SUPPLEMENTARY INFORMATION

Optimizing frequency sampling in CEST experiments

Nicolas Bolik-Coulon, D. Flemming Hansen, Lewis E Kay

Relation between number of points in the CEST profile and acquisition time

Consider first a CEST profile comprising N_{cest} points in the frequency domain. Since only real points are available, a real Fourier transform of the data preserves all of the information. The resulting time domain comprises approximately $\frac{N_{cest}}{2}$ complex points, corresponding to an acquisition time (AT) of ($\frac{N_{cest}}{2}$)/SW, where SW is the spectral width in the CEST frequency dimension. If we choose $N_{cest} = \frac{SW}{\kappa \Delta \upsilon_{1/2}^{min}}$ then $\kappa = \frac{sw}{N_{cest}\, \Delta \upsilon_{1/2}^{min}}$, where κ is the number of CEST points per minimum dip linewidth at half height. As discussed in the main text, κ =1 in our work, so that at least two points are defined per CEST dip. The acquisition time in the transformed CEST profile then corresponds to $\frac{1}{2\Delta \upsilon_{1/2}^{min}}$. Assuming that the dips are Lorentzian (which they most certainly are not), AT = $\frac{\pi T_2^{max}}{2} \approx 1.5 T_2^{max}$.

By means of example consider Figure S1. Here 1000 complex points in the time domain are created, with a sampling rate corresponding to SW = 1000 Hz. Starting from the frequency domain (comprising only real points, as would be the case in a CEST experiment; blue peak in middle panel) an inverse real Fourier transfer (iFT) results in 1000 unique points (501 points in the real and 499 in the imaginary channels; note that the imaginary components for the first and last points are both zero). Shown in Figure S1 are profiles with 1000 points each (red and green, first line), but only the first 501 points are unique, with the remaining last half of the data obtained from the first half using the relation $f_k = f_{N-k}^*$, where f_i is the ith complex point, * denotes complex conjugate, and *N*=1000 (Karunanithy et al., 2021). In this case AT = 0.5 s for the non-redundant time domain data, obtained by transformation of the 1000 real points in the frequency domain (blue profile). Suppose we wish to record data out to AT $= 1.5 T_2$=0.1s. Then 200 real points would be required in the frequency domain, corresponding to approximately 100 complex time domain points and $\kappa = \frac{sw}{N_{cest}\, \Delta \upsilon_{1/2}^{min}} = \frac{1000}{200 x \frac{1}{\pi T_2}} \approx 1$.

