Classical Simulations with PJNMR

The program PJNMR (Pure Java NMR) was written by Paul-Jean Letourneau as a joint project between the NANUC (National High Field Nuclear Magnetic Resonance Centre) resource of Canada and the former PENCE (Protein Engineering Network of Centres of Excellence) initiative of Canada. The PJNMR program is program simulates the effects of radio frequency pulses, gradient pulses and evolution periods on bulk magnetization vectors for spin-1/2 nuclei. The output is a graphical depiction describing the movement of the bulk vector and the density matrix elements associated with each state. It allows up to three spins and provides a history and schematic of the pulses applied, thus creating a 'pulse program', which can be saved as a macro. Below is a link to the original manuscript in the Journal of Magnetic Resonance describing the program.

https://www.sciencedirect.com/science/article/pii/S1090780703000089?via%3Dihub

PJNMR is a java applet and so can be run on any platform. If you want a copy for your own computer, follow the download instructions under "download" on the NANUC website:

http://www.nanuc.ca/downloads/downloads_login.php

You will have to register, but this is free. Follow the installation instructions. There is an install script which will set up appropriate links, etcetera.

In a separate document ("pjnmr-download-install.pdf") we've written explicit instructions for installation of PJNMR version 3.0.4 on MacOS and LINUX systems.

Running PJNMR 3.0.4

If you follow the instructions for downloading and installing PJNMR 3.0.4 ("pjnmr-download-install.pdf"), then the program can be started using the following commands:

MacOS:

```
/Users/UserName/pjnmr-v3.0.4/pjnmr/pjnmr
```

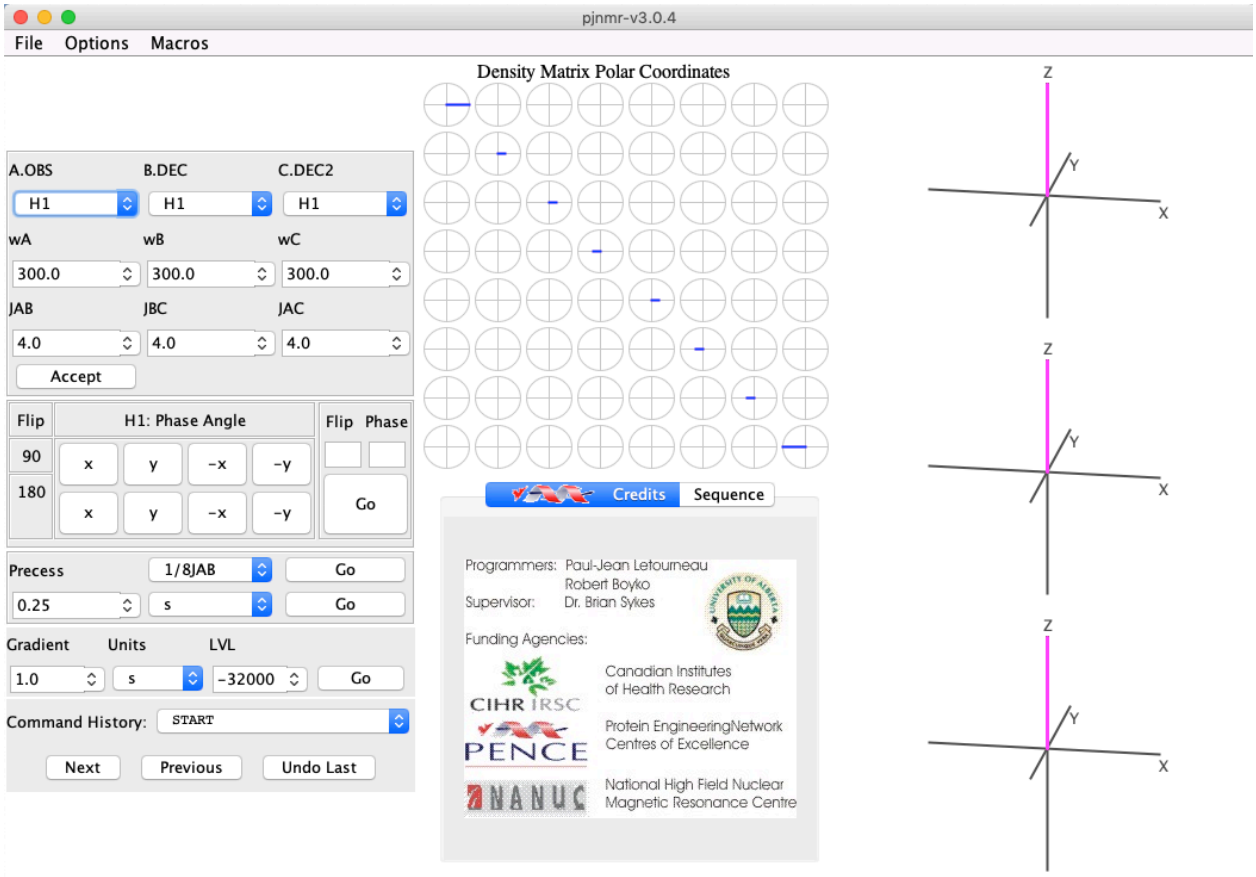
LINUX:

```
/home/UserName/pjnmr-v3.0.4/pjnmr/pjnmr
```

