シミュレーション物理8

磁性

今回の授業の目的

- 磁石が温度によって磁化をもったり，もたなかったりする様を計算機シミュレーションで調べる
- これは本当に数値実験。これを発展させて，脳のニューロンの発火具合などのシミュレーションも可能となる。

基本となる物理

熱平衡状態では自由エネルギー最小が実現している。
（等重率の原理から導くことが出来る。）

$$F = E - TS$$

内部エネルギーEを小さくするためには，ある特定の状態を選ぶ必要がある
→エントロピーが小さい
エントロピーの大きな状態→一般にない部エネルギーが大きい

自由エネルギーFを小さくするには，高温ではEを損してもSを大きくし，低温ではエントロピーとは関係なくEを小さくすればよい

相転移現象

- あるパラメータ（温度，圧力など）を変えていったとき，物理量が不連続に変化する現象
- 氷ー水，水ー水蒸気，強磁性ー常磁性，常伝導ー超伝導など
- ここではスピン系で記述される強磁性ー常磁性転移をシミュレーションする。これは非常に簡単なモデルなので，応用範囲も広い

スピン系

- スピン（磁気モーメント）をもったスピンが配置しているモデル
  - 簡単のため，スピンがいる格子点は規則的なものとする
  - ここでは2次元を扱う
- Hamiltonianはまずは単純に（イジング・モデル，Ising model）

$$H = -J \sum_{<i,j>} S_i S_j \qquad (S_i = \pm 1)$$

その他のスピン系のモデル

- ハイゼンベルク･モデル (Heisenberg model)

$$H = -J\sum_{<i,j>}\vec{S}_i\vec{S}_j \qquad (\vec{S}_i = (S_x, S_y, S_z), S_x{}^2 + S_y{}^2 + S_z{}^2 = 1)$$

- *XY*モデル

$$H = -J\sum_{<i,j>}\vec{S}_i\vec{S}_j \qquad (\vec{S}_i = (S_x, S_y), S_x{}^2 + S_y{}^2 = 1), \text{ or } H = -J\sum_{<i,j>}\cos(\theta_i - \theta_j)$$

- *n*-ベクトルモデル：これらを一般の成分にしたもの，*n*=1がイジング，2が*XY*，3がハイゼンベルク･モデル。

このハミルトニアンをスケール

統計力学ではボルツマン因子exp(-*E*/*kT*)が重要。よって*kT*/|*J*|を無次元の温度として，エネルギーは|*J*|でスケール。

$$H = \pm \sum_{<i,j>} S_i S_j \qquad (S_i = \pm 1)$$

-はスピンがそろうとエネルギーが下がるので強磁性
+はスピンが互いに反対を向くとエネルギーが下がるので反強磁性
ここでは強磁性のみを扱う

熱平衡状態

ではどのようにして，ある温度での状態を求めればよいか?
スピンに運動方程式があるわけではない。スピンは熱浴からランダムな力を受けて，平衡状態に達している。

熱平衡では状態i,jの間に以下の関係が成立

$$P(i \leftarrow j)P(j) = P(j \leftarrow i)P(i)$$

$$\therefore \frac{P(j)}{P(i)} = \frac{P(j \leftarrow i)}{P(i \leftarrow j)}$$

これが実現するように系を決めてやればよい

メトロポリス法

- 平衡状態では $\dfrac{P(j)}{P(i)} = \exp[-(E_j - E_i)/kT]$
- そこで

$$\frac{P(j \leftarrow i)}{P(i \leftarrow j)} = \exp[-(E_j - E_i)/kT]$$

- 一番簡単に

$$E_j - E_i > 0 \rightarrow$$

$$P(j \leftarrow i) = \exp[-(E_j - E_i)/kT], P(i \leftarrow j) = 1$$

プログラムの手順

- 1次元（統計力学の授業で解く），相転移を起こさないのでここではやらない
- 2次元スピンを考える $s(i,j)$, integer
- 始め$s(i,j)=1$に揃えておく
- 端から順にスピンを試しに反転させる
  - 反転してエネルギーが下がる→その反転を採用
  - 反転してエネルギーがδE上がる→その反転を確率$\exp(-\delta E/kT)$の確率で採用
- この手続きを延々と繰り返す
- 十分時間が経ったら$s(i,j)$の合計をとる。この合計の温度依存性を見る。

```
program ising
!------------------------
!  This is a program to simulate the Ising model
!  2005/6/10  Written by T. Ohtsuki
!------------------------
use KindNumbers
use randomnumber2
implicit none  ! Always begin with this statement
real(kind=double), parameter::zero=0.0_double,one=1.0_double
integer::i,lx,ly,ix,iy,isweep,nsweep,ixplus,ixminus,iyplus,iyminus
integer::dE
real(kind=double),dimension(5)::BoltzmannFactor
integer,allocatable::spin(:,:)
real(kind=double)::temperature,magnetization
integer::iseed,errorcode,isample,nsample

lx=10   ! X方向のサイズ
ly=10   ! Y方向のサイズ
nsweep=1000  !何回もスピンを試しに反転させたり戻したりする回数
nsample=50     !サンプル平均回数
open(1,file="magnetization.txt") !outputをこのファイルに
allocate(spin(lx,ly),stat=errorcode)  !サイズを割り当てる
if(errorcode/=0) print *,'Fail to allocate, status=',errorcode

iseed=2311  ! Initializing random number
call rndtsini(iseed)

TemperatureLoop:do temperature=1._double,3.5_double,0.1_double !温度を1-3.5まで，0. 1刻みで
```

```
magnetization=zero
sample: do isample=1,nsample   !サンプル平均

spin=1 !initial spins all up
Sweep:do isweep=1,nsweep
      do ix=1,lx
         do iy=1,ly
         ixminus=mod(lx+ix-2,lx)+1   !(ix,iy)の左側
         ixplus=mod(ix,lx)+1  !(ix,iy)の右側
         iyminus=mod(ly+iy-2,ly)+1 ! (ix,iy)の下側
         iyplus=mod(iy,ly)+1   !(ix,iy)の上側
        spin(ix,iy)=-spin(ix,iy)  ! Spinを試しに反転させる
        dE=-2*spin(ix,iy)*(spin(ixminus,iy)+spin(ixplus,iy)+&
                         spin(ix,iyminus)+spin(ix,iyplus)) ! 反転前後のエネルギー差
         if(exp(-dble(dE)/temperature).lt.drndts()) spin(ix,iy)=-spin(ix,iy)
         end do
      end do

   end do Sweep
   magnetization=magnetization+dble(sum(spin))/dble(lx*ly*nsample)
end do sample
write(1,'(2f14.7)') temperature,magnetization
end do TemperatureLoop

close(1)
deallocate(spin)
stop
end
```

実行のさせ方

- ソースファイルを作る。
- 実行ファイルを作る
  - f90 –o ising   ising.f90 random.o KindNumbers.o
  - isingとタイプして実行。(2,3分かかる)

課題（今週と来週）

- 温度と磁化の関係をプロット。温度がどの付近で磁化が有限になるか，調べる。
- プログラムに磁場を入れてみる。
- 磁化が有限の領域（強磁性），磁化が0（常磁性），ちょうどその境（転移点）において磁化の磁場依存性を調べる
- 反強磁性では磁化の磁場依存性は温度によってどう変わるか調べる
- 3次元にも拡張してみる