

PRODUCT OPERATORS AND MAPLE

The product operators that we have introduced, and transformations that they undergo in the presence of scalar coupling, chemical shift offsets and rf pulses, represent an algebra that can be coded and applied in the environment of a symbolic problem solver on various computer platforms. Packages such as MATHEMATICA and MAPLE are capable of symbolic manipulation as well as evaluation of various functions in interactive algebra, graphics and discrete math. In 1993 Peter Guntert et al produced POMA (Product Operators in Mathematica) (JMR A 101, 103–105 (1993)) and Kanters et al produced a similar treatment run under MAPLE (JMR A 101, 23–29 (1993)). These groups wrote notebooks or procedure files that define various functions such as pulse(target, spins, angle) that transform variables (product operators) according to the transformation rules we have introduced. We will learn to use one of these packages (Kanters' MAPLE package, pof (product operator functions)) to analyze various 1D and 2D NMR experiments.

1) Getting Started:

You can log onto one of the computers in the computer lab which has MAPLE installed for these exercises. Activate the maple leaf symbol on the "dock" bar of your computer. This will start a version of MAPLE in a window on your computer. At the end of your session in Maple you may have to type 'quit' to get out of the program. There is also an X window version that can be run on LINUX computers. This can be used from remote computers by typing 'xmaple'. This will only work from a remote login if you enable a remote display first with a command such as setenv DISPLAY 'IP address':0 from an rsh or ssh window. You may also have to identify the host computer to your computer with the command xhost 'IP address'. There is a useful introduction to MAPLE that you can find under the 'help' button on the maple menu bar.

For those who may want to set up MAPLE on their own computer, UGA has a site license and those at UGA can easily acquire the basic software. Others should go to the maplesoft website: www.maplesoft.com. You may need to load the 'share' library that contains the 'pof' procedures used in this exercise. Send inquiries about the availability of this library to lmorris@ccrc.uga.edu.

2) Learning about MAPLE:

Most versions of MAPLE, test or menu driven, will do for our exercises. There are good help functions accessible via either the menu button or text commands. On either version to get help on any command, precede it with a '?' and end with a semicolon ie '?'

Help;'. A description of the command is displayed. To see a list of functions type '? index[function];'.

There are five basic rules to Maple V syntax according to one book: Maple V by Example, Abell and Braselton, Academic Press, 1994.

1. The arguments of functions are given in parentheses (...).
2. A semicolon (;) or colon (:) must be included at the end of each command (the colon gives no output)
3. Multiplication is represented by *.
4. Powers are denoted by ^.
5. If you get no response or an incorrect response, you probably entered the command incorrectly.

We will just do some basic things here to give you an idea of how MAPLE works before getting into product operators. Let's multiply some matrices. The operations for matrices are in a library called linalg. We tell Maple that we will be using functions in this library by typing:

```
with(linalg);
```

Maple will respond with a list of functions.

Define a matrix, A, by typing:

```
A:=array([[1,1,0],[1,1,1],[0,1,1]]);
```

:= is the general way you define a variable.

Maple will respond with a matrix form.

Find the inverse of this matrix by typing:

```
B:=inverse(A);
```

Maple will respond with the inverse of A

Multiply A by its inverse by typing:

`A&*B;`

If the form is still symbolic. Try typing:

`evalm(A&*B);`

Note, `&*` denotes matrix multiplication and `evalm` is the matrix analog of `evalf` for functions. It gives the numeric representation of the result.

We can define our own functions (nmr, for example) in terms of 'procedures' using a series of commands like:

```
nmr:= proc(X,A) A&*X&*inverse(A) end;
```

If we now define two matrices C and R and type:

'nmr(C,R)' the procedure will be executed. If R is a rotation operator and C is a density matrix, the result would be a transformed density matrix. Try this with the rotation operator for a 90x pulse and an equilibrium density matrix for a single spin $\frac{1}{2}$ spin (note, the symbol for the square root of minus one is 'I').

There are more basic algebra and graphics examples in the supplementary file labeled introduction on the class web page.

3) Using Product operator Formalism:

Kanter has defined procedures for rf pulse effects and evolution of density matrices represented as linear combinations of product operators, and put these in a library called 'pof'. Worksheets giving examples of application are available at the, URL: www.maplesoft.com/applications. Search for 'pof' and several examples should appear. These can be downloaded by simply registering at the site or using the guest privileges. We have also provided copies of similar sheets on our class website and they should already be in a folder called "Maple worksheets" on your desktop. Load the file called 'My_CS_refocussing.mws' using options in the "folder" icon near the top left of the Maple window. This worksheet illustrates the effect of a 90-tau-180-tau sequence, the same sequence that removes the effects of chemical shift evolution and can generate a t1 point for a 2D J resolved spectrum. It is illustrated for both an isolated spin and a coupled pair of spins. You can learn to use MAPLE by typing the red text on the worksheet into a blank MAPLE window or simply by placing the cursor at the end of any command line and hitting the enter key. If you delete the blue text following the command first, you will see it reappear on executing the command. If you make a mistake or get an error message, simply edit the line and repeat the enter key operation.

The following explains some of the operations in the worksheet:

restart – clears variables in Maple and starts with a clean sheet.

with(share): with(pof); – tells Maple to look in the share library and that we are going to use procedure definitions provided by Kanters in the pof library (ignore the error message you may get at this point – the library will work for our purposes). The ‘:’ suppressed output from the first command, the ‘;’ let the functions be listed.

step1:=spinsystem([X]); defines this to be a one spin system, defines the basis set for the computation and assigns the variable step1 to the equilibrium density matrix. Note that this is given in terms of the product operator I_zX.

step2:=ypulse(step1, {X}, Pi/2); applies rotation operators for a 90y pulse to the X spin represented by the density matrix step1. Note that the result is given in terms of the product operator I_zX.

The following steps show evolution under chemical shift for period tau, evolution under a 180x pulse, and evolution for tau. Note that each operates on the density matrix from the previous operation. At the end the density matrix is simply I_zX. Note that there has been no net evolution due to chemical shift although there was evolution at intermediate steps.

Next, the sequence is restarted for a two spin system, assuming they are coupled with coupling constant J. Note that the final result, step5, now contains evolution under the coupling constant J, but not chemical shift offsets from either spin. This is the basis of J resolved spectroscopy when tau is incremented as the indirect dimension in a 2D experiment.

Now open and examine the My_cosy worksheet. This describes the basic 90–t1–90–t2 COSY experiment. Operations are similar to those described above until step5 where the observe function generates evolution in a second time domain, t2. In step6 the evalc function returns the result of Fourier Transformation (in algebraic form) in both dimensions. Note that there are 32 terms offset by chemical shifts of A and X and by +/- J/2 in both dimensions. They describe line shapes (absorption or dispersion) in each dimension (v1 or v2). These are the auto and cross peaks expected for the AX spin system. The table definitions that follow set values for T2 relaxation times, chemical shifts and coupling constants to be used in subsequent plot functions. A couple of different plotting functions are illustrated in what follows. The variables in the plot functions that set boundaries of the regions to be plotted and the number of points to be used.