大規模並列粒子シミュレーションコード開発用フレームワークFDPSの惑星形成・リング計算向け最適化

牧野淳一郎 (神戸大/理研), 岩澤全規 (理研), 行方大輔 (理研), 似鳥啓吾 (理研), Long Wang(理研), 谷川衝 (東大), 村主崇行 (理研), 細野七月 (京大)

2017/9/13 天文学会年会

講演概要

- 大体どういう話か
- 何を考える必要があるか
- 必要そうな変更
- 実現できそうな性能
- まとめ

大体どういう話か

- 計算機ハードウェアは速くなって、惑星形成・リング系シミュレーションを $O(N \log N)$ アルゴリズムでできる目処も大体たったので、ポスト京とかを使うと「すごく大きな」シミュレーションができる。
- リングだと 10^{12} 粒子とか。細いリングなら物理描像に近いところ。
- 但し、ノード数 10 万以上、コア数 1000 万以上とかになるので、まともに計算コードが動くかどうか分からない。
- 色々、おきそうな問題に対応し、現状で使える最大の計算機でテストした。
- 一応、なんとかなりそう。

何を考える必要がありそうか

- 結構近い将来に、計算機は数十万ノード、数千万コアに。
- リング系だと 1 兆粒子以上、惑星形成の長時間計算でも 100 億粒子以上が扱える。
- 計算アルゴリズムは、(P^{3}T が使えるとして) 基本的には Barnes-Hut ツリー。

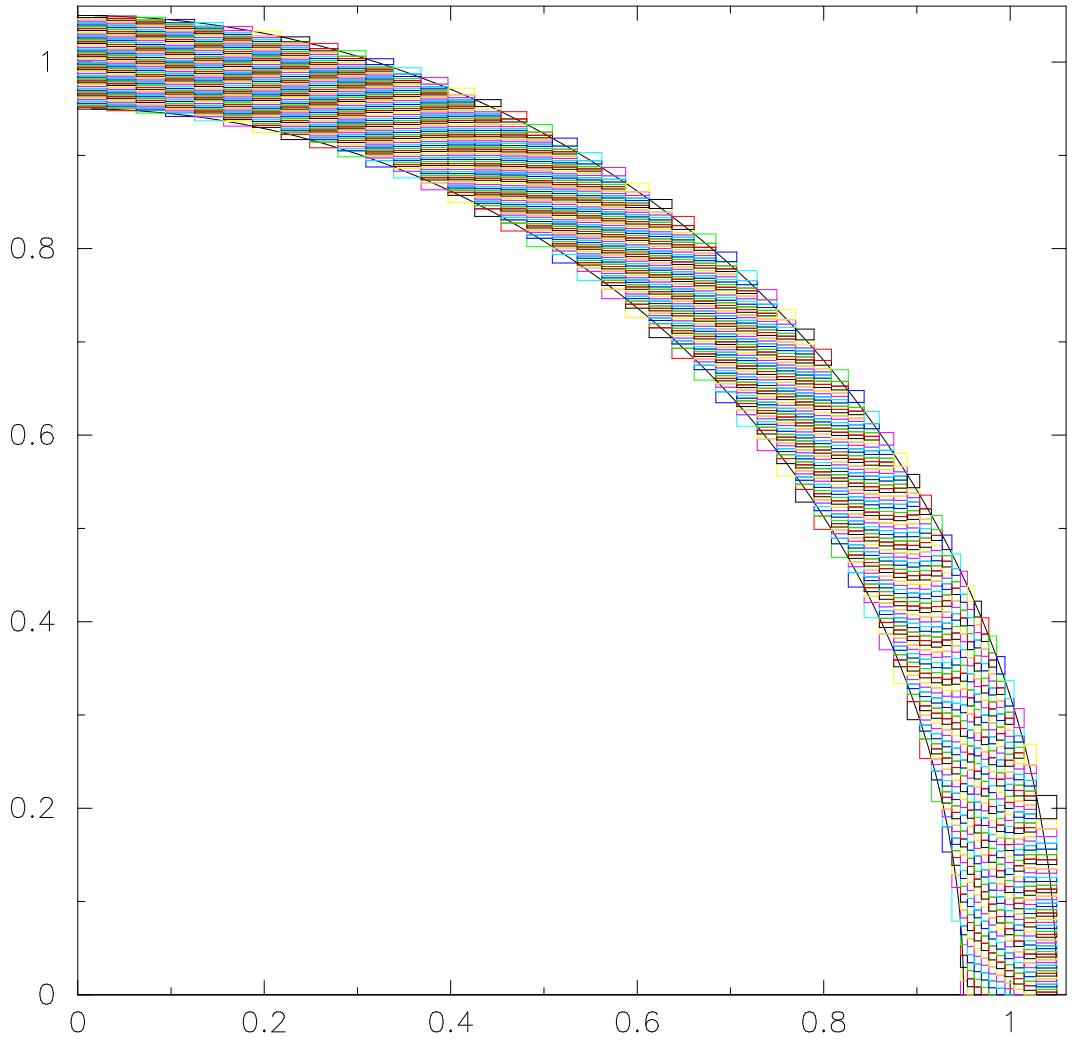
そもそもの問題: 1000 万コアで (あと色々駄目な計算機で)(HPL 以外に) 動く計算コードはあるのか？

