

```
#!/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 $mdir=$ARGV[1];
my $sch, my $chnum;
foreach my $r ($sch) {
```

Лекция 5. Введение в квантовую химию

Курс: Молекулярное моделирование в применении к биомолекулам

Головин А.В. ¹

¹МГУ им М.В. Ломоносова, Факультет Биоинженерии и Биоинформатики

Москва, 2013

```
my %qwa=find_quart( $coor{"O"} ); 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";
$filename="$mdir".$filename.".dat";
print "$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 }{"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;
```

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

```
}
```

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

Содержание

```
my ($my $coor, my $schnum)=read_pdb($ARGV[0]);
my $my $coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
```

Базисы

```
my $my $im;
my $my $keys %{$coor{"0"}}; my $my $ggg=substr($r,0,1); if ( $my $ggg ne $sch){ $my $schnum++; $my $my $ggg };
```

Корреляционные методы

```
my $my $filename=$ARGV[0];
$my $filename=~ s/^\.//;
$my $filename=~ s/\./_/;
$my $my $filename="$my $schnum." $my $my $filename ".dat";
```

Метод самосогласованного поля

```
print "filename: ", $my $filename;
open OUT, ">$my $filename";
print OUT "#INFO chain $my $schnum qnum $my $qnum\n";
```

Теория возмущений Меллера-Плессета

```
my $my $q= find_q( $coor{$my $m}, );
```

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

Теория функционала плотности, DFT

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

DFT: Функционалы

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

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

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

Напоминание:

```

#(my %coor,my $chnum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $dir=".";
my $q="n";
foreach my $c (sort keys %$coor{"O"}){
    my %qwa=(sort keys %$coor{"O"});
    my %qwa=(sort keys %$coor{"O"});
    my %qwa=(sort keys %$coor{"O"});
}

```

- **Базисный набор это:** набор математических функций используемых для описания электронных орбиталей атомов в молекуле.

```

if ($qnum > 0){
    #system("rm -f $ARGV[0].*");
    my $filename=$ARGV[0].".dat";
    $filename=$ARGV[0].".dat";
    $filename=$ARGV[0].".dat";
    print "filename=$filename\n";
    open OUT ">$filename";
    print OUT "#INFO chain $chnum qnum $qnum\n";
}

```

- **Ограниченный базис:** базисные функции в которых **не происходит** изменение параметров в функции в ходе расчёта молекулярных орбиталей.

```

foreach my $c (sort keys %$coor{"O"}){
    my %qartets= %qwa; #find quart $coor{$m};
    my %qartets= %qwa; #find quart $coor{$m};
    my %qartets= %qwa; #find quart $coor{$m};
}
# foreach my $c (sort keys %$coor{"O"}){
#     foreach my $q ( keys %qartets){
#         my $sr;
#         foreach my $res ( @{$qartets{$q}} ){
#             print "$q $coor{$m} {$res} {"$q"}->x,\n";
#             $nx=$nx+ $coor{$m} {$res} {"$q"}->x;
#             $ny=$ny+ $coor{$m} {$res} {"$q"}->y;
#             $nz=$nz+ $coor{$m} {$res} {"$q"}->z;
#             $ox=$ox+ $coor{$m} {$res} {"O6"}->x;
#             $oy=$oy+ $coor{$m} {$res} {"O6"}->y;
#             $oz=$oz+ $coor{$m} {$res} {"O6"}->z;
#         }
#     }
# }

```

- **Неограниченный базис:** базисные функции в которых **происходит** изменение параметров в функции в ходе расчёта молекулярных орбиталей.

