



素粒子の世界をあらわす方程式を解くことは簡単ではありません。そこで、コンピュータを利用して、素粒子の法則をさぐることが研究されています。ここでは、素粒子の間にはたらく強い力の例を見えます。

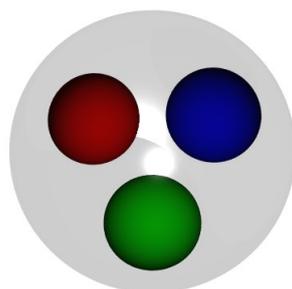
クォークとグルーオンの間の強い力

現在知られている素粒子のうち、クォークの仲間あいだには強い力が働きます。陽子、中性子や中間子などのハドロンと呼ばれる粒子は、いくつかのクォークがこの力で結合したものです。この力は大変強いために、ハドロンからクォークを単独で取り出すことはできません。そのために私たちはクォークを直接見る事ができないのです。

クォークの間のこの強い力のもとになっている(力を媒介している)粒子がグルーオンと呼ばれるものです。

クォーク：ハドロンを構成する素粒子

グルーオン：クォークの間の強い力を媒介する素粒子



クォークとグルーオンの方程式

上の描像をきちんとした形にあらわし、さらに実験や観測と比較するには、クォークとグルーオンの運動を支配する方程式を考える必要があります。

いろいろな理論的考察と実験の分析などから、その方程式は

$$S = \int d^4x \left\{ \frac{1}{4} (\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + g_0 f_{abc} A_\mu^b A_\nu^c) (\partial_\mu A_\nu^a - \partial_\nu A_\mu^a + g_0 f_{abc} A_\mu^b A_\nu^c) + \sum_i \bar{q}_i [\gamma_\mu (\partial_\mu + g_0 A_\mu^a T^a) + m_{i0}] q_i \right\} \quad (1)$$

であると考えられています。ここでこの方程式の細部を説明することはできませんが、クォークは q と \bar{q} 、グルーオンは A という記号で書かれた“場”というもので表されています。これは電場や磁場の考え方を拡張したものです。また m_{i0} はクォークの質量を、 g_0 は強い力の強さを表す数です。

さらに、上の方程式で表される運動を、量子力学の立場から考える必要があります。そのためには

$$\mathcal{Z} = \int dA_\mu^a(x) dq_i(x) d\bar{q}_i(x) e^{-S} \quad (2)$$

という積分を計算する必要があります。この積分が計算できれば、この理論に基づいてさまざまな量が予言でき、それを実験と比較することもできるのです。



モンテカルロ法による積分

積分の例として

$$I = \int_0^{\infty} dx e^{-x} \sin x$$

を考えてみます。この積分は（部分積分を使って）きちんと計算できて、答えは

$$I = 0.5$$

です。また、積分変数を x から y へ $x = -\log y$ と変数変換することで

$$I = \int_0^1 dy \sin(-\log y)$$

とあらわすこともできます。

このような積分をコンピュータで計算する一つの方法として、モンテカルロ法があります。モンテカルロというのはモナコ公国にある有名なとばく場の場所です。この方法では、0 から 1 の間に全くばらばらに分布する N 個の数（乱数といいます） y_1, y_2, \dots, y_N をコンピュータで作リ、その乱数での平均

$$I_N = \frac{1}{N} \sum_{i=1}^N \sin(-\log y_i)$$

として、上の積分のおおよその値（近似値）を求めます。コンピュータでさいころをふって計算するわけです。乱数の数 N を増やせば増やすほど正しい値に近づいていくはずで

プログラムはC++で書くと次のようになります。

```
#include <iostream>
#include <cstdlib>
#include <math.h>
using namespace std;

int main() {

    int N; cout << " N = ? "; cin >> N;

    double random, sum=0.0;
    for(int i=0; i<N; i++){
        random = (double) rand() / (double) RAND_MAX;
        sum = sum + sin(-log(random));
    }

    cout << " I_N = " << sum/N << endl;
    return 0;
}
```



結果は下のようになりました。

$N = 10$	$I_N = 0.364812$
$N = 20$	$I_N = 0.324211$
$N = 100$	$I_N = 0.443788$
$N = 1000$	$I_N = 0.508482$
$N = 10000$	$I_N = 0.499522$
$N = 100000$	$I_N = 0.499664$
$N = 1000000$	$I_N = 0.500262$

