

PROGRAPEによる 銀河シミュレーション

中里直人
理化学研究所

No.2

銀河という複雑な天体

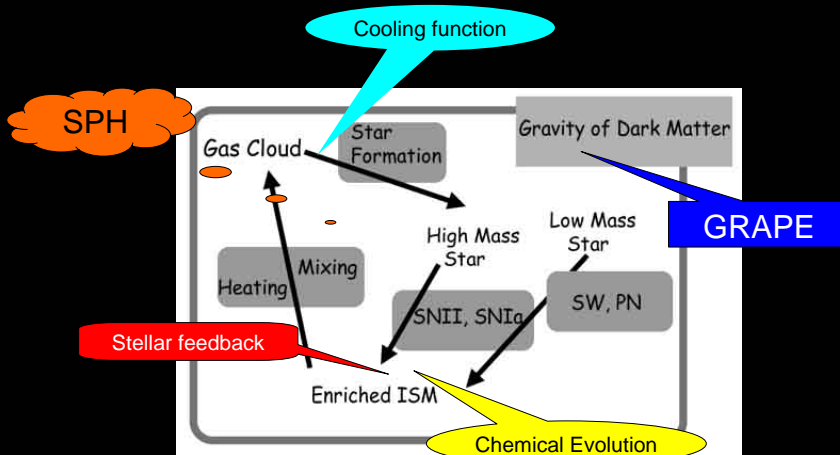


Multiwavelength astronomy

銀河は様々な構成要素からなっている
主に、恒星の集団 + ガス雲

GENSO (Galaxy EvolutionN with the SPH methOd)

GRAPEを活用した化学力学進化シミュレーションコード



SPH百万粒子の計算を一ヶ月で

- 目的: SPH&N-bodyシミュレーションで銀河進化を精密に調べたい
- 現状
 - $N_{\text{sph}} \sim 60\text{k model} : 10 \text{ days (PC + GRAPE6)}$
- SPHの部分を10倍以上高速にしたい
 - 見込みのありそうな手法:
 1. Massively Parallel Computer
 2. “PC+GRAPE” クラスタ
 3. **ハードウェアによる加速**

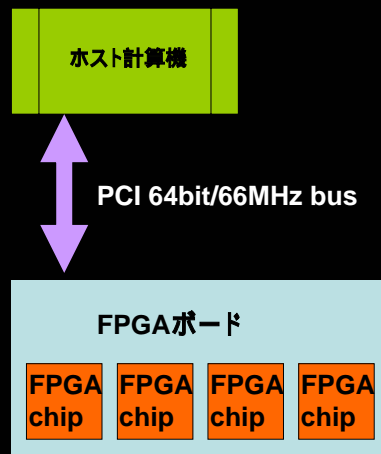
GRAPEからPROGRAPEへ

- GRAPEでは**カスタムLSI**を利用
 - 任意の機能を持ったLSIを発注できる
 - 利点
 - 高速に動作する
 - チップ単価が安い～数万円/チップ
 - 欠点
 - 一度設計を決めたら、機能は変化できない
 - 固定パイプラインからSIMD型GRAPE-DRへ
 - 半導体技術の微細化により設計コスト増大
 - GRAPE-6 ～1億円
- **GRAPEのよい所を残して、さらに汎用を持たせたい！ PROGRAPE**

現状のシステム: PROGRAPE-3



カスタムLSI



FPGA

低コストPROGRAPEボード

- PGRを様々なハードウェアに移植すると同時に、新規PROGRAPE-4ボードを開発
 - 低価格FPGAチップ(Spartan-3)の採用
 - 無駄な回路を省いた設計
 - PCI-Xを採用 (理論性能 1GB/sec)
 - XC3S5000 4個搭載
 - PROGRAPE-3と同等のスライス数
 - PROGRAPE-3より大幅に安い予定

本研究の一部は、IPA未踏ソフトウェア開発事業の援助を受けています

FPGAでの数値計算の現状

- これまで整数(固定小数)演算が主流
- ようやく最近、浮動小数点演算が実用的になりつつある
 - ただし、倍精度の計算は現状ではまだ難しい
 - 換言すると、並列可能な演算器数が少ないため、ピーク性能がでない
- 数TflopsまではカスタムLSIより低コスト
- FPGAは汎用演算器(FPU)として使える
 - 原理的には粒子法に限らず
- ただし、「プログラム」するのは結構難しい

