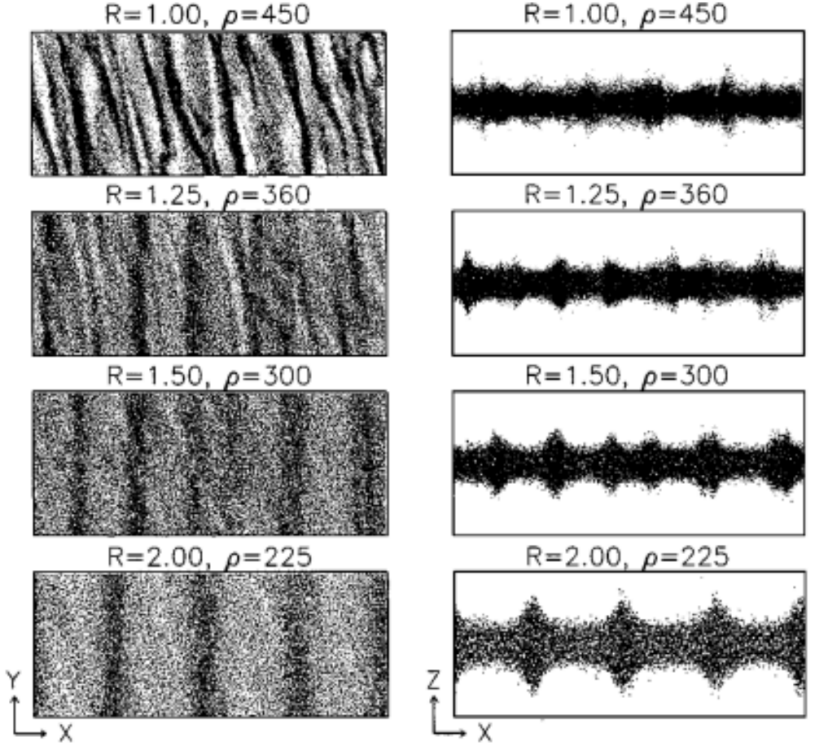
1000 万粒子で大体月ができたところ。
内側に縞々がある？なんか可視化のせい？本物？論文には特になにも書いてないようにみえる。

山岡さん修士論文



山岡さん神戸大学修士論文 (2022 年度)
1000 万粒子の計算。$r-\theta$ 平面で面密度をプロット。
明らかに縞々がある。
leading spiral になっている。
時間がたつと (=面密度が下がると) 縞々の間隔が小さくなる

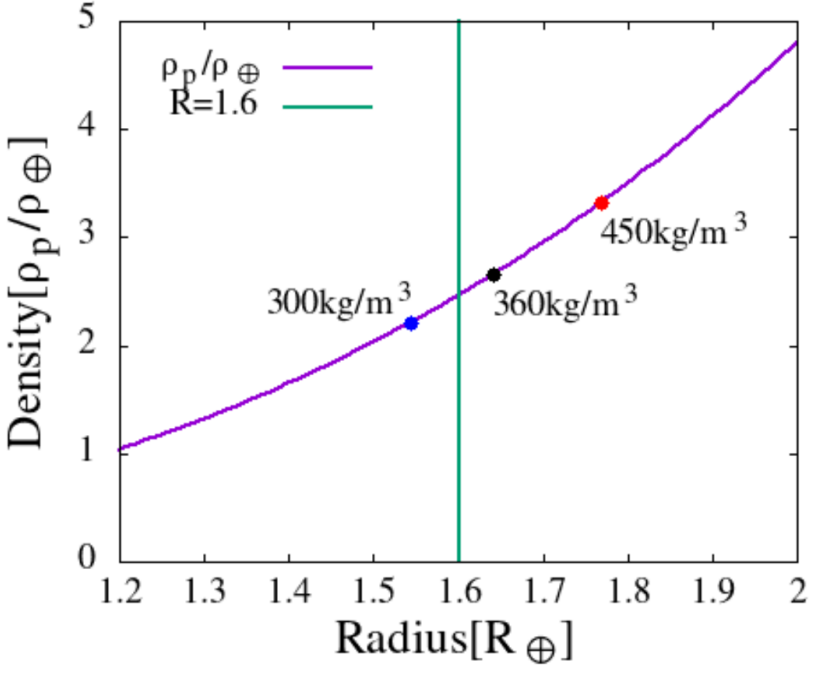
Viscous Overstability



Salo et al. 2001, Icarus, 153, pp. 295–315 Viscous Overstability in Saturn's B Ring. I. Direct Simulations and Measurement of Transport Coefficients.