Часто такими математическими функциями является гауссиан:

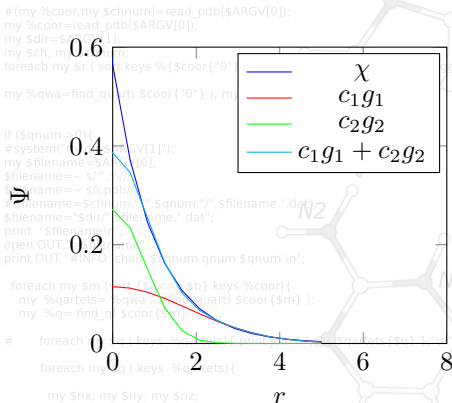
```

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

```

$$\psi = de^{-\alpha r^2}$$

Пример STO-2G для H:



$$\psi = \sum_1^2 ce^{-\alpha r^2}$$

$$\psi = c_1g_1 + c_2g_2$$

	α	c
1	0.151623	0.678914
2	0.851819	0.430129

$\epsilon = -0.4665819$ a.u. = 12.697 eV.

В реальности 13.606 eV. И ошибка 87.7 кДж/моль

Гауссианы:

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

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

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

$$1s = Ne^{ar^2}; \quad 2p_x = Ne^{ar^2}x; \quad 2p_y = Ne^{ar^2}y; \quad 2p_z = Ne^{ar^2}z;$$

```
foreach my $m ( keys %$qartets ){
my %$qartets=$qartets{$m};
my %$q=find_q( $coor{$m} );
# 3dxx = Near2x2; 3dxy = Near2xy; 3dxz = Near2xz;
```

$$3d_{xx} = Ne^{ar^2}x^2; \quad 3d_{xy} = Ne^{ar^2}xy; \quad 3d_{xz} = Ne^{ar^2}xz;$$

```
3dyy = Near2y2; 3dyz = Near2yz; 3dzz = Near2z2;
```

$$3d_{yy} = Ne^{ar^2}y^2; \quad 3d_{yz} = Ne^{ar^2}yz; \quad 3d_{zz} = Ne^{ar^2}z^2;$$

```
# 4fxxx = Near2x3; 4fxyx = Near2x2y; 4fxxz = Near2x2z;
```

$$4f_{xxx} = Ne^{ar^2}x^3; \quad 4f_{xyx} = Ne^{ar^2}x^2y; \quad 4f_{xxz} = Ne^{ar^2}x^2z;$$

И так далее.

```
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;
```

Описания базисных наборов для программы GAUSSIAN:

Общий вид обозначений от Поупл и коллег: **M-ijk..G**

- **M** – количество ограниченных гауссианов на один не валентный электрон
- Наличие двух и более букв после “-” означает, что валентные электроны описываются 2 и более функциями, каждая из которых состоит из линейной комбинации **ijk** гауссианов
- * - Означает, что для тяжёлых атомов используются не только гауссианы характерные для конкретной орбитали, но и гауссианы следующей орбитали.

Например для углерода в **3-21*G**: у валентных электронов с 3 гауссианами прибавляется 6 гауссианов для d-орбиталей.

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

Описания базисных наборов для программы GAUSSIAN:

```
my ($coor,$schnum)=read_pdb($ARGV[0]);
my $coor=read_pdb($ARGV[0]);
my $dir=$ARGV[1];
my $sch,$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;
```

- ****** то же самое, что и *****, но добавляются 3 гауссиана для р-орбиталей к гауссианам N и Ne.

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

- **+** Означает добавление дополнительных гауссианов тех же орбиталей, но с маленьким значением α . Этот шаг нужен для точного счёта систем где значительная электронная плотность удалена от ядра: электронные пары, анионы.

```
foreach my $r ( sort keys %{$coor{"0"}} ){
# print "$r $coor{$r}";
my $nx=$coor{$r}{"x"};
my $ny=$coor{$r}{"y"};
my $nz=$coor{$r}{"z"};
my $sx=$coor{$r}{"x"};
my $sy=$coor{$r}{"y"};
my $sz=$coor{$r}{"z"};
my $r;
foreach my $res ( @{$qartets{$r}} ){
# print "$r $coor{$r} {"$res"} {"$r"}->x,"n";
$nx=$nx+ $coor{$r} {"$res"} {"$r"}->x;
$ny=$ny+ $coor{$r} {"$res"} {"$r"}->y;
$nz=$nz+ $coor{$r} {"$res"} {"$r"}->z;
$ox=$ox+ $coor{$r} {"$res"} {"06"}->x;
$oy=$oy+ $coor{$r} {"$res"} {"06"}->y;
$oz=$oz+ $coor{$r} {"$res"} {"06"}->z;
```

