Computer Lab 4a – Peptide Monte Carlo Simulations

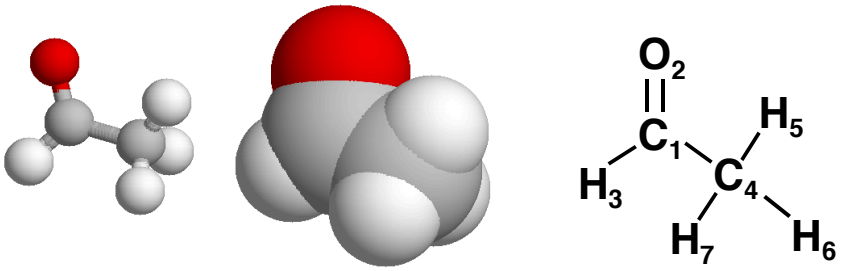
In this lab, we will become familiar with the chemical structures of polypeptides by building and analyzing peptide models. The relationship between **internal coordinates** and **Cartesian coordinates** will be explored. The role of steric restriction in determining polypeptides will be investigated.

I. Z-matrix description of polypeptide molecule.

In a collection of single atom molecules such as Argon the position of each atom in space was the only molecular information needed to completely describe the system and calculate the interaction energy. Now we consider molecules which have bonded atoms and the energy of the molecule will include terms for bond lengths, bond angles and rotations around bonds (dihedral or torsion angles). Such a molecule can be described either as a collection of coordinates in space or as a path of bond lengths, angles and rotations extending from the first atom in the molecule. This latter description, called **internal coordinates (IC),** is defined by a **Z-matrix**. The use of internal coordinates has advantages for manipulating molecular coordinates while retaining chemical relationships as we will see in this lab.

A. Simple Example

An example will illustrate the concept of a Z-matrix. The figure below shows three different depictions of the acetaldehyde molecule.



A Z-matrix for acetaldehyde is contained in the following table.

Atom Number	Atom Name	Bond Distance	Bond Angle	Dihedral Angle	Atom Connect	Angle Connect	Dihedral Connect
1	C_1	-	-	-	-	-	-
2	O_2	1.2	-	-	1 C_1	-	-
3	H_3	1.1	120	-	1 C_1	2 O_2	-
4	C_4	1.5	120	180	1 C_1	2 O_2	3 H_3
5	H_5	1.1	110	0	4 C_4	1 C_1	2 O_2
6	H_6	1.1	110	120	4 C_4	1 C_1	2 O_2
7	H_7	1.1	110	-120	4 C_4	1 C_1	2 O_2

In a Z-matrix the first atom is considered the origin, it usually is given the *x*, *y*, *z* coordinates [0.0, 0.0, 0.0]. Here the carbonyl carbon **C_1** is chosen as the first atom. The second atom is placed at the required bond distance apart from the first atom; here the second atom is the carbonyl oxygen **O_2** and it is 1.2 Å from the **C_1**. The Z-matrix specifies its connection to atom 1 (**C_1**) and its distance (1.2 Å) from that atom. Usually this second atom is placed along the *x* axis so in this case it would have *x*, *y*, *z* coordinates 1.2, 0.0, 0.0. The third atom requires *both a distance and angle* (planar angle) to describe its position relative to the first and second atoms. In our acetaldehyde example the third atom (**H_3**) is also connected to atom 1 (**C_1**), is 1.1 Å away from atom 1 and makes an angle of 120° with atom 1 and atom 2 (**O_2**). Convention places the third atom in the *x*, *y* plane. The fourth atom (**C_4**) *requires a distance, a planar angle and a dihedral angle* to describe its position relative the the first three atoms; it is 1.5 Å from atom 1, makes an angle of 120° with atoms 2 and 1 and makes a dihedral angle of 180° with atoms 3, 2 and 1. **From here to the end of the molecule the positions of atoms are described by a distance, planar angle and dihedral angle to previously defined atoms.**

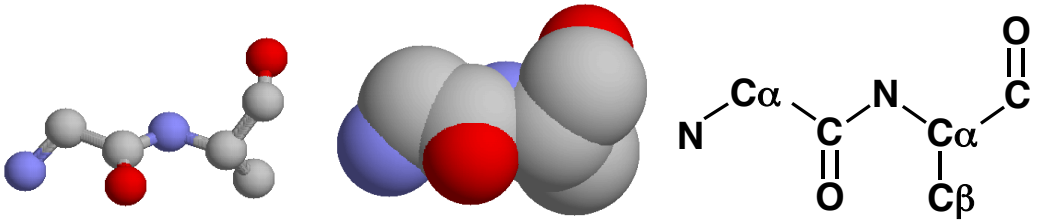
When talking about Z-matrices we frequently use the following terminology for the preceding or **antecedent atoms**: The immediately preceding atom (-1) is the **first parent**, the second (-2) is the **second parent**, etc.