- 若干楽観的に、1 兆粒子 100 万 MPI プロセスとしても、コアあたり 100 万粒子しかない。
- 実効性能 300PF、相互作用数 1000(=3 万演算) として、1 兆粒子で 1 ステップ 0.1 秒、100 億なら 1 ミリ秒。
- 容易に通信リミットになる。

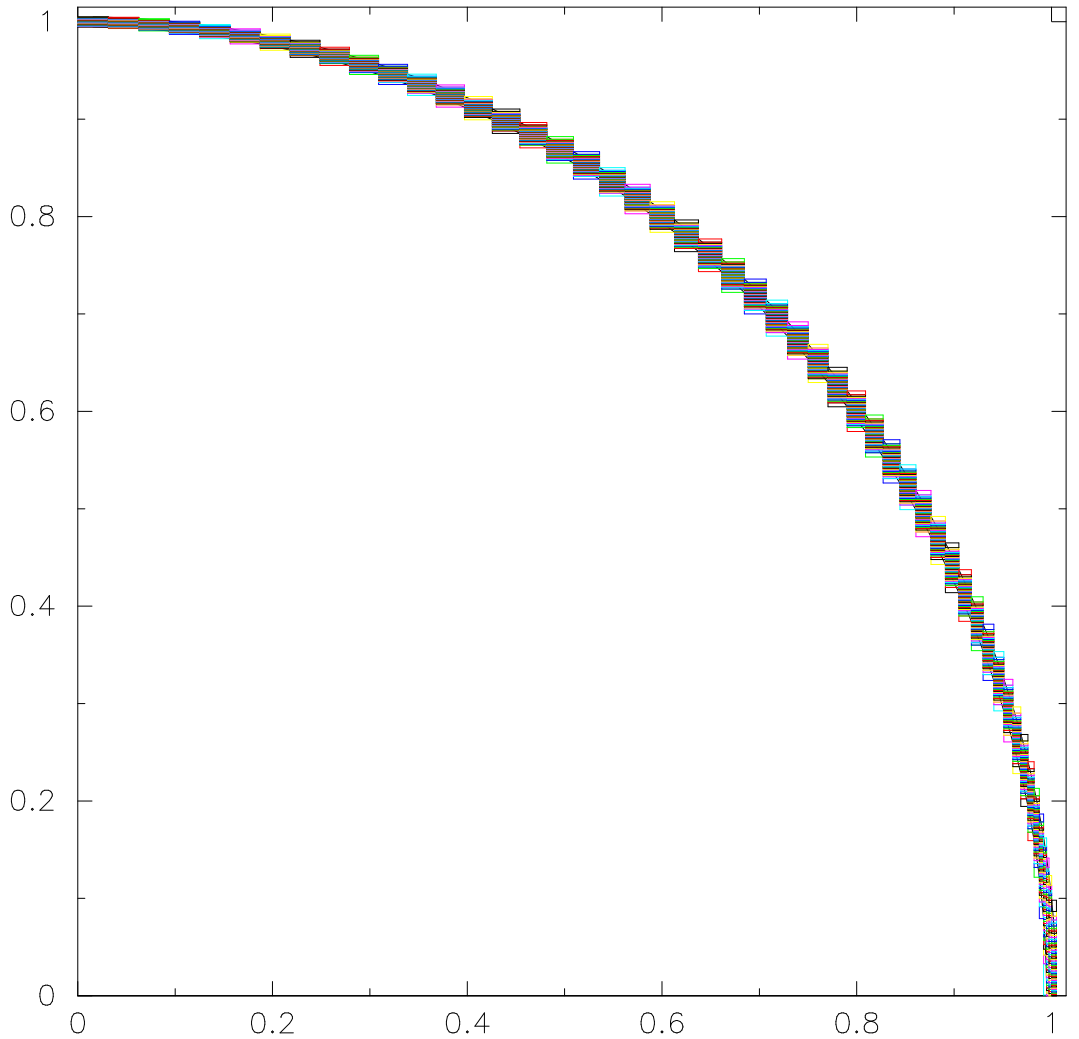
必要そうな変更

- とにかく通信を減らす必要がある。
  - 基本的に 2 次元分割 (縦は薄い):1 プロセスが担当する領域がすごく小さくなる
  - 通常の BH 木むけ空間分割ではドメイン形状も非常に悪くなる。特にリングが細いと大変。
- ~~駄目な~~将来の計算機の場合、遠距離通信が特に問題になる可能性がある。大域通信を呼ぶとありえないような時間かかったりする。

