Matrix-Assisted Diffusion-Ordered NMR Spectroscopy with an Invisible Matrix: a Vanishing Surfactant

Electronic Supplementary Information

Robert Evans[a], Aaron Hernandez-Cid[b], Guilherme Dal Poggetto[b], Ashley Vesty[b], Stephan Haiber[c], Gareth A. Morris[b] and Mathias Nilsson[b].

Contents

---

1. Use of CTAB in matrix-assisted DOSY

The use of CTAB in matrix-assisted DOSY experiments has already been demonstrated in a number of articles. A further demonstration is shown in Figure SI.1, below, where a test mixture of the isomers of dihydroxybenzene (hydroquinone, resorcinol and catechol, signals labelled H, R and C respectively in Fig. SI.1) is partially resolved by the addition of 150 mM CTAB, with no co-solutes present. These signals are far from the alkyl region, so there is no overlap with surfactant signals. As expected, the more hydrophobic isomers, resorcinol and catechol, have lower diffusion coefficients in the micellar solution, reflecting increased association with the (normal, spherical) micelles formed.

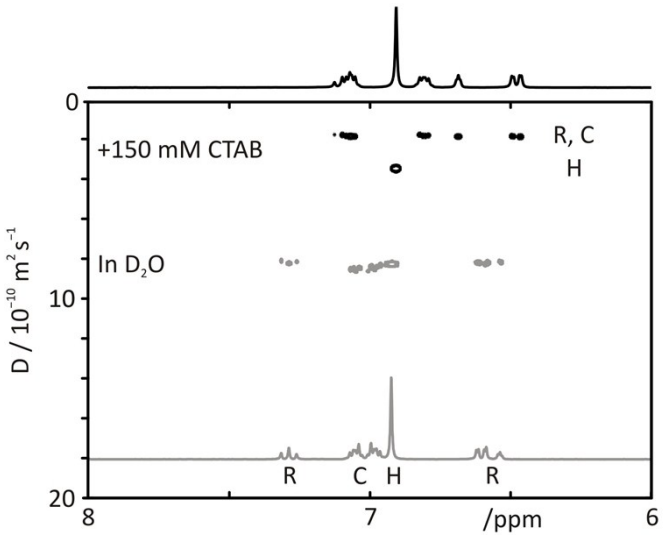


Figure SI.1: Combined proton and Oneshot DOSY spectra of a mixtures containing 10 mM of each dihydroxybenzene in D_2O (grey spectra) and in D_2O and 150 mM CTAB (black spectra). Signals due to hydroquinone, resorcinol and catechol are labelled H, R and C respectively.

2. Effect of catechol and buffer solution on CTAB micelles

In order to quantify the effect of catechol on CTAB micelles, increasing concentrations of catechol were added to CTAB solutions in D_2O. Both T_2 and diffusion measurements were acquired alongside standard proton spectra.

^1H spectra with catechol added

Figure SI.2A shows the effect of increasing catechol concentration on the ^1H spectra on CTAB samples in the absence of phosphate buffer. Very little effect is observed until 10 mM of catechol is present. Above this concentration, all of the CTAB signals broaden.