Given a complete Z-matrix and coordinates of the first atom one can calculate the Cartesian coordinates using geometry. Notice that the spatial position of the whole molecule is set by the direction of the first bond. This may be chosen arbitrarily.

The advantage of using a Z-matrix is that one can make changes to the bond angles and dihedral angles and change the conformation of the molecule while at the same time staying within chemically prescribed stereochemistry. For example, the torsion angles may be constrained to known acceptable ranges. **This is an efficient way to explore the conformational freedom of a large molecule in a Monte Carlo simulation.**

B. POLYPEPTIDE Z-MATRIX

In this lab, we will build peptides using a Z-matrix description of the peptide and specified bond rotations or dihedral angles. Below are three depictions of the dipeptide GLY-ALA. No hydrogens are shown and only the two amino acid residue units are shown as they exist in a continuing polypeptide; i.e. there is no C-terminal OH.



A Z-matrix for the GLY-ALA dipeptide is shown in the following table.

Atom Name*	Bond Distance	Bond Angle	Dihedral Angle	Atom Connect	Angle Connect	Dihedral Connect
N (1)	-	-	-	-	-	-
Cα (2)	1.451	-	-	1 N(i)	-	-
C (3)	1.516	112.5	-	2 Cα(i)	1 N(i)	-
O (4)	1.231	120.6	137	3 C(i)	2 Cα(i)	1 N(i)
N (1)	1.329	116.2	180	-3 C(i-1)	-2 Cα(i-1)	-4 N(i-1)
Cα (2)	1.458	121.7	180	1 N(i)	-3 C(i-1)	-2 Cα(i-1)
C (3)	1.525	111.2	-64	2 CA(i)	1 N(i)	-3 C(i-1)
O (4)	1.231	120.8	137	3 C(i)	2 Cα(i)	1 N(i)
Cβ (5)	1.521	110.4	-122	2 Cα(i)	1 N(i)	3 C(i)

*Notice that we have numbered the atoms for each residue separately. Some antecedent atoms are in the previous (*i*-1) residue and some are in the current residue (*i*). If the connecting atom is in a previous residue the numbering counts negatively upstream (*i*-1, toward the N-terminus) from the current atom.

II. Build a dipeptide from a Z-matrix.
Answer the questions highlighted in yellow and send the answers to **achin14@jhu.edu** either in the body of an email or as an attached file with your JHEDID in the name (e.g. *JHEDID_lab1b.txt* if made with vi or *JHEDID_lab1b.docx* if made with MSWord).

We will use a Python program called **ribosome.py** to build a dipeptide from the above Z-matrix. The program and Z-matrix are stored in a subdirectory of the Python installation on the cluster, you will not have to understand the code in these programs. We will create a PDB file of the dipeptide on the cluster and **sftp** the file back to the Mac for analysis.

1. Log on to a Mac, change to your permanent home directory or flashdrive, make a new directory called *lab4a*, change to that directory. This will be your main Mac working directory for the lab.

2. Open a second terminal window on the local machine and **`ssh`** to your account on the cluster (**`compbio2@kirin.kit.jhu.edu`**), then **`cd [yourJHEDID]`**.

3. To set up your UNIX shell environment on the cluster just enter the following,

```
tcsh
```

Check your account for an **environmental variable**. An environmental variable

can define a shortcut so you won't have to type the full directory path to a program every time you want to run the program. You already have several environmental variables set. Type **env** to see what they are. You will see that the home directory for the class is defined as the HOME variable, etc. You should also see the following in the list of your environmental variables,

```
LINUS= /home/apps/lib/python2.7/site-packages/pylinus_1_0
```

To use an environmental variable, you prepend a dollar sign to the variable name. You can see the definition of a specific variable by echoing it to the screen. Type **echo $LINUS**. You should see the definition of the LINUS environmental variable again.

To see the utility programs that are available in this subdirectory of the Python installation type **ls $LINUS/utils/**. There should be one called **ribosome.py**.

4. Use **vim** to create a file called *gly_ala.rib* containing the following three lines:

```
TITLE GLY ALA
RES GLY
RES ALA
```

The file *gly_ala.rib* is a simple command file that tells the ribosome program what residue types to use in building a peptide.

