

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

```
#{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{$r}};
```

```
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
```

```
# 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 }{"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;
```

```
$r=$res;
```

```
}
```

# Структурная Биоинформатика

## Лекция 12. Макромолекулярные взаимодействия.

### Возможность предсказания.

Головин А.В. <sup>1</sup>

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

Москва, 2012

R

R

```
#!/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 } };

```

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

```

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

```

## Введение

## Макромолекулярный докинг

```

foreach my $m (sort { $a<=>$b } keys %coor){
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 }{"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 );
```

# Human "Interactome"

```

#(my %coor,my $chnum
my %coor=read_pdb($
my $dir=$ARGV[1];
my $ch, my $chnum;
foreach my $r ( sort key
```

```
my %qwa=find_quart( $
```

```

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

```

foreach my $m (sort {
my %qartets= %qwa
my %q= find_q( $coor
```

```
# foreach my $q { k
```

```
foreach my $q { k
```

```

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

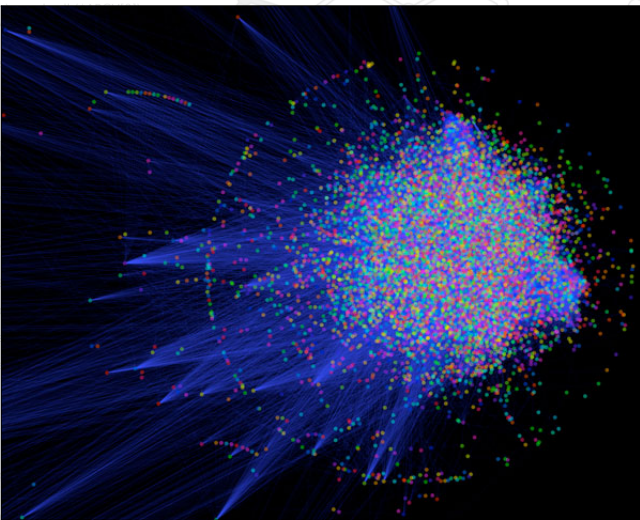
```
foreach my $res
```

```

# print "$
$nx=$nx+ $
$ny=$ny+ $
$nz=$nz+ $
```

```

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



# Способы предсказания белок-белковых взаимодействий

Взаимодействующие белки возможно ко-эволюционируют.

- Филогенетический профайлинг.  
Поиск пар белковых семейств среди широкого ряда видов.
- Появление и исчезновение пар семейств возможно указывает на взаимодействие.
- Предсказание на основе подобия филогенетических деревьев.
- Методы на основе классификации.
- Поиск гомологичных мест контакта.
- Ассоциативные методы. Это поиск характеристических последовательностей на основе профилей и мотивов.





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

# Поиск наименьшего $\Delta G$

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

- Суть метода основывается на поиске соответствия поверхностей для достижения максимальной поверхности контакта.

- После нахождения возможных конфигураций происходит ранжирование.

```
if ($qnum > 0) {
  #system("ls -l $dir/$schnum/$qnum");
  my $filename = "s/" . $qnum;
  $filename = "l/" . $qnum;
  # $filename = "schnum/" . $qnum;
  $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_quartl $coor
  my %q = find_ql $coor{$m};
```

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

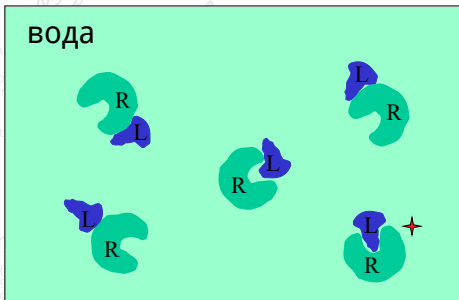
```
my $nx; my $ny; my $nz;
my $ox; my $oy; my $oz;
my $r;
```

```
foreach my $res (@{ $qartets{$q} }) {
  print "$q $coor{$m} {$res} {\n";
  $nx = $nx + $coor{$m} {$res} {"N1"}->x;
```

