

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
use Math::Trig ;
use strict;
```

```
my ($coor,$schnum)=read_pdb($ARGV[0]);
my $coor=load_pdb($ARGV[0]);
my $mdir=$schnum;
my $sch=$schnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=subst($r,0,1); if ( $ggg ne $sch){ $schnum++; $sch=$ggg } ;
```

```
my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;
```

```
if ($qnum >0){
#system("mkdir -p $mdir");
my $filename=$ARGV[0];
$filename-- s/^.*\.\.//;
$filename-- s/\.\.//;
#$filename=$schnum.".".$qnum.".".$filename.".dat";
$filename="$mdir"."$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $schnum qnum $qnum\n";
```

```
foreach my $m (sort {$a=<=>$b} keys %coor){
my %qartets= %qwa; #find_quart( $coor{$m} );
my %q=find_q( $coor{$m} );
```

```
# foreach my $m (keys %qartets){
foreach my $q ( keys %qartets){
```

```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res (@{ $qartets{$q} ){
# print "$q $coor{$m}{$res}{\"N\"}->x,\"n\";
$nx=$nx+ $coor{$m}{$res}{\"N9\"}->x;
$ny=$ny+ $coor{$m}{$res}{\"N9\"}->y;
$nz=$nz+ $coor{$m}{$res}{\"N9\"}->z;
```

```
$ox=$ox+ $coor{$m}{$res}{\"O6\"}->x;
$oy=$oy+ $coor{$m}{$res}{\"O6\"}->y;
$oz=$oz+ $coor{$m}{$res}{\"O6\"}->z;
$r=$res;
}
```

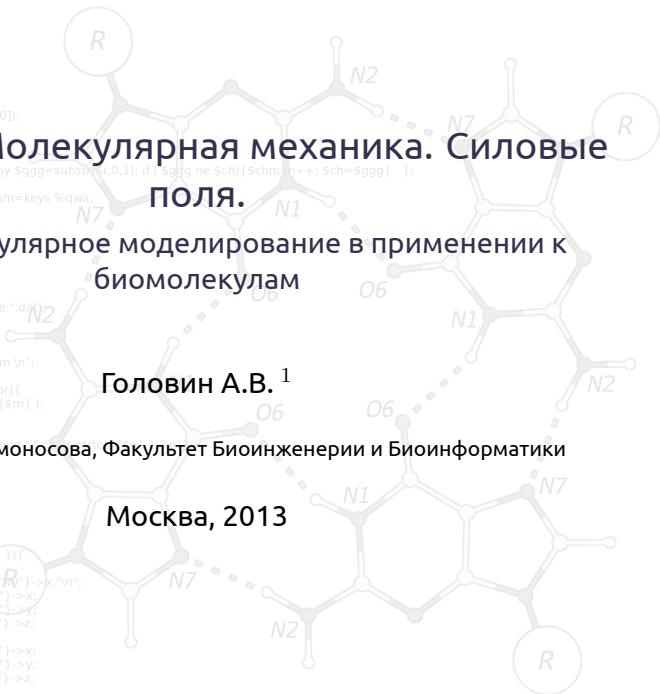
# Лекция 6. Молекулярная механика. Силовые поля.

## Курс: Молекулярное моделирование в применении к биомолекулам

Головин А.В. <sup>1</sup>

<sup>1</sup>МГУ им М.В. Ломоносова, Факультет Биотехнологии и Биоинформатики

Москва, 2013



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Содержание

```
#!/(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){ $chnum++; $ch=$ggg } ;
```

## Введение

```
if ($qnum > 0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename=~ s/^\.//;
$filename=~ s/\ /_/;
$filename="$dir"/.$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $chnum qnum $qnum\n";
```

## Ковалентные взаимодействия

```
foreach my $m (sort { $a<=>$b } keys %coor){
my $q=$m.$m.$coor{$m};
# foreach my $q ( keys %qartets){ print join " ",@{$qartets{$q}},"\n";
foreach my $q ( keys %qartets){
```

## Нековалентные взаимодействия

```
my $r;
foreach my $res (@{ $qartets{$q} }){
# print "$q $coor{$m} {$res} {"$r"}->x,\n";
$nx=$nx+ $coor{$m} {$res} {"$r"}->x;
$ny=$ny+ $coor{$m} {$res} {"$r"}->y;
$nz=$nz+ $coor{$m} {$res} {"$r"}->z;
$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;
```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Уравнение Шредингера

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %coor ) {
    my %qwa=find_quart( $coor{$r},0.1);
}

```

$$\left( -\frac{\hbar^2}{m} \left[ \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right] + V \right) \Psi(r, t) = i\hbar \frac{\partial \Psi(r, t)}{\partial t}$$

Или:

$$H\Psi = E\Psi; \quad H = \frac{-\hbar^2}{m} \nabla^2 - \frac{Ze^2}{4\pi\epsilon_0 r}$$

Современные базисы предполагают примерно 60 функций на атом. Итого: 900 функций на аминокислоту.

- Можно аппроксимировать электронную плотность уравнениями классической физики.

```
#!/usr/bin/perl  
use Math::VectorReal qw( :all );
```

# Молекулярная механика (ММ)

```
#!/my %coor,my $chnum)=read_pdb($ARGV[0]);
```

```
my %coor=read_pdb($ARGV[0]);
```

```
my $dir=$ARGV[1];
```

```
my $cni=$ARGV[2];
```

```
foreach my $r ( sort keys %{$coor{"0"} } ){ my $sggq=substr($r,1); if ($sggq ne $cni){ $cni=$sggq; $r=$sggq; } }
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

```
my %qwa;
```

- В ММ электронная структура атома замещается на достаточно простые уравнения с параметрами.
- Наборы параметров называются силовыми полями.
- Используется допущение Борна-Оппенгеймера (электроны быстро адаптируются к движению ядер)
- Расчёт энергии происходит на основе положения ядер.
- Упрощения позволяют работать с большими системами
- В некоторых случаях ММ подходы могут давать результаты, сравнимые по точности с методами QM.

# Простое уравнение силового поля (СП)

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

$$U = \sum_{\text{bonds}} \frac{k_i}{2} (l_i - l_0)^2 + \sum_{\text{angles}} \frac{k_i}{2} (\phi_i - \phi_0)^2 + \sum_{\text{torsions}} \frac{V_n}{2} (1 + \cos(n\omega - \gamma)) +$$

$$+ \sum_{i=1}^N \sum_{j=i+1}^N \left( 4\epsilon_{ij} \left[ \left( \frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{ij}}{r_{ij}} \right)^6 \right] + \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}} \right)$$

