

# 並列計算機とシミュレーション

中里直人

分散平行処理研究室

242-B

[nakasato@u-aizu.ac.jp](mailto:nakasato@u-aizu.ac.jp)

# 自己紹介

- 中里直人 (なかさと なおひと)
  - 会津大学 准教授 博士(理学)
  - 専門分野 天文シミュレーションとHPC
    - SPH法による銀河形成シミュレーション
    - FPGAボードによる専用計算
    - 専用計算機の研究開発
    - 専用計算機用ソフトウェアの研究開発
    - メニーコア計算機による高性能計算
  - Web <http://galaxy.u-aizu.ac.jp/trac/note/>

# 全てが並列計算の時代

- 私に関係ないなどということはありません
  - 携帯電話 : 1 – 4 並列な計算機
  - パソコン : 2 – 6 並列な計算機
  - ゲーム機 : 3 – 6 並列な計算機
  - GPU : ~ 100 – 1000 並列な計算機
  - スパコン : ~ 1000 - 100000 並列な計算機
  - Google : ~ 10000 並列な計算機
  - インターネット : ~ 1000000000 並列な計算機

# 卒研のプロジェクトについて

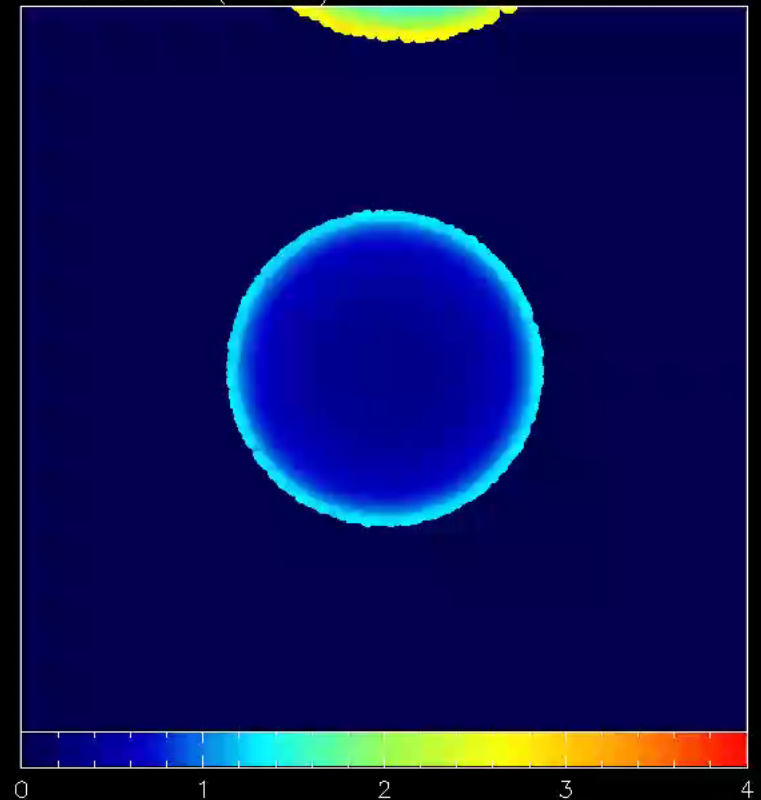
- **メニーコア計算機の活用**
  - GPU, Cell, FPGA
- **GRAPE-DR/GRAPE-MP**
  - 手作り計算機
- **新規コンパイラ開発**
  - これまでにないプログラミング手法の研究
- **京コンピュータ (予定)**
  - 最先端のマルチコアプログラミング