```
$ny = $ny + $coor{$m} {$res} {"N2"}->y;
  $nz = $nz + $coor{$m} {$res} {"N3"}->z;
```

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

вода



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

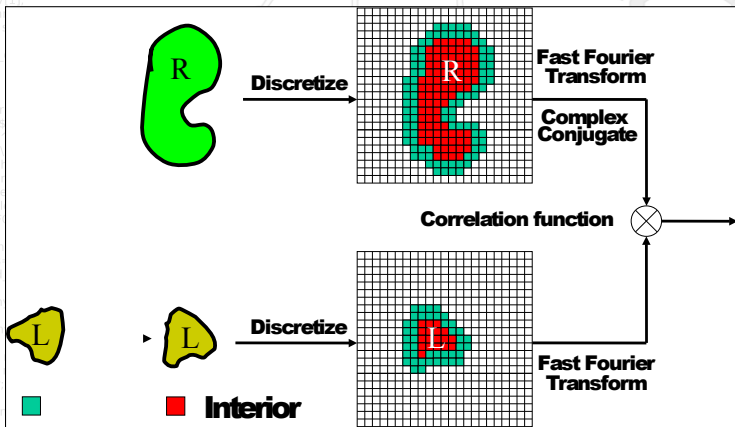
# Белковый докинг с использованием FFT

```
my (%coor,%schnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $sch, my $sch;
foreach my $r (
my %qwa=find_
```

```
if ($qnum > 0){
#system("mkdir
my $filename=
$filename="-- s/
$filename="-- s/
#$filename=$c
$filename="$di
print "$filenam
open OUT,">$fil
print OUT "#INF
```

```
foreach my $m
my %qartets;
my %q= find
# foreach m
foreach m
my $nx;
my $ox;
my $r;
foreach
```

```
#
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
```





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

# Оценка производительности

```
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 $sggg=substr($r,0,1); if ( $sggg ne $sch ){ $schnum++; $sch=$sggg } ;
```

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

```
if ($sqnum > 0){
#system("cat $mdir/$ARGV[0].pdb > $mdir/$ARGV[0].dat");
my $filename="--s/c/~/V/";
$filename=$filename.$schnum;
$filename=$mdir.$filename.$schnum;
print "$filename";
open OUT, ">$filename";
print OUT "#INFO: chain $schnum group $sqnum ln";
foreach my $m ( sort { $a->{ $coor{"$m"} }->{ $coor{"$m"} } } keys %$coor ){
my $qwa=$qwa; #find_quart( $coor{"$m"} );
my $q=$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"} {"N9"}->x, "\n";
$nx=$nx+ $coor{"$m"} {"$res"} {"N9"}->x;
$ny=$ny+ $coor{"$m"} {"$res"} {"N9"}->y;
$nz=$nz+ $coor{"$m"} {"$res"} {"N9"}->z;
```

```
# 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"} {"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;
```

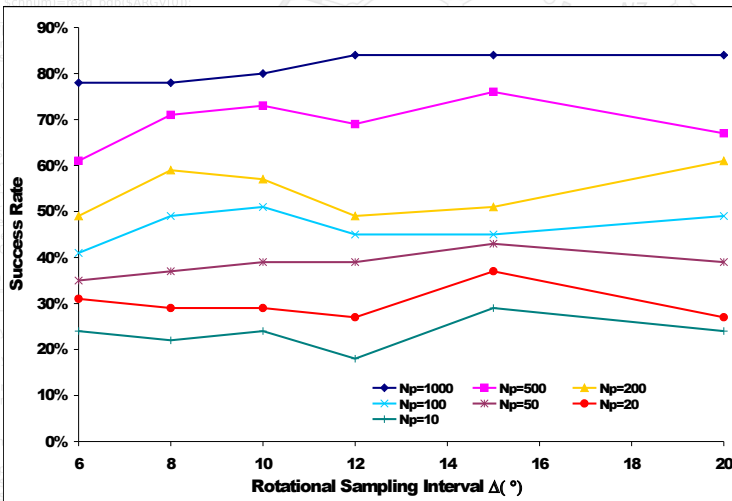
```
$oy=$oy+ $coor{"$m"} {"$res"} {"O6"}->y;
```

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

- **Success Rate:** для данного количества предсказаний ( $N_p$ ), это процент структур для которых был найден как минимум один удачный результат

- **Hit Count:** среднее количество хитов при данном значении  $N_p$ .

# Зависимость Success Rate от шага вращения



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

# Зависимость Hint count от шага вращения

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

