Arrayed NMR Experiments at Constant Temperature
Lingyang Zhu, Dean Olson, and MCC; 5 Feb 2019

Purpose

To acquire NMR data at certain time intervals without temperature changes.

Example – Constant Temperature

Suppose you want to collect proton NMR spectra every 10 sec for a total of 2 min without using temperature control. If you acquire at time = 0, this results in a total of 13 proton spectra each separated by 10 sec. The **pad** parameter is automatically in sec.

Rough Setup – Getting Started

- Lock and shim as usual.
- For your nucleus and solvent, set values for **nt** and **d1** as usual.
- Set up an array for the pre-acquisition delay, **pad** . This is a hidden parameter not listed when you type **dg** . This is roughly the amount of time between acquisitions.
- Type **array**
- Answer the questions to set up an array:
    - Parameter to be arrayed **pad**
    - Enter number of steps in array **13**
    - Starting value 1**0**
    - Array increment **0**

After the array setup is complete, type **pad[1]=0** ; this ensures that the first arrayed **pad** value = 0, as desired. This way, an initial data set is acquired.

Type **da** at any time to display the set of arrayed values:

pad = 0,10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 10

They will appear listed next to an array increment.

Type **time** to learn the total amount of time the acquisition will take, but the value will not make sense, and will be an unexpected amount of time.

After you submit the experiment (type **ga** or **go**), the experiment will start normally, and acquiring a spectrum until all transients are collected. Then, there is a 10 sec wait before acquisition of the second spectrum, and so on, until all 13 spectra are collected.

After the arrayed experiment is completed, type **ds(1)** to display the first spectrum or any other spectrum number by typing **ds(#)**.

You can also display all the spectra by typing **dssh dssl** . This displays stacked spectra horizontally, and displays stacked spectra labels (which are really indexes for each arrayed increment). You can save the entire collection of spectra by typing **svf** and declaring a file name, as usual.

But, we have a problem.

Refined Setup – For Serious Kinetic Data Acquisition

The above approach has a weakness. The hint was that the **time** command returned a total time that was not 2 min, nor was it 2 min 10 sec; it was a total time that surprised us and did not make sense.

The time *between* acquisitions is a constant interval of 10 sec. However, the time at which each data set is acquired is *not* 10 sec, but the sum of **pad** + **d1** + **at** . Typically, in a kinetic experiment, we want the time at which data is collected to be constant, and a nice, round number, such as 10 sec. The above procedure achieved a constant interval, but it was not the nice, even 10-sec interval that we had hoped. We can correct this by simply typing:

pad = 10-d1-at

Then, type **pad?** To see the value calculated for **pad** . Use this value to set up the array again. Use of this **pad** value will yield an array increment that is more smartly chosen, and does more carefully what you really want.

Also, it is a good idea to set the gain to a fixed value. Type this:

gain?
gain = gain - 6

Now, type **array** and set things up as above, but use the newly computed value for **pad** . This will yield NMR data at evenly spaced intervals, just as intended.

Once the array is set up, type **time** to confirm the total experiment time to collect all the individual data sets in the array. Within about 1 sec, it should be exactly as you intended.

HOWEVER, the time command often returns the *wrong* total time. Before you begin an actual experiment, watch carefully the data collection rate yourself to confirm that acquisition is proceeding at the desired rate.

Note. You might wonder why **pad** was used for a separation interval instead of just **d1**. The reason is that the **pad** occurs between spectra, but the **d1** occurs between transients (**nt** times) within a spectrum. If you always had **nt** = 1, then you could just use **d1** to set an interval.

Data analysis note: when you save your arrayed spectra using **svf** and open it in Mnova, it will be a single file with all the spectra stacked in it. To extract all or some of your spectra from the combined file, use the ‘stack’ tools in Mnova to choose which spectra you want to extract. The extracted spectra will then open in the same Mnova file for manipulation and analysis.