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

Итак:

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

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

```
my $d
my $c
foreach
```

```
my %c
```

```
if ($q
```

```
#syste
```

```
my $fi
```

```
$filen
```

```
$filen
```

```
#$fili
```

```
$filen
```

```
print
```

```
open C
```

```
print C
```

```
forea
```

```
my
```

```
my
```

```
#
```

```
#
```

Запись Базиса	Количество гауссианов на орбиталь, для C	Применение
STO-3G	3	Большие системы
6-31G	6 - не валентные 3+1 - валентные	Системы без поляризации
6-31*G	То же самое + 6 функций типа l+1	Системы с анизотропией заряда
6-31**G	см. выше + 3 p-функции для H и He	Водородная связь
6-31+G(2df)	Поляризационные: 2*6 d-функций + 7 f- функций (см *) Диффузных : 4	Нужно там, где важно точно рассчитать высокую плотность электронов
6-311++ G(3df,3pd)	Диффузные на все атомы и поляризационные на C: 3*6 d типа + 7 f типа H: 3*3 p типа + 1 d типа	Если всё, что было до этого было не достаточно точным.

```
$ny=$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 );
```

Как не надо:

```

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

```

• 3-21++G(2df,2pd)

Только две функции на валентную орбиталь и 3 поляризационных набора функции с кучей гауссианов и диффузные наборы.

```

foreach my $m ($m=substr($s,$coor{"$m"});
my %q=keys %qwa;
my %q=keys %qwa;

```

• 6-311+G(2df)

3 набора на валентную орбиталь, 3 поляризационных + диффузный набор для тяжёлых атомом и отсутствие поляризации для водорода?

```

#
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 );
```

Недостаток подхода Фока:

```
my ($SARGV,$chain,$qnum)=read_pdb($ARGV[0]);
```

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

Метод Фока работает для систем близких к минимуму энергии. И не работает для возбуждённых систем, систем где наблюдается переход из одного стабильного состояния в другое стабильное и т.д.

$$\Psi(1, 2 \dots N) = \sum_{k_1} \sum_{k_2} \dots \sum_{k_n} c(k_1 \dots k_N) \psi_{k_1}(1) \psi_{k_2}(2) \dots \psi_{k_N}(N)$$

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

```
my %qartets = %qwa; #find quart( $ccoor{$m} );
```

```
my %q = find_q( $ccoor{$m} );
```

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

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

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

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

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

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

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

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

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

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

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

```
};
```

```
};
```

```
};
```

```
};
```

```
};
```

```
};
```

```
};
```

```
};
```

```
};
```

Это переводится в бесконечный ряд и в случае Фока мы учитываем только один член.

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

Недостаток подхода Фока:

```

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

```

Повысить точность счёта мы можем удачно учтя некоторые компоненты ряда. Значит надо минимизировать:

$$E = E_{real} - E_{HF}$$

Существуют три основных метода:

- Конфигурационного взаимодействия
- Метод самосогласованного поля
- Теория возмущение Меллера-Плессе

```

foreach my $m (sort { $a<=>$b } keys %coor){
  my %qartets = %qwa; #find quartl %coor{$m} );
  my %sq=kind of %coor{$m} );

```

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

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

```
my $m;
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 );
```

Метод конфигурационного взаимодействия

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

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

Мы нашли M спиновых орбиталей и N из них заняты электронами, именно их мы и используем для расчётов и подменим некоторые орбитали в найденной матрице новыми орбитальями. Т. е. создадим возбуждённую молекулу.

```
if ($sqm == 0) {
    $filename="-- s/^.*V//";
    $filename="-- s/\.pdb//";
    my $filename="$chnum.".$sqm.".dat";
    $filename="$dir".$filename.".dat";
    print "$filename\n";
    open OUT, ">$filename";
    print OUT "#INFO chain $chnum qnum $sqm\n";
}
```

$$\Psi_{CI} = \sum_{K=0}^{\infty} C_K \Psi_K$$

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

Применяем подход Рутхана-Хола и находим коэффициенты C . Этот подход и является методом конфигурационного взаимодействия (CI). Тут необходимо заметить, что для каждого возбуждённого состояния надо свой детерминант Слейтора и считать орбитали заново.