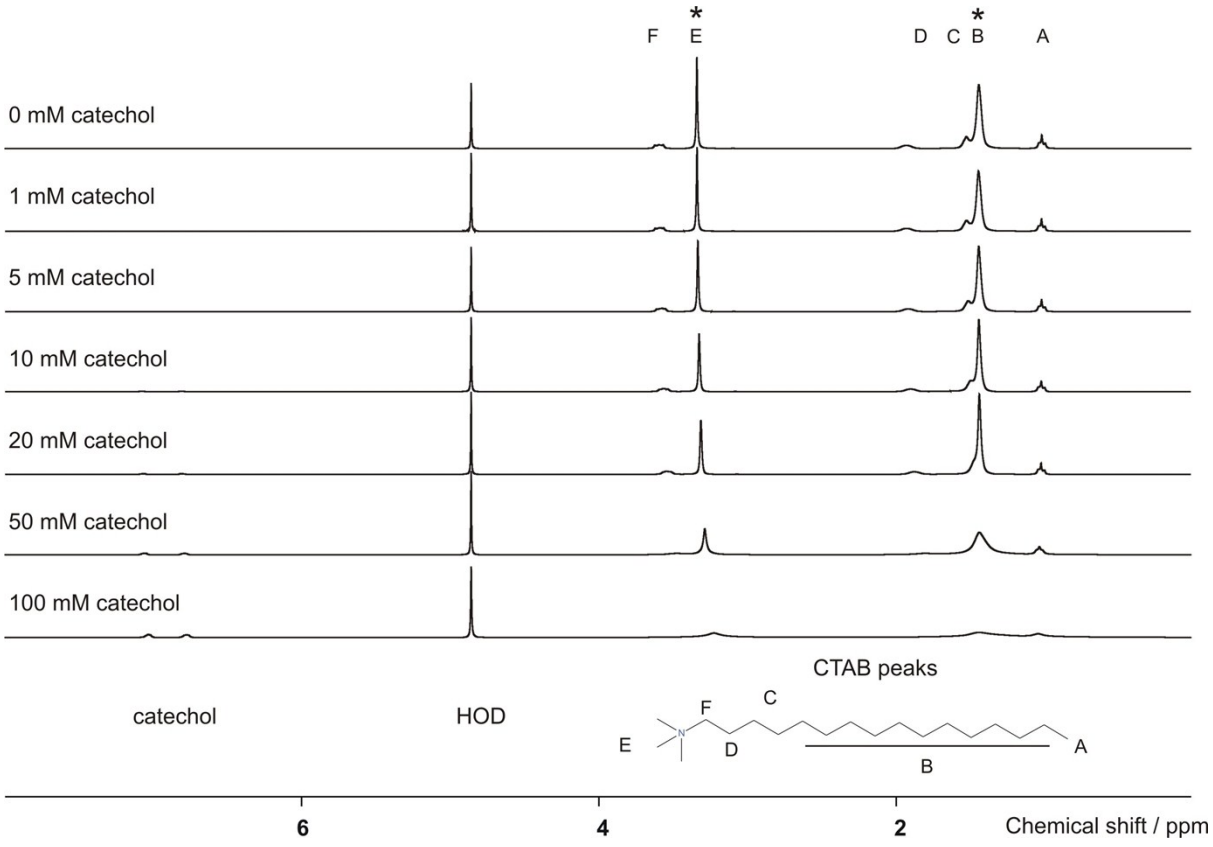


Figure SI.2A: Stacked proton NMR spectra of 150 mM CTAB and increasing concentrations of catechol as detailed in the figure.

Figure SI.2B shows the effect of increasing catechol concentration on the ^{1}H spectra of CTAB samples in the presence of phosphate buffer. Addition of the buffer solution has an immediate effect on the lineshapes and linewidths of the CTAB peaks, and the additional of catechol broadens them further.