PROGRAPEでできること

$$\bullet \text{GRAPE} \quad \frac{d\vec{v}_i}{dt} = -G \sum \frac{m_j (\vec{r}_i - \vec{r}_j)}{(|\vec{r}_i - \vec{r}_j|^2 + \varepsilon^2)^{3/2}}$$

$$\bullet \text{PGR} \quad \vec{F}_i^\perp = \sum f(\vec{X}_i^\perp, \vec{X}_j^\perp)$$

- 演算が複雑で、パラメーターが多い問題用
 - 天体シミュレーションでの応用想定
 - SPH法による流体シミュレーション
 - 重力相互作用も含む
 - 分子間力、多体力など、様々な可能性

SPH相互作用

- 総和形式の相互作用という意味では重力と似ているが、かなり”ややこしい”相互作用

$$\rho_i = \sum m_j W(\vec{r}_i - \vec{r}_j; h), h = \frac{1}{2}(h_i + h_j)$$

$$\rho_i (\nabla \cdot \vec{v}_i) = \sum (\vec{v}_i - \vec{v}_j) \cdot \nabla W(\vec{r}_i - \vec{r}_j; h)$$

$$\rho_i (\nabla \times \vec{v}_i) = \sum (\vec{v}_i - \vec{v}_j) \times \nabla W(\vec{r}_i - \vec{r}_j; h)$$

$$\frac{d\vec{v}_i}{dt} = -\sum m_j \left(\frac{P_i}{\rho_i^2} + \frac{P_j}{\rho_j^2} + \Pi_{ij} \right) \nabla W(\vec{r}_i - \vec{r}_j; h)$$

$$\frac{d\vec{u}_i}{dt} = \sum m_j \left(\frac{P_i}{\rho_i^2} + \frac{1}{2} \Pi_{ij} \right) (\vec{v}_i - \vec{v}_j) \cdot \nabla W(\vec{r}_i - \vec{r}_j; h)$$

演算精度のテスト(1)

- 演算精度がSodの衝撃波管問題にどの程度影響するのか？

– 浮動小数点形式: $s \times f \times 2^{\text{exp}}$

符号

指数部

仮数部

– CPUでの演算: 単精度 または 倍精度

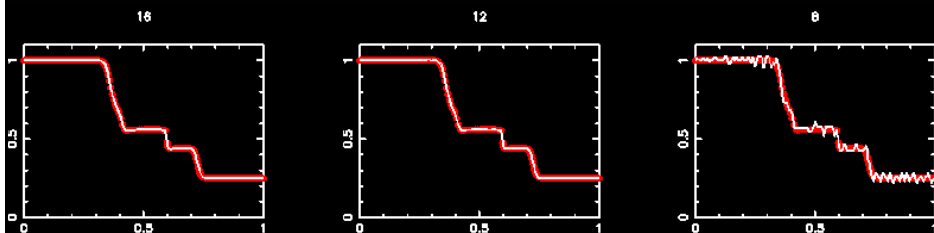
- 単精度では指数部8ビット 仮数部24ビット
- 倍精度では指数部11ビット 仮数部53ビット

– PROGRAPEでは任意の精度を使用可能

- 仮数部が小さいほどFPGAを有効利用できる

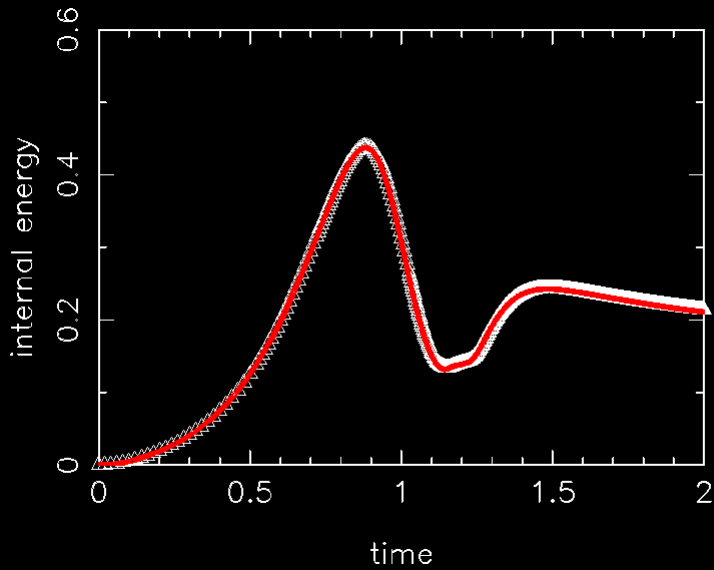
演算精度のテスト(2)