# 並列計算機を使う

- GPUという並列計算機のプログラミング

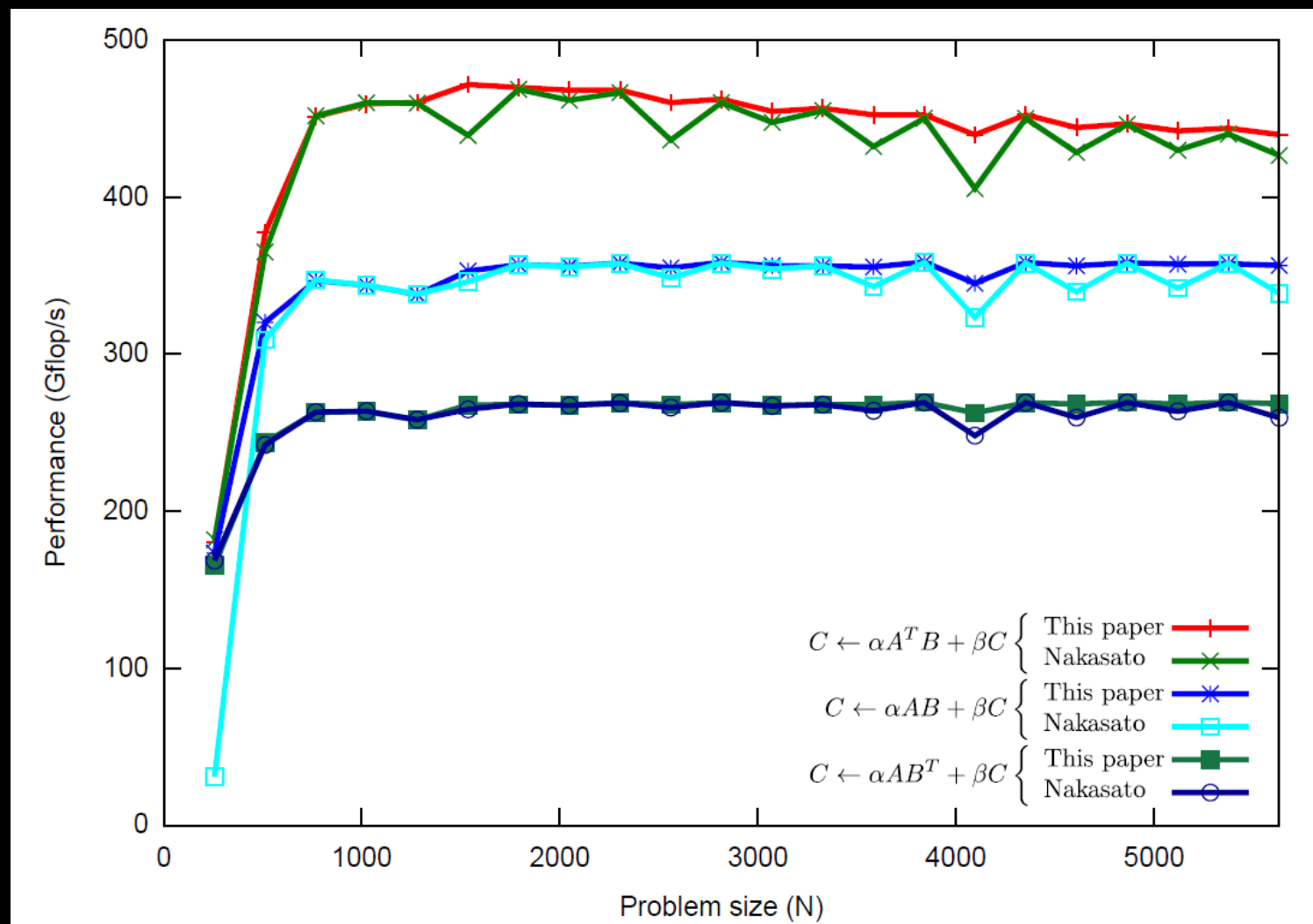


27839.6 (0.02) km : 0.25 sec



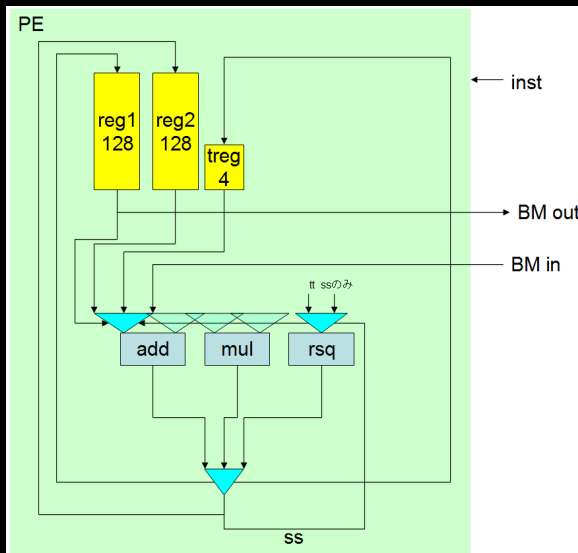
# GPUによる行列乗算: 世界一

- Matsumoto, NN, Sakai, Sedukhin (2010)