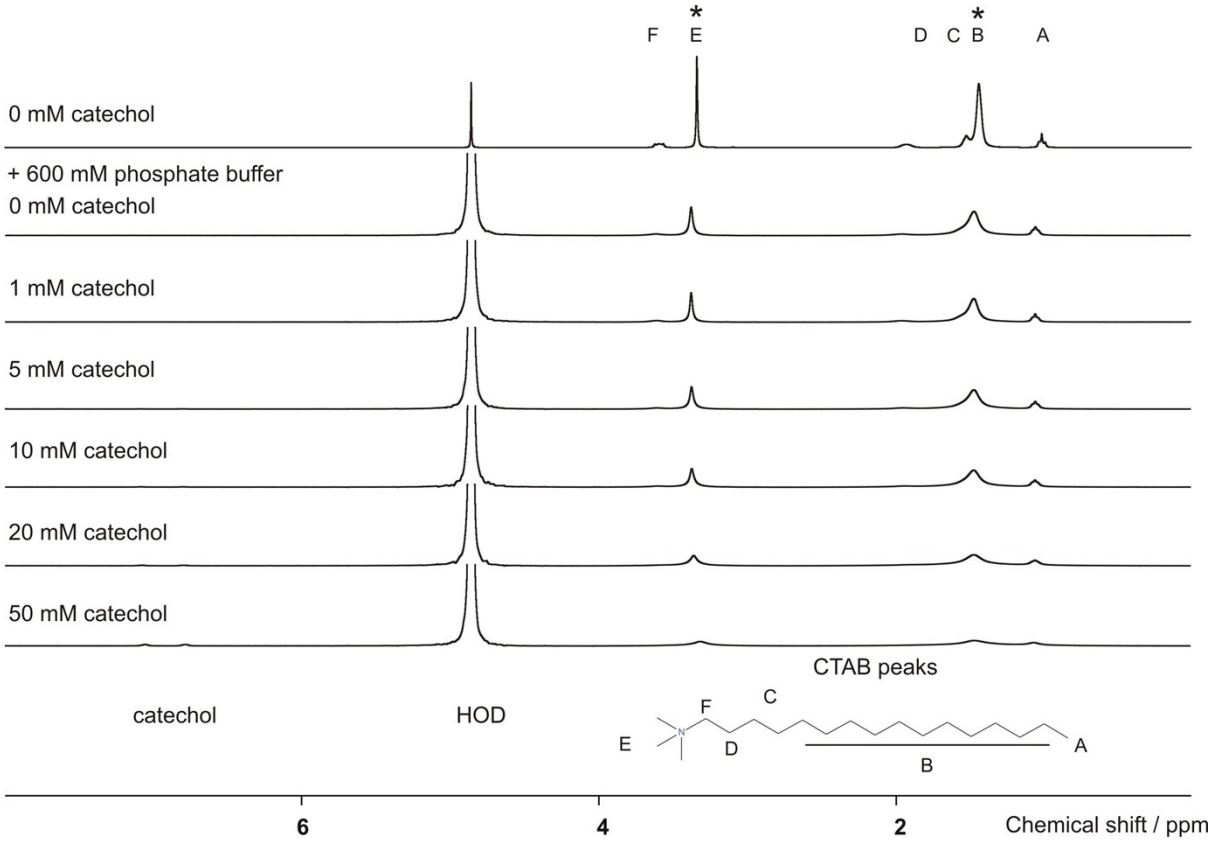


Figure SI.2B: Stacked proton NMR spectra of 150 mM CTAB and increasing concentrations of catechol as detailed in the figure, in solutions containing 600 mM total concentration of PBS buffer.

Relaxation and Diffusion Measurements

The transverse relaxation times, T_2, of the species were measured using the PROJECT sequence. The addition of catechol was observed to reduce T_2 from hundreds to tens of ms.

This decrease in relaxation time was accompanied by a large increase in apparent hydrodynamic radius, estimated from measured CTAB diffusion coefficients using the Stokes-Einstein equation. CTAB micelles formed in the absence of any extra species have been reported as being typically around 4 nm in size, consistent with the measurements obtained at low concentrations of catechol.

The addition of 600 mM PBS buffer solution had an immediate and large effect on the relaxation times. Further addition of catechol rapidly reduced the relaxation times below those observed in the absence of PBS. There was also a large increase in the measured size of the micelles. Figure SI.3E compares the relaxation times of CTAB in the absence and in the presence of added PBS buffer solution.

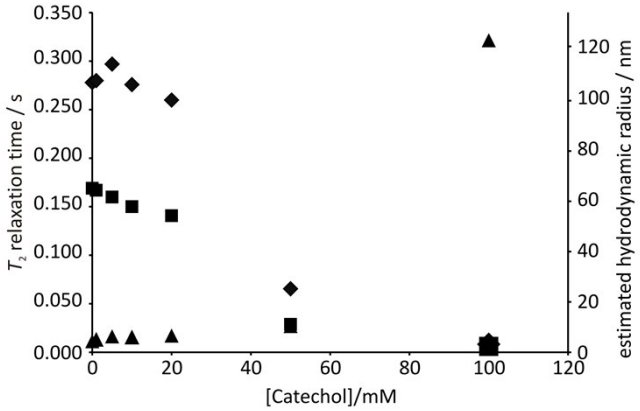


Figure SI.2C: Plot of transverse relaxation times and estimated hydrodynamic radii of micelles as a function of concentration of added catechol for CTAB micelles in D_2O. Filled diamond and square markers indicate transverse relaxation times of peaks at 3.34 and 1.4 ppm respectively. Filled triangles indicate estimated hydrodynamic radii of the micelles present.

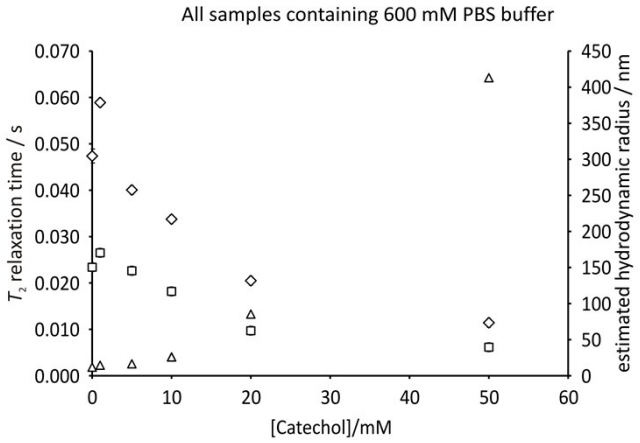


Figure SI.2D: Plot of transverse relaxation times and diffusion coefficients as a function of concentration of added catechol for CTAB micelles in PBS in H_2O/D_2O. Open diamond and square markers indicate transverse relaxation times of peaks at 3.34 and 1.4 ppm respectively. Open triangles indicate estimated hydrodynamic radii of the micelles present.

