Supporting Information

Exploring methods to expedite the recording of CEST datasets using selective pulse excitation

Tairan Yuwen,[1] Guillaume Bouvignies,[2] and Lewis E. Kay[1,3]

[1]Departments of Molecular Genetics, Biochemistry and Chemistry, University of Toronto, Toronto, Ontario M5S 1A8, Canada

[2]Laboratoire des biomolécules, LBM, Département de chimie, École normale supérieure, PSL University, Sorbonne Université, CNRS, 75005 Paris, France

[3]Hospital for Sick Children, Program in Molecular Medicine, 555 University Avenue, Toronto, Ontario M5G 1X8, Canada

Table of Contents

Supplementary Figures

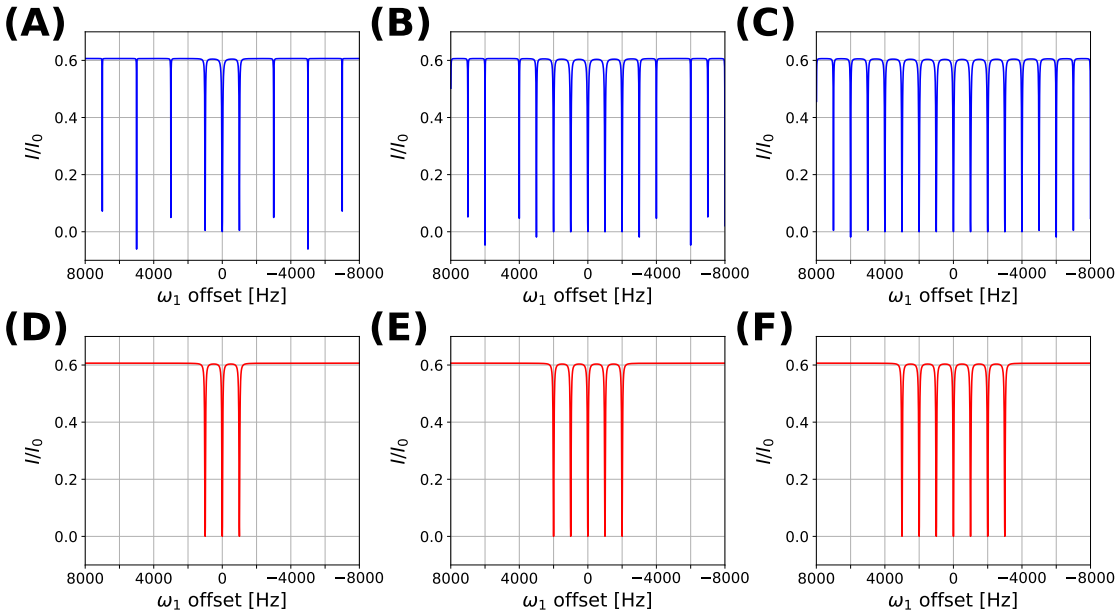

Fig. S1. Numerical simulations comparing D-CEST (A–C) and cos-CEST (D–F) schemes. Simulation are for a single-spin system and the following parameters are used: $R_1 = 1\ s^{-1}$, $R_2 = 10\ s^{-1}$, $B_1^{CEST} = 10$ Hz, $sw_{CEST} = 1000$ Hz, $T_{Ex} = 500$ ms, and *rf* inhomogeneity included as described previously [1]. The ground state is located at 0 Hz and exchange effects are not included in the simulations. Panel (A–C) are simulated for D-CEST with $B_1^{D\text{-}CEST} = 20$, 50 and 100 Hz, respectively. Panel (D–F) are simulated for cos-CEST with $N = 3$, 5 and 7, respectively. It is clear that in the D-CEST scheme it is impossible to restrict the excitation region by decreasing $B_1^{D\text{-}CEST}$.

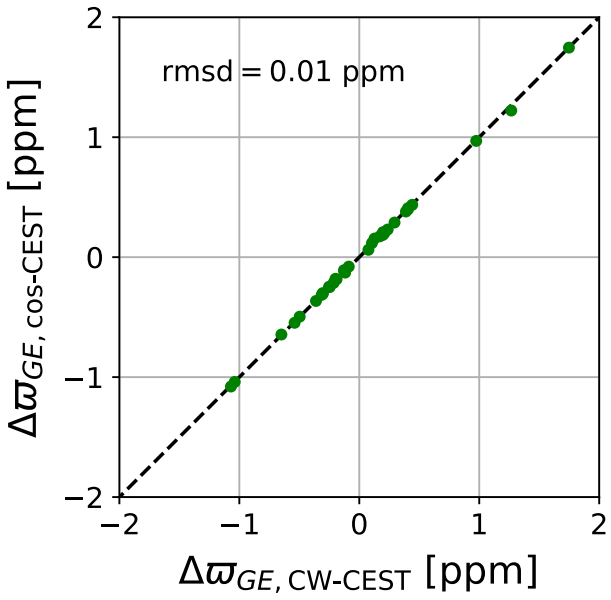

Fig. S2. Correlation of $\Delta\varpi_{GE}$ obtained with CW-CEST and cos-CEST schemes on a fully protonated apoSOD1^{2SH} sample. Values from CW-CEST have been obtained previously [2]. In total 38 residues that have clearly discerned excited state dips in both experiments are included in the comparison. Note that, in general, for cos-CEST it is necessary to measure two separates datasets with different sw_{CEST} in order to obtain $\Delta\varpi_{GE}$ unambiguously; in the present case sw_{CEST} = 1200 and 1260 Hz were obtained.