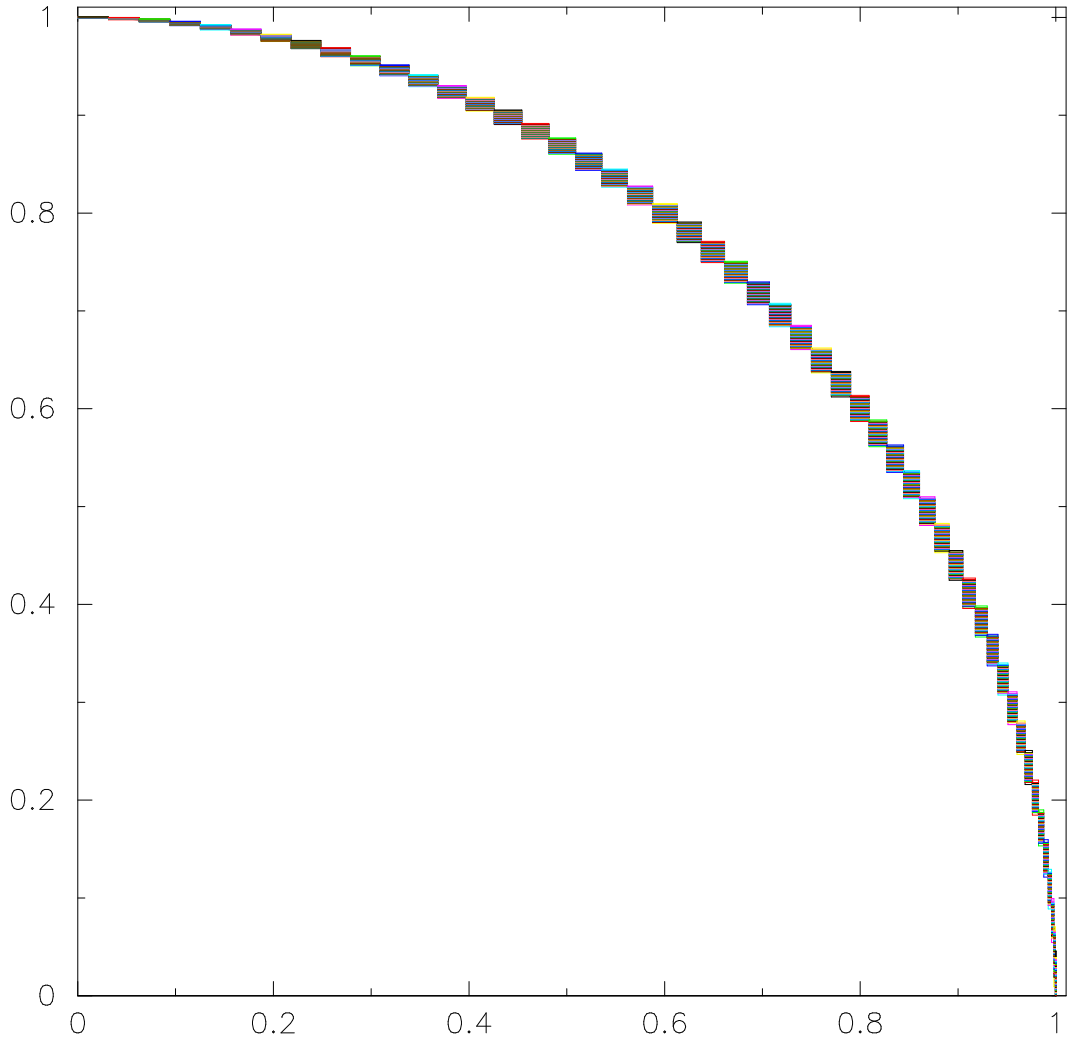
Ring width=0.1



Ring width=0.01



Ring width=0.001