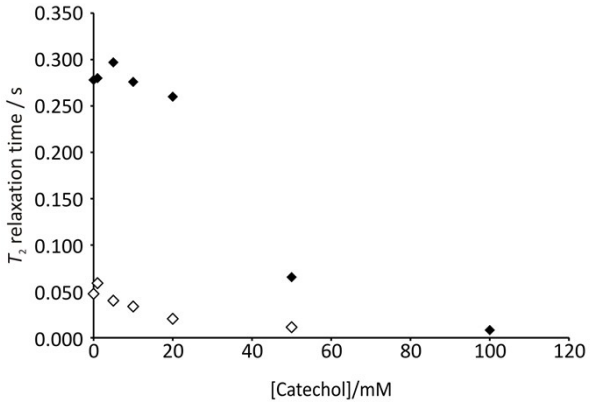


Figure SI.2E: Plot of transverse relaxation times of CTAB micelles as a function of concentration of added catechol for CTAB micelles in D_2O in the absence (filled) and in the presence (open) of added PBS buffer solution.

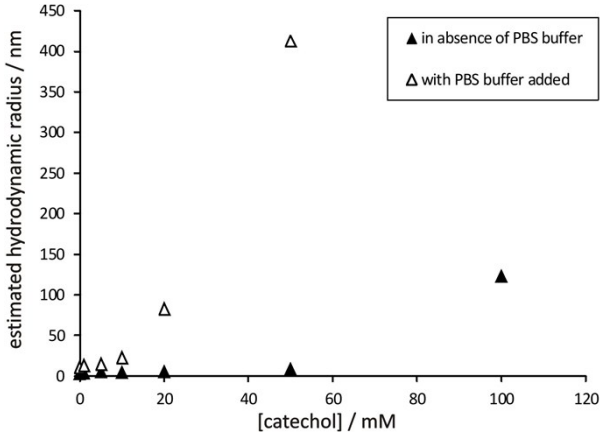
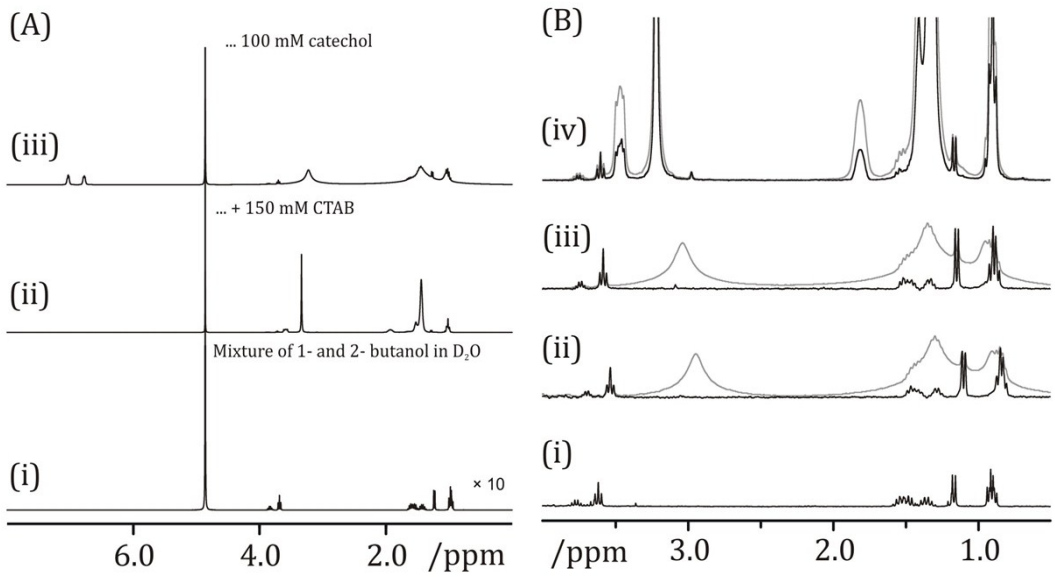


Figure SI.2F: Plot of estimated hydrodynamic radii of CTAB micelles as a function of concentration of added catechol for CTAB micelles in D_2O in the absence (filled) and in the presence (open) of added PBS buffer solution.