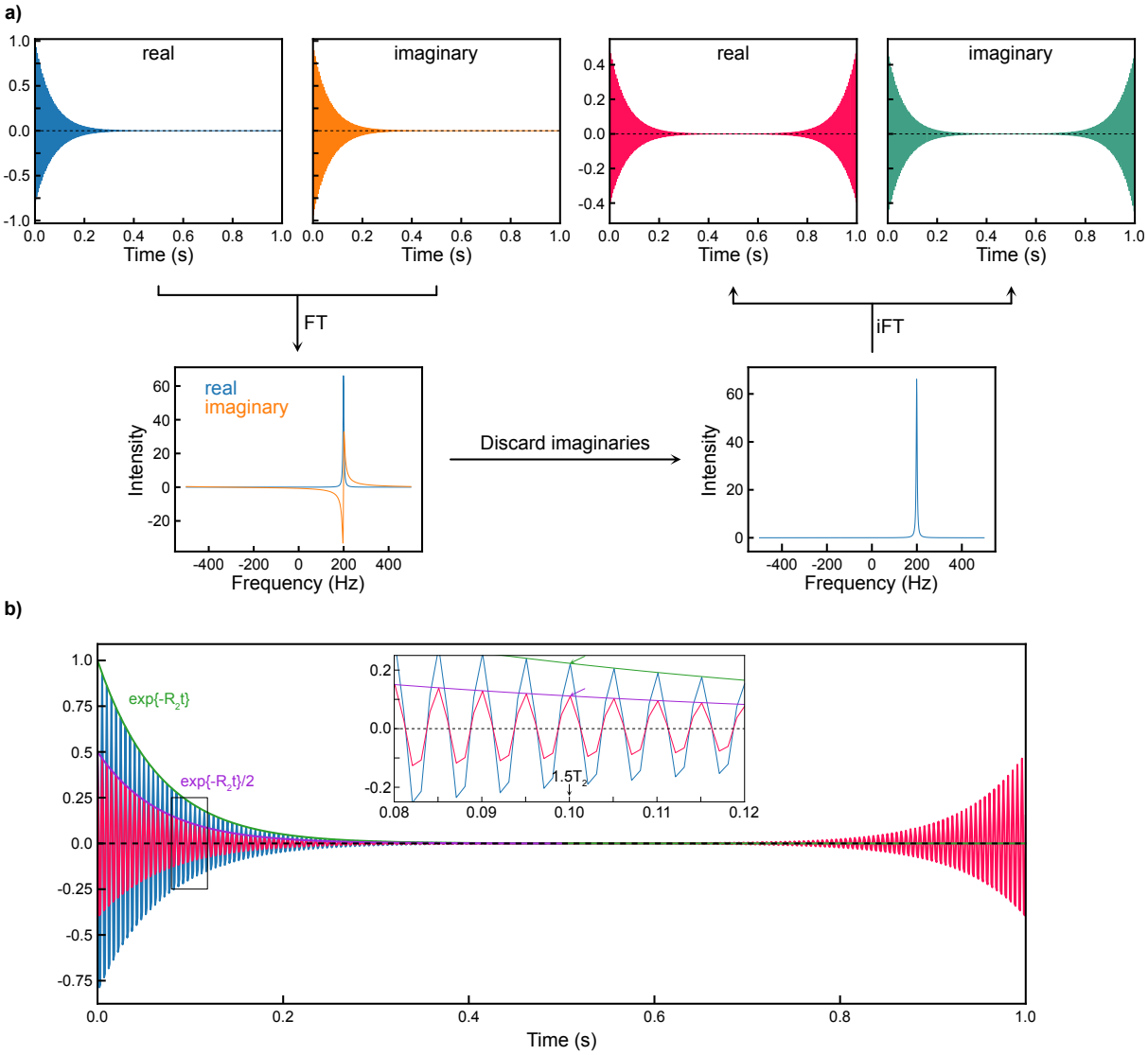

Figure S1. Time- and frequency-domain Fourier transforms. **a)** A Free Induction Decay (FID) was simulated as $e^{-R_2 t}e^{-i\omega_o t}$, R_2 = 15 s^{-1} and $\frac{\omega_o}{2\pi}$= 200 Hz. Both the real and imaginary time-domain components are shown (top panel left). The time-domain data is Fourier transformed (initially the first point is divided by 2 to eliminate the baseline offset in the frequency domain), generating real absorptive (blue) and imaginary dispersive (orange) peaks (middle panel, left). Only the real part is retained, in analogy to CEST data (middle panel, right). The absorptive peak is inverse Fourier transformed, producing a new FID with real and imaginary components (top, right; note that the complete time domain is retained, rather than just the first half shown in Figure 1). **b)** Overlay of the real part of the FID generated initially, ($e^{-R_2 t}cos\ (\omega_o t)$; blue, top panel, left), and the real FID obtained after the FT, discard imaginaries, iFT manipulation described above (red; top panel, right). For the first half of the time domain both FIDs have the same evolution profile (although the post-manipulation FID, red, is a factor of two smaller). The exponential decay of the signal amplitude is shown in green and in purple for the blue and red FIDs, respectively. The inset enlarges the region surrounding the time point corresponding to $1.5 \times T_2 = 0.1$ s and the green and purple arrows indicate the coordinate points (0.1, 0.22) and (0.1, 0.11) respectively, *i.e.* points at which the amplitude has decayed to $100e^{-1.5}$ % of the initial values for both FIDs. It must be noted

that when the time-domain is shortened, the decreasing and increasing parts of the red FID overlap such that the decay will not follow the expected exponential profile shown in purple.