If you are using an earlier version, or it has been installed differently, or if you are using a Windows system, you should consult with your computer administrator for running PJNMR.

PJNMR Graphical Interface

When you start PJNMR, the graphical interface will appear on the screen, as shown below.



The program can handle three spins. The dropdown menus under 'A.OBS', 'B.DEC', and 'C.DEC2' allow selection of the nucleus of interest (^{1}H, ^{13}C, ^{15}N, etcetera, including 'none'). The nucleus to be observed is always 'A.OBS', so there is not a 'none' option for 'A.OBS'.

The next line of dropdown menus allows selection of resonance frequency (in MHz). In the default startup screen shown above, all three nuclei are set to ^{1}H, and all are set to 300 MHz.

The next line of dropdown menus allows selection of coupling constants (in Hz).

The 'Accept' button reinitializes the system (reestablishes equilibrium bulk magnetization).

The next block allows for application of radiofrequency pulses. Clicking on the '*x*' box on the '90' Flip line results in a 90 degree pulse about the '*x*' axis. This moves the magenta magnetization vector 90 degrees onto the *-y* axis. The 'Flip' and 'Phase' boxes allow for application of pulses of any pulse angle and phase. For instance, Flip=90 and Phase=0 (followed by clicking the 'Go' button) gives the same result as clicking the 'x' box on the 90 degree line.

The ‘Precess’ block allows for evolution. In the default menu, the precession time is ‘0.25’, with units of ‘s’ (the units can be selected from the dropdown menu). If multiple spins are involved, convenient times like ‘1/8J’ can be selected (in the default menu above, ‘1/8JAB’ is shown).

The ‘Command History’ keeps track of commands that were entered. If a string of commands are entered, one can return to the result after any command by clicking on the ‘START’ dropdown menu, and locating the appropriate command. There are also ‘Next’ and ‘Previous’ buttons that allow the results of the ‘next’ and ‘previous’ steps to be examined. An ‘Undo’ button is also helpful. There is also the option to save a session, and it (a macro that runs the commands) can be reloaded and run again at a later time.

At the top of the center of the screen is a block showing the population of density matrix elements. This will become more useful to students later in the semester, so we’ll mostly ignore it in this tutorial for now.

In the center of the screen is a toggle between ‘Credits’, which just tells who wrote the program, etcetera, and ‘Sequence’, which gives a nice graphical depiction of commands and command history.

Also, the axes system(s) at the right can be rotated using the mouse and left mouse button, although this feature doesn’t work that well.

The ‘File’, ‘Options’, and ‘Macros’ menus include some useful features, such as saving sessions and macros, taking screenshots, changing the look (white versus black background), and others. You might want to experiment with these later.

A note about conventions. For organic chemists, and in NMR books written by and for organic chemists, a 90° *x* pulse on equilibrium ‘+*z*’ magnetization results in the bulk magnetization vector moving from the ‘+*z*’ axis to the ‘+*y*’ axis. PJNMR uses the “spectroscopists” convention, where a 90° *x* pulse on equilibrium ‘+*z*’ magnetization gives ‘-*y*’ magnetization.

Below are some exercises for you to try.