# 並列計算機を設計する実現する

- 商用の計算機に**立ち向かう**には？
  - 大量生産のため安い
  - なんでもできる
- 問題を特化した**効率の良い計算機**



```
s0 : swap_128 port map ( f => dx(11), x => x, y => y, xs => xx, ys => yy);

with dx(11) select
diff <= dy when '1',
dx when others;

e0 : extract_128_116_11 port map ( x => xx, s => sx, m => mx0, e => ex);
e1 : extract_128_116_11 port map ( x => yy, s => sy, m => my0, e => ey);

with ex select
mx <= (others => '0') when "00000000000",
mx0 when others;

with ey select
my <= (others => '0') when "00000000000",
my0 when others;

qx <= mx;

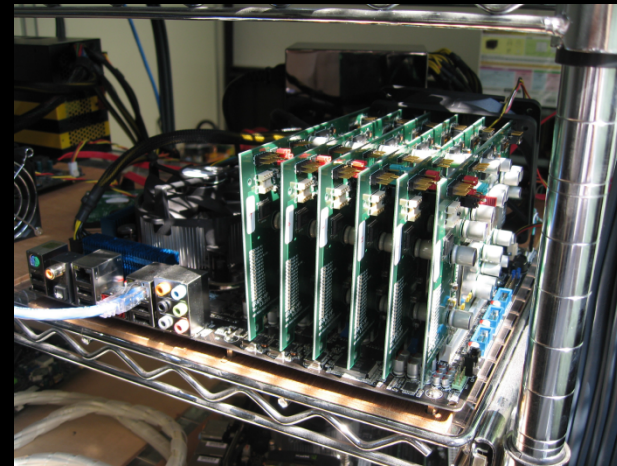
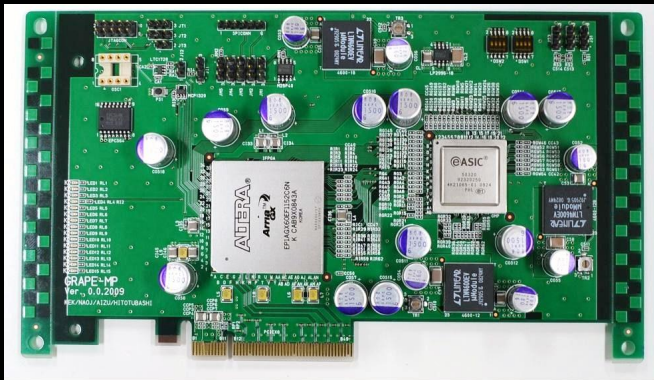
ff <= diff(10) or diff(9) or diff(8) or diff(7);
with ff select
shift <= "1111111" when '1',
diff(6 downto 0) when others;

rs : int_rshift_V port map ( c => shift, i => my, o => qy0, s => rf1);
qy(0) <= qy0(0) or rf1; -- force1 rounding
qy(116 downto 1) <= qy0(116 downto 1);
```