- 指数部は8ビットに固定して仮数部のビット幅を変えた場合の変化について調べた



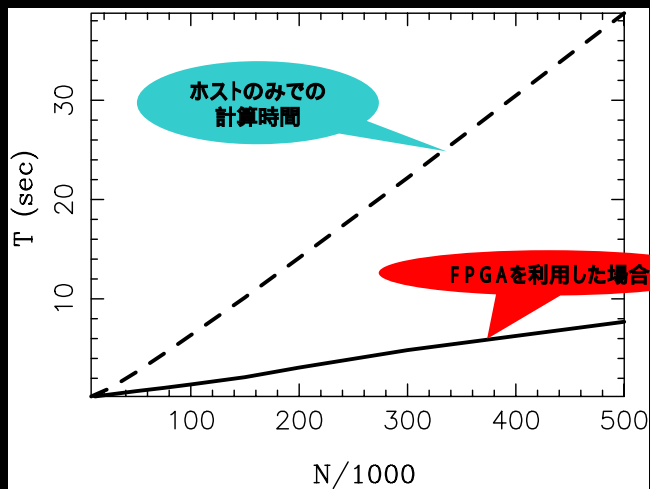
- 16ビット: ほぼ倍精度と違いがない
- 12ビット: 多少人工的な振動
- 8ビット: 人工的な振動が見られる

3次元テスト計算



性能の比較(1)

- 1ステップの計算時間(粒子分布は等温球)



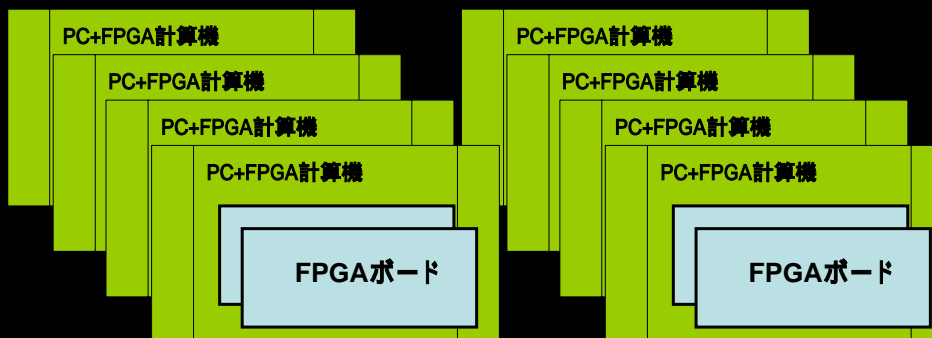
性能の比較(2)

自己重力計算も含んだ場合の現実的な比較

N	HOST	PROGRAPE 1	PROGRAPE 2	PROGRAPE 2T
25000	2.30	1.39 (1.6)	0.54 (4.3)	0.49 (4.7)
50000	6.63	3.35 (2.0)	1.30 (5.1)	0.97 (6.8)
100000	17.8	7.96 (2.2)	3.47 (5.1)	1.95 (9.1)
500000	107	45.7 (2.3)	51.9 (2.1)	9.62 (11.1)

ホスト(倍精度演算)より、5倍から11倍速い

将来のシステム



「PC+FPGA 2枚」を8組あわせてクラスターとする
 1組あたりの**実効性能**: 重力計算 ~ 300GFLOPS
 SPH計算 ~ 80GFLOPS
 8組で重力計算を行えば、1TFLOPS以上実行性能

まとめ

- **FPGAによるスーパーコンピューティング**
 - 低精度浮動小数点演算が実行可能になった
 - 汎用的なGRAPE型計算機の構築
 - プログラミングシステムの開発
 - SPH法ぐらい複雑な演算も可能
 - 演算性能は銀河シミュレーションに最適
 - 重力 約300GFLOPS
 - SPH 約80GFLOPS
- **SPH法による天体物理シミュレーション**
 - より高解像度のシミュレーション
 - 100万體銀河シミュレーションが現実的に