3. Overlap of butanol and CTAB signals, and filtration of the latter

Diffusion experiments can be rendered uninformative by overlap between NMR signals. Figure SI.3A shows that there is significant overlap between the peaks of butan-1-ol and butan-2-ol and CTAB at both high and low concentrations of catechol. The T_2 relaxation time of CTAB, in the absence of any additives, is not nearly short enough for a total spin echo time of 20 ms to attenuate it by more than 50 %. However, addition of catechol and buffer solution, as in Figure SI.3B.ii and iii causes large micelles to form and the signals of these can be effectively and completely removed by appropriate choice of experimental parameters in the diffusion experiments.

Figure SI.3C shows the effectiveness of the filtration for the system presented in the paper. There is significant overlap between the peaks of the butanol isomers with CTAB in D_2O. In a saturated NaCl sample, all of the CTAB peaks are reduced in size, but there is still overlap, particularly with the methyl peaks of butanol. The T_2 filter of the PROJECT-Oneshot sequence completely removes the CTAB signals from the spectrum.



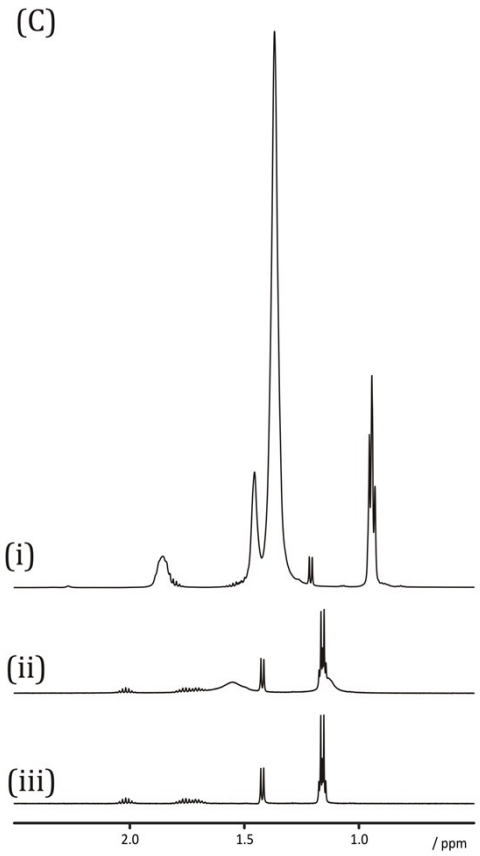


Figure SI.3: A Proton NMR spectra of a test mixture of 10 mM each of butan-1-ol and butan-2-ol (i) in the absence of CTAB (ii) in the presence of CTAB and (iii) in the presence of CTAB and 100 mM catechol. The spectrum of the sample with no additives has been magnified ten-fold to show the alcohol signals.

B: Spectra of mixtures containing butan-1-ol and butan-2-ol in D_2O in different CTAB solutions. Sample (i) contains only the alcohol mixture, (ii) the alcohol mixture plus an 'invisible matrix' solution of CTAB, 100 mM catechol and PBS, (iii) the alcohol mixture, CTAB and 100 mM catechol and (iv) the alcohol mixture and CTAB only. Black spectra are of the first increment of the Oneshot45 spectra and the grey spectra are ^1H spectra of the two solutions.

C: ^1H NMR spectra of the first gradient level of the DOSY datasets used in Fig. 2 of the main paper, for a solution containing 10 mM each of 2-methylpropan-1-ol (isobutanol) and butan-2-ol (*sec*-butanol) in i) D_2O and 150 mM CTAB, ii) the 'invisible matrix' solution, containing 150 mM CTAB in a saturated solution of NaCl in D_2O, both acquired using the Oneshot pulse sequence; and iii) the 'invisible matrix' solution, containing 150 mM CTAB in a saturated solution of NaCl in D_2O, acquired using the PROJECT-Oneshot sequence.

4. Matrix-assisted DOSY using CTAB/catechol/PBS matrix

An 'invisible matrix' solution can be produced by using both additional salt and hydrotropes can also be used in matrix-assisted DOSY experiments. Figure SI.4 compares DOSY spectra of 10 mM each of butan-1-ol and butan-2-ol in (a) D_2O only and (b) a D_2O solution containing 150 mM CTAB, 200 mM catechol and 600 mM total concentration of PBS. The intermediate experiments, performed without T_2 filtration, are not included here. The main drawback with this approach is that the catechol signals found at 7 ppm are not removed by the filtration, i.e. the matrix is not completely invisible. The separation can be interpreted on the basis of the hydrophobicities of the species in the sample. butan-1-ol (log P = 0.84) interacts with the surfactant more strongly than butan-2-ol (log P = 0.68) and the effective diffusion coefficients observed in Fig. SI.4 reflect this.