```
if ($qnum > 0){
#system("mkdir SARGV[1]");
my $filename=SARGV[0];
$filename="-- s/~/V//;
$filename="-- s/~/pdb//;
# $filename="$qnum."."$qnum";
$filename="$dir/$filename";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $qnum qnum $qnum\n";
```

```
foreach my $m (sort {$a-<=>$b} keys %coor){
my %qartets = %qwa; #find quart( $coor($m) );
my %q = find_q( $coor($m) );
```

```
# foreach my $q { keys %qartets } {
foreach my $q { keys %qartets){
```

Связи

Углы

Торсионные углы

Ван-дер-Ваальсовы взаимодействия

Кулоновские взаимодействия

```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;

foreach my $res (@{ $qartets[$m] }) {
print "$q $coor($m) ($res) -> x, y, z\n";
$nx=$nx+$coor($m){$res}{x};
$ny=$ny+$coor($m){$res}{y};
$nz=$nz+$coor($m){$res}{z};
$ox=$ox+$coor($m){$res}{"O6"}->x;
$oy=$oy+$coor($m){$res}{"O6"}->y;
$oz=$oz+$coor($m){$res}{"O6"}->z;
```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Пропан

```
#!/(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $ch ){ $ch=$ggg; $chnum=$ggg } ;
```

```
my %qwa=find_quart( %coor{"0"} ); my $qnum=keys %qwa;
```

```
if ($qnum > 0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/^.*\//";
$filename="-- s/\.pdb//";
#$filename=$chnum."_$qnum.".$filename.".dat";
$filename="$dir"/.$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $chnum qnum $qnum\n";
```

```
foreach my $m (sort { $a<=>$b } keys %coor){
my %qartets = %qwa; #find_quart( %coor{$m} );
my %q = find_q( %coor{$m} );
```

```
# foreach my $q ( keys %qartets ){ print join " ", @{$qartets{$q}} ;
```

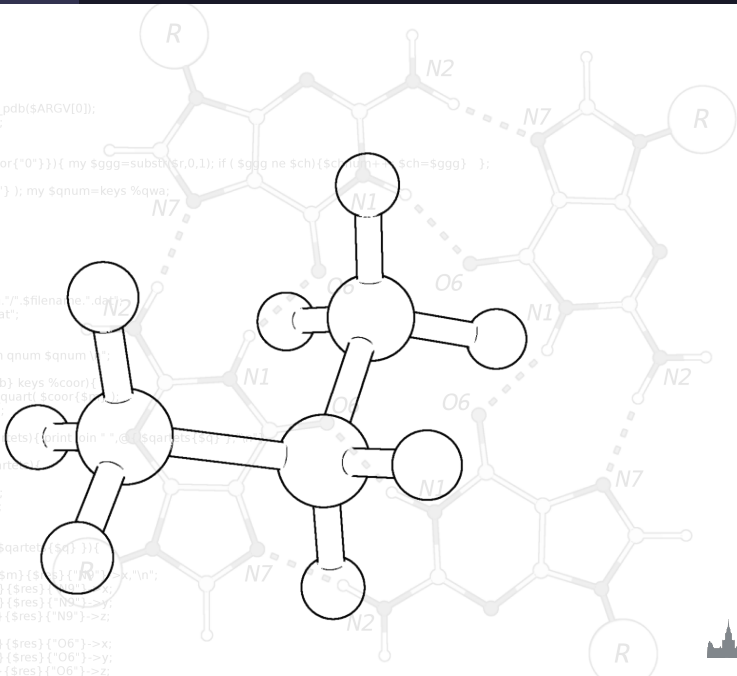
```
foreach my $q ( keys %qartets ){
```

```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res ( @{$qartets{$q}} ){
```

```
# print "$q %coor{$m} {"$res"} {"N"}->x,"n";
$nx=$nx+ %coor{$m} {$res} {"N"}->x;
$ny=$ny+ %coor{$m} {$res} {"N"}->y;
$nz=$nz+ %coor{$m} {$res} {"N"}->z;
```

```
$ox=$ox+ %coor{$m} {$res} {"O6"}->x;
$oy=$oy+ %coor{$m} {$res} {"O6"}->y;
$oz=$oz+ %coor{$m} {$res} {"O6"}->z;
```



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Основные особенности силовых полей

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"O"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){$chnum++; $ch=$ggg} };

```

```
my %qwa=find_quart( %coor{"O"} ); my $qnum=keys %qwa;
```

- Большинство параметров неотделимо от поля.

```
if ($qnum > 0){
```

```

#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];

```

- Параметризация ММ сильно зависит от целей исследования.

```

$filename=$ch.$chnum;
$filename=$dir.$filename;
$filename=$ch.$chnum.".".$qnum.".".$filename;
$filename=$dir.$filename;
print "file $filename\n";
open OUT ">$filename";
print OUT "#INFO: chain $chnum qnum $qnum ln";

```

- Большинство силовых полей параметризованы для воспроизведения структуры.

```

foreach my $m ( keys %coor ){
my %qartets= %qwa; #find_quart( %coor{$m} );
my %q= find_q( %coor{$m} );

```

- Силовые поля — это результат оптимизации параметров.

```

# foreach my $q ( keys %qartets ){
foreach my $q ( keys %qartets ){

```

- Силовые поля — это эмпирически найденные данные.

```

my $ox, my $oy, my $oz;
my $r;

```

```

foreach my $res ( @{$qartets{$q}} ){
# print "$q $coor{$m} {$res} {"$r"}->x,"n";
$nx=$nx+ $coor{$m} {$res} {"$r"}->x;
$ny=$ny+ $coor{$m} {$res} {"$r"}->y;
$nz=$nz+ $coor{$m} {$res} {"$r"}->z;

```

```

$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;

```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );

```

# Типы атомов в СП

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,3); if ( $ggg ne $ch ){ $chnum++; $ch=$ggg } };
my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;

```

```

if ($qnum >0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/^.*\//";
$filename="-- s/\.pdb//";
#$filename=$chnum.".".$qnum.".dat";
$filename="$dir".$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $chnum qnum $qnum\n";

