

PICプログラム最適化実用報告

佐野孝好

PICコードのベクトル化

- 止まり木法によってベクトル化は可能
- ただし、演算数を増やしてベクトル化を実現
- (ベクトル化率の向上) ≠ (計算の高速化)

1次元プラズマ粒子シミュレーション

- Particle-In-Cell (PIC) コード

- 電磁場の時間発展

$$\nabla \times \mathbf{B} = \mu_0 \mathbf{J} + \frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t}$$

- マックスウェル方程式

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$$

- 荷電粒子の時間発展

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \quad \nabla \cdot \mathbf{B} = 0$$

- 相対論的運動方程式

$$\frac{d}{dt} (\gamma m \mathbf{v}) = q (\mathbf{E} + \mathbf{v} \times \mathbf{B})$$

計算規模を特徴付ける2つの量

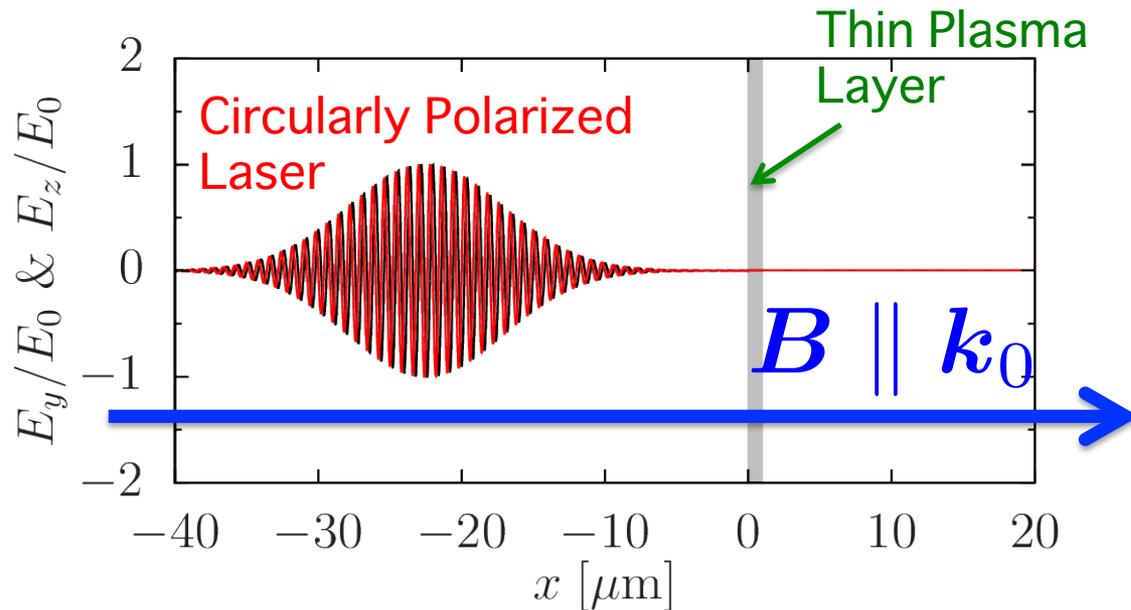
1. グリッド数(電磁場用)

2. 粒子数(荷電粒子用)

- 止まり木法の余分な演算数はグリッド数に比例
- つまり、グリッド数が大きいと不利

1 D PIC Simulations

- Whistler Wave + Thin Plasma Foil
- Key Parameters
 - Plasma Density
 - External Magnetic Field Strength