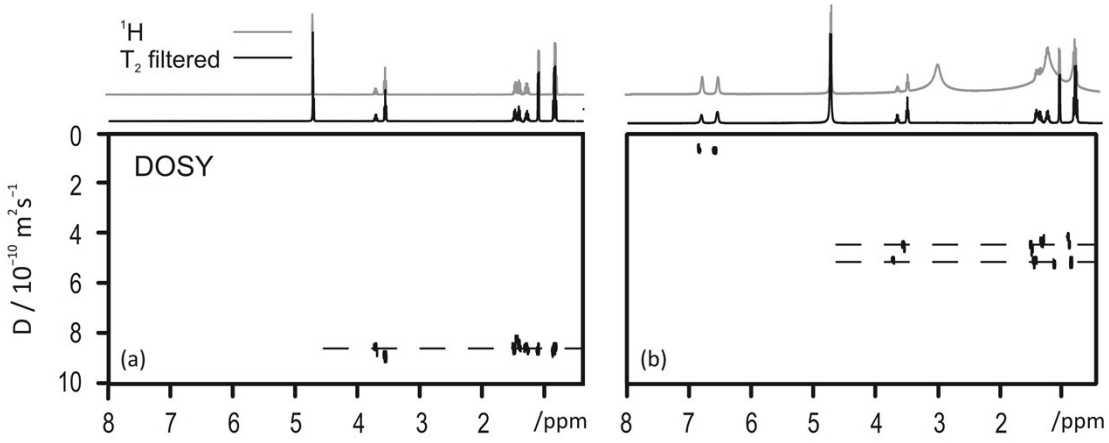


Figure SI.4: Combined proton, T_2-filtered Oneshot and DOSY spectra for a mixture containing 10 mM each of butan-1-ol and butan-2-ol in D_2O, without (a) and with (b) the addition of CTAB 'invisible matrix' solution consisting of 150 mM CTAB, 200 mM catechol and 600 mM total concentration of PBS. Black spectra are of the first increment of T_2 filtered Oneshot spectra and the grey spectra are ^{1}H spectra of the two solutions.