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

Корреляция электронов

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

- Расчёт электронных взаимодействий по Хартри-Фоку игнорирует мгновенное кулоновское отталкивание, что приводит к ошибкам примерно в 1% общей энергии и это химически значимо.
- CI конечно ограничен из-за сложности счёта, но обычно учёт однократно- и двухкратновозбужденные конфигурации позволяет покрыть 95% корреляционной энергии и понизить энергию системы на 200-300 кДж.
- Таким образом можно правильно описать диссоциацию N_2

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

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

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

Метод самосогласованного поля

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

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

Метод CI хорош когда предсказанные методом Фока орбитали достаточно близки к реальности. В методе самосогласованного поля предлагается варьировать не только коэффициенты C , но и форму орбиталей:

$$\Psi \equiv \sum_{K=0}^{\infty} C_K \Psi_K(\delta\psi_1 \delta\psi_2 \dots \delta\psi_K)$$

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

Как и ранее мы не можем взять и использовать все комбинации, поэтому рассматривают только однократные и двухкратные состояния возбуждения.

Можно явно указать на какой уровень будет переходить возбуждаемый электрон.

```
my $r;
my $s;
my $n;
my $z;
my $x;
my $y;
my $z;
```

```
$s=$s+$coor{$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;
```

Теория возмущений Меллера-Плессета

Считаем, что метод фока хорош, только нужно корректировать поправкой n -ого порядка:

$$\Psi = \Psi_0 + \sum_K^{C_K} \Psi_K (\delta\psi_1 \delta\psi_2 \dots \delta\psi_K)$$

Тогда при $n=0$ эта поправка равна 0 и собственно имеем результат Фока. Меллер и Плессет предложили считать корреляционную составляющую как:

$$\hat{V} \equiv H - (F + \langle \Psi_0 | H - F | \Phi_0 \rangle) \approx \frac{1}{2} \sum_N \frac{1}{r_{ij}} - \sum_N (J_j - K_j)$$

Поправки первого и второго порядка:

$$E_1 = \langle \Phi_0 | \hat{V} | \Phi_0 \rangle; \quad E_2 = \sum_{j < > 0} \frac{|\langle \Phi_0 | \hat{V} | \Phi_j \rangle|^2}{E_0 - E_j}$$

```
#!/usr/bin/perl
use Math::VectorReal qw( :all );
use IO::File;
my $f = IO::File->new($ARGV[0], "r");
my $q = $f->readline();
```

Теория возмущений Меллера-Плессе

```
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;
```

Ограничения:

```
my $system("mkdir $ARGV[1]");
my $filename=$ARGV[0];
$filename="-- s/^.*\//";
$filename="-- s/\.pdb//";
my $filename="$schnum."."$sqnum."."$filename."."dat";
my $filename="$mdir."$filename."dat";
my $f = IO::File->new($filename, "w");
print OUT "INFO chain $schnum $sqnum $schnum\n";
```

Состояние молекулы достаточно хорошо описано одним определителем Слэйтора и основной вклад в энергию вносят возбуждения n- порядка, т. е. для MP2 не более чем двукратные возбуждения.

```
my %my%qarts = %qwa; #find quart( %coor{$m} );
my @my%qarts = keys %qarts; { print join " ", @my%qarts{$sq} }, "\n";
foreach my $q ( keys %qarts ){
    my $nx; my $ny; my $nz;
    my $ox; my $oy; my $oz;
    my $r;

    foreach my $res ( @my%qarts{$sq} ){
        #
        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;
```