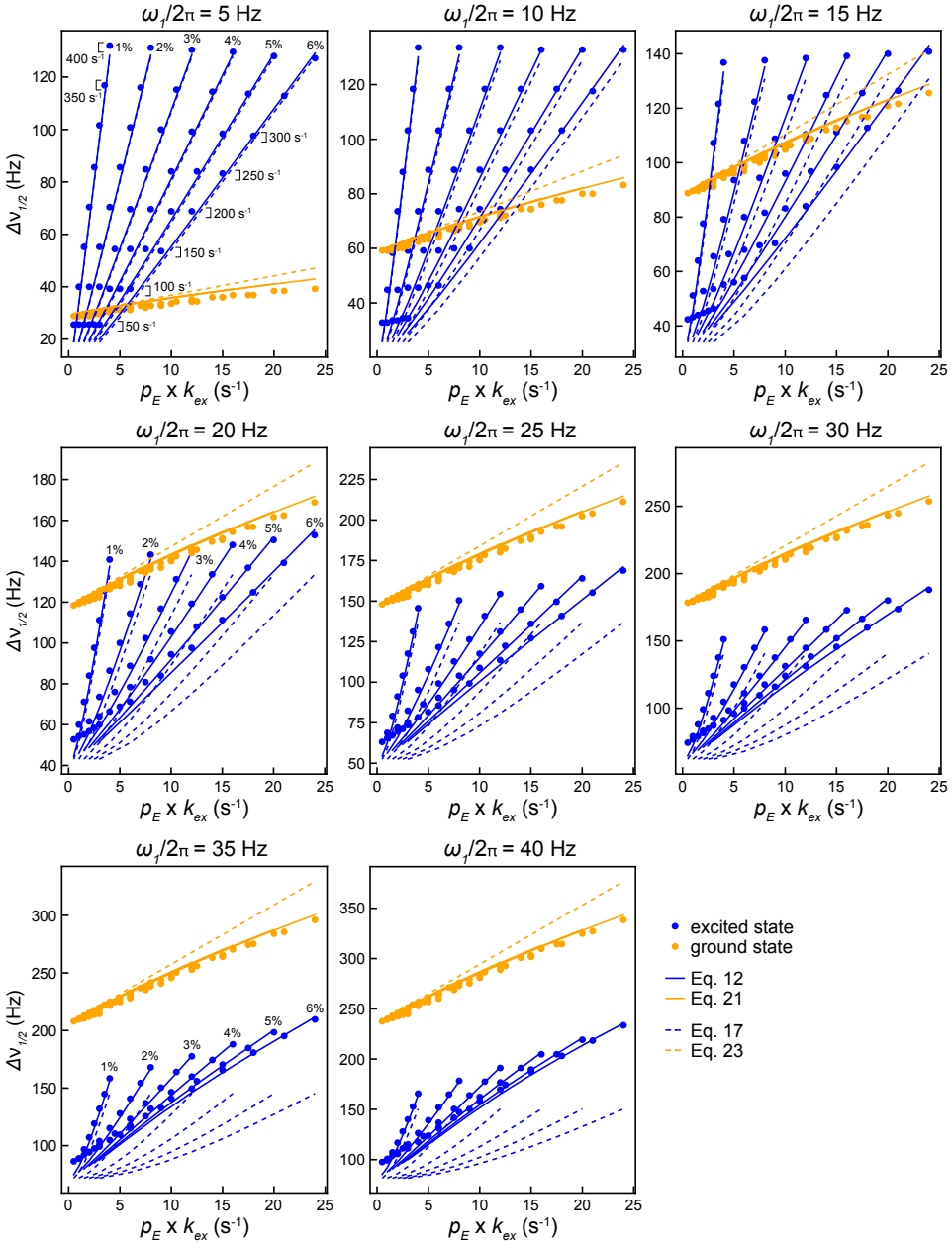

Figure S2. Linewidths at half height ($\Delta\nu_{1/2}$) for major and minor dips in CEST profiles as a function of $\frac{\omega_1}{2\pi}$ calculated from simulated CEST profiles. Details can be found in the legend to Figure 2. Values of p_E and k_{ex} are listed, and lines denote predicted linewidths from Eq. 12 (blue, solid lines), Eq. 17 (blue, dashed lines), Eq. 21 (orange, solid lines) or Eq. 23 (orange, dashed lines).