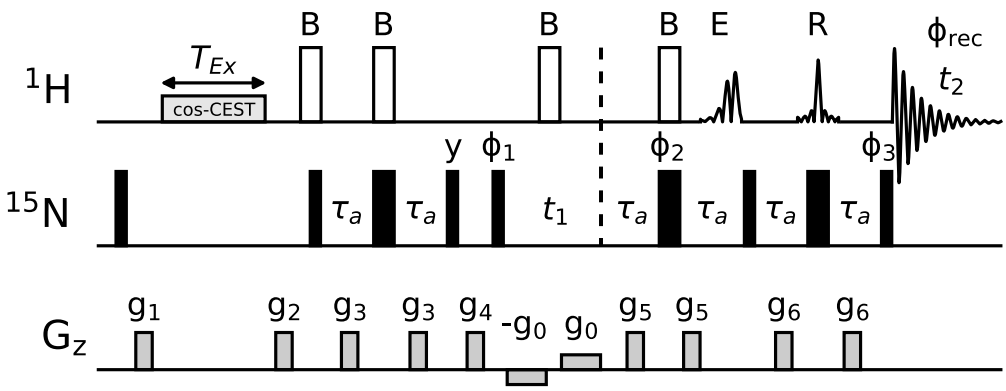

Fig. S3. IPAP-based L-optimized ^{1}H-CEST (BEST-CEST) experiment for the study of slow chemical exchange processes in fully protonated proteins, with an NH_2 filter to suppress side-chain signals from Asn/Gln. Details are as for Fig. 3. Gradients are applied with the following durations (ms) and strengths (% maximum): g_0: 0.25%, g_1: (0.4, −25%), g_2: (1.0, 15%), g_3: (1.0, −60%), g_4: (1.5, −70%), g_5: (1.0, 60%), g_6: (0.8, 80%). For small to moderately sized proteins ($<$ 15 kDa) the signal intensity is typically 20–30% lower than the version without the NH_2 filter, which is due to additional transfer steps converting anti-phase ^{15}N magnetization to in-phase and back.

References

[1] P. Vallurupalli, G. Bouvignies, L. E. Kay, Studying "invisible" excited protein states in slow exchange with a major state conformation, J. Am. Chem. Soc. 134 (2012) 8148–8161.