土星の B リング想定の局所シミュレーション。粒子の密度が低いと普通の wake ではなくリングがでる。

「密度」と親天体からの距離



Salo et al. の「密度」を山岡さんの計算での中心からの距離に対応するように変換したもの。
リング構造がみえるかどうかの境界が地球半径の 1.6 倍あたり。
これが Salo et al. の計算では密度 $340\mathrm{kg/m^3}$ あたりに対応し、その外側では wake、内側ではリング (leading spiral) 。
この構造は viscous overstability によるものである。

大域シミュレーションで初めて viscous overstability を確認。

未解明の問題

- viscous overstability はこの領域での角運動量輸送にどの程度影響するか？
- 粒子数もっと増やした時にそもそも月は本当にできるのか？

計算方法

各粒子の運動方程式は以下で与えられる

$$\frac{dv_i}{dt} = -G\frac{M}{r_i^3}r_i + \frac{1}{m_i}\sum_{j\neq i} F_{ij} \tag{1}$$

ここで、r_i 、v_i, 、bm_i は粒子 i の位置、速度、質量、t は時間、 G は重力定数、M は中心天体の質量である。

計算方法(続き)

F_{ij} は粒子 j が粒子 i に及ぼす力であり、以下で与えられる。

$$
\boldsymbol{F}_{ij} = \begin{cases} G\dfrac{m_i m_j}{r_{ij}^3}\boldsymbol{r}_{ij} & (r_{ij} > r_{\text{coll}}) \\ \left[G\dfrac{m_i m_j}{r_{\text{coll}}^3} + \dfrac{m_j}{m_i + m_j}\left(\kappa\dfrac{r_{ij} - r_{\text{coll}}}{r_{ij}} + \eta\dfrac{\boldsymbol{r}_{ij}\cdot\boldsymbol{v}_{ij}}{r_{ij}^2}\right)\right]\boldsymbol{r}_{ij} & (r_{ij} \le r_{\text{coll}}) \end{cases} \tag{2}
$$

ここで $\boldsymbol{r}_{ij} = \boldsymbol{r}_j - \boldsymbol{r}_i, \boldsymbol{v}_{ij} = \boldsymbol{v}_j - \boldsymbol{v}_i, r_{ij} = \|\boldsymbol{r}_{ij}\|$ である。また、 r_{coll} は粒子の半径の 2 倍である (全ての粒子の質量、半径は同じとした)。

計算方法(続き)

κ と η は非弾性衝突を表現する係数であり、今回のシミュレーションでは衝突の法線方向の速度に対する弾性係数が 0.95 となり、振動周期が時間ステップの 32 倍となるようにこれらを決めている。接線方向には摩擦は働かず、弾性係数は 1 である。このように、粒子間の力自体に接触した (めりこんだ) 時に反発する成分を加える方法を Soft particle 法と呼ぶ。めりこんだ時に直接速度・位置を修正する方法もあるが、大きなクランプが成長してくると計算速度が低下するため、Soft particle 法を採用した。

計算に使ったハードウエアとソフトウエア

- ハードウェア
- ソフトウェア

ハードウェア

今回の計算には、AMD Ryzen 9 7950X CPU のデスクサイド PC 1 台を用いた。

測定にあたっては CPU コア電圧とクロック上限を BIOS 設定で変化させ、

結果がよかったものを用いた。デフォルトではクロック自動、コア電圧 1.25V だが、
今回の速度ではクロック 3500MHz、コア電圧 0.85V とした。

メモリは DDR5 で、 4800 MHz 動作のものを 3200MHz で使用した。

ソフトウェア

シミュレーションに使用したソフトウェアは nbody-with-center[1] である。

これは、多体問題シミュレーションフレームワーク FDPS を用いて実装されている。円盤やリング系に対して極座標を使ってツリー構造を構築することで無駄を防ぐ等の様々な最適化が実装されている。

また、粒子間相互作用や粒子-超粒子の四重極相互作用は PIKG カーネル自動生成ツールを使って高レベル記述から AVX-512 のイントリンシックによる記述が自動生成されており、高い実行効率を実現している。

OS は Ubuntu 22.04 であり、コンパイラは gcc version 11.3.0 である。

[1] https://github.com/jmakino/nbody-with-center

PIKG コード全体(粒子-粒子)

```
EPI F32vec xi:pos
EPI F32vec vi:vel
EPI F32    mi:mass
EPI S32    iid:id

EPJ F32vec xj:pos
EPJ F32vec vj:vel
EPJ F32    mj:mass
EPJ S32    jid:id

FORCE F32vec acc:acc
FORCE F32    pot:pot
FORCE F32    dedtdumperi:dedtdumperi

F32 eps2
F32 r_coll
F32 r_coll_inv
```

```
F32 r_coll_sq
F32 kappa
F32 eta

mjlocal=mj
rij = xi - xj
m_r = mj/ (mi+mj)
r2 = rij * rij + eps2
r_inv_true = rsqrt(r2)
r = r2 * r_inv_true
r2_inv_true  = r_inv_true*r_inv_true
dr = r_coll-r
if r_coll < r
  r_inv = r_inv_true
  dr = 0.0f
else
  r_inv = r_coll_inv
endif
r2_inv =  r_inv * r_inv
```

```
if iid == jid
   m_r=0.0f
   mjlocal=0.0f
endif
potij = r_inv * mjlocal
acc -=  potij*r2_inv * rij
pot -= potij
kappacoef = kappa * m_r * dr*r_inv_true
acc += kappacoef* rij
pot += kappacoef*dr*r * 0.5f
vij = vi - vj
rv = rij*vij
if r_coll_sq < r2
   rv=0.0f
endif
etacoef = -eta * m_r * rv * r2_inv_true
acc += etacoef* rij
dedtdumperi -= etacoef * rv*m_r
```