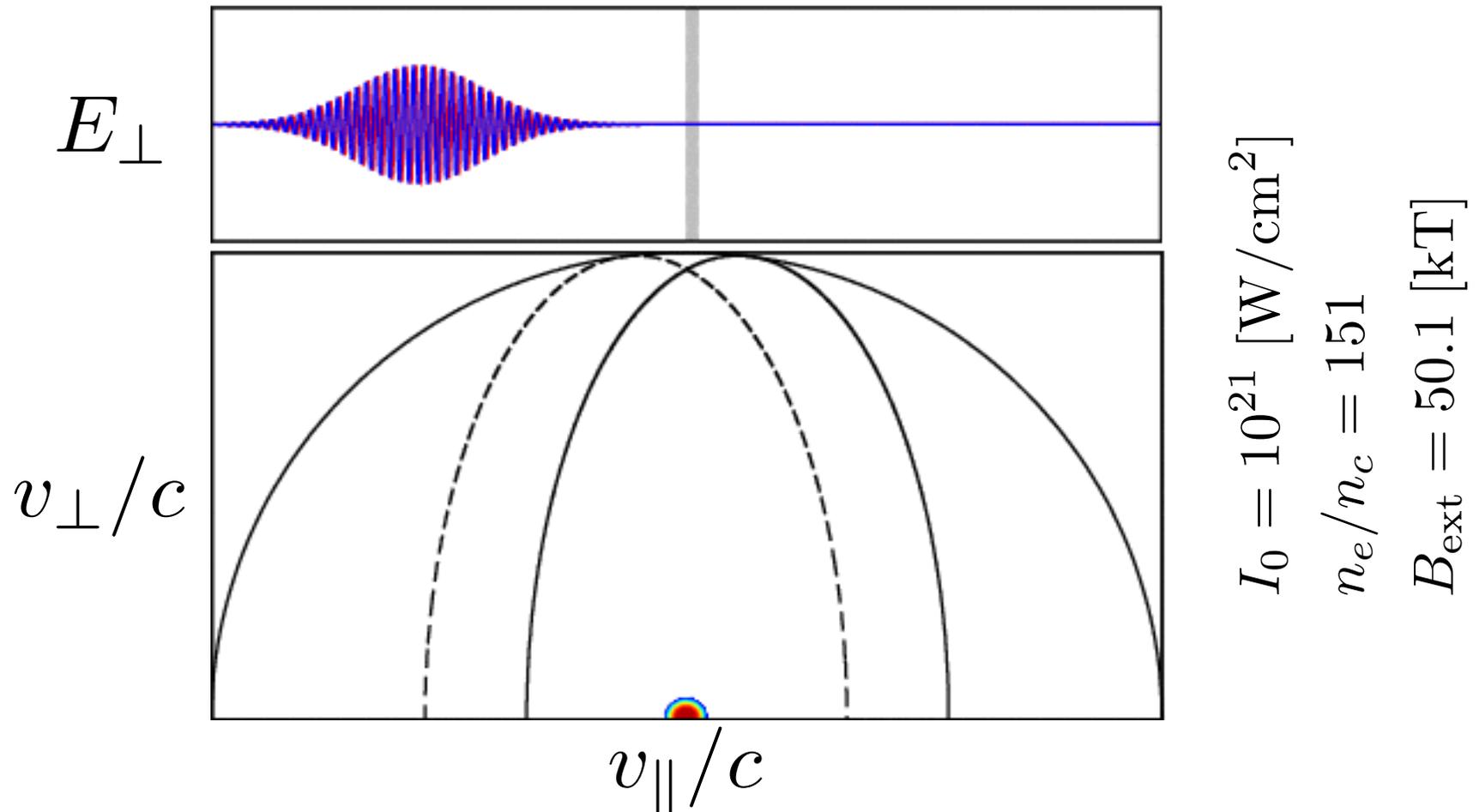
$$\lambda_0 = 0.8 [\mu\text{m}]$$

$$\tau_0 = 30 [\text{fs}]$$

$$t_f = 1 [\mu\text{m}]$$

Time Evolution of Electron Velocity

- Acceleration by Laser E Field → Further Pump up by Resonance



止まり木法の実装

- メインループにある電流密度を計算するサブルーチンを取り替えるだけ

```
=3
open(21,file='history.d',status='unknown')
do i = 1,iend
  t = dt*dble(i)
  if (mod(i,10000).eq.0) then
    write(*,'(a,1i9,a,1p1e15.7e2)') 'step =',i,', t =',t
  endif
  call bfield
  call velocity
  call position
  call current
  call bfield
  call efield
  if (mod(i,ihist).eq.0) then
    call history(t)
  endif
!   if (mod(i,iout).eq.0) then
!     call output(isnap)
!   endif
enddo
```