[2] T. Yuwen, L. E. Kay, Longitudinal relaxation optimized amide ^{1}H-CEST experiments for studying slow chemical exchange processes in fully protonated proteins, J. Biomol. NMR 67 (2017) 295–307.

```
#!/bin/csh

# Define variables
@ ni = 64
@ nfrq = 42
@ yT = $ni * $nfrq * 2
@ yN = $yT * 2
set frq_array = `seq 1 $nfrq`
set jhn_half = 46.3
set hi_cutoff = 5e6

# Clean the directory first
rm -rf test_*.fid data_* ft_* ft sim dif

# FID format conversion for IP and AP separately
bruk2pipe -in ./ser \
  -bad 0.0 -ext -aswap -AMX -decim 1664 -dspfvs 20 -grpdly 67.9841613769531  \
  -xN              1536  -yN               $yN  \
  -xT               768  -yT               $yT  \
  -xMODE            DQD  -yMODE        Complex  \
  -xSW        12019.231  -ySW         2108.815  \
  -xOBS         800.304  -yOBS          81.093  \
  -xCAR           4.662  -yCAR         119.000  \
  -xLAB              HN  -yLAB            15N  \
  -ndim               2  -aq2D          States  \
| nmrPipe -fn COADD -cList 1 1 -axis y -time    \
  -out ./test_1.fid -verb -ov

bruk2pipe -in ./ser \
  -bad 0.0 -ext -aswap -AMX -decim 1664 -dspfvs 20 -grpdly 67.9841613769531  \
  -xN              1536  -yN               $yN  \
  -xT               768  -yT               $yT  \
  -xMODE            DQD  -yMODE        Complex  \
  -xSW        12019.231  -ySW         2108.815  \
  -xOBS         800.304  -yOBS          81.093  \
  -xCAR           4.662  -yCAR         119.000  \
  -xLAB              HN  -yLAB            15N  \
  -ndim               2  -aq2D          States  \
| nmrPipe -fn COADD -cList 1 -1 -axis y -time   \
  -out ./test_2.fid -verb -ov

# Process series of 2D spectra for IP and AP separately
pseudo3D.com -in test_1.fid -outDir data_1 -tau $frq_array -time -inner
pseudo3D.com -in test_2.fid -outDir data_2 -tau $frq_array -time -inner

xyz2pipe -in ./data_1/test%03d.fid \
| nmrPipe -fn POLY -time -ord 2 -verb                       \
| nmrPipe -fn SP -off 0.45 -end 0.98 -pow 2 -c 0.5 -verb    \
| nmrPipe -fn ZF -zf 2 -auto -verb                         \
| nmrPipe -fn FT -verb                                     \
| nmrPipe -fn PS -p0 5.8 -p1 0.0 -di -verb                 \
| nmrPipe -fn EXT -x1 11.0ppm -xn 5.5ppm -sw -verb         \
| nmrPipe -fn TP -verb                                     \
| nmrPipe -fn LP -ord 16 -verb                             \
| nmrPipe -fn SP -off 0.45 -end 0.98 -pow 2 -c 1.0 -verb   \
| nmrPipe -fn ZF -zf 2 -auto -verb                         \
| nmrPipe -fn FT -verb                                     \
| nmrPipe -fn PS -p0 -90.0 -p1 180.0 -di -verb             \
| nmrPipe -fn TP -verb                                     \
| nmrPipe -fn POLY -auto -verb                             \
| pipe2xyz -ov -verb -out ./ft_1/test%03d.ft2

xyz2pipe -in ./data_2/test%03d.fid \
| nmrPipe -fn POLY -time -ord 2 -verb                       \
| nmrPipe -fn SP -off 0.45 -end 0.98 -pow 2 -c 0.5 -verb    \
| nmrPipe -fn ZF -zf 2 -auto -verb                         \
| nmrPipe -fn FT -verb                                     \
| nmrPipe -fn PS -p0 95.8 -p1 0.0 -di -verb                \
| nmrPipe -fn EXT -x1 11.0ppm -xn 5.5ppm -sw -verb         \
```

```
| nmrPipe -fn TP -verb                                          \
| nmrPipe -fn LP -ord 16 -verb                                  \
| nmrPipe -fn SP -off 0.45 -end 0.98 -pow 2 -c 1.0 -verb        \
| nmrPipe -fn ZF -zf 2 -auto -verb                              \
| nmrPipe -fn FT -verb                                          \
| nmrPipe -fn PS -p0 0.0 -p1 180.0 -di -verb                    \
| nmrPipe -fn TP -verb                                          \
| nmrPipe -fn POLY -auto -verb                                  \
| pipe2xyz -ov -verb -out ./ft_2/test%03d.ft2

# Combine and shift IP and AP spectra for IPAP virtual decoupling
mkdir ft_sum ft_dif ft

foreach n ($frq_array)
    set k = `printf "test%03d.ft2\n" $n`
    echo $k

    addNMR -in1 ft_1/$k -in2 ft_2/$k -out ft_sum/$k -add
    addNMR -in1 ft_1/$k -in2 ft_2/$k -out ft_dif/$k -sub

    nmrPipe -in ft_sum/$k \
    | nmrPipe -fn CS -ls "$jhn_half"Hz \
    | nmrPipe -out ft_sum_shift/$k -ov

    nmrPipe -in ft_dif/$k \
    | nmrPipe -fn CS -rs "$jhn_half"Hz \
    | nmrPipe -out ft_dif_shift/$k -ov

    addNMR -in1 ft_sum_shift/$k -in2 ft_dif_shift/$k -out ft/$k -add
end

# Fit relative peak intensities in the 2D series
peakHN.tcl -in ./ft/test001.ft2 -out ./test.tab -hi $hi_cutoff
autoFit.tcl -specName ./ft/test%03d.ft2 -series -inTab ./test.tab -outTab ./nlin.tab

# Extract autofit fitting results
getTabCol.tcl -in ./nlin.tab -var INDEX "Z*" > nlin_edit.tab
```

```
/* b_N15_1H_CEST_ipap_lek_800_cp