```

```

foreach my $m (sort { $a-<=>$b } keys %coor){
my %qartets = %qwa ; #find_quart( $coor{$m} );
my %q = find_q( $coor{$m} );

```

```
# foreach my $q ( keys %qartets ){ print join( " ", $qartets{$q} ), "\n"; }
```

```
foreach my $q ( keys %qartets ){
```

```

my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;

```

```
foreach my $res ( @{$qartets{$q}} ){
```

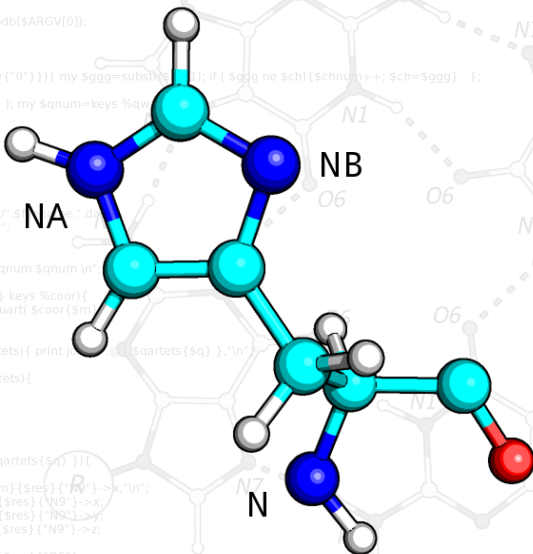
```
#
print "$q $coor{$m}{$res} {"$r"}->x,\n";
$nx=$nx+ $coor{$m}{$res}{"N9"}->x;
$ny=$ny+ $coor{$m}{$res}{"N9"}->y;
$nz=$nz+ $coor{$m}{$res}{"N9"}->z;

```

```

$ox=$ox+ $coor{$m}{$res}{"O6"}->x;
$oy=$oy+ $coor{$m}{$res}{"O6"}->y;
$oz=$oz+ $coor{$m}{$res}{"O6"}->z;

```





```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Потенциал для описания связи

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"O"}}){ my $ggg=substr($r,0,1); if ( $ggg eq "O" ){
my %qwa=find_quart( $coor{"O"} ); my $qnum=keys %qwa;
```

## Потенциал Морзе

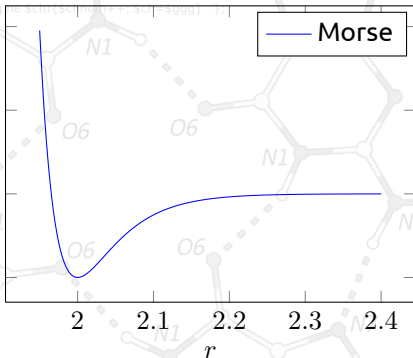
$$U(l) = D_e \{1 - e^{-a(l-l_0)}\}^2$$

## Более распространённый

$$U(l) = \frac{k_i}{2} (l_i - l_0)^2$$

```

foreach my $m (sort {$a<=>$b} keys %coor){
my %qartets = %qwa; #find_quart( $coor{$m} );
my %q = find_q( $coor{$m} );
# foreach my $q ( keys %qartets ){
foreach my $q ( keys %qartets ){
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
foreach my $res ( @{$qartets{$q}} ){
print "$q $coor{$m} {$res} {"N"}->x,"n";
$nx=$nx+ $coor{$m} {$res} {"N"}->x;
$ny=$ny+ $coor{$m} {$res} {"N"}->y;
$nz=$nz+ $coor{$m} {$res} {"N"}->z;
$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;
```



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Параметры при описании связи

```
my ($my $coor, my $schnum)=read_pdb($ARGV[0]);
my $my $coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
my $sch, my $schnum;
foreach my $r ( sort keys %{$coor{"0"}} ) { my $ggg=substr($r,0,1); if ( $ggg ne $sch ) { $schnum++; $sch=$ggg } ;
```

```
my $my $qwa=find_quart( $coor{"0"} ); my $sqnum=keys %qwa;
```

```
if ($sqnum > 0) {
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/^.*\//";
$filename="-- s/\.pdb//";
#$filename=$schnum.".".$sqnum.".dat";
$filename="$mdir/$filename.dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $schnum qnum
```

```
foreach my $m ( sort { $a <=> $b } keys %qwa ) {
my %qartets = %qwa; #find_quartets
my %q = find_q( $coor{$m} );
```

```
# foreach my $q { keys %qartets
```

```
foreach my $q { keys %qartets
```

```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res ( @ { $qartets
```

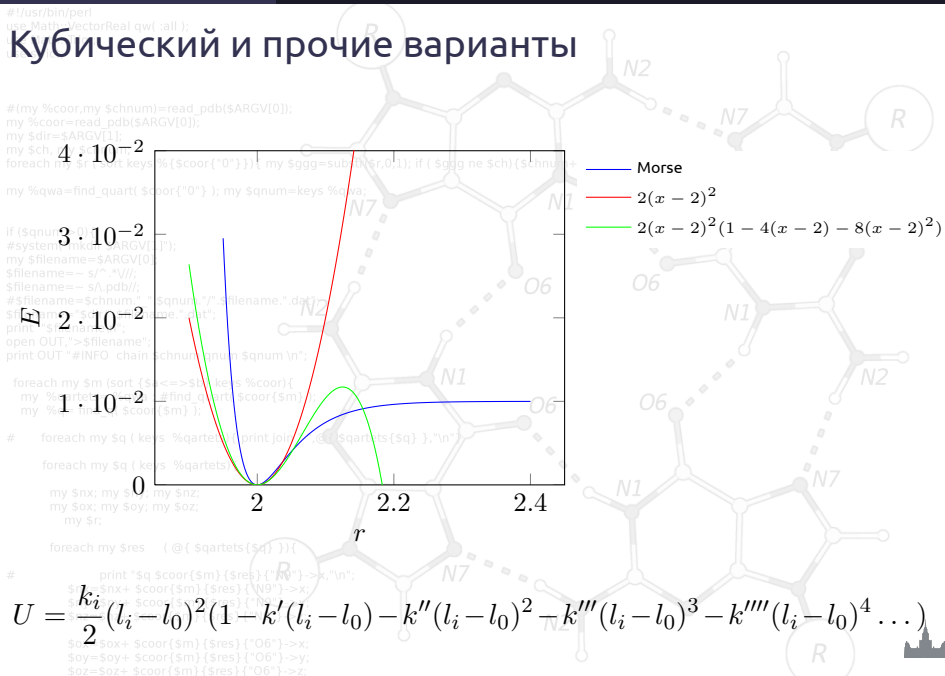
```
# print "$q $coor{$m} {$res} { "N9" }->x, "n";
$nx=$nx+ $coor{$m} {$res} {"N9"}->x;
$ny=$ny+ $coor{$m} {$res} {"N9"}->y;
$nz=$nz+ $coor{$m} {$res} {"N9"}->z;
```

```
$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;
```