# GRAPE-MP/Network board

- GRAPE-MP 四倍精度演算用計算機



- メッシュ型クラスター

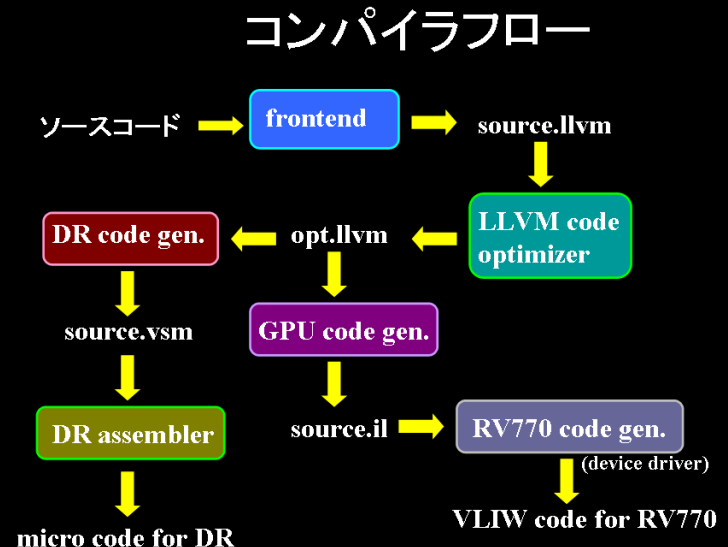




# コンパイラの開発と応用

- SIMD計算機の利用を簡単に
  - 計算モデルの考案・改良
  - コンパイラの実装 (LLVMの活用)

- 実装コンパイラの応用
  - 多倍長演算
  - 高速シミュレーション



# 計算したい数式

$$\begin{aligned} I &= \int_0^1 dx \int_0^{1-x} dy \int_0^{1-x-y} dz F(x, y, z), \\ F(x, y, z) &= D(x, y, z)^{-2} \\ D &= -xys - tz(1 - x - y - z) + (x + y)\lambda^2 \\ &\quad + (1 - x - y - z)(1 - x - y)m_e^2 \\ &\quad + z(1 - x - y)m_f^2. \end{aligned} \tag{2}$$

# LSUMPでの定義

## 効率のよい言語仕様を自分で定義

```
VARI xx, yy, cnt4;  
VARJ x30_1, gw30;  
VARF res;  
CONST tt, ramda, fme, fmf, s, one;
```

~ 10行

```
zz = x30_1*cnt4;
```

```
d = -xx*yy*s-tt*zz*(one-xx-yy-zz)+(xx+yy)*ramda**2 +  
      (one-xx-yy-zz)*(one-xx-yy)*fme**2+zz*(one-xx-yy)  
*fmf**2;
```

```
res += gw30/d**2;
```

# LLVMでのコード

```
define double @main0(double %xx, double %yy, double %cnt4, double %x30_1, double %gw30) {
    %multmp = mul double %x30_1, %cnt4                ; <double> [#uses=3]
    %negtmp = sub double -0.000000e+00, %xx          ; <double> [#uses=1]
    %multmp1 = mul double %negtmp, %yy               ; <double> [#uses=1]
    %gtmp = load double* @s                          ; <double> [#uses=1]
    %multmp2 = mul double %multmp1, %gtmp            ; <double> [#uses=1]
    %gtmp3 = load double* @tt                        ; <double> [#uses=1]
    %multmp4 = mul double %gtmp3, %multmp           ; <double> [#uses=1]
    %gtmp5 = load double* @one                      ; <double> [#uses=1]
    %subtmp = sub double %gtmp5, %xx                ; <double> [#uses=1]
    %subtmp6 = sub double %subtmp, %yy              ; <double> [#uses=3]
    %subtmp7 = sub double %subtmp6, %multmp         ; <double> [#uses=2]
    %multmp8 = mul double %multmp4, %subtmp7        ; <double> [#uses=1]
    %subtmp9 = sub double %multmp2, %multmp8        ; <double> [#uses=1]
    %addtmp = add double %xx, %yy                   ; <double> [#uses=1]
    %gtmp10 = load double* @ramda                   ; <double> [#uses=2]
    %exptmp = mul double %gtmp10, %gtmp10           ; <double> [#uses=1]
    %multmp11 = mul double %addtmp, %exptmp         ; <double> [#uses=1]
    %addtmp12 = add double %subtmp9, %multmp11      ; <double> [#uses=1]
    %multmp20 = mul double %subtmp7, %subtmp6        ; <double> [#uses=1]
    %gtmp21 = load double* @fme                     ; <double> [#uses=2]
    %exptmp22 = mul double %gtmp21, %gtmp21        ; <double> [#uses=1]
    %multmp23 = mul double %multmp20, %exptmp22    ; <double> [#uses=1]
    %addtmp24 = add double %addtmp12, %multmp23     ; <double> [#uses=1]
    %multmp28 = mul double %multmp, %subtmp6        ; <double> [#uses=1]
    %gtmp29 = load double* @fmf                     ; <double> [#uses=2]
    %exptmp30 = mul double %gtmp29, %gtmp29        ; <double> [#uses=1]
    %multmp31 = mul double %multmp28, %exptmp30    ; <double> [#uses=1]
    %addtmp32 = add double %addtmp24, %multmp31     ; <double> [#uses=2]
    %exptmp33 = mul double %addtmp32, %addtmp32    ; <double> [#uses=1]
    %divtmp = fdiv double %gw30, %exptmp33         ; <double> [#uses=1]
    %accumtmp = tail call double @accum(double %divtmp) ; <double> [#uses=1]
    store double %accumtmp, double* @res
    ret double 0.000000e+00
}
```