5. PROJECT-Oneshot Pulse Sequence

Sequence based on cpmg-dosy experiment (W. H. Otto and C. K. Larive, J. Magn. Reson., 2001, 153, 273 – 276) using Oneshot (Ref. 28) and PROJECT (Ref. 29).

```
#ifndef LINT
static char SSCCSid[] = "@(#)PROJECT_Doneshot.c 18-iii-2016";
#endif
/*******************************************************
PROJECT-Oneshot - This is the same layout of Oneshot DOSY sequence
but adding
a PROJECT element after the first 90 degree pulse. Oneshot is based
on modifications to the bipolar pulse stimulated echo (BPPSTE)
sequence.


Parameters:
        delflag   - 'y' runs the Doneshot sequence
                    'n' runs the normal s2pul sequence
        del       -  the actual diffusion delay
        gt1       - total diffusion-encoding pulse width
        gzlvl1    - diffusion-encoding pulse strength
        gstab     - gradient stabilization delay (~0.0002-0.0003
sec)
        gt3       - spoiling gradient duration (in sec)
        gzlvl3    - spoiling gradient strength (destroys transverse
                        magnetisation during the diffusion delay)
        gzlvl_max - maximun accepted gradient strength
                       32767 with PerformaII, 2047 with PerformaI
        kappa     - unbalancing factor between bipolar pulses as a
                      proportion of gradient strength (~0.2)
        pasecycleflag - flag to turn on and off the phase cycle
        fn2D      - Fourier number to up the 2D display in F2
        wet       - 'y' turns wet flag on
        satmode   - 'y' turns on presaturation
        Dtaub     - Tau delay between pulses in PROJECT element
The parameters for the heating gradients (gt4, gzlvl4) are
calculated
in the sequence. They cannot be set directly.
tau defined as time between the mid-points of the bipolar diffusion
encoding
gradient pulses


        Constant energy dissipation calculation corrected 8vii99

17xi08 GAM      Add warning if phasecycleflag='n'  !

15iii11   GAM/AC        Correct syntax for presat; remove spurious
else
                        and split gradient pulses either side of
satdly
```

```
*********************************************************/

#include <standard.h>

static int ph1[64] =
{0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2
,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2,0,2},
           ph2[64] =
{1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1
,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1},
           ph3[64] =
{1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3
,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3},
           ph4[64] =
{0,0,0,0,1,1,1,1,2,2,2,2,3,3,3,3,0,0,0,0,1,1,1,1,2,2,2,2,3,3,3,3,0,0
,0,0,1,1,1,1,2,2,2,2,3,3,3,3,0,0,0,0,1,1,1,1,2,2,2,2,3,3,3,3},
           ph5[64] =
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2
,2,2,2,2,2,2,2,2,2,2,2,2,2,2,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3},
           ph6[64] =
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0},
           ph7[64] =
{0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3
,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3,0,1,2,3},
          ph8[64] =
{0,0,0,0,2,2,2,2,0,0,0,0,2,2,2,2,1,1,1,1,3,3,3,3,1,1,1,1,3,3,3,3,2,2
,2,2,0,0,0,0,2,2,2,2,0,0,0,0,3,3,3,3,1,1,1,1,3,3,3,3,1,1,1,1};


pulsesequence()
{

double  kappa     = getval("kappa"),
        gzlvl1    = getval("gzlvl1"),
        gzlvl3    = getval("gzlvl3"),
        gzlvl_max = getval("gzlvl_max"),
        gt1       = getval("gt1"),
        gt3       = getval("gt3"),
        del       = getval("del"),
        gstab     = getval("gstab"),
        gzlvl4,gt4,Dtau,Ddelta,dosytimecubed, dosyfrq,
        satpwr = getval("satpwr"),
        satdly = getval("satdly"),
        satfrq = getval("satfrq"),
        Dtaub,     /*time interval between pulses in PROJECT*/
        looptime = getval("looptime"); /*time of one single loop*/

int     cycles = getval("cycles"), /*number of CPMG in PROJECT
element*/
```

```
     AHCph = getval("AHCph"); /*Defines the number of step cycles
to use*/


 /*ph1 = (0 2)32, ph2 = [1]64, ph3 = (1 3)32, ph4 = (0 1 2 3)16, ph5
= [0]16,[1]16,[2]16,[3]16, ph6 = [0]64, ph7 = (0 1 2 3)16 */

char delflag[MAXSTR],phasecycleflag[MAXSTR],
wet[MAXSTR],satmode[MAXSTR];
     gt4 = 2.0*gt1;
     getstr("delflag",delflag);
     getstr("wet",wet);
     getstr("satmode",satmode);

/* Decrement gzlvl4 as gzlvl1 is incremented, to ensure constant
   energy dissipation in the gradient coil
   Current through the gradient coil is proportional to gzlvl */

   gzlvl4 = sqrt(2.0*gt1*(1+3.0*kappa*kappa)/gt4) *
            sqrt(gzlvl_max*gzlvl_max/((1+kappa)*(1+kappa))-