  Used for measurement of amide 1H CEST in small proteins so dont use TROSY

  This pulse sequence will allow one to perform the following experiment:
  1H CEST when magnetization is of the form Iz with 15N-1H HSQC readout.

  To callibrate B1 field: set -Dcal_HB1

  Min phase cycle is 2

  Use gen_fq1list.py to generate the frequency list
  Use gen_vclist.py to generate the vc list for B1 calibration
  Use gen_dante.py to generated cos-modulated shaped-DANTE pulses

  Notes: If an even number of excitation bands is used (l7%2==0) then after a time of
         2/swcest the waveform repeat. After each 1/swcest the waveform changes sign.
         This is taken into account by ipp25.

  USAGE:

         Generate cos-shaped pulse using gen_dante. Typically use an odd number of
          frequencies; This generates a waveform for 1/swcest duration that is then
          repeated via a loop counter to generate a long excitation of time_T1. It
          extends in 1 us steps from 0<=t<=1/swcest.


         Calibrate B1
         (0) Choose an isolated amide proton and set cnst1; Need to sit on a multiplet
             component. Best to record a 1D using this sequence which is a coupled
             spectrum and generate the frequency this way.
         (1) set d1 = 1 td3=1 td2 = number of time points td1 =1
         (2) unlike CW-CEST where calibration uses vd list, here a vc list is used
             gen_vclist : Generates a list of numbers for a loop counter that loops
                          over the cos-waveform n times. The duration of the
                          cos-waveform is 1/swcest = 1/cnst8. So if cnst8=1000 Hz
                          then each waveform is 1 ms. Using 0,10,20,30,...,300 will
                          go out to 300 ms in the time domain (31 points). Each dwell
                          is 10/swcest so the total spectral width in the spectrum
                          is swcest/10, 100 Hz = +/- 50 Hz.
         (3)  To run expt: set d1=0, td3=2 td2=number of frequency points td1 =
                           2*complex points.  Suppose you want N frequency components.
                           Then swcest = sw/N, where sw is the desired total range to
                           excite, when sweeping over swcest. If you want sw = 5 ppm
                           (from 11 to 6) = 4000 Hz the swcest=4000/3 Hz. Sweep from
                           +4000/6 to -4000/6 about the center of the amides, 8.5 ppm.
                           Use gen_fq1list 44 xx -B1Hz where B1Hz is the strength of
                           the weak B1 field (- sign as scan from down to up field)
                           and xx = (swcest/6 + (8.5-o1p)*800.20).

*/

;prosol relations=<triple>

#include <Avance.incl>
#include <Grad.incl>
#include <Delay.incl>

/***********************/
/*   Define phases     */
/***********************/
#define zero ph=0.0
#define one ph=90.0
#define two ph=180.0
#define three ph=270.0

/************************/
/*    Define pulses     */
/************************/
```

```
define pulse dly_pg1     /* Messerle purge pulse */
  "dly_pg1=5m"
define pulse dly_pg2     /* Messerle purge pulse */
  "dly_pg2=dly_pg1/1.62"
define pulse pwh
  "pwh=p1"               /* 1H hard pulse at power level pl1 (tpwr) */
define pulse pwn
  "pwn=p3"               /* PW90 for N pulse at power level pl3 (dhpwr2)  */

#ifndef cos_flg
  define pulse pwh_dante
    "pwh_dante=p15"
#endif

#ifdef c_flg
  define pulse pwc_ad
    "pwc_ad=p22"         /* PW180 for adiabatic pulse at pl22 (d_ad)   */
#endif

#ifdef N_sel
  define pulse pwn_sl
    "pwn_sl=p32"
#endif

/*********************/
/*    Define lists    */
/*********************/
#ifdef cal_HB1
  define list<loopcounter> ncyc_cal=<$VCLIST>
#else
#ifdef cal_HR1
  define list<delay> time_cal=<$VDLIST>
#else
  define list<frequency> H_offset=<$FQ1LIST>
#endif
#endif

/*******************************************************/
/* Assign cnsts to check validity of parameter range */
/*******************************************************/
#ifdef fsat
  "cnst10=plw10"         /* tsatpwr - set max at 0.00005W */
#endif

#ifdef mess_flg
  "cnst11=plw11"         /* tpwrmess - set max at 2.0W */
#endif

#ifdef c_flg
  "cnst22=spw22"         /* d_ad - set to 100 W */
#endif

#ifdef cal_HB1
  "cnst31=plw31"         /* dpwr2 - set max at 5.8W */
#endif

#ifdef N_sel
  "cnst32=spw32"         /* power level for 15N selective pulse */
#endif

#ifdef cal_HB1
#ifdef cos_flg
  "cnst51 = cnst1 + ((l7+1)%2)*cnst8*0.5" ; if even number must shift by swcest/2
#else
  "cnst51 = cnst1"
#endif
#endif

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