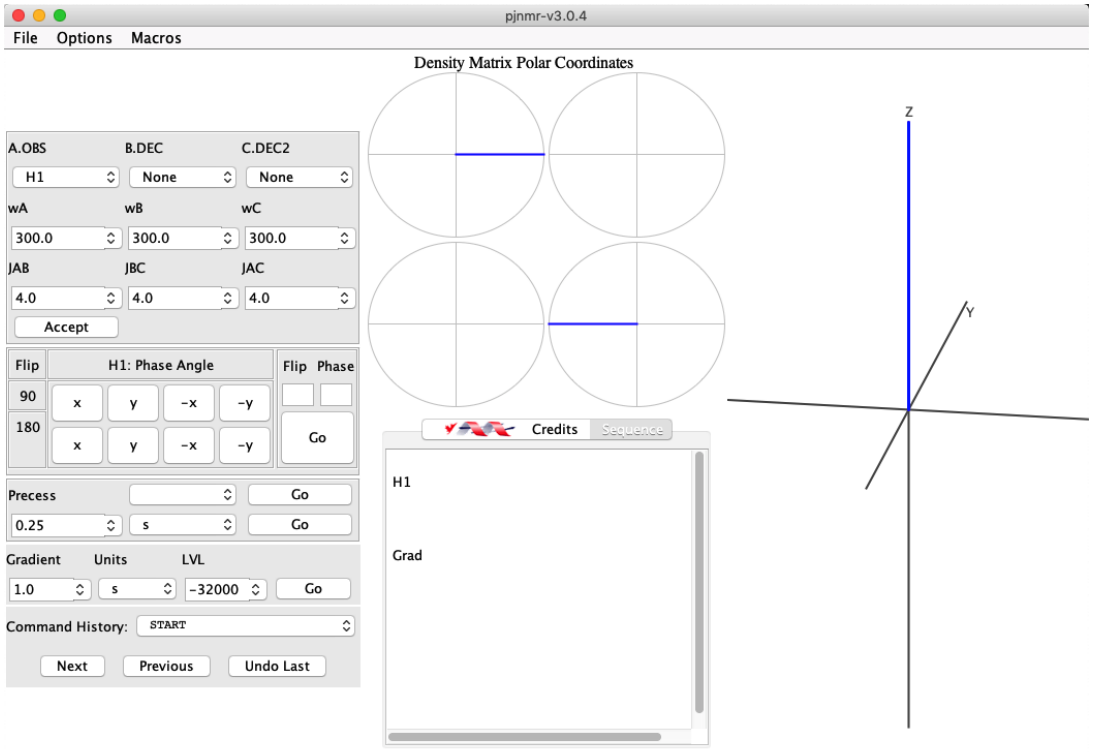
Exercise 1.

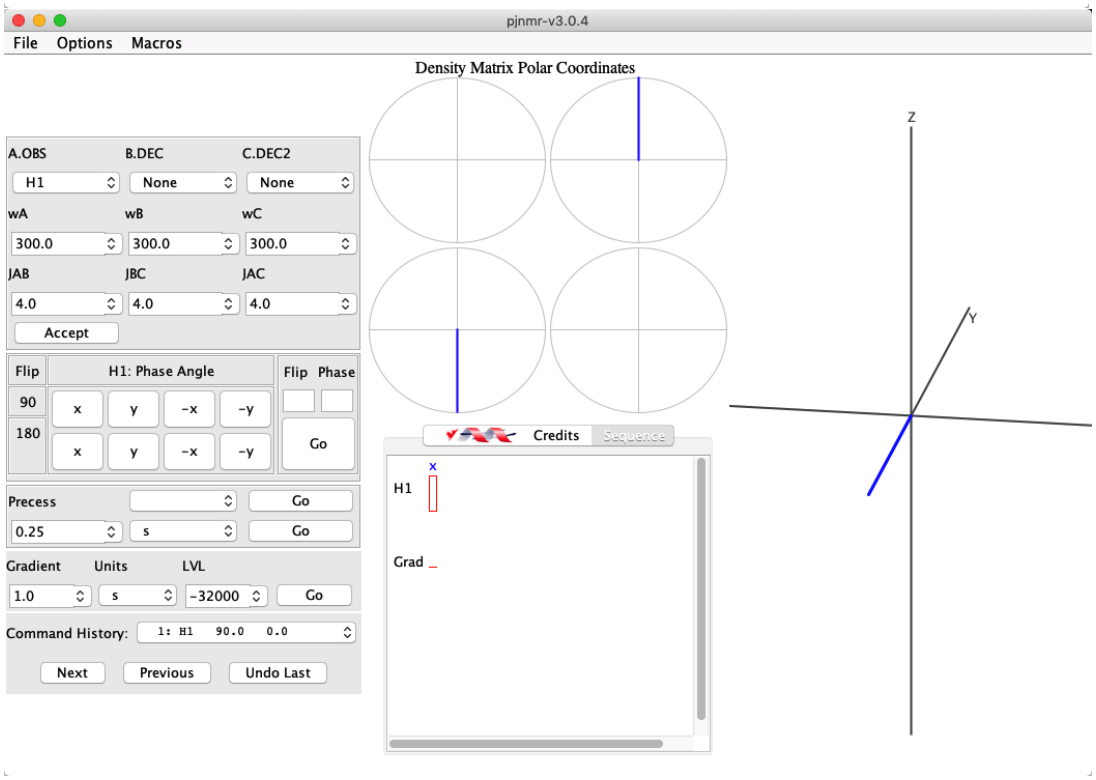
We’ll learn some of the basic features of applying pulses and visualizing the results.

