

# Fortran90を用いたお手軽grid model構築ツールgmsの開発

九大院・理・地球惑星  
中野満寿男・中島健介  
masuo@geo.kyushu-u.ac.jp

地球惑星科学合同大会・情報地球惑星科学J031-007

2004.5.10

# はじめに

- **お手軽**に数値実験を行える環境が整った
  - PCの高速化・低価格化
  - 気象分野では高機能モデルの登場・公開
- 高機能モデルで**お手軽**に      が再現できる  
る

# 再現 理解

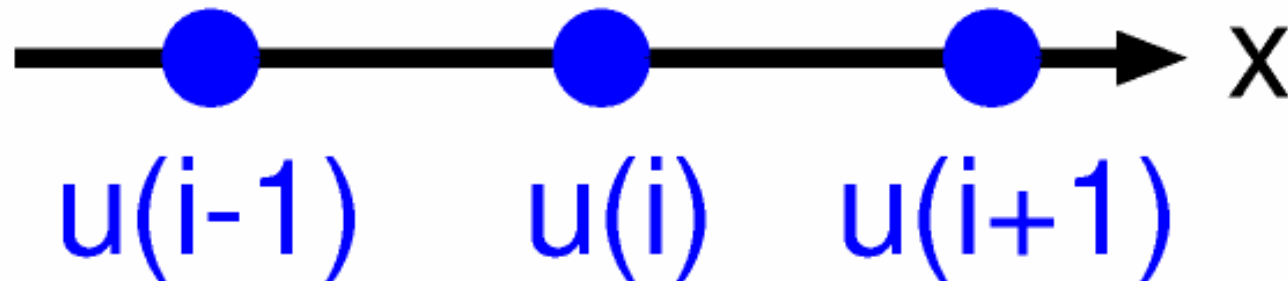
- 重要な物理機構は何ですか？？
  - 理解のために簡単モデルによる計算が必要
    - 高機能モデルの改変は大変！
      - 読めない。読む気にならない。
    - ゼロから自作するとなると
      - 重労働！

# 目的

- 格子点モデルを**お手軽に**構築したい
  - 可読性もよくなるとうれしい
  - お手軽で可読性がよければバグも減るだろう
- スペクトルモデルでの試み
  - `smodel` (竹広ら、2004)

# なぜ読めない？ 重労働？

似たような配列添え字を何度も書かないといけない

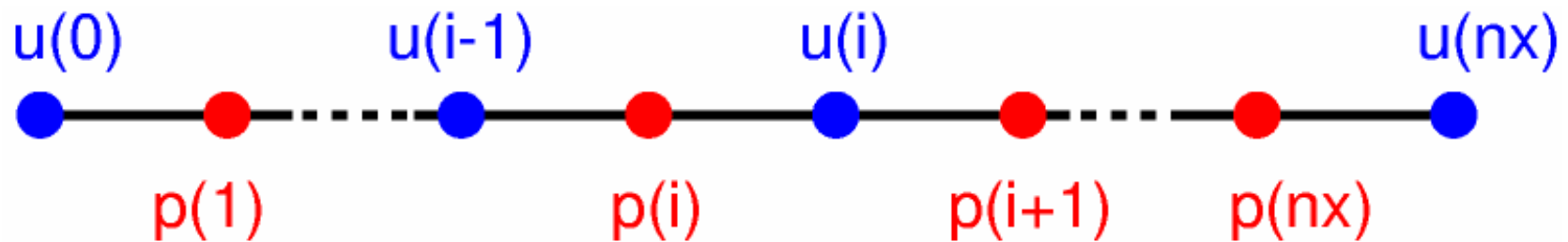


$\frac{\partial u}{\partial x}$  をfortranで2次精度中央差分すると

```
do i = 1, nx
  dudx(i) = ( u(i+1) - u(i-1) ) / dx
end do
```