```
/* Calculate parameters for shaped pulses */
/****************************************/
#ifdef calc_SP
  /*  BIP 720,50,20.1 (p6, sp6)   */
  "p6=0.12/(bf1/1e6)"   /*  BIP pulse length   */
  "spw6=plw1*(pow((p1*2.0/p6)/0.25,2))"   /* BIP power level  */

  /*  PC9 (p7, sp7)   */
  "p7=7.2/(cnst6*bf1/1e6)"   /*  PC9  pulse length  */
  "spw7=plw1*(pow((p1/p7)/0.125,2))"   /* PC9  power level  */

  /*  REBURP (p8, sp8)   */
  "p8=4.875/(cnst6*bf1/1e6)"   /* REBURP pulse length  */
  "spw8=plw1*(pow((p1*2.0/p8)/0.07981,2))"   /* REBURP power level  */

  /*  EBURP & EBURP_TR  (p9, sp9, sp19)   */
  "p9=4.6/(cnst6*bf1/1e6)"   /*  EBURP pulse length   */
  "spw9=plw1*(pow((p1/p9)/0.06103,2))"   /* EBURP power level  */
  "spw19=plw1*(pow((p1/p9)/0.06103,2))"   /* EBURP_TR power level  */
#endif

/*************************************/
/*    Define shaped pulses on 1H     */
/*************************************/
define pulse pw_bip
  "pw_bip=p6"             /* PW180 for BIP pulses */
define pulse pw_pc9
  "pw_pc9=p7"             /* PW90 for PC9 pulses */
define pulse pw_reburp
  "pw_reburp=p8"          /* PW180 for REBURP pulses */
define pulse pw_eburp
  "pw_eburp=p9"           /* PW90 for EBURP pulses */
define pulse pw_eburptr
  "pw_eburptr=p9"         /* PW90 for EBURP_TR pulses */

/*************************/
/*    Define delays       */
/*************************/
define delay hscuba       /* length of 1/2 scuba delay */
  "hscuba=30m"
define delay taua         /* 1 / 4J(XH)                */
  "taua=d3"

#if !defined(cal_HB1) && !defined(cal_HR1)
  define delay time_T1
    "time_T1=d17"         /* CEST duration - typically 100-500 ms */

  define delay time_T1_adj

  "l8=(trunc((time_T1/1s)*cnst8+0.1))"

#ifdef eta_flg
  "l5=trunc(larger(l4,1)+0.2)"
  "l6=(trunc(l8/(l5*2)))"
  "time_T1_adj=1s*((l5*l6*2)/cnst8)"

  define delay taud
    "taud=d1-time_T1_adj"
  define delay taue
    "taue=1s*(l6/cnst8)"
#else /*eta_flg*/
  "time_T1_adj=1s*(l8/cnst8)"
#endif /*eta_flg*/
#endif