~ 50行

# GPUで動作するコード

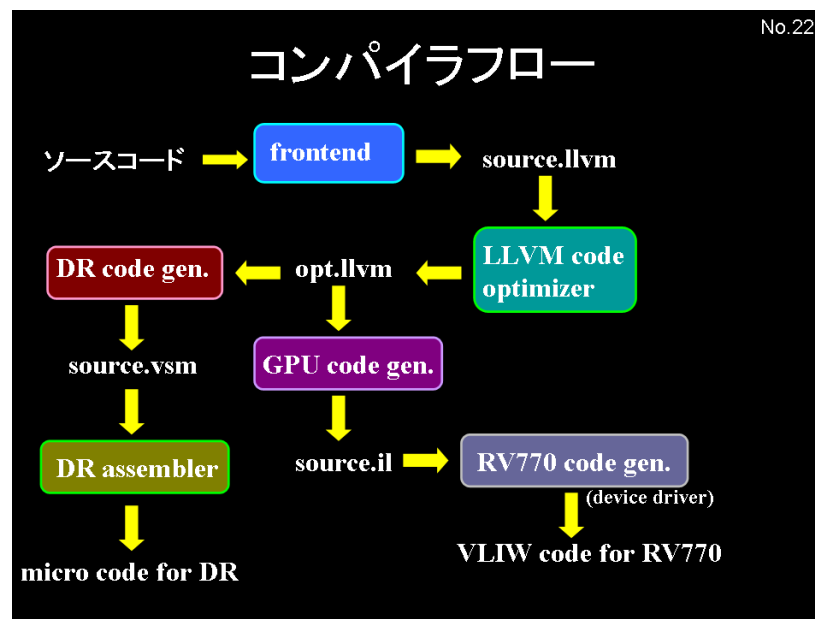
```
sample_resource(0)_sampler(0) r100, v0.xy
sample_resource(1)_sampler(1) r101, v0.xy
sample_resource(2)_sampler(2) r102, v0.xy
;load constants
mov r500, cb0[0]
mov r501, cb0[1]
mov r502, cb0[2]
mov r503, cb0[3]
mov r504, cb0[4]
mov r505, cb0[5]

; zero accum
mov r300, 110.xxxx

mov r200.xy, 111.xy
mov r201.w, cb10[0].w
mov r201.z, cb10[0].z
ixor r201.x, r201.x, r201.x
itof r203.x, r201.z

whileloop
  sample_resource(3)_sampler(3) r120, r200.xy
  sample_resource(4)_sampler(4) r121, r200.xy
  mov r30, r120
  mov r31, r102
  call 2
  mov r310, r32
  mov r320, r100_neg(yw)
  mov r30, r320
  mov r31, r101
  call 2
  mov r311, r32
  mov r30, r311
  mov r31, r504
  call 2
  mov r313, r32
  mov r30, r500
  mov r31, r310
  call 2
  mov r318, r32
  mov r30, r505
  mov r31, r100
  call 1
```

~ 600行



# 配属に必要な条件

- プログラミングが好きである得意であること
- 最先端の研究に取り組みたい
- 大学院への進学希望者を歓迎
- [nakasato@u-aizu.ac.jp](mailto:nakasato@u-aizu.ac.jp) 242-B
  - 研究室 242-E

# 配属後の進め方

- 週に1度のセミナー(90分)
  - 教科書の輪講(3年終わりから4年前期)
    - 2008年 **Structure and Interpretation of Computer Program**
    - 2009,2010,2011年 **Algorithm Design Manual**
  - 卒研の研究テーマについてのセミナー(4年後期)
- 研究テーマの決定
  - 4年の夏休みころまでに