We’ll consider a single ^{1}H spin.

a) For ‘A.OBS’ leave it set to ‘H1’. For ‘B.DEC’ and ‘C.DEC2’ set both of these to ‘none’ using the pulldown menus.
b) Click on ‘Accept’.
c) Click on the ‘Sequence’ button. The display should be the top figure on the next page (note, the density matrix has only diagonal elements, i.e. no coherences yet).

d) Click on '*x*' on the 90 'Flip' line. As shown in the bottom figure, the bulk vector has moved to the -*y* axis (note, off-diagonal density matrix elements indicate coherences generated by the pulse).





Exercise 2.

Examining and predicting the effects of other pulses on equilibrium '+*z*' magnetization.

Repeat the exercise above, starting with equilibrium '+*z*' magnetization, and examining the effects of other pulse angles and phases (for instance, 180° *x*, 90° -*y*, 270° *y*, etcetera). *Before you apply each pulse, you should be able to predict the outcome*.

Next, repeat this exercise using the 'Flip' and 'Phase' boxes (and the 'Go' button) to implement the pulses. Enter '90' in the 'Flip' box and '0' in the 'Phase' box, then click the 'Go' button. You'll see that this represents a 90° *x* pulse. Likewise, 90° pulses with phases of '90', '180', and '270' represent 90° *y*, 90° -*x*, and 90° -*y* pulses, respectively. You should confirm this. You can also make the pulse angle very small, like 5°, and then repeatedly click 'Go' to watch the vector move away from the +*z* axis and into the transverse plane

Exercise 3.

Examine the effects of pulses on transverse magnetization.

Beginning with equilibrium '+*z*' magnetization, use any 90° pulse to generate transverse magnetization (+*x*, -*x*, +*y*, or -*y*). You should be able to predict the outcome of any 90° pulse on equilibrium '+*z*' magnetization. THEN, with the magnetization vector on some axis in the transverse plane, apply any 90° pulse and see what the effect is. Again, you should be able to predict the outcome. Repeat with 180°, 270° pulses, etcetera, with different phases.

Exercise 4.

In NMR, 180° pulses are used to invert magnetization. For instance, beginning with equilibrium '+*z*' magnetization, a 180° pulse converts the '+*z*' magnetization to '-*z*' magnetization. If the 180° pulse is perfect (perfectly calibrated), then complete inversion occurs. If not, then some magnetization is not inverted.

So-called "composite" 180° pulses, or "compensated" 180° pulses were designed to mitigate the effects of imperfectly calibrated pulses. A composite 180° inversion pulse, for instance, is often implemented as a three-pulse sequence, such as 90° *x*, 180° *y*, 90° *x*. Other phases can also be used (90° *y*, 180° *x*, 90° *y*, or 90° *y*, 180° -*x*, 90° *y*, etcetera). *Here we'll examine if composite 180° pulses perform better than simple 180° inversion pulses when the pulses are imperfectly calibrated*.

First, using the 'Flip' and 'Phase' boxes (and 'Go' button), compare a perfect 180° x pulse to a miscalibrated or imperfectly calibrated 180° x pulse. Here we are assuming that a 180° pulse is imperfectly calibrated, such that it is really only a 170° pulse.

"perfect" 180° x pulse: (Flip=180, Phase=0, Go)
"imperfect" 180° x pulse: (Flip=170, Phase=0, Go)

Now compare a perfect composite 180° x pulse to an imperfect one.

"perfect" 180° x pulse: (Flip=90, Phase=0, Go; Flip=180, Phase=90, Go; Flip=90, Phase=0, Go)
"imperfect" 180° x pulse: (Flip=85, Phase=0, Go; Flip=170, Phase=90, Go; Flip=85, Phase=0, Go)

You should be able to see that, despite the fact that the individual pulses are imperfectly calibrated, that the *miscalibrated* 180° pulse comes much closer to giving complete inversion compared to the single miscalibrated 180° pulse.