呼ぶ側のコード

(PIKG の中でもできるが今回は外側で)

- **位置、速度座標を適当な相対値にしてから単精度に変換**
- **求まった加速度・ポテンシャルを倍精度に変換**

```
template <class TParticleJ>
void CalcForceEp(const FPGrav * pi,
                 const PS::S32 ni,
                 const TParticleJ * pj,
                 const PS::S32 nj,
                 ForceGrav * force) {
    PS::F32 eps2 = FPGrav::eps * FPGrav::eps+1e-30;
    PS::F32 kappa = FPGrav::kappa;
    PS::F32 eta   = FPGrav::eta;
    PS::F32 r_coll   = FPGrav::rcoll*2;
    auto r_coll_inv  = 1.0/r_coll;
```

```
auto r_coll_sq  = r_coll*r_coll;

PPEpi epi[ni];
PPForce f[ni];
for(int i=0;i<ni;i++){
    epi[i].pos = (PS::F32vec)(pi[i].pos_car - pi[0].pos_car);
    epi[i].vel = (PS::F32vec)(pi[i].vel - pi[0].vel);
    epi[i].mass = pi[i].mass;
    epi[i].id = pi[i].id;
    f[i].acc = 0.0;
    f[i].pot = 0.0;
    f[i].dedtdumperi = 0.0;
}
PPEpj epj[nj];
for(int i=0;i<nj;i++){
    epj[i].pos = (PS::F32vec)(pj[i].pos_car - pi[0].pos_car);
    epj[i].vel = (PS::F32vec)(pj[i].vel - pi[0].vel);
    epj[i].mass = pj[i].mass;
    epj[i].id = pj[i].id;
```

```
    }
    epep_pikg(eps2, r_coll, r_coll_inv, r_coll_sq, kappa, eta)
        (epi,ni, epj,nj,f);
    for(int i=0;i<ni;i++){
        force[i].acc += f[i].acc;
        force[i].pot += f[i].pot;
        force[i].dedtdumper += f[i].dedtdumperi;
    }
}
```

達成した電力性能の値とそれを計算した手法

性能と消費電力の計測に使ったコマンドラインパラメータは以下の通りである。

```
env OMP_NUM_THREADS=4 mpirun -n 4 ./nbody-with-center-quad-mpi-pig \
  -T 0.25 -r 0.003 -K 32 -i moontest200k.in -o result -D 0.125 \
  -g 0.95 -S 0.9 -R 1 -t 0.3 -s 0.00048828125 -n 512 -d 0.03125 -u 8 -b
```

ツリーの Opening parameter は 0.3、相互作用リストを共有するグループの最大粒子数は 512 である。

また、粒子数は 20 万である。
2 万粒子のアニメーション

達成した電力性能の値とそれを計算した手法(続き)

1 タイムステップの平均実行時間は 0.03513 秒、1 タイムステップあたりの平均相互作用計算数は粒子-粒子が 1.778×10^8、粒子-超粒子が 1.404×10^8 であった。1 粒子あたりの相互作用数は 600 くらいである。
相互作用の演算数は、粒子-粒子が 49、粒子-超粒子が 70 とした。これは、逆数平方根の計算を 12 演算としたものである。

これらから、演算速度は 527.76 Gflops となった。一方、消費電力は、YOKOGAWA 製電力計 WT210 を使用した (計算機による計測ではなく、表示値を目で読んでいるので若干過大になっているが) 83W であった。

従って、電力性能は単精度で 6.359 Gflops/W、倍精度換算では 3.179 Gflops/W となる。

ちなみに

クロック・電圧等デフォルト設定だと消費電力 195W で、測定に使った設定の 2.3 倍、速度は測定に使った設定の 1.3 倍程度にしかならなかった。

最近のベンチマーク結果を見ると AMD の CPU は Intel の CPU に比べて典型的には 2 倍程度電力効率がよいが、デフォルト設定ではそれでも非常に高電圧にして高クロックにしていることがわかる。

実際に 3.5GHz で動作しているとすると単精度での理論ピーク性能は 1792Gflops であり、実測した性能は 528Gflops であったので実行効率は 29.5% となる。比較的低いのは、円盤系では相互作用の計算量が少ないため、他の部分がみえることが大きな要因である。

まとめ

- デブリ円盤からの月形成過程の N 体シミュレーションについて、電力性能を測定した結果を報告した。
- 支配方程式は重力相互作用+物理的に接触した粒子の非弾性反発である。計算コードは並列粒子法計算コード開発フレームワーク FDPS を使ったもので、相互作用記述には PIKG を利用した。測定に使ったマシンの CPU は AMD Ryzen 9 7950X である。
- 演算速度は 527.76GF、消費電力は 83W、電力性能は単精度で 6.359 Gflops/W、倍精度換算では 3.179 Gflops/W となる。