"d11=30m"
"in0=inf1/2"              /* t1/2 increment */

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

```
/* f1180: Start t1 at half dwell to get -90/180 phase correction */
/*        in F1 (15N) dim - set zgoptns -Df1180                  */
/*****************************************************************/
#if defined(f1180) && !defined(cal_HB1)
  "d0=(in0)/2"
#else
  "d0=0.2u"
#endif

/***********************************/
/* Calculate parameters for D-CEST */
/***********************************/
#ifdef cos_flg
  define pulse p_dante
    "p_dante = 1s/cnst8 - 0.1u"

  "spw5=pow((4.0*pwh*l7*cnst9)/1s,2)*plw1"
#else
  define pulse p_dante
    "p_dante = 4*pwh_dante*cnst9/cnst8"

  define delay d_dante
    "d_dante = 1s/cnst8 - p_dante"
#endif

/************************/
/* Initialize variables */
/************************/
"l3=1"
"plw2=0"
"spoal5=0.5"
"spoff5=0"
"spoal6=0.5"
"spoff6=0"
"spoal7=0.5"
"spoff7=bf1*(cnst5/1e6)-o1"
"spoal8=0.5"
"spoff8=bf1*(cnst5/1e6)-o1"
"spoal9=1"
"spoff9=bf1*(cnst5/1e6)-o1"
"spoal19=0"
"spoff19=bf1*(cnst5/1e6)-o1"
"spoal22=0.5"
"spoff22=0"

;"acqt0=0"
;baseopt_echo

1 ze
/*****************************************/
/*   Check Validity of Parameter Range   */
/*****************************************/
#ifdef fsat
  if "cnst10 > 0.00005" {
    2u
    print "error:tsatpwr pl10 too large; < 0.00005W !!!"
    goto HaltAcqu
  }
#endif

#ifdef mess_flg
  if "cnst11 > 2.0" {
    2u
    print "error:tpwrmess pl11 too large; < 2W !!!"
    goto HaltAcqu
  }
#endif

#ifdef c_flg
```

```
  if "cnst22 > 45.0" {
    2u
    print "error:don't fry the probe, spw22 dpwr too large; < 45W !!!"
    goto HaltAcqu
  }
#endif

#ifdef N_sel
  if "cnst32 > 60" {
    2u
    print "error: spw32 power is too high; < 35W !!!"
    goto HaltAcqu
  }
#endif

#if !defined(cal_HB1) && !defined(cal_HR1)
  if "time_T1 > 1.0s" {
    2u
    print "error: time_T1 is too long; < 1.0s !!!"
    goto HaltAcqu
  }

#ifdef eta_flg
  if "time_T1 > d1" {
    2u
    print "error: d1 should be longer than time_T1 !!!"
    goto HaltAcqu
  }
#endif
#endif

#ifdef cal_HB1
  if "cnst31 > 9" {
    2u
    print "error:don't fry the probe, pl31 dpwr2 too large; < 9W !!!"
    goto HaltAcqu
  }

  if "d1 < 1s" {
    2u
    print "error: d1 is too short; > 1s !!!"
    goto HaltAcqu
  }

  if "aq > 64m" {
    2u
    print "error: aq is too long; < 64ms !!!"
    goto HaltAcqu
  }
#endif

#ifdef cal_HB1
2 d11 do:f3
#else
2 d11
#endif
  2u rpp25

/*****************************************/
/* Continue to check real time variables */
/*****************************************/
#ifdef cal_HB1
  if "ncyc_cal/cnst8 > 1.8" {
    2u
    print "error: time_cal during B1 calib is too long !!!"
    goto HaltAcqu
  }
#endif
```

```
/*************************************************/
/*   Presaturation Period                        */
/*   option for Messrlie purge zgoptn -Dmess_flg  */
/*************************************************/
#ifdef mess_flg
  4u pl11:f1
  (dly_pg1 ph26):f1
  2u
  (dly_pg2 ph27):f1
#endif

#ifdef fsat
  4u pl10:f1
#if !defined(eta_flg) || defined(cal_HB1) || defined(cal_HR1)
  d1 cw:f1 ph26
#else
  taud cw:f1 ph26
#endif
  2u do:f1
#ifdef fscuba
  hscuba
  (pwh ph26):f1
  (pwh*2 ph27):f1
  (pwh ph26):f1
  hscuba
#endif /*fscuba*/
#else
#if !defined(eta_flg) || defined(cal_HB1) || defined(cal_HR1)
  d1
#else
  taud
#endif
#endif /*fsat*/

  20u UNBLKGRAD

/***************************************/
/* Erase 15N equilibrium magnetization */
/***************************************/
  4u pl3:f3
  (pwn ph26):f3

  2u
  p51:gp1
  d16

/*************************************************/
/*  +/- phase cycle on 1H prior to CEST period    */
/*************************************************/
#ifdef Hsteady_flg
  10u fq=0:f1
  (pw_eburp:sp9 ph26):f1
  2u
  (pw_eburptr:sp19 ph0):f1

  2u
  p51:gp1
  d16
#endif /*Hsteady_flg*/

/***************************************/
/*  D-CEST portion on Hz magnetization  */
/***************************************/
#ifndef cos_flg
  4u pl15:f1
#endif

#ifdef cal_HB1
  10u fq=cnst51:f1
```

```
  if "ncyc_cal > 0" {