```
my %coor=read_nrb($ARGV[0]);
my $dir=$ARGV[1];
my $sch, my $sch;
foreach my $r (1..$chnum){
    my %qwa=find_nrb($coor,$r);
```

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

```
foreach my $m (1..$qnum){
    my %qartets=find_nrb($coor,$m);
    my %q= %qwa; %qartets;
```

```
# foreach my $r (1..$chnum){
```

```
foreach my $m (1..$qnum){
```

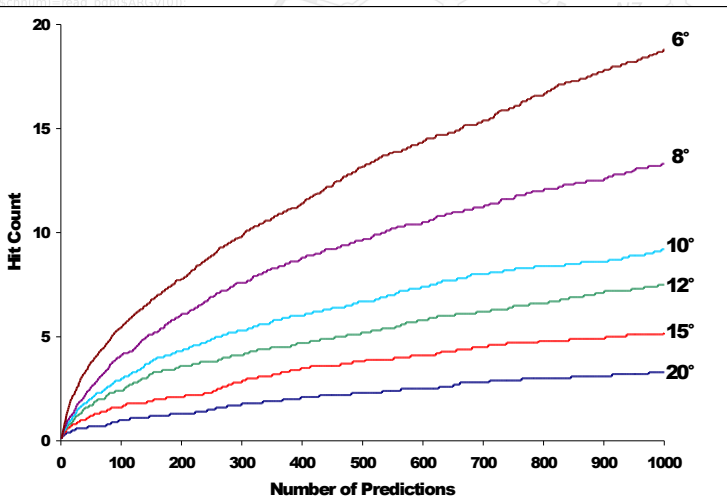
```
my $nx;
my $ox;
my $r;
```

```
foreach my $m (1..$qnum){
```

```
#
```

```
$nx=$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 );
```

# Решеточная комплементарность поверхности

```
#!/(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"},0)} my $gq($ch,$dir,0,1); if ($gq ne $ch){ $chnum++; $ch=$gq; } );
```

```
my %qwa=find_qwa($coor,$ch,$chnum);

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

foreach my $m (sort {$a-$b} keys %qwa){
my %qartets={};
my %q=find_q($coor,$ch,$chnum,$m);
# foreach my $q ( keys %q ){
foreach my $q ( keys %q ){
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;
```

	1	1	1	1	1	
1	9i	9i	9i	9i	9i	1
1	9i	9i	9i	9i	9i	1
1	9i	9i	9i	1	1	
1	9i	9i	9i	1	1	
1	9i	9i	9i	9i	9i	1
1	9i	9i	9i	9i	9i	1
1	1	1	1	1	1	

		1	1		
1	1	9i	9i	1	
1	1	9i	9i	1	
		1	1		

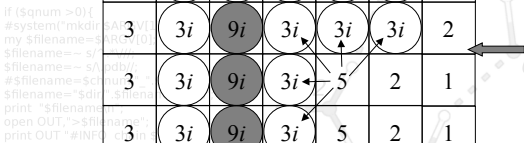
```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
```

# Парная комплементарность поверхности