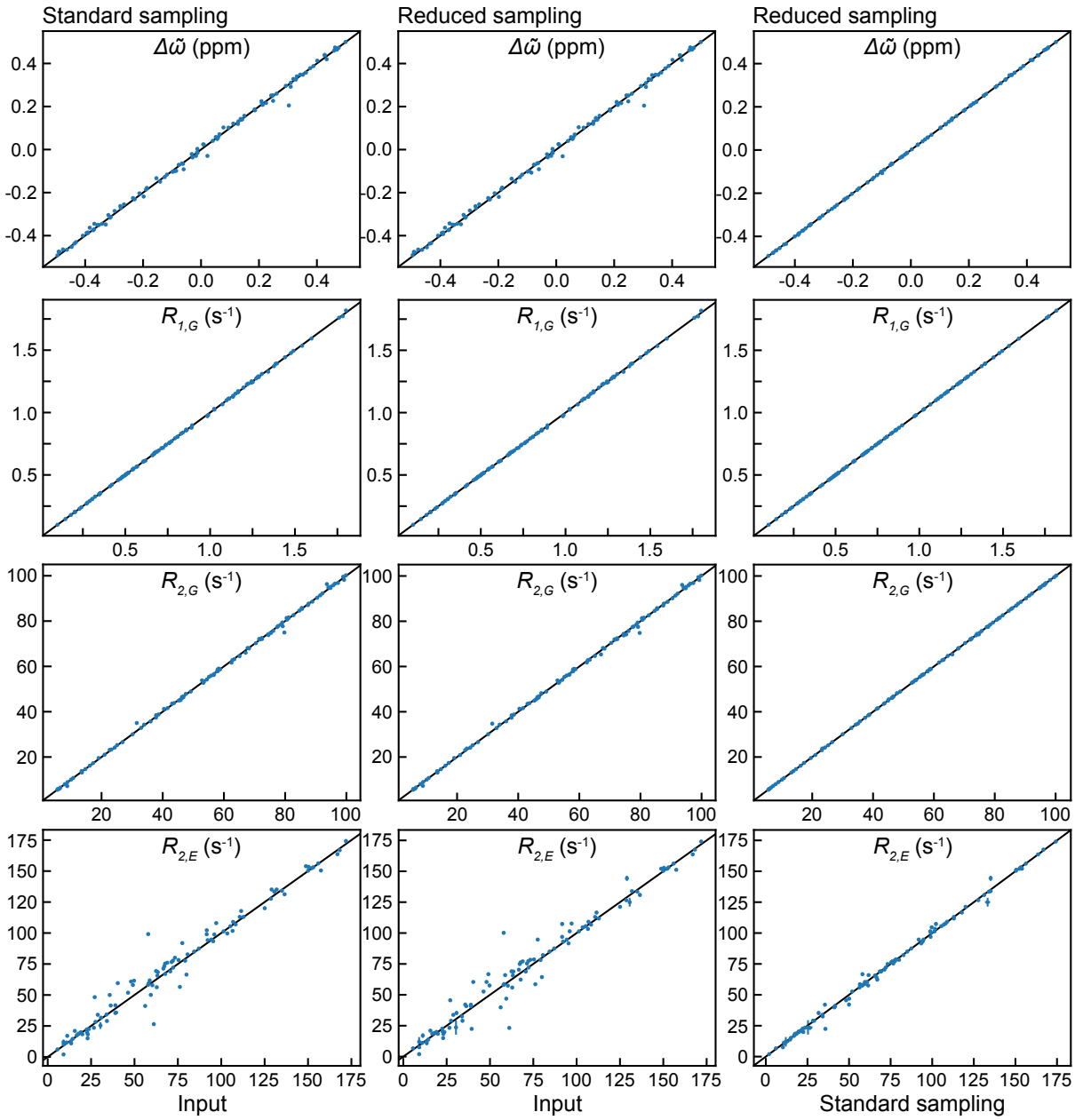

Figure S3. Linear correlation plots of exchange parameters fit from simulated CEST profiles generated as described in Materials and Methods; for all CEST profiles $|\Delta\varpi| \leq 0.5$ ppm. The relation, $\Delta\upsilon_{1/2}^{min}(Hz) = 2.15\frac{\omega_1}{2\pi} + 11.41$, was used along with Eq. 24 to obtain the number of CEST points and hence the frequency spacing for reduced sampling, while points were computed every $\frac{\omega_1}{2\pi}$ Hz for standard sampling. Each correlation panel is based on 100 profiles (circles; a pair of CEST curves with different B_1 values for each profile was fit) with the y=x line shown in black. A two-site exchange model was used both for generating the CEST profiles and for data fitting; the exchange and relaxation parameters used to generate the CEST curves are listed in Table 1.