gzlvl1*gzlvl1);

/* In pulse sequence, del>4.0*pw+3*rof1+2.0*gt1+5.0*gstab+gt3 */

   if ((del-(4*pw+3.0*rof1+2.0*gt1+5.0*gstab+gt3)) < 0.0)
   {  del=(4*pw+3.0*rof1+2.0*gt1+5.0*gstab+gt3);
      text_message("Warning: del too short; reset to minimum
value");
   }

   if ((d1 - (gt3+gstab) -2.0*(gt4/2.0+gstab)) < 0.0)
   {  d1 = (gt3+gstab) -2.0*(gt4/2.0+gstab);
      text_message("Warning: d1 too short;  reset to minimum
value");
   }

   if ((gzlvl1*(1+kappa)) > gzlvl_max)
   {  abort_message("Max. grad. amplitude exceeded: reduce either
gzlvl1 or kappa");
   }

   if (ni > 0.0)
   {  abort_message("This is a 2D, not a 3D dosy sequence; please
set ni to zero");
   }

   Dtaub=((looptime-5.0*pw)/4.0);
   Ddelta=gt1;
   Dtau=2.0*pw+gstab+gt1/2.0+rof1;
   dosyfrq = sfrq;
   dosytimecubed = Ddelta*Ddelta*(del+(Ddelta/6.0)*(kappa*kappa-
2.0)+(Dtau/2.0)*(kappa*kappa-1.0));
   putCmd("makedosyparams(%e,%e)\n",dosytimecubed,dosyfrq);
   initval(cycles,v20);
```

```
/* phase cycling calculation */

   if (AHCph >= 64)
   {
   settable(t1,64,ph1);
   settable(t2,64,ph2);
   settable(t3,64,ph3);
   settable(t4,64,ph4);
   settable(t5,64,ph5);
   settable(t6,64,ph6);
   settable(t7,64,ph7);
   settable(t8,64,ph8);
   }

   if (AHCph==32)
   {
   settable(t1,32,ph1);
   settable(t2,32,ph2);
   settable(t3,32,ph3);
   settable(t4,32,ph4);
   settable(t5,32,ph5);
   settable(t6,32,ph6);
   settable(t7,32,ph7);
   settable(t8,32,ph8);
   }

   if (AHCph==16)
   {
   settable(t1,16,ph1);
   settable(t2,16,ph2);
   settable(t3,16,ph3);
   settable(t4,16,ph4);
   settable(t5,16,ph5);
   settable(t6,16,ph6);
   settable(t7,16,ph7);
   settable(t8,16,ph8);
   }

   if (AHCph==8)
   {
   settable(t1,8,ph1);
   settable(t2,8,ph2);
   settable(t3,8,ph3);
   settable(t4,8,ph4);
   settable(t5,8,ph5);
   settable(t6,8,ph6);
   settable(t7,8,ph7);
   settable(t8,8,ph8);
   }
```

```
   if (AHCph<=4)
   {
   settable(t1,4,ph1);
   settable(t2,4,ph2);
   settable(t3,4,ph3);
   settable(t4,4,ph4);
   settable(t5,4,ph5);
   settable(t6,4,ph6);
   settable(t7,4,ph7);
   settable(t8,4,ph8);
   }

   getelem(t1,ct,v1);
   getelem(t2,ct,v2);
   getelem(t3,ct,v3);
   getelem(t4,ct,v4);
   getelem(t5,ct,v5);
   getelem(t6,ct,v6);
   getelem(t7,ct,v7);
   getelem(t8,ct,oph);              /*Receiver phase*/

status(A);
   delay(rof1);
   zgradpulse(-1.0*gzlvl4,gt4/2.0);     /* 1st dummy heating pulse
*/
   delay(gstab);
   zgradpulse(gzlvl4,gt4/2.0);          /* 2nd dummy heating pulse
*/
   delay(gstab);

if (satmode[0] == 'y')
    {
    if (d1-satdly > 0)
       delay(d1 - satdly);
    else
    delay(0.02);
    obspower(satpwr);
     if (satfrq != tof)
      obsoffset(tof);
     rgpulse(satdly,zero,rof1,rof1);
     if (satfrq != tof)
      obsoffset(tof);
    obspower(tpwr);
    delay(1.0e-5);
    }
    else
    { delay(d1); }

   zgradpulse(-1.0*gzlvl3,gt3);         /* Spoiler gradient
balancing pulse */
   delay(gstab);
```

```
status(B);                              /* first part of sequence */

   rgpulse(pw, v1, rof1, 0.0);          /* first 90, v1 */

   starthardloop(v20);                  /*Loops start here*/

   delay(Dtaub-rof1);

   rgpulse(2*pw,v2,rof1,rof1);           /*1st 180 degree pulse in
PROJECT, v2*/

   delay(Dtaub-2*rof1);

   rgpulse(pw,v3,rof1,rof1);         /*90 degree pulse in PROJECT,
v3*/

   delay(Dtaub-2*rof1);

   rgpulse(2*pw,v2,rof1,rof1);          /*2nd 180 degree pulse in
PROJECT, v2*/

   delay(Dtaub-rof1);

   endhardloop();                                        /*loops
stop here*/

   zgradpulse(gzlvl1*(1.0-kappa),gt1/2.0);               /*1st main
gradient pulse*/
   delay(gstab);
   rgpulse(pw*2.0, v4, rof1, 0.0);                      /* first
180, v4 */

   zgradpulse(-1.0*gzlvl1*(1.0+kappa),gt1/2.0);          /*2nd main
grad. pulse*/
   delay(gstab);
   rgpulse(pw, v5, rof1, 0.0);                           /* second
90, v5 */

   zgradpulse(gzlvl1*2.0*kappa,gt1/2.0);               /* Lock
refocussing pulse*/
   delay(gstab);

   zgradpulse(gzlvl3,gt3);                            /* Spoiler
gradient balancing pulse */
   delay(gstab);

   delay(del-4.0*pw-3.0*rof1-2.0*gt1-5.0*gstab-gt3); /* diffusion
delay */

   zgradpulse(2.0*kappa*gzlvl1,gt1/2.0);             /*Lock
refocussing pulse*/
   delay(gstab);
```

```
   rgpulse(pw, v6, rof1, 0.0);                         /* third 90, v6
*/

   zgradpulse(gzlvl1*(1.0-kappa),gt1/2.0);           /*3rd main
gradient pulse*/
   delay(gstab);
   rgpulse(pw*2.0, v7, rof1, rof2);                 /* second 180,
v7 */

   zgradpulse(-1.0*(1.0+kappa)*gzlvl1,gt1/2.0); /*4th main grad.
pulse*/
   delay(gstab);

status(C);

}
```