#ifdef cos_flg
3   0.1u ipp25        ; number of excitation bands (l7) is even
    if "l7%2 == 0" {
      (p_dante:sp5 ph25):f1
    }
    else {
      (p_dante:sp5 ph26):f1
    }
#else /*cos_flg*/
3   (p_dante ph26):f1
    d_dante
#endif /*cos_flg*/
    lo to 3 times ncyc_cal
  }
#else /*cal_HB1*/
#ifdef cal_HR1
  time_cal
#else /*cal_HR1*/
  10u fq=H_offset:f1
#ifdef eta_flg
4 2u
  if "abs(H_offset) > 10000Hz" {
    taue
  }
  else {
#ifdef cos_flg
5   0.1u ipp25
    if "l7%2 == 0" {
      (p_dante:sp5 ph25):f1
    }
    else {
      (p_dante:sp5 ph26):f1
    }
#else /*cos_flg*/
5   (p_dante ph26):f1
    d_dante
#endif /*cos_flg*/
    lo to 5 times l6
  }

  (pwn ph27 pwn*2.0 ph26 pwn ph27):f3
  taue
  taue
  (pwn ph27 pwn*2.0 ph26 pwn ph27):f3

  if "abs(H_offset) > 10000Hz" {
    taue
  }
  else {
#ifdef cos_flg
6   0.1u ipp25
    if "l7%2 == 0" {
      (p_dante:sp5 ph25):f1
    }
    else {
      (p_dante:sp5 ph26):f1
    }
#else /*cos_flg*/
6   (p_dante ph26):f1
    d_dante
#endif /*cos_flg*/
    lo to 6 times l6
  }
  lo to 4 times l5
#else /*eta_flg*/
  if "abs(H_offset) > 10000Hz" {
    time_T1_adj
  }
```

```
  else {
    if "l8 > 0" {
#ifdef cos_flg
8     0.1u ipp25
      if "l7%2 == 0" {
        (p_dante:sp5 ph25):f1
      }
      else {
        (p_dante:sp5 ph26):f1
      }
#else
8     (p_dante ph26):f1
      d_dante
#endif
      lo to 8 times l8
    }
  }
#endif /*eta_flg*/
#endif /*cal_HR1*/
#endif /*cal_HB1*/
  10u fq=0:f1

  2u
  p52:gp2
  d16

/*****************************************/
/* Special setup for the reference plane */
/*****************************************/
#ifndef cal_HB1
#ifdef cal_HR1
  "l3=0"
#else
  if "abs(H_offset) > 10000Hz" {
    "l3=0"
  }
  else {
    "l3=1"
  }
#endif /*cal_HR1*/
#endif /*cal_HB1*/

  if "l3 == 0" {
    (pw_pc9:sp7 ph26):f1

#ifdef N_sel
    "DELTA = taua - pw_pc9*0.51 - larger(pw_reburp,pwn_sl)*0.46"
#else
    "DELTA = taua - pw_pc9*0.51 - pw_reburp*0.475"
#endif
    DELTA

#ifdef N_sel
    (center (pw_reburp:sp8 ph26):f1 (pwn_sl:sp32 ph26):f3)
#else
    (center (pw_reburp:sp8 ph26):f1 (pwn*2 ph26):f3)
#endif

    DELTA pl3:f3

    (pw_pc9:sp7 ph27):f1
  }

  2u
  p54:gp4
  d16

/*****************************************************/
/*  Option to purge NH2 peaks to get cleaner spectra  */
```

```
/*******************************************************/
#ifdef NH2_purge
  (pw_bip:sp6 ph26):f1
  (pwn ph26):f3

  2u
  p53:gp3
  d16

  "DELTA = taua - 2u - p53 - d16"
  DELTA

  (center (pw_bip:sp6 ph26):f1 (pwn*2 ph26):f3)

  2u
  p53:gp3
  d16

  DELTA

  (pwn ph27):f3

  2u
  p54:gp4
  d16
#endif

/*********************/
/*   15N labeling    */
/*********************/
#ifndef Hsteady_flg
  (pwn ph1):f3
#else
  (pwn ph11):f3
#endif

#ifndef NH2_purge
  "DELTA = pw_eburp*0.5*(1+0.72) - 32u - 100u + de*0.5"
  DELTA
#endif

  32u gron0*-1.0
  d0
  100u groff

#ifdef c_flg
  (center (pw_bip:sp6 ph26):f1 (pwc_ad:sp22 ph26):f2)
#else
  (pw_bip:sp6 ph26):f1
#endif

  32u gron0*1.0
  d0
  100u groff

#ifdef NH2_purge
  2u
  p55:gp5
  d16

#ifdef c_flg
  "DELTA = taua - 2u - p55 - d16 - larger(pw_bip,pwc_ad)*0.5 - 32u - 100u + pw_eburp*0
.5*(1-0.72) + de*0.5"
#else
  "DELTA = taua - 2u - p55 - d16 - pw_bip*0.5 - 32u - 100u + pw_eburp*0.5*(1-0.72) + d
e*0.5"
#endif
  DELTA
```

```
  (center (pw_bip:sp6 ph26):f1 (pwn*2 ph2):f3)

  2u
  p55:gp5
  d16

#ifdef c_flg
  "DELTA = taua - 2u - p55 - d16 + larger(pw_bip,pwc_ad)*0.5 + 32u + 100u - pw_eburp*0
.5*(1+0.72) - de*0.5"
#else
  "DELTA = taua - 2u - p55 - d16 + pw_bip*0.5 + 32u + 100u - pw_eburp*0.5*(1+0.72) - d
e*0.5"
#endif
  DELTA
#else /*NH2_purge*/
  "DELTA = pw_eburp*0.5*(1-0.72) + de*0.5"
  DELTA

  (pwn*2 ph2):f3

  "DELTA = 32u + 100u"
  DELTA

#ifdef c_flg
  (center (pw_bip:sp6 ph26):f1 (pwc_ad:sp22 ph26):f2)
#else
  (pw_bip:sp6 ph26):f1
#endif
#endif /*NH2_purge*/

/*************************/
/*   N->H back transfer   */
/*************************/
  (pw_eburp:sp9 ph26):f1

  "DELTA = de"
  DELTA

  (pwn ph26):f3

  2u
  p56:gp6
  d16

  "DELTA = taua - 2u - p56 - d16 - pw_reburp*0.475"
  DELTA

  (center (pw_reburp:sp8 ph26):f1 (pwn*2 ph26):f3)