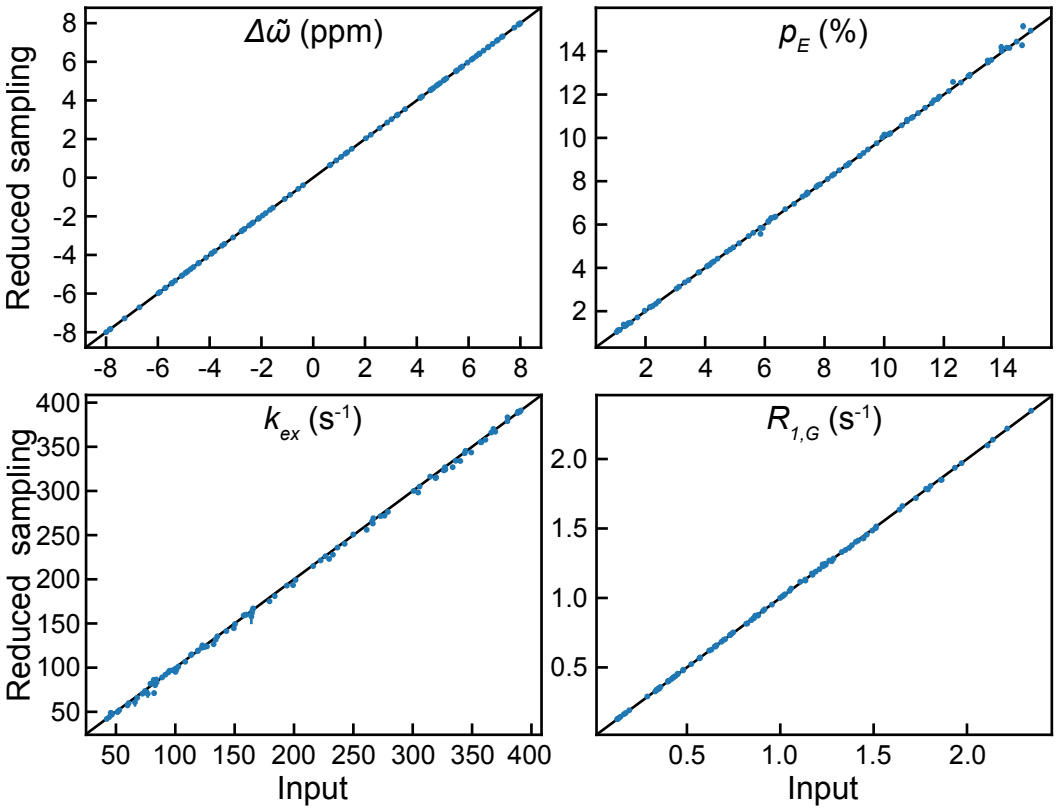

Figure S4. Linear correlation plots of exchange parameters fit from simulated CEST profiles generated as described in Materials and Methods and discussed in the text, using a reduced sampling approach *vs* input parameters. The intrinsic R_2 of the ground and excited states are both set to 5 s^{-1}. The relation, $\Delta v_{1/2}^{min}(Hz) = 2.15\frac{\omega_1}{2\pi} + 11.41$, was used along with Eq. 24 to obtain the number of CEST points and hence the frequency spacing for each CEST profile. Each correlation panel is based on analysis of 100 profiles (circles; a pair of CEST curves with B_1 values of 15 and 30 Hz for each profile were fit together). The y=x line is shown in black. A two-site exchange model was used both for generating the CEST profiles and for fitting the data; parameters are listed in Table 1. Only profiles for which $|\Delta\varpi| > 0.5$ ppm were considered.