```
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} ) {
    my ($sch, my $schnum) = ($r =~ /(\d+)(\d+)/ );
    ($schnum += $sch) ($schnum += $sch);
}
```

1	2	3	3	3	2	1
2	3i	3i	3i	3i	3i	2
3	3i	9i	3i	3i	3i	2
3	3i	9i	3i	5	2	1
3	3i	9i	3i	5	2	1
3	3i	9i	3i	3i	3i	2
2	3i	3i	3i	3i	3i	2
1	2	3	3	3	2	1

		1+3i	1+3i		
1+3i	1+3i	1+9i	1+9i	1+3i	
1+3i	1+3i	1+9i	1+9i	1+3i	
		1+3i	1+3i		



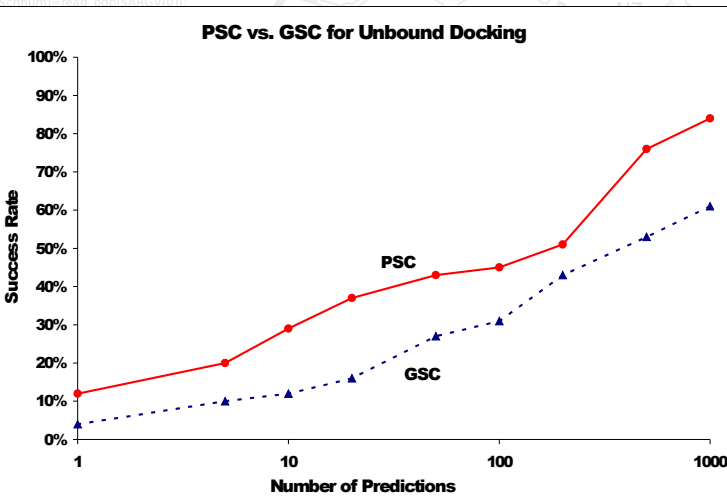
```
foreach my $res ( @{$schnum{$sch}} ) {
    print "$r $coor{$sch}{$res}";
    $nx = $nx + $coor{$sch}{$res}{"N1"}->x;
    $ny = $ny + $coor{$sch}{$res}{"N2"}->y;
    $nz = $nz + $coor{$sch}{$res}{"N9"}->z;
    $ox = $ox + $coor{$sch}{$res}{"O6"}->x;
    $oy = $oy + $coor{$sch}{$res}{"O6"}->y;
    $oz = $oz + $coor{$sch}{$res}{"O6"}->z;
}
```

**R<sub>PSC</sub>**

**L<sub>PSC</sub>**



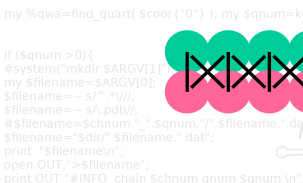
## PSC vs. GSC и Success Rate



# Почему так?

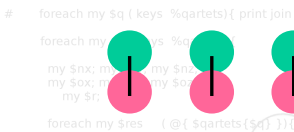
```
#!/usr/bin/perl
use Math::VectorReal qw( :all );

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $ch, my $chn;
```



```
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";
```



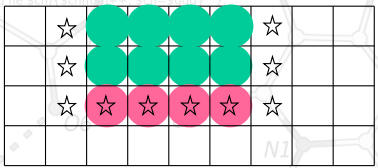
```
foreach my $m (sort { $a<=>$b } keys %coor){
    my %qartets = %qwa; #find_quart( %coor{$m} );
    my %q = find_q( %coor{$m} );

    # foreach my $q ( keys %qartets ){ print join " ", @{$qartets{$q}}, "\n";
    foreach my $q ( keys %qartets ){
        my $nx; my $ny; my $nz;
        my $ox; my $oy; my $oz;
        my $r;

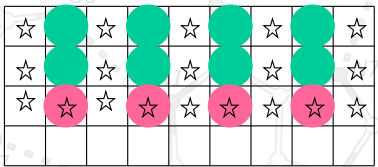
        foreach my $res ( @{$qartets{$q}} ){
            print "$q %coor{$m}{ $res } { \"$R\" }->x, \"\n\";
            $nx=$nx+ %coor{$m}{ $res } { \"$R\" }->x;
            $ny=$ny+ %coor{$m}{ $res } { \"$R\" }->y;
            $nz=$nz+ %coor{$m}{ $res } { \"$R\" }->z;