  2u
  p56:gp6
  d16

  "DELTA = taua - 2u - p56 - d16 - pw_reburp*0.475 - 4u"
  DELTA

  4u BLKGRAD

  (pwn ph3):f3

#ifdef cal_HB1
  4u pl31:f3
#endif

/*************************************/
/*   Test how much water left          */
/*   set d13=AQ and zgoptn -Dwater     */
/*   also use small value of rg        */
/*************************************/
```

```
#ifdef water
  if "nsdone == 0" {
    d13
    d1 pl1:f1
    (pwh*0.1 ph26):f1
  }
  else {
    d13
  }
#endif

/*************************************/
/*   Signal detection and looping      */
/*************************************/
#ifdef cal_HB1
  go=2 ph31 cpds3:f3
  d11 do:f3 mc #0 to 2
#else
  go=2 ph31
  d11 mc #0 to 2
#endif

  F3QF(calph(ph3, +180))
#ifdef cal_HB1
  F2QF(calclist(ncyc_cal,1))
#else
#ifdef cal_HR1
  F2QF(calclist(time_cal,1))
#else
  F2QF(calclist(H_offset,1))
#endif /*cal_HR1*/
#endif /*cal_HB1*/
  F1PH(calph(ph1, +90), caldel(d0, +in0) & calph(ph1, +180) & calph(ph11, +180) & calp
h(ph31, +180))

HaltAcqu, 1m
exit

ph0=2 0
ph1=0 2
ph2=0 0 1 1 2 2 3 3
ph3=1
ph11=0
ph25=2 0
ph26=0
ph27=1
ph28=2
ph29=3
ph31=0 2 2 0

;d1: Repetition delay d1
;d3: taua =1/4J(NH) 2.68ms
;d11: delay for disk i/o, 30ms
;d13: set to acq time for testing water preservation
;d16: gradient recovery delay, 200us
;d17: time_T1, time for CEST, typically several 100s of ms
;pl1: tpwr - power level for 1H pulses
;pl3: dhpwr2 - power level for 15N pulse pwn
;pl10: tsatpwr - power level for water presat
;pl11: tpwrmess - power level for Messerle purge
;pl15: power level for 1H DANTE pulses
;pl31: dpwr2 - power level for 15N cpd3
;sp22: power level for 13C adiabatic pulse
;sp32: power level for 15N selective pulse
;spnam5: shape for shaped DANTE excitation
;spnam6: Bip720,50,20.1
;spnam7: Pc9_4_90.1000
;spnam8: Reburp.1000
;spnam9: Eburp2.1000
```

```
;spnam19: Eburp2tr.1000
;spnam22: shape for 13C adiabatic pulse
;spnam32: shape for 15N selective pulse to remove Arg
;p1: pwh
;p3: pwn
;p6: f1 channel - 180 degree shaped pulse for inversion
;                    Bip720,50,20.1            (200us at 600.13 MHz)
;p7: f1 channel -  90 degree shaped pulse for excitation
;                    Pc9_4_90.1000             (3.0ms at 600.13 MHz)
;p8: f1 channel - 180 degree shaped pulse for refocusing
;                    Reburp.1000               (2.0ms at 600.13 MHz)
;p9: f1 channel -  90 degree shaped pulse for excitation
;                     Eburp2.1000/Eburp2tr.1000 (1.92ms at 600.13 MHz)
;p15: pw_dante - 90 degree pulse at pl15
;p22: pwc_ad
;p32: pwn_sl for selective 15N pulse on amides
;p51: gradient pulse 51                                [400 usec]
;p52: gradient pulse 52                                [1000 usec]
;p53: gradient pulse 53                                [1000 usec]
;p54: gradient pulse 54                                [1500 usec]
;p55: gradient pulse 55                                [1000 usec]
;p56: gradient pulse 56                                [800 usec]
;cpdprg3: 15N decoupling program during t2 [waltz16]
;pcpd3: 1/dmf2 - 90 degree pulse for cpd3
;cnst1: offset (Hz) for B1 calibration from o1
;cnst5: H(N) chemical shift (ppm)
;cnst6: H(N) excitation bandwidth (ppm)
;cnst8: Frequency period (Hz) between adjacent excitation bands
;cnst9: Average H1 B1 (Hz) during D-CEST
;l4: number of pairs of 15N 180o pulses to apply for suppressing cross-correlation
;l7: number of excitation bands for D-CEST
;fq1list: frequency list (Hz) of 1H offsets in D-CEST
;vclist: variable counter list for 1H B1 calibration in D-CEST
;vdlist: variable delay list for 1H R1 measurement in D-CEST
;inf1: 1/SW(X) = 2*DW(X)
;in0: 1/(2*SW(x))=DW(X)
;nd0: 2
;ns: 2*n
;FnMODE: States in F1
;FnMODE: QF in F2

;for z-only gradients:
;gpz0: 0.25%
;gpz1: -25%
;gpz2: 15%
;gpz3: -60%
;gpz4: -70%
;gpz5: 60%
;gpz6: 80%

;use gradient files:
;gpnam1: SMSQ10.32
;gpnam2: SMSQ10.32
;gpnam3: SMSQ10.32
;gpnam4: SMSQ10.32
;gpnam5: SMSQ10.32
;gpnam6: SMSQ10.32

;zgoptns: Dfsat, Dfscuba, Dmess_flg, Df1180, Dc_flg, DN_sel, Dcal_HB1, Dcal_HR1, Dcalc
_SP, DHsteady_flg, Deta_flg, Dcos_flg, DNH2_purge
```