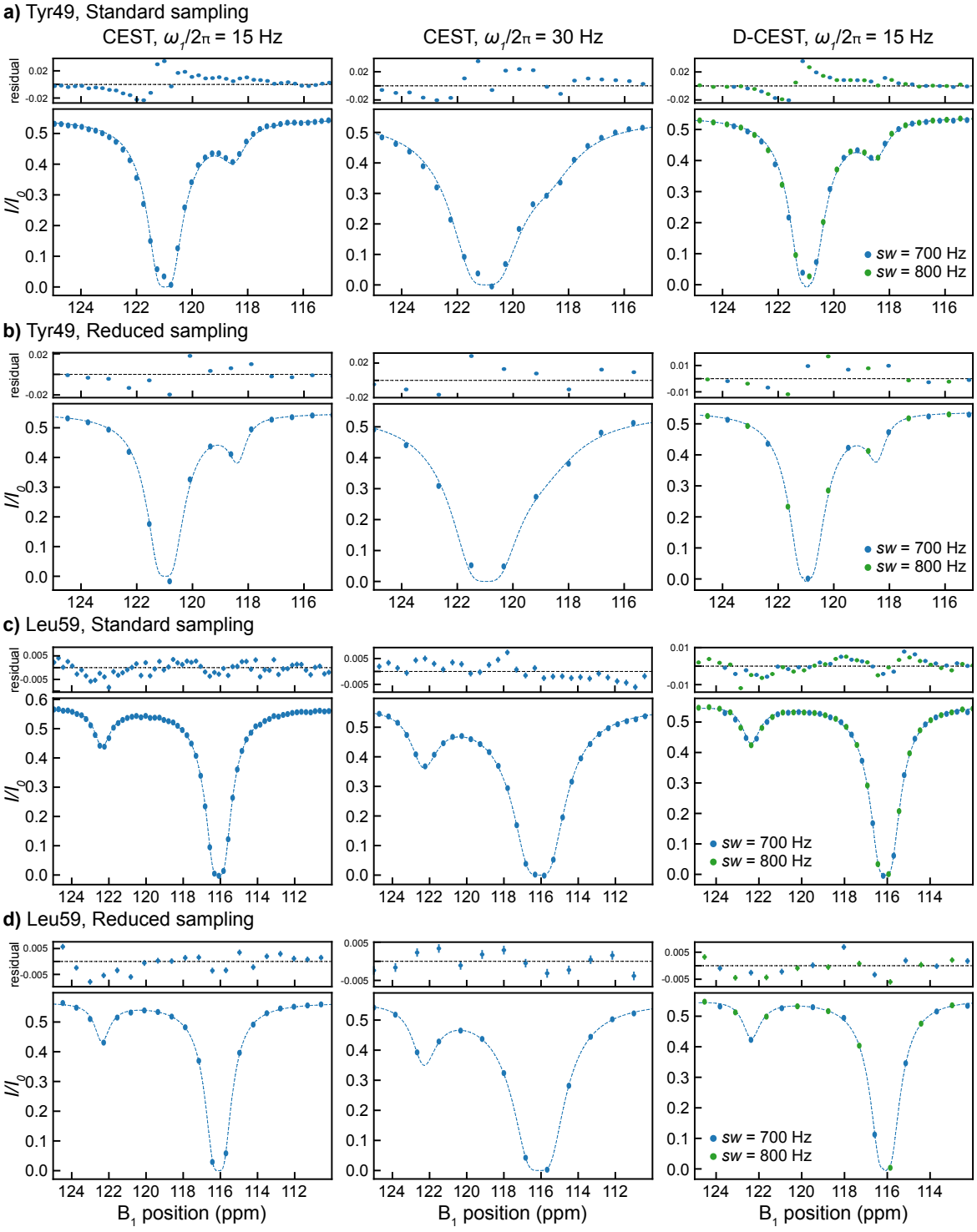

Figure S5. Comparison of CEST profiles recorded with a B_1 field of 15 Hz (left) and 30 Hz (middle) as well as D-CEST profiles recorded with a B_1 field of 15 Hz (right) for (**a**) Tyr49 showing the relatively poor fitting quality (dashed lines) and for (**b**) Leu59 where an "average" fit quality was obtained. Fit residuals are shown above each profile. For the D-CEST experiments two profiles were measured, corresponding to spectral widths of 700 Hz (blue) and 800 Hz (green). The data from both profiles have been merged in the D-CEST panels.