 **call current_v**

“余分な”計算部分

```
=3do ii = 1,nw  
  do i = 0,ng+1  
    jx(i) = jx(i)+jxw(i,ii)  
    jym(i) = jym(i)+jymw(i,ii)  
    jz(i) = jz(i)+jzw(i,ii)  
  enddo  
enddo
```

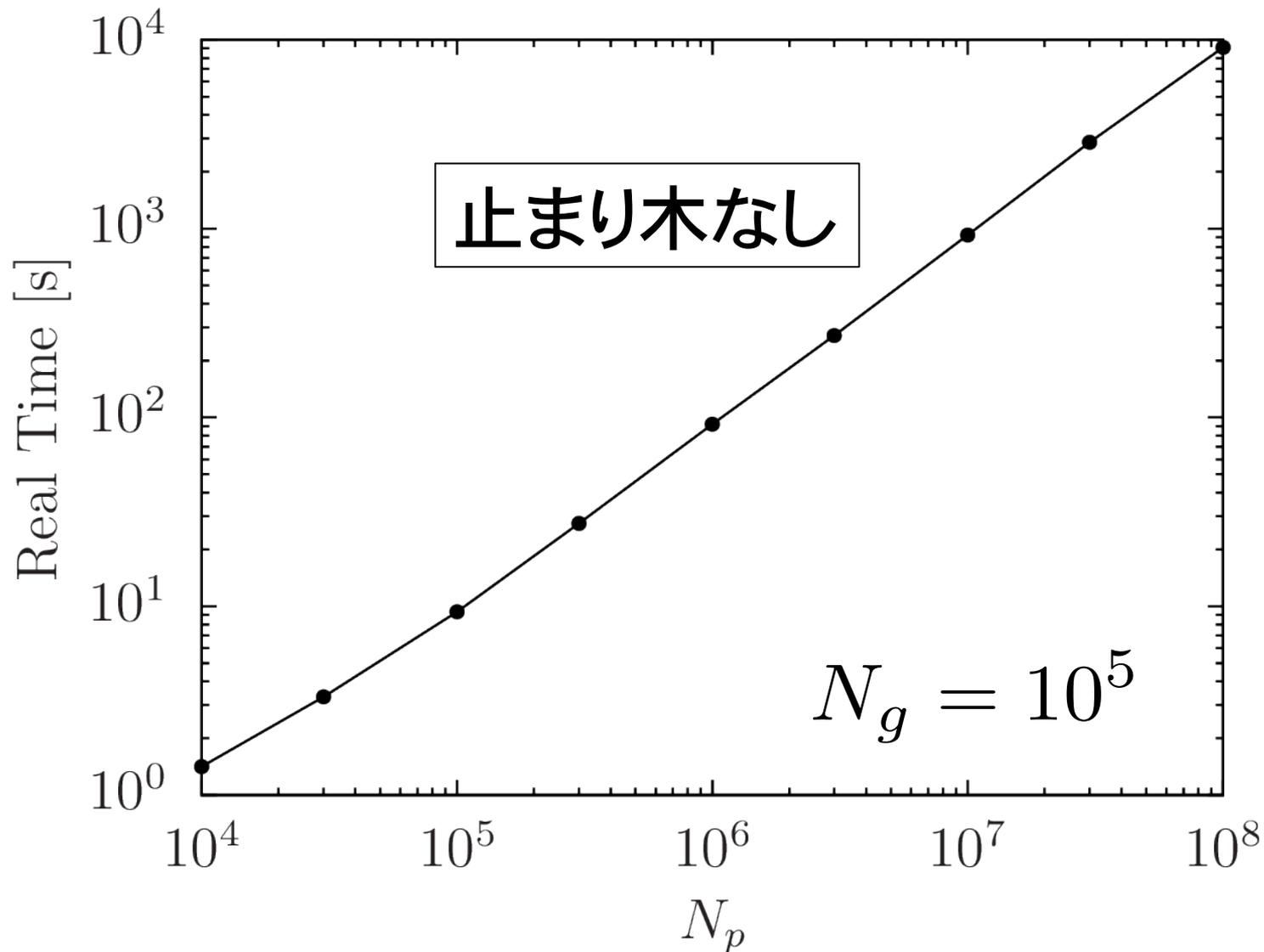
ベクトル化率の向上

- (グリッド数)=(粒子数)= 10^5 の場合

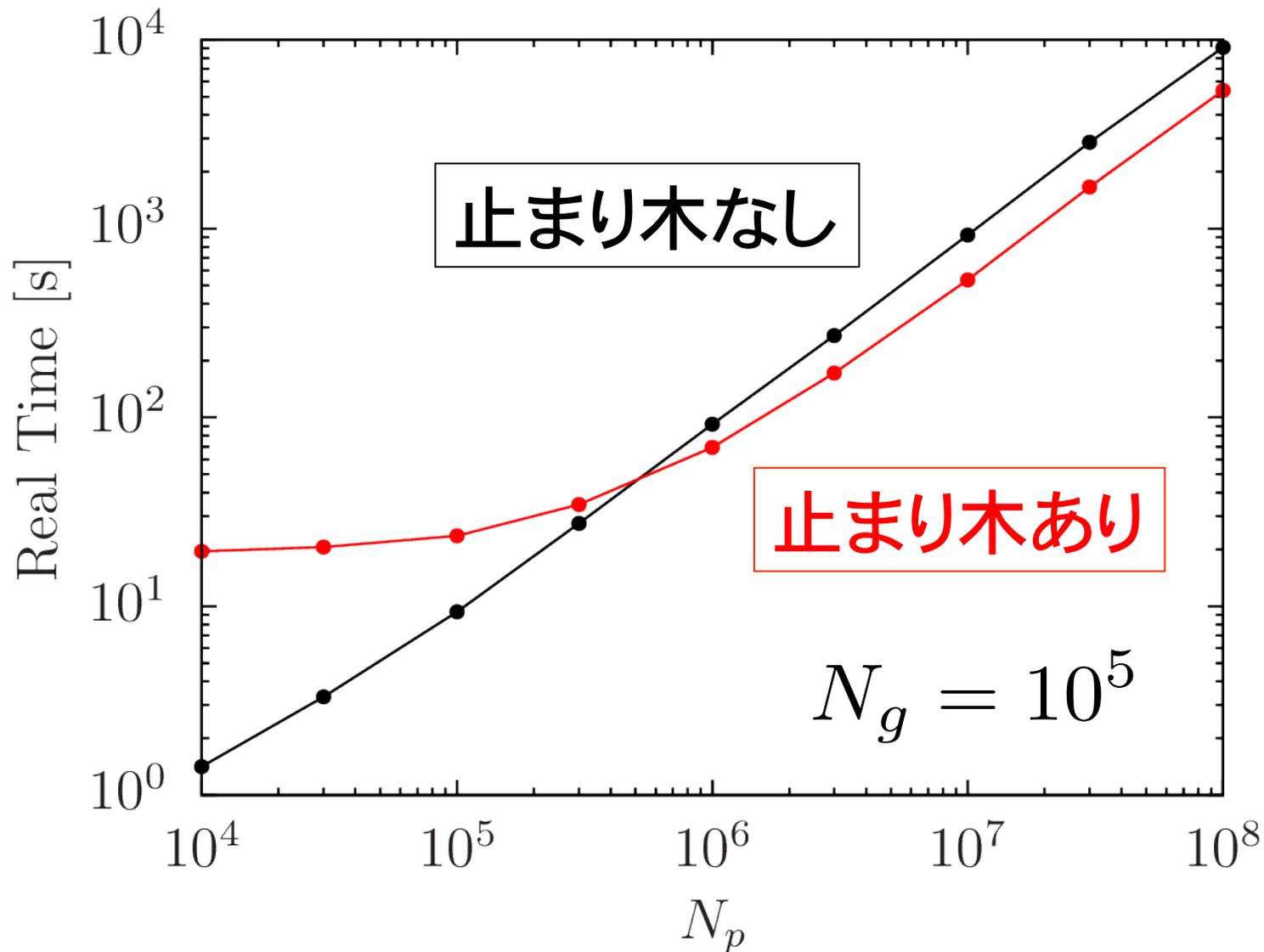
90%

→ 99.4%

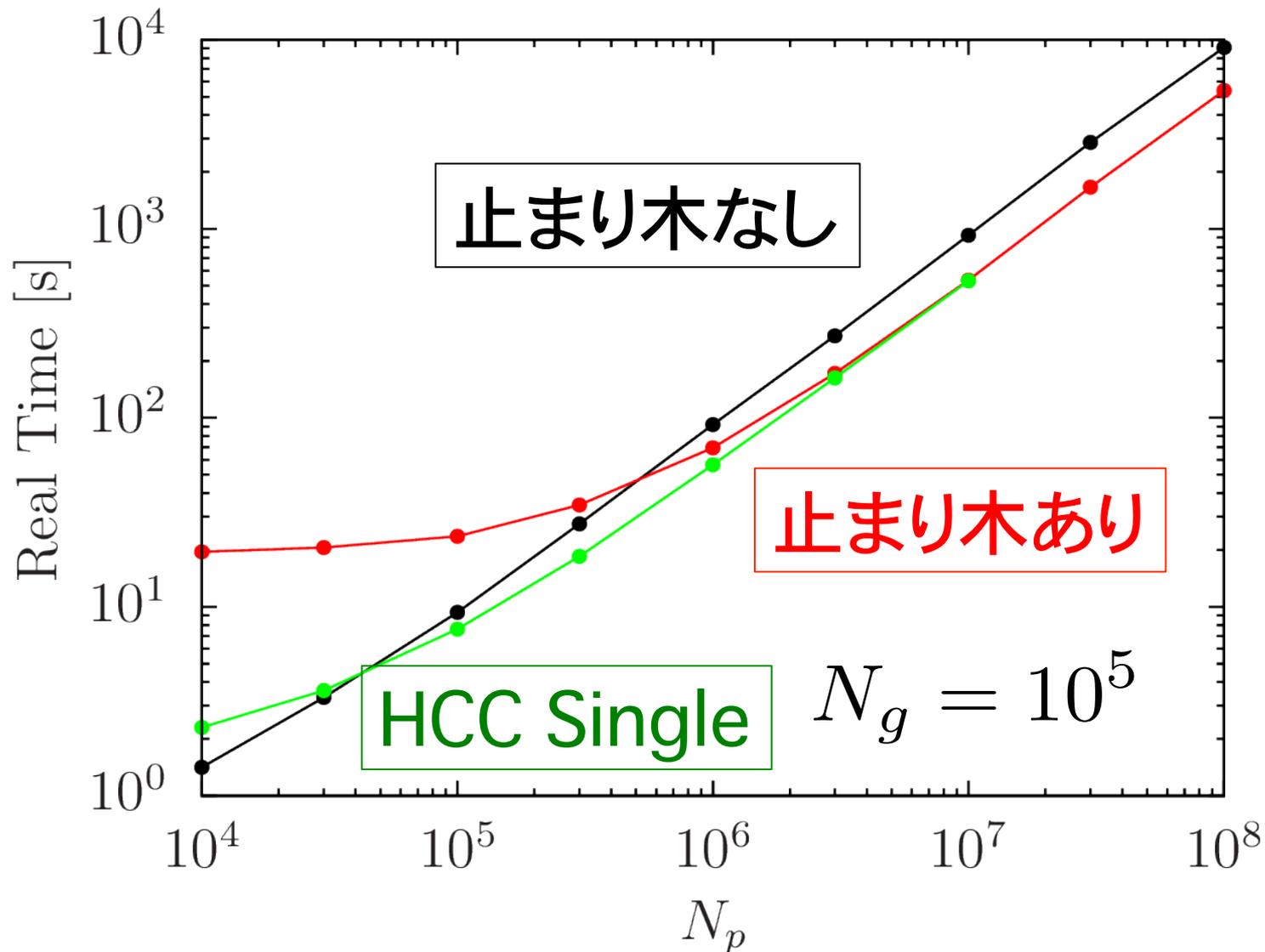
SXでの計算速度は？



SXでの計算速度は？



SXでの計算速度は？



まとめ(計算規模の小さい場合)

- 結論として、PICコードはベクトル計算に向いているとは言えない。。。
- 現在私がやりたい計算
- 小規模計算を大量にパラメータサーベイ
- HCCにシングルジョブを大量投入して実行(1日に500から1000モデルを計算)
- 大規模計算が必要になればVCCに移行
- あくまで私の場合ということですので。。。