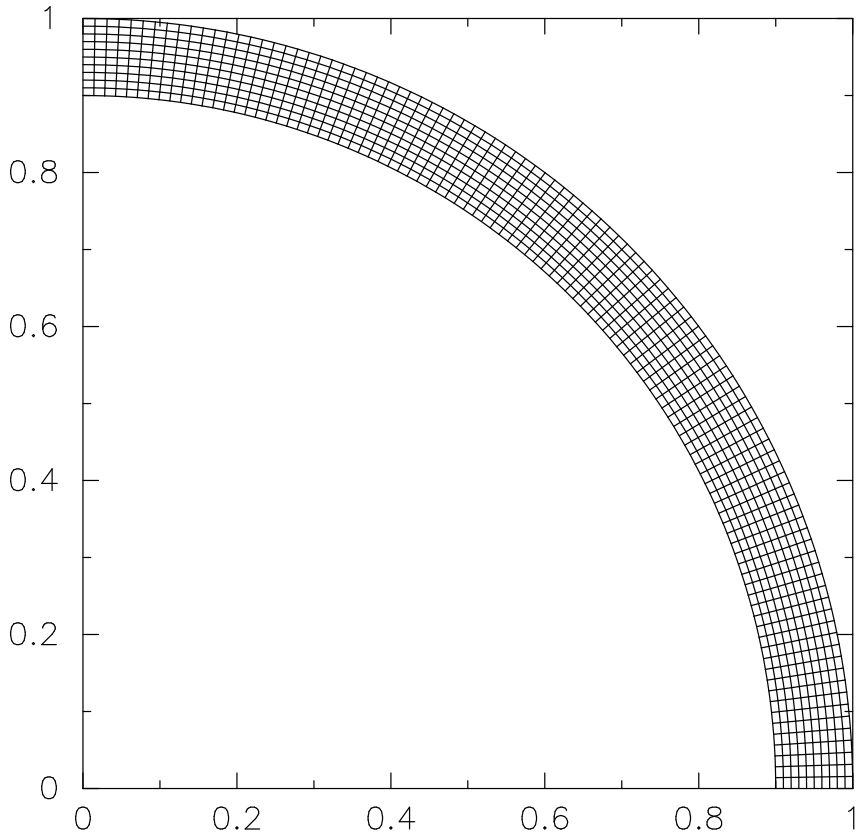
問題と対応

- リングが細くなると、計算領域も細長くなる。単純にアスペクト比に比例
- 具合が悪いプロセスで通信量がすごく増える。

対応:

- (比較的に) 細いリングなんだから、円筒座標系 $(\boldsymbol{R}, \boldsymbol{\theta}, \boldsymbol{z})$ で空間分割やツリー構築をすればいい。
- 空間分割は $\boldsymbol{\theta}$ 方向に細かく、$\boldsymbol{R}$ 方向はあらくして、領域がほぼ正方形になるようにする。
- (リングの半径が $\mathbf{1}$ くらいとして) ツリー構築から相互作用リスト構築までは全部円筒座標 $(\boldsymbol{R}, \boldsymbol{\theta}, \boldsymbol{z})$ をデカルト座標だと思ってやる。
- 本当の相互作用計算はもちろんデカルト座標でやる。ツリーのモーメントとかもそっちで計算。

分割例



幅 0.1 のリングを 400×10 に分割。領域のサイズは 0.016×0.01
ほぼ理想的な空間分割になる。

粒子移動

- ある領域の全粒子が、1 タイムステップ毎に隣とかもっと遠くのプロセスまで移動する。
- プロセス数が非常に大きいので起こる問題

姑息な対応:典型的なケプラー回転の速度で座標系を回転 (逆回転？) させる。

- リング中央では粒子は動かなくなる
- リング両端では、1 軌道周期でリングの幅くらい動く＝リングが細いほど具合がよくなる。

その他色々

- 大域通信
  - 我々の並列ツリー実装では、 $\mathbf{MPI\_AlltoAllv}$ を使っている。これは駄目な計算機ではまともに動かない。また、プロセス数に通信量が比例するわけで本質的に問題。
  - リングの場合、角度方向に離れた領域は半径方向にまとめてよいのは明らかなので、そういう「超領域」を作ってそれだけを通信するように。
  - まだ計算時間にプロセス数の平方根に比例する成分があるが、原理的には θ 方向もまとめた超領域をつくれば定数時間にできる。
- 駄目な計算機では相互作用計算以外のあらゆるものがすごく遅いとかあるので、ツリー構築・相互作用リスト構築を複数ステップに 1 度にして相互作用リストを使い回すことを可能にした。

実現できそうな性能 (1)

将来の計算機 (婉曲表現) のモデルとして、中国の **Sunway TaihuLight** に実装。ベースのコードは **FDPS** だが、まだ公開版に全部実装されているわけではない。

- **40960** ノード、**1** ノード **3TF**
- **1** ノード $= 4 * (1 + 64)$ プロセッサ
- **1** プロセッサ **(MPE)** はデータキャッシュあり、あと **(CPE)** はデータキャッシュなしでローカルメモリ **64KB**。主記憶には基本的に **DMA** アクセス
- **64 CPE** の間は低レイテンシのレジスタ-レジスタ通信あり

実現できそうな性能 (2)

改良前 (複数ステップ使い回す、座標回転まで実装)

- そもそも細いリングでは計算動かなかった。幅 0.1 くらいの太いリングではなんとかとまらないで動く。
- 実行効率 10% くらい

改良後 (演算カーネル、ロードバランスの改良の効果もあるが)

- 実行効率 30% くらい。カーネル自体が 50% くらいなので結構限界。
- まだ全ノード測定ができてないが、計算上は 1 兆粒子 1 ステップ 1 秒くらいでできる。

実現できそうな性能 (3)

将来のいろいろな計算機で、ある程度の粒子数があればこの程度の性能が実現可能と考えられる

- ある程度の粒子数: 現状では、1 タイムステップに 0.1 秒以上程度、、、
- 1 日で 100 万ステップ回るのでまあ実用的。

まとめ

- 非常にコア数、ノード数の多い、「将来の計算機」での惑星形成計算、リング系計算のために必要になる改良をおこなった。
- 円筒座標系での空間分割、ツリー構築、座標回転、通信量低減のための超領域構築等である。
- **Sunway TaihuLight** で、そもそも計算が動かなかった細いリングでも計算可能で、ほぼ理論限界に近い性能がでるようになった。
- (もうちょっとまともな計算機なら将来でも十分性能が出せると期待する)
- **FDPS** の次期バージョンではこの辺使えるように