Electronic Supplementary Material (ESI) for Analyst.


# Robust NMR water signal suppression for demanding chemometric applications.

Juan A. Aguilar and Simon J. Kenwright

Dr Aguilar: Department of Chemistry. Durham University. South Road, Durham, DH1 3LE (UK).
j.a.aguilar@durham.ac.uk
Simon J. Kenwright current address: The Queen's College, University of Oxford, Ox1 4AW (UK).

This supplementary information section contains the Robust-5 code for Varian spectrometers.

```
#ifndef LINT
static char SCCSid[] = "@(#)Robust5";
#endif

/* J. A. Aguilar. Durham University (UK). 20/12/2012.
 J.a.aguilar@durham.ac.uk
This code is provided as a record of the pulse programmes used to obtain the spectra reported in this publication.  There is no warranty (implied or explicit) that it is optimal or bug-free.  Anyone using this code does so at their own risk.


gt2 - gradient duration (1 ms)
gzlvl1 - gradient power for the 1st solvent suppression echo
gzlvl2 - gradient power for the 2nd solvent suppression echo
A good gzlvl1/gzlvl2= 5.8. See below.
gstab  - gradient stabilization delay (1 ms)
pe - flag for perfect echo pulse. Default="y"
d3 - delay in WATERGATE. Position of the secondary notches=1/d3 from the centre of the spectrum.
alt_grd. Flag to alternate the pulsed field gradients every other scan. Default="y". This seems to improve the suppression slightly. To determine whether your spectrometer benefits from this option, just run two experiments setting alt_grd="y","n"

Typical conditions as used in the manuscript:
The phase cycle requires 32 scans and that is what we recommend, but often a minimum of 8 produces good results. Always set the number of scans to a multiple of two.
If very high levels of suppression are sought, it is important to optimize the duration and strength of the pulsed field gradients, as well as their ratios. We have found that a ratio of 5.8 between the first and the second gradient pairs produces excellent results when 1 ms long pulses of 28.3 G cm-1 (first pair) are used. The optimum gradient stabilization delay varies among probes but 0.5 to 1 ms delays are often adequate. It was found that in order to minimize signal distortions the pre-focusing gradient pules and the first radio-frequency pulse should be separated by at least 1.5 ms, although this may vary among probes. Finally, alternating the polarity of the gradient pulses every other scan seems to slightly improve results. Again, the latter is probably probe dependent.
Because the W5 elements produces extra notches separated by 1/d3 Hz, the W5 interpulse delay (d3) was set to 240 µs when using the 600 MHz spectrometer, and to 287 µs when using the 700 MHz one.
*/

#include <standard.h>

pulsesequence()

{
static int          ph1[2]  = {0,2},
ph2[32] = {0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2,3,3,3,3,3,3,3,3},
ph3[32] = {2,2,2,2,2,2,2,2,3,3,3,3,3,3,3,3,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1},
ph4[1]  = {1},
ph5[8]  = {0,0,1,1,2,2,3,3},
```

```
ph6[8]  = {2,2,3,3,0,0,1,1},
rec[16] = {0,2,2,0,0,2,2,0,2,0,0,2,2,0,0,2},
ph9[2] = {0,1};

        double  gstab = getval("gstab"),
        gzlvl1 = getval("gzlvl1"),
        gt2 = getval("gt2"),
        gzlvl2 = getval("gzlvl2"),
        d3 = getval("d3"),
        ddrtc = getval("ddrtc");

        char    sspul[MAXSTR],
         pe[MAXSTR],
        alt_grd[MAXSTR];

        rof1 = getval("rof1"); if(rof1 > 2.0e-6) rof1 = 2.0e-6;
        getstr("pe",pe);
        getstr("sspul",sspul);
        getstr("alt_grd",alt_grd);

        settable(t1,2,ph1);
        settable(t2,32,ph2);
        settable(t3,32,ph3);
        settable(t4,1,ph4);