# なぜ読めない？ 重労働？

互い違い格子を使うと同じ添え字のものが違う場所に配置される



さっきと同様に  $\frac{\partial p}{\partial x}$ 、 $\frac{\partial u}{\partial x}$  は

$$dpdx(i) = ( p(i+1) - p(i) ) / dx$$

$$dudx(i) = ( u(i) - u(i-1) ) / dx$$

•端っこの添え字も違う

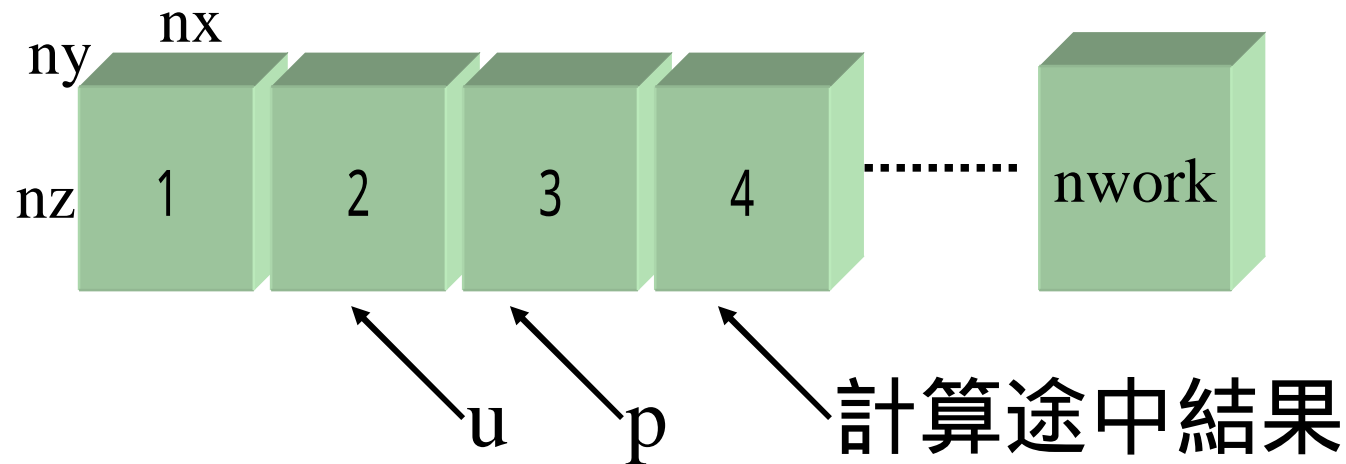
# gmsの特徴

- 配列添え字の追放
  - 構造体を定義する
    - 変数自身が自分の配置情報を持っている
  - 構造体に対して差分などの関数を定義
    - 差分をとられると配置場所がずれるといったことも考慮
- メモリー管理
  - fortran90にGC機能はない！！
    - リファレンスカウンタ方式によって実装

# 構造体とメモリー管理

```
type var_xyz
  integer :: id           !メモリーブロック上の「アドレス」
  integer, dimension(3) :: grid !配置情報
end type var_xyz
```

値そのものはユーザーが確保するメモリーブロックに格納



ほかにもvar\_x, var\_y, var\_z, var\_xy, var\_xz, var\_yzの計7つがある



# g m s 的書法

$\frac{\partial u}{\partial x}$  2次精度中央差分

$$dudx(i) = ( u(i) - u(i-1) ) / dx$$

24文字

$$dudx = d_x(u)$$

11文字  
( 46% )

# g m s 的書法

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) \quad \text{2次精度中央差分}$$

$$d2udx2(i) = ( u(i+1) - 2.0 * u(i) + u(i-1) ) / (2.0 * dx)$$

43文字

$$d2udx2 = d\_x(d\_x(u))$$

18文字  
( 42% )

**直感的で読める！**

タイプ数が少ない

•バグの混入が防げる 省力化・効率アップ

# gmsの落とし穴

構造体の演算を定義しなければいけない

- たとえば「+」の場合

- 異なる構造体同士の演算を許可している 49通り
- 構造体と実数の演算 14通り

計 63通

り！！

四則演算の定義だけで6000行！！！！

300行弱のRubyで自動生成

- メンテナンスが楽

# まとめ

fortran90を用いてgrid model構築ツールを開発

- 変数に配置情報を持たせるため、構造体を定義
- 互い違い格子を意識した差分・平均演算を定義

## 利点

- 配列添え字を追放することが可能
  - 可読性の飛躍的な向上！
- わずらわしい配置情報管理から開発者を解放

モデルの開発と改変が  
お手軽に行えるようになると期待される

# まとめ

## 弱点

- 動作が遅い
  - PCだと約5倍のCPU時間

# 今後の課題

- 高速化
  - 並列化？
- もっと難しいモデルでの実績
  - 現在は水なしの3次元モデルまで
  - 水ありは？複雑な境界条件では？
- マニュアルの整備



g m s