5. Now run the ribosome program by entering the following command:

```
python2 $LINUS/utils/ribosome.py < gly_ala.rib > gly_ala.pdb
```

1. What is the meaning of the redirect symbols **<** and **>**?
Notice that by using the environmental variable **$LINUS** you did not have to type **/home/apps/lib/python2.7/site-packages/pylinus_1_0/**. Since we will run LINUS programs many times having this variable will save time. Inspect the new file *gly_ala.pdb* to make sure it looks like a PDB file.

7. On the Mac in the *lab4a/* directory use **sftp** to bring the newly created PDB file, *gly_ala.pdb* back from the cluster to the Mac and view it with PyMOL.

2. What are the coordinates of the first atom (GLY, N)?

```
PyMOL> select resn gly and name N
PyMOL> print cmd.get_atom_coords('sele')
```

3. What are the coordinates of the second atom in the structure (GLY, CA)? Compare these values to the Z-matrix above. 4. Do these coordinates agree with the Z-matrix?

There is an alternate syntax for selecting atoms in PyMOL that is shorter and we

will use this syntax to display the peptide torsion angle. Here is the generic syntax,

```
/object/segid/chain/resid/name
```

If there is only one *object* in the PyMOL window you can leave that item blank, if there is only one *segid* in the object you can leave that item blank, etc. To see an example, enter the following in the PyMOL command window (atom order is same as in a row in the above Z-matrix),

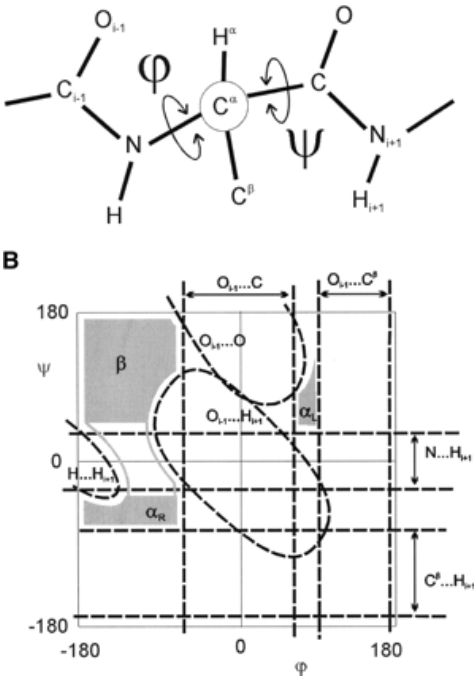
```
get_dihedral ////2/C, ////2/CA, ////2/N, ////1/C
```

5. What is the torsion angle value and does it agree with the Z-matrix above?

III. Peptide conformations are sterically restricted.

A. RAMACHANDRAN PLOT

The main angles which determine the conformation of a polypeptide backbone are the dihedral, or torsion, angles called ϕ and ψ. These are defined at the top of the figure at right. Free rotation is not allowed for these dihedral angles because for some combinations of ϕ and ψ atoms along the polypeptide peptide will clash as shown in the bottom part of the figure. For example, ϕ may not be in most of the region 90° to 170° because the carbonyl oxygen O(i-1) will clash with Cβ(i). Certain allowed combinations of ϕ, ψ are given special names, β, α_R and α_L. Such a plot of the allowed ϕ,ψ combinations is called a Ramachandran plot.



You will now determine for yourself whether a ϕ between 90° and 170° is possible. You can build polypeptides with specified ϕ,ψ angles by including them in the **.rib* file.

1. Go to the cluster, copy *gly_ala.rib* file to *gly_ala_ala.rib* and edit the new file to contain the following information. These instructions will allow you to build a tripeptide with the central residue having a ϕ = 150, supposedly in violation of the steric rules indicated above. The other ϕ,ψ angles are in the allowed regions as shown above.

```
TITLE      GLY ALA ALA
RES GLY PHI -120 PSI 130
RES ALA PHI  150 PSI 130
RES ALA PHI -120 PSI 130
```

2. Run the **ribosome.py** program as before (but redirect the output to a file called *gly_ala_ala.pdb*) and bring the new PDB file back to the Mac. (Try the command, **`!sftp`** in the Mac terminal window you previously used for sftp.)

3. View the tripeptide GLY-ALA-ALA in PyMOL and measure the distance between the GLY carbonyl oxygen and the following ALA Cβ. To label the atoms in PyMOL click on the following in the command menu,

   gly_ala_ala L | atom name

   Then use the Wizard to display the distance between two atoms,

   Wizard | Measurement [click on the two atoms]
   Click Done

4. Assume that the VDW radii for oxygen and carbon are 1.35Å and 1.65Å, respectively.
   6. Are these two atoms in van der Waals overlap?
   7. Would this conformation of the peptide be the lowest energy conformation given the Lennard-Jones description of atomic interaction energies?