$$E = \frac{k}{2} (r - r_0)^2$$

СВЯЗЬ	$r_0, \text{Å}$	$k, \text{kcal mol}^{-1} \text{Å}^{-2}$
$C_{sp^3}-C_{sp^3}$	1.523	317
$C_{sp^2}-C_{sp^2}$	1.337	690
$C_{sp^2}-O_{sp^2}$	1.208	777
$C_{sp^3}-N_{sp^3}$	1.438	367

## Кубический и прочие варианты



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Потенциал валентного угла

```
#!/(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $ch ){ $chnum++; $ch=$ggg } ;
```

```
my %qwa=find_quart( %coor{"0"} ); my $qnum=keys %qwa;
```

```
if ($qnum >0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename-- s/^.*\//;
$filename-- s/\.pdb//;
```

```
или $filename=$chnum." ".$qnum." ".$filename." .dat";
$dir="$dir"/$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO ch
```

```
foreach my $m ( sort { $a-<=>$b } keys %coor ){
my %qartets = %qwa; #find_quart( %coor{$m} );
my %q = find_q( %coor{$m} );
```

```
# foreach my $q ( keys %qartets ){
foreach my $q ( keys %qartets ){
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res ( @ { $qartets{$q} } ){
# print "$q %coor{$m} {$res} {"R"}->x,"n";
$nx=$nx+ %coor{$m} {$res} {"N9"}->x;
$ny=$ny+ %coor{$m} {$res} {"N9"}->y;
$nz=$nz+ %coor{$m} {$res} {"N9"}->z;
```

```
$ox=$ox+ %coor{$m} {$res} {"O6"}->x;
$oy=$oy+ %coor{$m} {$res} {"O6"}->y;
$oz=$oz+ %coor{$m} {$res} {"O6"}->z;
```

$$U(\phi) = \frac{k_i}{2} (\phi_i - \phi_0)^2$$

$$U(\phi) = \frac{k_i}{2} (\phi_i - \phi_0)^2 (1 - k'(\phi_i - \phi_0) - k''(\phi_i - \phi_0)^2 -$$

$$-k'''(\phi_i - \phi_0)^3 - k''''(\phi_i - \phi_0)^4 \dots)$$



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Потенциал торсионного угла

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $idir=$ARGV[1];
my $sch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=$coor{"0"}{$r}; if ( $ggg ne $sch ){ $chnum++; $sch=$ggg };
}
my %qwa=find_quart( $coor{"0"} ); my $sqnum=keys %qwa;

```

$$U(\omega) = \sum_{\text{torsions}} \frac{V_n}{2} (1 + \cos(n\omega - \gamma))$$

```

if ($sqnum > 0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];

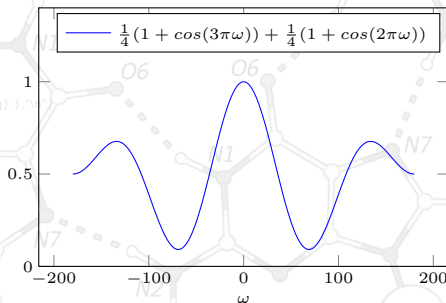
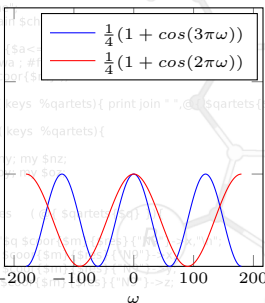
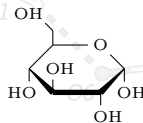
```

## Рассмотрим О-С-С-О (сахар в ДНК)

```

#$filename=$schnum." ".$sqnum." ".$filename.".d";
$filename="$idir"."$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO char sc

```



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Потенциал торсионного угла

```
my ($my $coor, my $schnum)=read_pdb($ARGV[0]);
my $my $coor=read_pdb($ARGV[0]);
my $my $dir=$ARGV[1];
my $my $sch, my $schnum;
foreach my $fr ( sort keys %{$coor{"0"}} ){ my $my $ggg=subst($r,0,1); if ( $my $ggg ne $sch ){ $my $schnum++; $my $sch=$ggg } ;
```

```
my $my $qwa=find_quart( $coor{"0"}, my $my $qnum=keys %$qwa;

$$U(\omega) = \frac{V_1}{2}(1 + \cos\omega) + \frac{V_2}{2}(1 + \cos 2\omega) + V_3 \frac{2}{3}(1 + \cos 3\omega) \dots$$

```

```
if ($qnum > 0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/^.*\./";
$filename="-- s/\.pdb//";
#$filename="$schnum.".$qnum.".dat";
$filename="$schnum.".$qnum.".dat";
print "filename=$filename";
open OUT, ">$filename";
print OUT "#INFO chain $schnum qnum $qnum\n";
```

- Для поля MM2 используют три члена.

- Поле OPLS использует ряды с 4-ю слагаемыми.

```
foreach my $fr ( keys %$qwa ){
my $my $qartets = %qwa; #find_quart( $coor{$m} );
my $my $q = find_q( $coor{$m} );
# foreach my $q { keys %$qartets } { print join " ", @{$qartets{$q}}, "\n";
```

```
foreach my $q { keys %$qartets } {

$$U(\omega) = \frac{1}{2} [F_1(1 + \cos\omega) + F_2(1 - \cos 2\omega) + F_3(1 + \cos 3\omega) + F_4(1 - \cos 4\omega)]$$

```

```
foreach my $res ( @{$qartets{$q}} ){
# print "$q $coor{$m} {$res} {"$res"}->x, "\n";
$nx=$nx+ $coor{$m} {$res} {"$res"}->x;
$ny=$ny+ $coor{$m} {$res} {"$res"}->y;
$nz=$nz+ $coor{$m} {$res} {"$res"}->z;