        settable(t5,8,ph5);
        settable(t6,8,ph6);
        settable(t7,16,rec);

        mod2(ct,v9);

status(A);

delay(rof1);
delay(d1-gt2*4.0-gstab*2.0);
obspower(tpwr);
obsoffset(tof);

if (alt_grd[0] == 'y')
{
ifzero(v9); rgradient('z',gzlvl1);
elsenz(v9); rgradient('z',-1.0*gzlvl1);
endif(v9);
}

else  rgradient('z',gzlvl1);
        delay(2.0*gt2);

if (alt_grd[0] == 'y')
```

```
{
ifzero(v9); rgradient('z',gzlvl2);
elsenz(v9); rgradient('z',-1.0*gzlvl2);
endif(v9);
}
else  rgradient('z',gzlvl2);
            delay(2.0*gt2);
            rgradient('z',0);
            delay(1.5e-3);


status(B);
            rgpulse(pw, t1, rof1, rof1);         /* 90d initial excitation pulse */

if (alt_grd[0] == 'y')
{
ifzero(v9); zgradpulse(-1.0*gzlvl1,gt2);
elsenz(v9); zgradpulse(gzlvl1,gt2);
endif(v9);
}

else zgradpulse(-1.0*gzlvl1,gt2);
      delay(gstab-rof1-rof1);
      rgpulse(pw*0.087,t2,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.087 + pw*0.206)/3.1416);
      rgpulse(pw*0.206,t2,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.206 + pw*0.413)/3.1416);
      rgpulse(pw*0.413,t2,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.413 + pw*0.778)/3.1416);
      rgpulse(pw*0.778,t2,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.778 + pw*1.491)/3.1416);
      rgpulse(pw*1.491,t2,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*1.491 + pw*1.491)/3.1416);
      rgpulse(pw*1.491,t3,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*1.491 + pw*0.778)/3.1416);
      rgpulse(pw*0.778,t3,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.778 + pw*0.413)/3.1416);
      rgpulse(pw*0.413,t3,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.413 + pw*0.206)/3.1416);
      rgpulse(pw*0.206,t3,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.206 + pw*0.087)/3.1416);
      rgpulse(pw*0.087,t3,rof1,rof1);

if (alt_grd[0] == 'y')

{
ifzero(v9); zgradpulse(-1.0*gzlvl1,gt2);
elsenz(v9); zgradpulse(gzlvl1,gt2);
```

```
endif(v9);
}

else zgradpulse(-1.0*gzlvl1,gt2);
      delay(gstab-rof1);

 if (pe[A] == 'y')                    /* Use perfect echo pulse */
  {
           rgpulse(pw,t4,rof1,rof1);           /* 90d refocusing pulse */
  }

if (alt_grd[0] == 'y')
{
ifzero(v9); zgradpulse(-1.0*gzlvl2,gt2);
elsenz(v9); zgradpulse(gzlvl2,gt2);
endif(v9);
}

else zgradpulse(-1.0*gzlvl2,gt2);


/* Set up hard second WATERGATE */

      delay(gstab-rof1-rof1);
      rgpulse(pw*0.087,t5,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.087 + pw*0.206)/3.1416);
      rgpulse(pw*0.206,t5,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.206 + pw*0.413)/3.1416);
      rgpulse(pw*0.413,t5,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.413 + pw*0.778)/3.1416);
      rgpulse(pw*0.778,t5,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.778 + pw*1.491)/3.1416);
      rgpulse(pw*1.491,t5,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*1.491 + pw*1.491)/3.1416);
      rgpulse(pw*1.491,t6,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*1.491 + pw*0.778)/3.1416);
      rgpulse(pw*0.778,t6,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.778 + pw*0.413)/3.1416);
      rgpulse(pw*0.413,t6,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.413 + pw*0.206)/3.1416);
      rgpulse(pw*0.206,t6,rof1,rof1);
      delay(d3-rof1-rof1-2*(pw*0.206 + pw*0.087)/3.1416);
      rgpulse(pw*0.087,t6,rof1,rof2);


if (alt_grd[0] == 'y')

{
```

```
ifzero(v9); zgradpulse(-1.0*gzlvl2,gt2);
elsenz(v9); zgradpulse(gzlvl2,gt2);
endif(v9);
}

else zgradpulse(-1.0*gzlvl2,gt2);
     delay(gstab-rof2+ddrtc);

setreceiver(t7);
status(C);
}
```