確かに、乱数の数 N を増やせば増やすほど本当の答え 0.5 に近づいていることがわかります。この例は、コンピュータを使わなくても正しい答えが計算できる例ですが、紙と鉛筆では計算できない複雑な積分を求めるのにもモンテカルロ法を使うことができます。

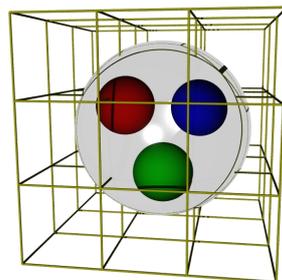


格子ゲージ理論

コンピュータで積分を計算する方法がわかりましたから、これをクォークとグルーオンの運動を調べるのに使ってみたいと思います。具体的には、(2) 式の積分を計算したいのです。

ここで問題になるのは、**実は、(2) 式の積分は無限大であるということです。**これを積分が発散するといいます。この発散の理由は、我々の住む**空間の点の数が無限大であるところ**にあります。そのために、(2) 式そのままでは意味のある計算ができないのです。

それで、考えだされたのが、格子ゲージ理論という方法です。これは、我々の住む**空間を格子目に切って、空間の点の数を有限にして考えます。**そして、クォークやグルーオン（をあらわす場）は、この格子目の点上だけにおかれているものとして考えます。



式であらわすと

$$Z = \int dU(x, \mu) dq_i(x) d\bar{q}_i(x) e^{-S}$$

$$S = \frac{1}{g_0^2} \sum_{x, \mu, \nu} \text{tr}\{1 - U(x, \mu)U(x + \hat{\mu}, \nu)U(x + \hat{\nu}, \mu)^{-1}U(x, \nu)^{-1}\} + \sum_{x, i} \bar{q}_i(x)(D + m_{i0})q_i(x)$$

となります。これで、コンピュータで計算できる積分の形になりました。



計算プログラム

実際にここで使っているコンピュータのプログラムの一部分は次のように書かれています。

```
forallsites(x){
  for(int mu=0;mu<4;mu++){

    mdp_matrix trial=mylattice.random(x).SU(Nc);

    // metropolis test
    if(exp(minus_delta_action(beta_d_Nc,trial,x,mu,u))
    >= mylattice.random(x).plain()){
      u(x,mu)=trial;}

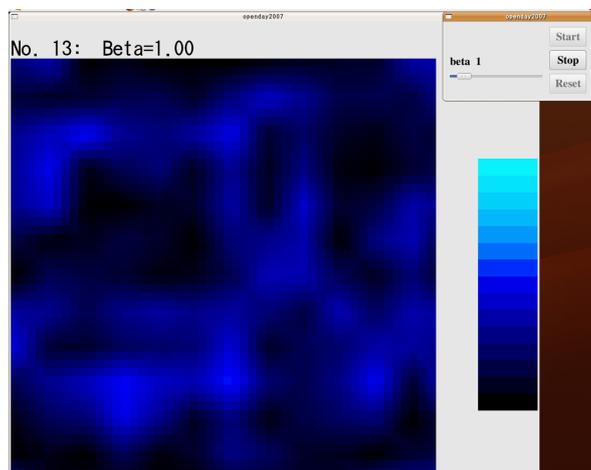
  }
}
```

基本的には、上で述べた数式をプログラムであらわし、モンテカルロ法（ここで使っているのはメトロポリス法と呼ばれる種類のものです）でランダムにグルーオンの場の値を作らせています。

スクリーンでのデモンストレーション

このようにしてコンピュータによって作ったグルーオンの場が持つ“エネルギー”の分布をスクリーンに映しています。時間1次元と空間3次元の4次元時空間を $12 \times 12 \times 12 \times 12 = 20736$ 個の格子目で近似して、そのうちのある平面内の様子を表示しています。また、ここでの計算ではクォークの効果は（計算量が莫大になるので）取り入れていません。

明るい色の部分が高いエネルギーの場所をあらわします。beta というのは $2/g_0^2$ のことです。beta の値をいろいろ変えて遊んでみて下さい。



このデモンストレーションは、温度0の真空中でのグルーオン場に対応するものです。これを見ると、真空が空っぽなものではなく、たえず細かい変化がおきていることがわかります。これは、量子力学において場が持つ量子ゆらぎと言われるものです。

原理的には、このようにコンピュータでつくった場の値を使うことで、ハドロンの質量などを計算することができます。それらを観測と比べることで、クォークとグルーオンを支配する方程式の正しさを検証することができます。