$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;
```

# "Неправильные" торсионные углы

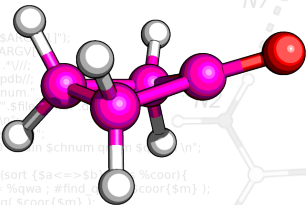
```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
use List::Util qw( min );
use List::MoreUtils qw( first );

my ($p,$q,$r,$s) = @ARGV;

my ($my $coor,$my $schnum)=read_pdb($ARGV[0]);
my $coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $sch,$my $schnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $sch ){ $schnum++; $sch=$ggg } ;
```

```
my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;
```

```
if ($qnum > 0){
#system("mkdir $dir/$schnum");
my $filename=$ARGV[0];
$filename="-- s/\\.\\/";
$filename="-- s/\\.\\/";
# $filename=$schnum;
$filename="$dir/$schnum";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO: $schnum $qnum\n";
```



```
foreach my $m ( sort { $a-<=>$b } keys %coor ){
my %qartets = %qwa; #find quartets
my %q = find_q( $coor{$m} );
```

```
# foreach my $q { keys %qartets } { print join " ", @qartets{$q} ,"\n";
```

```
foreach my $q { keys %qartets }
```

Для циклобутанона кислород должен находиться в одной плоскости с c1,c2,c3.

Используют потенциал, где перечисление не 1-2-3-4, а 1-4-2-3

```
# print "$q $coor{$m} ($res){ "N9"}->x, "\n";
$nx=$nx+ $coor{$m} {$res}{ "N9"}->x;
$ny=$ny+ $coor{$m} {$res}{ "N9"}->y;
$nz=$nz+ $coor{$m} {$res}{ "N9"}->z;

$ox=$ox+ $coor{$m} {$res}{ "O6"}->x;
$oy=$oy+ $coor{$m} {$res}{ "O6"}->y;
$oz=$oz+ $coor{$m} {$res}{ "O6"}->z;
```

$$U(\omega) = V_1(1 - \cos\omega)$$

$$U(\omega) = V_1(\omega - \omega_0)^2$$

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Кросс-составляющие в силовых полях

```
my ($my $coor, my $schnum)=read_pdb($ARGV[0]);
my $my $coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
my $sch, my $schnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $sch ){ $schnum++; $sch=$ggg } ;
```

Кросс-составляющие отражают зависимость состояния одной связи или угла от состояния соседней связи.

Существуют: stretch-strech, stretch-bend, stretch-torsion



```
# foreach my $q ( keys %qartets ){ print join " ", $qartets{$q}, "\n";
foreach my $q ( keys %qartets ){
```

```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res ( @{$ $qartets{$q} } ){
# print "$q $coor{$m} {$res} {"N7"}->x, "\n";
$nx=$nx+ $coor{$m} {$res} {"N9"}->x;
$ny=$ny+ $coor{$m} {$res} {"N9"}->y;
$nz=$nz+ $coor{$m} {$res} {"N9"}->z;
```

```
$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;
```

$$U(l_1, l_2) = \frac{K_{l_1 l_2}}{2} (l_1 - l_{1,0})^2 (l_2 - l_{2,0})^2$$

$$U(l_1, l_2, \phi) = \frac{K_{l_1 l_2 \phi}}{2} [(l_1 - l_{1,0})^2 + (l_2 - l_{2,0})^2] (\phi - \phi_0)$$



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Нековалентные взаимодействия

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"O"}}){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){ $chnum++; $ch=$ggg } };
my %qwa=find_quart( $coor{"O"} ); my $qnum=keys %qwa;

```

- Нековалентные взаимодействия являются определяющими в формировании структуры биополимеров.
- Так как эти взаимодействия реализуются через пространство, то часто они описываются как функции, обратно пропорциональные расстоянию между двумя атомами.

```

if ($qnum > 0){
#system("if [ -d $dir ]; then mv $filename $dir; else mv $filename $dir/$filename; fi");
$filename=$dir"/$chnum"/$qnum"/$filename";
# $filename=$dir"/$chnum"/$qnum"/$filename";
$filename="$dir"/$chnum"/$qnum"/$filename".dat";
print "$filename\n";
open OUT ">$filename";
print OUT "### O Chain Scoring Quantum Scoring\n";
foreach my $m ( keys %coor ){
my %qartets = %qwa; #find quart( $coor{$m} );
my %qartets = %qwa; #find quart( $coor{$m} );
# foreach my $q ( keys %qartets ){ print join " ", @qartets{$q} }, "\n";
foreach my $q ( keys %qartets ){
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
foreach my $res ( @qartets{$q} ){
# print "$q $coor{$m} {$res} {"$ch"}->x,\"n\";
$nx=$nx+ $coor{$m} {$res} {"$ch"}->x;
$ny=$ny+ $coor{$m} {$res} {"$ch"}->y;
$nz=$nz+ $coor{$m} {$res} {"$ch"}->z;
$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;

```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Электростатические взаимодействия

```
#!/my $coor,my $chnum)=read_pdb($ARGV[0]);
my $coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"O"}} ){ my $ggg=subst($r,0,1); if ( $ggg ne $ch ){ $chnum++; $ch=$ggg } ;
```

Допустим, что поверхность единичного потенциала можно представить зарядами в центрах атомов.

Тогда электростатические взаимодействия будут описываться по закону Кулона:

$$U(q_1, q_2) = \frac{q_1 q_2}{4\pi\epsilon_0\epsilon_r r_{ij}}$$

$$U = \sum_{i=1}^{N_A} \sum_{j=1}^{N_B} \frac{q_i q_j}{4\pi\epsilon_0\epsilon_r r_{ij}}$$

```
foreach my $m (sort {$a=<=>$b} keys %coor){
  my %qartets = %qwa; #find_quart($coor{$m});
  my %q = find_q($coor{$m});
```

```
# foreach my $q ( keys %qartets ){ print join " ", @qartets{$q}, "\n";
```

```
foreach my $q ( keys %qartets ){
```

```
  my $nx; my $ny; my $nz;
  my $ox; my $oy; my $oz;
  my $r;
```

```
  foreach my $res ( @{$qartets{$q}} ){
```

```
#    print "$q $coor{$m}{$res}{\"N\"}->x,\"n\";
```

```
    $nx=$nx+ $coor{$m}{$res}{\"N9\"}->x;
```

```
    $ny=$ny+ $coor{$m}{$res}{\"N9\"}->y;
```

```
    $nz=$nz+ $coor{$m}{$res}{\"N9\"}->z;
```

```
    $ox=$ox+ $coor{$m}{$res}{\"O6\"}->x;
```

```
    $oy=$oy+ $coor{$m}{$res}{\"O6\"}->y;
```

```
    $oz=$oz+ $coor{$m}{$res}{\"O6\"}->z;
```