Теория функционала плотности, DFT

- В DFT система описывается не волновой функцией, а функцией электронной плотности, включающей вклад всех электронов.
- Предполагается, что для любой реальной системы с потенциалом и плотностью существует такая воображаемая «невзаимодействующая» система (т.е. система, в которой отсутствует межэлектронное взаимодействие) с некоторым одноэлектронным потенциалом, электронная плотность которой ($\rho_s(r)$) совпадает с точной электронной плотностью реальной системы.
- Точное решение многоэлектронного уравнения Шредингера представляется слэтеровским детерминантом $\Psi(r)$, состоящим из одноэлектронных орбиталей.

$$\rho(r) = \rho_s(r) = \sum_i^N |\chi_i|^2$$

Теория функционала плотности

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

Основная идея теоремы Хоэнберга-Кона:

Если в системе постоянное кол-во электронов, то их взаимное взаимодействие не зависит от внешнего потенциала.

Тогда разбиваем энергию электронов на:

$$E = E^T + E^V + E^J E^{XC}$$

- T кинетическая составляющая
- V потенциальная энергия (ядра)
- J отталкивание электронов
- XC обменно-корреляционная составляющая.

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

Теория функционала плотности

Хоненберг и Кохн показали, что E^{XC} зависит только от электронной плотности.

Обычно E^{XC} разбивают: $E^{XC} = E^X + E^C$

Обе части зависят от плотности и их часто представляют как:

- локальные, которые зависят только от плотности

$$E_{LDA}^X = 3/2 \left(\frac{3}{4\pi} \right)^{1/3} \int \rho^{4/3} \partial^3 r$$

- Градиент-корректированные, они зависят от плотности и её градиента

Широко используемый сегодня функционал Векке 88 года:

$$E_{Bekke88}^X = E_{LDA}^X - \gamma \int \frac{\rho^4 / 3x^2}{(1 + 6\gamma \sin h^{-1} x)} \partial^3 r; \quad x = \frac{|\nabla^2 \rho|}{\rho^{4/3}}$$

Теория функционала плотности

```
#!/usr/bin/perl
use Math::VectorReal qw( ! );
my $coor;
my $snum;
my ($my $coor,$my $snum)=read_pdb($ARGV[0]);
my %coor=read_pdb($ARGV[0]);
my $mdir=$ARGV[1];
```

Появились корреляционные функционалы:

- Pedrew и Wang (1991)
- Vosko, Wilk и Nusair (1980)

И так далее. Часто применяются гибридные функционалы:

$$E_x^{HF} = \frac{1}{2} \sum_{i,j} \int \int \psi_i^*(r_1) \psi_j^*(r_1) \frac{1}{r_{12}} \psi_i(r_2) \psi_j(r_2) dr_1 dr_2$$

$$E_{xc}^{B3LYP} = E_{xc}^{LDA} + a_0(E_x^{HF} - E_x^{LDA}) + a_x(E_x^{GGA} - E_x^{LDA}) + a_c(E_c^{GGA} - E_c^{LDA})$$

где:

$$E_{XC}^{GGA}[n_{\uparrow}, n_{\downarrow}] = \int \epsilon_{XC}(n_{\uparrow}, n_{\downarrow}, \vec{\nabla} n_{\uparrow}, \vec{\nabla} n_{\downarrow}) n(\vec{r}) d^3 r$$

```
# 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;
```

Теория функционала плотности

Применение:

- Современные функционалы дают хорошую производительность при хорошей точности результатов.
- В расчётах квантовой химии одним из распространённых является виды обменного функционала BLYP, B3LYP.
- В DFT есть проблемы в описании Ван-дер-Ваальсовых взаимодействий или дисперсионного взаимодействия, т.е. стэкинг и $\pi - \pi$ взаимодействия. Их в принципе можно компенсировать аналитическими потенциалами.
- «В целом, текущее состояние метода теории функционала плотности таково, что невозможно оценить погрешность расчёта, не сравнивая его результаты с другими подходами или с результатами экспериментов” ru.wikipedia.org

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

Для самостоятельного изучения:

```

#(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 } ;

```

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

- Методы получения значений физико-химических свойств на основе известной электронной структуры.
- Метод сопряжённых кластеров.
- Плоско-волновые базисы их применение.
- Современная теория валентной связи.

Доклад на любую из этих тем и зачёт автоматом.

```

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

```