   Display the molecule as spheres. Use,

   Display | Quality | Maximum Quality

   8. Do the VDW surfaces of the above carbonyl oxygen and Cβ carbon atoms overlap?
   9. Do any other non-bonded atoms overlap? (Note: We usually count "bonded" atoms as those that have three or fewer bonds between them. In other words, the *i* and *i*+1, *i* and *i*+2, and *i* and *i+3* atoms may have slight overlap because they are "sharing" some of the electron density.)

Now we will build polyalanine peptides in the two most frequently observed backbone conformations, α-helix and β–strand.

1. Go back to the cluster and make two new **.rib* files called *helix.rib* and *strand.rib*. Their contents should be as follows on the next page.
   (Shortcut: In **vim** after you have typed the first **RES ALA** and escaped the insert mode, enter the keys, **yy** and then **p** (yank a line and put it). Now you can continue to put (enter **p**) the yanked line until you have thirteen such lines).

```
TITLE  HELIX
DEFAULT PHI -60.0
DEFAULT PSI -40.0
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
```

```
TITLE   STRAND
DEFAULT PHI -120.0
DEFAULT PSI  130.0
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
```

Notice that we can set the default dihedral angles at the top rather than specifying the angles for each residue as in the previous example.

2. Run **ribosome.py** on these *.rib* input files and redirect the output to appropriate PDB files (e.g. *helix.pdb* and *strand.pdb*).
3. Fetch both new PDB files and the python script, */home/compbio2/Shared/phipsi.py*, back to the Mac and view the helix PDB file in PyMOL first.

10. What are ϕ,ψ values for the helix structure?

PyMOL> **run phipsi.py**
PyMOL> **phipsi helix**

Click the following items in the command menus to the right of the helix object name.

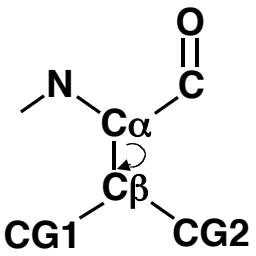
A | find | polar contacts | within selection

11. Does the helix structure have hydrogen bonds between the O(i) and N(i+4)?

12. What are ϕ,ψ values for the strand structure?

B. SIDECHAIN ROTAMERS

In the first lecture, we mentioned sidechain **rotamers**. This word refers to the concept that the alkane chains of amino acid sidechains have preferred conformations. Similar to the familiar butane torsion angle plot we have used several times in this course each torsion bond in a sidechain usually has three energy minima at the *gauche+*, *trans*, and *gauche-* conformations. These are the most likely conformations. For example, the amino acid valine has three rotamers with energetically optimal dihedral angles –60°, 180° and 60° around the Cα-Cβ bond. The bond we are talking about is illustrated in the next figure.



These sidechain dihedral angles are called χ (chi) angles. In terms of the Z-matrix this angle is defined as CG1(i)-Cβ(i)-Cα(i)-N(i). We can build peptides with specified χ angles by including them in the **.rib* files. On the cluster edit your *helix.rib* and *strand.rib* files to include three valine residues with the following χ angles,

```
TITLE  HELIX
DEFAULT PHI -60.0
DEFAULT PSI -40.0
RES ALA
RES ALA
RES ALA
RES ALA
RES VAL CHI1 -60
RES VAL CHI1 180
RES VAL CHI1  60
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
```

```
TITLE   STRAND
DEFAULT PHI -120.0
DEFAULT PSI  130.0
RES ALA
RES ALA
RES ALA
RES ALA
RES VAL CHI1 -60
RES VAL CHI1 180
RES VAL CHI1  60
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
RES ALA
```

Build these peptides, bring them back to the Mac and view them in PyMOL. Then select the sidechains of valine residues and color them cyan. This can be accomplished by entering the following for each structure (you may want to open another window on the Mac and launch two PyMOL sessions.

```
PyMOL> show spheres
PyMOL> select bb, name c+o+n+ca
PyMOL> select sc, resn VAL and not bb
(sc)      C l cyans l cyan
```

13. Are all rotamers equally sterically probable in **both** helix and strand conformations?
14. Which backbone conformation, helix or strand, is preferred for valine?
15. Explain your last answer in terms of Boltzmann statistics.

Go back and review your answer to the last question – it illustrates the crux of the big idea for this lab.