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Двойное обрезание

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,3); if ( $ggg =~ $ch ){ $chnum++; $ch=$ggg } ;

```

```
my %qwa=find_quart( $coor{"0"} ); my $qnum=keys %qwa;
```

```

if ($qnum > 0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/^.*\//";
$filename="-- s/\.pdb//";
#system("cp $filename $dir");
print "$filename";
open OUT ">$dir/$filename";
print OUT "#INFO chain $chnum qnum $qnum\n";

```

```

foreach my $m (sort { $a-<=>$b } keys %coor){
my %qartets= %qwa; #find_quart( $coor{$m} );
my %q= find_q( $coor{$m} );

```

```
# foreach my $q { keys %qartets } { print join " ", @{$qartets{$q}}, "\n";
```

```
foreach my $q { keys %qartets } {
```

```

my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;

```

```
foreach my $res ( @{$qartets{$q}} ) {
```

```

# print "$q $coor{$m} {$res} {"$r"}->x,\n";
$nx=$nx+ $coor{$m} {$res} {"N9"}->x;
$ny=$ny+ $coor{$m} {$res} {"N9"}->y;
$nz=$nz+ $coor{$m} {$res} {"N9"}->z;

```

```

$ox=$ox+ $coor{$m} {$res} {"O6"}->x;
$oy=$oy+ $coor{$m} {$res} {"O6"}->y;
$oz=$oz+ $coor{$m} {$res} {"O6"}->z;

```

Для сферы A мы считаем все частичные заряды, а для сферы A-B мы будем считать взаимодействие групп зарядов с нашим атомом.

$$U_1 = \sum_{i=1}^{N_A} \frac{q_1 q_i}{4\pi\epsilon_0\epsilon_r r_{1i}} + \sum_{j=1}^{N_{group}} \frac{q_1 q_j}{4\pi\epsilon_0\epsilon_r r_{1j}}$$

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Потенциал реакционного поля

```
#!/(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"O"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $ch){ $chnum++; $ch=$ggg } ;
```

```
my %qwa=find_quart( $coor{"O"} ); my $qnum=keys %qwa;
```

**Основная идея: мы считаем, что за некоторым расстоянием плотность заряда одинаковая, и, следовательно, известна некая диэлектрическая проницаемость среды.**

```
if ($qnum > 0){
    $filename="-- s/" . $ch . "/";
    $filename=$ch . "dir/" . $qnum . "/" . $filename . ".dat";
    $filename="$dir" . $filename . ".dat";
    print "$filename\n";
    open OUT,">$filename";
    print OUT "#INFO chain $chnum qnum $qnum\n";
```

```
foreach my $m (sort { $a<=>$b } keys %coor){
    my %qartets= %qwa; #find quartets
    my %q= find_quart($m);
```

$$U_{ij} = \frac{q_1 q_i}{4\pi\epsilon_0\epsilon_r r_{i1}} \left[ 1 + \frac{\epsilon_{rf} - \epsilon_r}{2\epsilon_{rf} + \epsilon_r} \frac{r_{ij}^3}{r_c^3} \right] - \frac{q_1 q_i}{4\pi\epsilon_0\epsilon_r r_c} \frac{3\epsilon_{rf}}{2\epsilon_{rf} + \epsilon_r}$$

```
# foreach my $q ( keys %qartets){
```

```
    my $nx; my $ny; my $nz;
    my $ox; my $oy; my $oz;
    my $r;
```

```
    foreach my $res ( @{$qartets{$q}} ){
```

```
#         print "$q $coor{$m}{$res}{"R"}->x,"n";
```

```
        $nx=$nx+ $coor{$m}{$res}{"N9"}->x;
```

```
        $ny=$ny+ $coor{$m}{$res}{"N9"}->y;
```

```
        $nz=$nz+ $coor{$m}{$res}{"N9"}->z;
```

```
        $ox=$ox+ $coor{$m}{$res}{"O6"}->x;
```

```
        $oy=$oy+ $coor{$m}{$res}{"O6"}->y;
```

```
        $oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Суммирование Эвальда

```
#!/(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $ch ){ $chnum++; $ch=$ggg } ;
```

```
my %qwa=find_quart( %coor{"0"} ); my $qnum=keys %qwa;
```

**Основная идея: нам нужно учитывать не только заряды в ближайшем окружении, но и, как в кристалле, заряды, находящиеся в соседних ячейках.**

```
if ($qnum > 0){
    $filename="--s/~/VC/";
    # $filename=$chnum."_qnum/".$qnum."_";
    $filename="$dir".$filename.".dat";
    print "$filename\n";
    open OUT,">$filename";
    print OUT "#INFO chain $chnum qnum $qnum\n";
```

$$U_{ij} = \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} \sum_{z=1}^{N_z} \sum_{i=1}^{N_1} \sum_{j=1}^N N \frac{q_i q_j}{4\pi\epsilon_0 \epsilon_r r_{ij}}$$

```
foreach my $q ( keys %qartets ){ print join " ",@{$qartets{$q}} ,"\n";
```

**Это сходится, но очень медленно.**

```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res ( @{$qartets{$q}} ){
    # print "$q %coor{$m}{$res} {"$R"}->x,\n";
    $nx=$nx+ %coor{$m}{$res} {"$R"}->x;
    $ny=$ny+ %coor{$m}{$res} {"$R"}->y;
    $nz=$nz+ %coor{$m}{$res} {"$R"}->z;
```

```
$ox=$ox+ %coor{$m}{$res} {"$R"}->x;
$oy=$oy+ %coor{$m}{$res} {"$R"}->y;
$oz=$oz+ %coor{$m}{$res} {"$R"}->z;
```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Суммирование Эвальда

```
#!/my $coor,my $chnum)=read_pdb($ARGV[0]);
```

Эвальд предложил перевести этот ряд в сумму 2-ух быстро сходящихся рядов и константы.