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

C



D







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

# Влияние на Success rate

```
my %coor,my %Schnum; read_nrb($ARGV[0]);
```

```
my %coor=read
my $dir=$ARGV
my $ch, my $ch
foreach my $r (
```

```
my %qwa=find_
```

```
if ($qnum > 0){
#system("mkdir
my $filename=$
$filename=- s/
$filename=- s/
# $filename=$c
$filename="$dir
print "$filename
open OUT,">$fil
print OUT "#INF
```

```
foreach my $m
my %qartets;
my %q= find
```

```
# foreach m
```

```
foreach m
```

```
my $nx;
my $ox;
my $r;
```

```
foreach r
```

```
#
```

```
$nx=
```

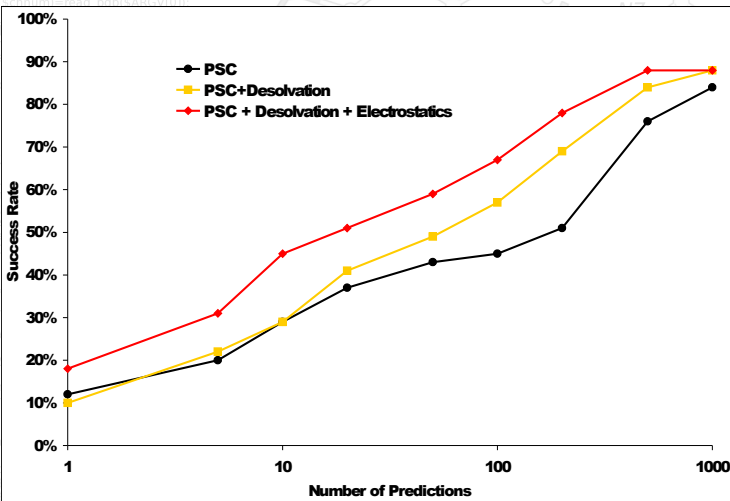
```
$ny=
```

```
$nz=$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 );
```

# Влияние на Hit Count

```
my ($coor,$schnum)=read_nhb($ARGV[0]);
```

```
my $coor=read_nhb($ARGV[0]);
my $dir=$ARGV[1];
my $sch,$schnum=$ARGV[2];
foreach my $r ($schnum) {
    my $qwa=find_nhb($coor,$sch,$r);
```

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

```
foreach my $m ($schnum) {
    my %qartets;
    my %q= find_nhb($coor,$sch,$m);
```

```
# foreach my $m ($schnum) {
```

```
foreach my $m ($schnum) {
```

```
my $nx;
my $ox;
my $r;
```

```
foreach my $m ($schnum) {
```

```
#
```

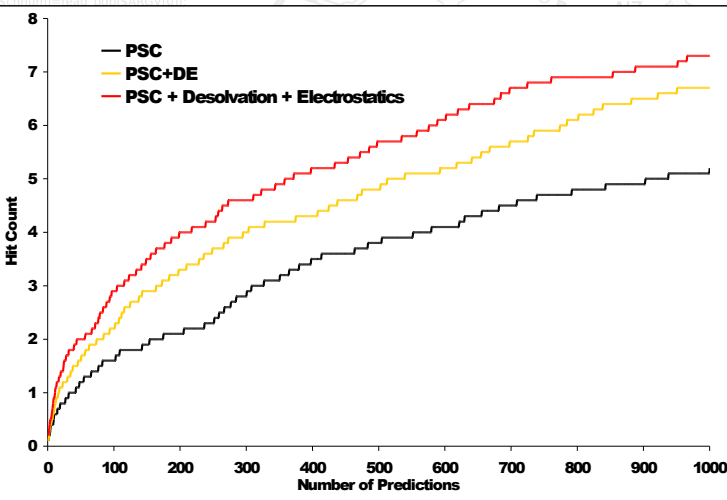
```
$nx=$nx+$qartets{$m};
```

```
$ny=$ny+$qartets{$m};
```

```
$ox=$ox+$qartets{$m};
```

```
$oy=$oy+$qartets{$m};
```

```
$oz=$oz+$qartets{$m};
```



# Вопросы?

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

#(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 $ARGV[1]");
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";

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;
```