Exercise 5:

Here we'll use PJNMR to observe precession in the transverse plane.

Keep the settings above ('A.OBS' = 'H1', 'B.DEC' and 'C.DEC2' = 'none'), but change the Larmor frequency for A to 100 ('wA = 100'). *Make sure to click 'Accept'*.

Now, apply a 90° x pulse to place the magnetization into the transverse plane.

You'll see the value of 'Precess' is 0.25 s. On the same line is a small 'Go' button. Click the button several times to see what happens (you should see that nothing appears to happen….why?).

Now change the value of 'Precess' to 0.025 s (remember to click 'Accept'). Apply a 90° x pulse to place the magnetization into the transverse plane. Then click the small 'Go' button on the same line several times. What happens? Why?

Repeat with 'Precess' = 0.0025 s and 0.00025 s.

So, you should be able to rationalize what you observe. When the Larmor frequency is 100 Hz (i.e., 100/s), a precession time of 0.25 s (each click of 'Go') causes the vector to precess exactly 25 times around the *z*-axis. Thus, it appears that nothing is happening each 0.25 s (each click of 'Go').

When the precession time is 0.025 s (with a Larmor frequence of 100 Hz), then each 0.025 s (each click of 'Go'), the vector precesses 2.5 times about the *z*-axis. Thus, the appearance is that the vector alternates between the +*y* and -*y* axis each time the 'Go' button is clicked.

When the precession time is 0.0025 s (with a Larmor frequence of 100 Hz), then each 0.0025 s (each click of 'Go'), the vector precesses 0.25 times about the *z*-axis. Thus, the appearance is that the vector moves 90° each time 'Go' is clicked.

When the precession time is 0.00025 s (with a Larmor frequence of 100 Hz), then each 0.00025 s (each click of 'Go'), the vector precesses 0.025 times about the *z*-axis. Thus, it takes 40 clicks of 'Go' to get the vector to make one revolution about the *z*-axis.

Problem Set:

For problems 1-4, you should be able to first predict the results/answers, and then use PJNMR to confirm your predications.

1). Starting with equilibrium '+*z*' magnetization, what is the final state (where does the bulk magnetization vector end up) after:
a) a 90° *x* pulse
b) a 270° *x* pulse
c) a 720° *x* pulse

2). Starting with equilibrium '+*z*' magnetization, what is the final state (where does the bulk magnetization vector end up) after:
a) a 90° *y* pulse
b) a 270° *y* pulse
c) a 90° -*y* pulse
d) a 270° -*x* pulse

3). Starting with equilibrium '+*z*' magnetization, a 90° pulse is applied that causes the bulk magnetization vector to end up on the '+*x*' axis.
a) what was the phase of the 90° pulse (*x*, *y*, -*x*, or -*y*)
b) if a 90° *y* pulse is applied to the '+*x*' magnetization, where does the bulk magnetization vector end up?
c) if a 90° *x* pulse is applied to the '+*x*' magnetization, where does the bulk magnetization vector end up?

4). Starting with equilibrium '+*z*' magnetization, a 90° pulse is applied that causes the bulk magnetization vector to end up on the '+*y*' axis.
a) what was the phase of the 90° pulse (*x*, *y*, -*x*, or -*y*)
b) if a 270° *y* pulse is applied to the '+*y*' magnetization, where does the bulk magnetization vector end up?
c) if a 90° -*x* pulse is applied to the '+*y*' magnetization, where does the bulk magnetization vector end up?

5). So-called 'volume-selective' 90° composite pulses (90° *y*, 90° -*x*, 90° -*y*, 90° *x*) were designed for removing remnants of strong solvent signals that arise from sample outside of the nmr coil volume. These signals are poorly shimmed and have very low pulse angles (e.g < 20°). For strong solvent signals, they can interfere with weaker signals of interest (e.g. broad asymmetric base of solvent peak). Try comparing a 'good' 90° *x* pulse (found inside the coil volume) with a 'poor' 90° *x* pulse (e.g. a 20° *x* pulse) found outside the coil volume: i.e. compare (Flip=90 Phase=90 Go; Flip=90 Phase=180 Go; Flip=90 Phase=270 Go; Flip=90 Phase=0 Go) to (Flip=20 Phase=90 Go; Flip=20 Phase=180 Go; Flip=20 Phase=270 Go; Flip=20 Phase=0 Go).

How does this composite pulse solve the problem?