$$U = U_{dir} + U_{rec} + U_0$$

$$U_{dir} = f/2 \sum_{i,j}^N \sum_{x=1}^{N_x} \sum_{y=1}^{N_y} \sum_{z=1}^{N_z} q_i q_j \frac{\text{erfc}(\beta r_{ij,n})}{r_{ij,n}}$$

$$U_{rec} = \frac{f}{2} \pi V \sum_{i,j}^N q_i q_j \sum_{m_x} \sum_{m_y} \sum_{m_z} \frac{\exp(-\pi m/\beta)^2 + 2\pi i m (r_i \cdot r_j)^2}{m}$$

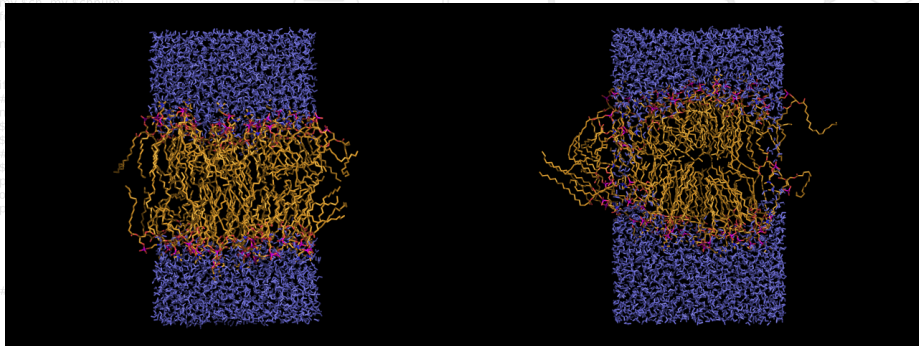
$$U_0 = \frac{f\beta}{\sqrt{\pi}} \sum_i^N q_i^2$$

Где бета - это параметр, определяющий соотношение прямого и обратного взаимодействий

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Суммирование Эвальда vs двойное обрезание

```
#{my %coor,my $chnum}=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
```



```
my $ox;my $oy;my $oz;
foreach my $res ( @({ $qartets($q) }) ) {
```

## Self-assembly with PME

```
# print "$q $coor{$m}{$res}{"N"}->x,"n";
$nx=$nx+ $coor{$m}{$res}{"N9"}->x;
$ny=$ny+ $coor{$m}{$res}{"N9"}->y;
$nz=$nz+ $coor{$m}{$res}{"N9"}->z;

$ox=$ox+ $coor{$m}{$res}{"O6"}->x;
$oy=$oy+ $coor{$m}{$res}{"O6"}->y;
$oz=$oz+ $coor{$m}{$res}{"O6"}->z;
```

## Self-assembly with Cut-off



# Ван-дер-Ваальсовы взаимодействия

- В основе природы Ван-дер-Ваальсовых взаимодействий лежат электронные эффекты: дисперсионные и обменные.
- В принципе, рассчитать такие эффекты можно в QM, но это далеко не тривиальная задача.
- В ММ нам надо считать такие взаимодействия быстро, на сегодняшний день наиболее часто используют потенциал Леонарда-Джонса:

$$U_{VdW} = \sum_{i=1}^N \sum_{j=i+1}^N 4\epsilon_{ij} \left[ \left( \frac{\sigma_{ij}}{r_{ij}} \right)^{12} - \left( \frac{\sigma_{ij}}{r_{ij}} \right)^6 \right]$$

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Ван-дер-Ваальсовы взаимодействия

```
my ($my $coor, my $schnum)=read_pdb($ARGV[0]);
my $my $coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
my $sch, my $schnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $sch ){ $schnum++; $sch=$ggg } ;
```

```
my $my $qwa=find_quart( $coor{"0"} ); my $sqnum=keys %qwa;
```

```
if ($sqnum > 0){
```

Наряду с потенциалом Леонарда-Джонса используют потенциал Букингама:

```
$filename=$schnum . $sqnum . ". $filename." . dat";
$filename="$mdir". $filename." . dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $schnum qnum $sqnum\n";
```

```
foreach my $m ( sort { $a-<=>$b } keys %coor ){
  my %qartets = %qwa; #find_quart( $coor{$m} );
  my %q = find_q( $coor{$m} );
```

```
# foreach my $q ( keys %qartets ){ print join " ", @{$qartets{$q}}, "\n";
```

```
foreach my $q ( keys %qartets ){
```

```
  my $nx; my $ny; my $nz;
  my $ox; my $oy; my $oz;
  my $r;
```

```
  foreach my $res ( @{$qartets{$q}} ){
```

```
# print "$q $coor{$m} {$res} { "N"}->x, "\n";
```

```
$nx=$nx+ $coor{$m} {$res} {"N"}->x;
```

```
$ny=$ny+ $coor{$m} {$res} {"N"}->y;
```

```
$nz=$nz+ $coor{$m} {$res} {"N"}->z;
```

```
$ox=$ox+ $coor{$m} {$res} {"O"}->x;
```

```
$oy=$oy+ $coor{$m} {$res} {"O"}->y;
```

```
$oz=$oz+ $coor{$m} {$res} {"O"}->z;
```

$$V_{bh}(r_{ij}) = A_{ij} \exp(-B_{ij} r_{ij}) - \frac{C_{ij}}{r_{ij}^6}$$

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Взаимодействия между разными типами атомов

```
my ($coor,$schnum)=read_pdb($ARGV[0]);
my $coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $sch, my $schnum;
foreach my $r ( sort keys %{$coor{"O"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $sch ){ $schnum++; $sch=$ggg };
```

Константы для разных типов атомов будут разные. Для их определения существуют правила смешивания:

```
my $filename=$ARGV[0];
$filename="-- s/^.*\//";
$filename="-- s/\.pdb//";
#$filename=$schnum.".".$qnum.".".$filename.".dat";
$filename="$dir"."$filename.".dat";
print "$filename\n";
open OUT,">$filename";
print OUT "#INFO chain $schnum qnum $qnum\n";
```

$$\sigma_{AB} = 1/2(\sigma_{AA} + \sigma_{BB})$$

$$\epsilon_{AB} = \sqrt{\epsilon_{AA}\epsilon_{BB}}$$

Это не единственный вариант правила смешивания, но такой подход наиболее распространён для моделирования биологических систем

```
foreach my $m ( sort { $a-<=>$b } keys %coor ){
  my %quartets = %qwa; #find quartet($coor{$m});
  foreach my $q ( keys %quartets ){ print join(" ", $quartets{$q});
  my $ox; my $oy; my $oz;
  my $r;

  foreach my $res ( @{$quartets{$q}} ){
    print "$q $coor{$m} {$res} {"$R"}->x,\n";
    $nx=$nx+ $coor{$m} {$res} {"N9"}->x;
    $ny=$ny+ $coor{$m} {$res} {"N9"}->y;
    $nz=$nz+ $coor{$m} {$res} {"N9"}->z;

    $ox=$ox+ $coor{$m} {$res} {"O6"}->x;
    $oy=$oy+ $coor{$m} {$res} {"O6"}->y;
    $oz=$oz+ $coor{$m} {$res} {"O6"}->z;
```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Различия для 1-4 взаимодействий

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $ch ){ $chnum++; $ch=$ggg } };
my %qwa=find_quart( %coor{"0"} ); my $qnum=keys %qwa;

```

- Так как 1-4 взаимодействия могут уже быть учтены в описании торсионного угла, то может быть, что силовых полях такие нековалентные взаимодействия не учитываются.

- В полях семейства AMBER, 1-4 VdW взаимодействия всё-таки учитываются, но их потенциал делится на 2.

```

# foreach my $q ( keys %qartets ){ print join " ",@{$qartets{$q}} , "\n";
foreach my $q ( keys %qartets ){
    my $nx; my $ny; my $nz;
    my $ox; my $oy; my $oz;
    my $r;
    foreach my $res ( @{$qartets{$q}} ){
        print "$q $coor{$m}{$res} {"$R"}->x,"n";
        $nx=$nx+ $coor{$m}{$res} {"$R"}->x;
        $ny=$ny+ $coor{$m}{$res} {"$R"}->y;
        $nz=$nz+ $coor{$m}{$res} {"$R"}->z;
        $ox=$ox+ $coor{$m}{$res} {"O6"}->x;
        $oy=$oy+ $coor{$m}{$res} {"O6"}->y;
        $oz=$oz+ $coor{$m}{$res} {"O6"}->z;
    }
}

```

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Водородные связи

```
my ($my $coor, my $snum)=read_pdb($ARGV[0]);
my $my $coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
my $sch, my $snum;
foreach my $r ( sort keys %{$coor{"0"}} ){ my $ggg=substr($r,0,1); if ( $ggg ne $sch ){ $snum++; $sch=$ggg } ;
```

```
my %qwa= read_quart( $coor{"0"} ); my $snum=keys %qwa;
```

```
if ($snum > 0){
```

```
  #system("
  my $filename=$ARGV[0];
```

```
  $filename=$ARGV[0];
```

```
  $filename=$ARGV[0];
```

```
  # $filename=$snum.".".$snum.".".$filename." dat";
```

```
  $filename=$mdir.$filename." dat";
```

```
  print "file
  open OUT, ">";
```

```
  print OUT "#!perl chain $sch num $snum ln";
```

```
  foreach my $m ( sort keys %{$coor{"0"}} ){
```

```
    my %qwa= read_quart( $coor{"$m"} );
```

```
    my %q= %{$qwa{"$m"} };
```

```
  # foreach my $q ( keys %qartets ){ print join " ", @{$qartets{$q}} , "\n";
```

```
  foreach my $q ( keys %qartets ){
```

```
    my $nx; my $ny; my $nz;
```

```
    my $ox; my $oy; my $oz;
```

```
    my $r;
```

```
    foreach my $res ( @{$qartets{$q}} ){
```

```
      # print "$q $coor{"$m"} {"$res"} {"N"}->x, "\n";
```

```
      $nx=$nx+ $coor{"$m"} {"$res"} {"N9"}->x;
```

```
      $ny=$ny+ $coor{"$m"} {"$res"} {"N9"}->y;
```

```
      $nz=$nz+ $coor{"$m"} {"$res"} {"N9"}->z;
```

```
      $ox=$ox+ $coor{"$m"} {"$res"} {"O6"}->x;
```

```
      $oy=$oy+ $coor{"$m"} {"$res"} {"O6"}->y;
```

```
      $oz=$oz+ $coor{"$m"} {"$res"} {"O6"}->z;
```

$$U_{HB} = \frac{A^{10}}{r} - \frac{C^{12}}{r}$$

- В силовых полях водородная связь часто описывается как комбинация Ван-дер-Ваальсовых и Кулоновских взаимодействий
- Существуют силовые поля, где водородная связь задаётся своим потенциалом на основе потенциала Леонарда-Джонса 10-12:



```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Водородные связи

```
#!/my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
```

Для точного описания водородной связи вносят поправки, учитывающие геометрию водородной связи:

```
if ($snum > 0){
#system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/~/./";
$filename="-- s/\./";
#$filename=$chnum.".".$snum.".".$filename.".dat";
$filename="$dir".$filename;
print "$filename\n";
open OUT ">".$filename;
print OUT "chain $chnum $snum $snum\n";
```

$$U_{HB} = \left( \frac{C}{d^6} - \frac{D}{d^4} \right) \cos^m \theta$$

```
foreach my $m (sort {$a<=>$b} keys %coor){
my %qartets = %qwa ; #find quart( $coor{$m} );
my %q = find_q( $coor{$m} );
```

```
# foreach my $q ( keys %qartets){ print join " ",@{$qartets{$q}},"n";
```

```
foreach my $q ( keys %qartets){
```

```
my $nx; my $ny; my $nz;
```

```
my $ox; my $oy; my $oz;
```

```
my $r;
```

$$U_{HB} = \left( \frac{A}{r_{H...Ac}^{10}} - \frac{C}{r_{H...Ac}^{12}} \right) \cos^2 \theta_{Don-H...Acc} \cos^4 \omega_{LP-Acc...H}$$

```
foreach my $res ( @{$qartets{$q}} ){
```

```
print "$q $coor{$m} {$res} { "N9"}->x,"n";
```

```
$nx=$nx+ $coor{$m} {$res} {"N9"}->x,"n";
```

```
$ny=$ny+ $coor{$m} {$res} {"N9"}->y,"n";
```

```
$nz=$nz+ $coor{$m} {$res} {"N9"}->z,"n";
```

```
$ox=$ox+ $coor{$m} {$res} {"O6"}->x,"n";
```

```
$oy=$oy+ $coor{$m} {$res} {"O6"}->y,"n";
```

```
$oz=$oz+ $coor{$m} {$res} {"O6"}->z,"n";
```