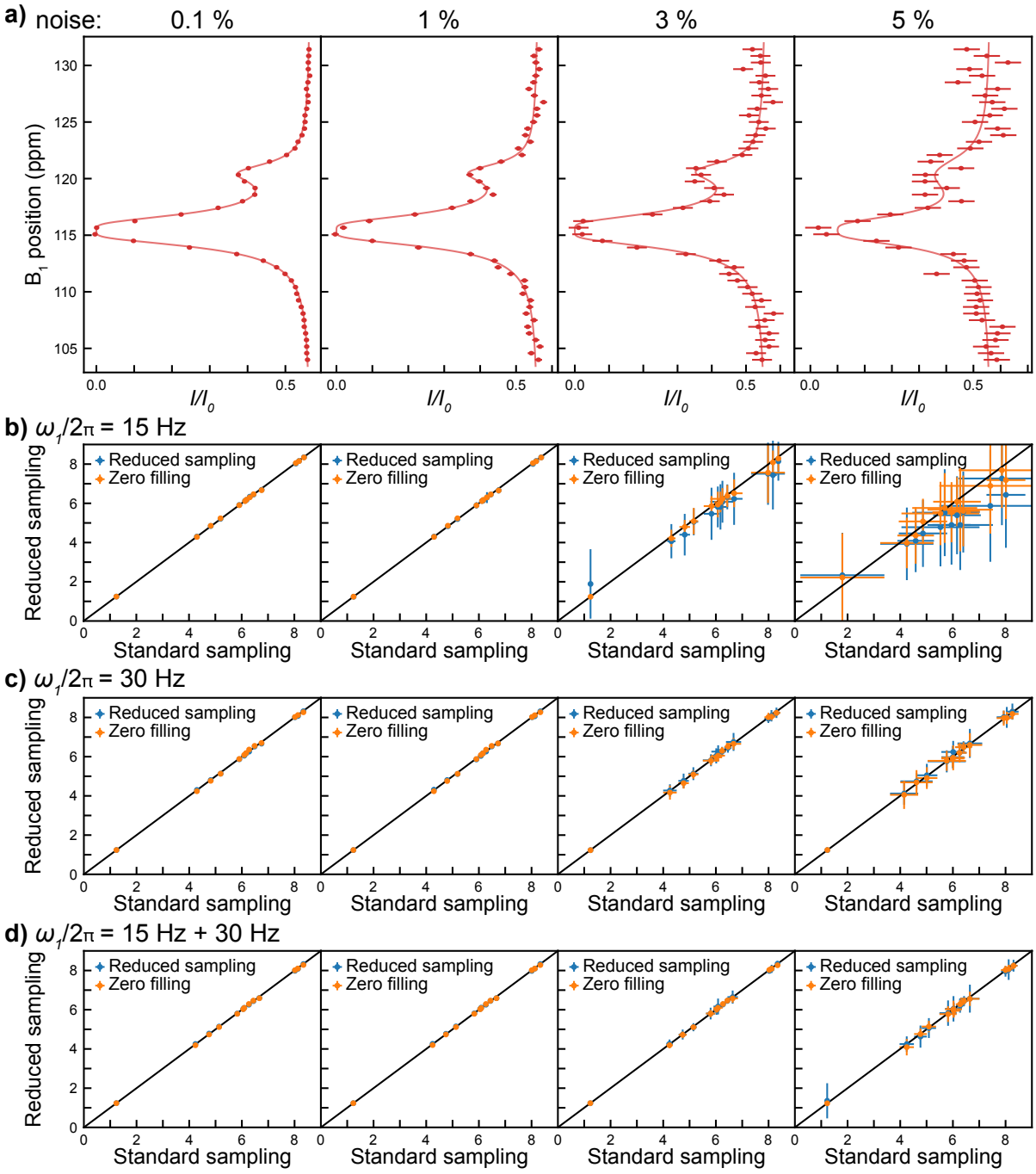

Figure S6. Effects of zero-filling and noise on fitted chemical shifts from CEST profiles. (**a**) CEST profiles were obtained via post-acquisition processing (Figure 1a) involving inverse Fourier transformation, doubling of the time domain by zero-filling, and re-transformation. By means of example, profiles for Lys26 obtained via reduced sampling and a B_1 field strength of 30 Hz and processed as described above are shown. To each profile has been added a different level of noise (based on the intensity of the Lys26 peak in the reference spectrum I_o; note that the experimental noise is close to 0.1% of I_o). For each of 13 residues that were included in the analysis (residue number: 26, 28, 29, 33, 37, 41, 42, 43, 50, 52, 55, 59, 67) one hundred profiles have been generated for each noise level (see text). Each profile was generated starting from experimental datasets recorded on A39G FF (600 MHz, 2ºC; both standard and reduced sampling, as well as reduced sampled data that had been zero-filled as described above) and then subsequently globally fit with a two-site exchange model. Shown are linear

correlation plots as a function of noise level for extracted $\Delta\varpi$ values obtained from analysis of datasets recorded with B_1 field strengths of 15 Hz (**b**), 30 Hz (**c**), and both 15 Hz and 30 Hz (**d**). The y=x line is shown in black. Correlations between values fitted from standard and reduced sampling profiles and standard and zero-filled reduced sampled curves are shown in blue and orange circles, respectively. Error bars denote one standard deviation from mean $\Delta\varpi$ values extracted in fits based on the analysis of the 100 profiles for each noise level (for each of the 13 residues).

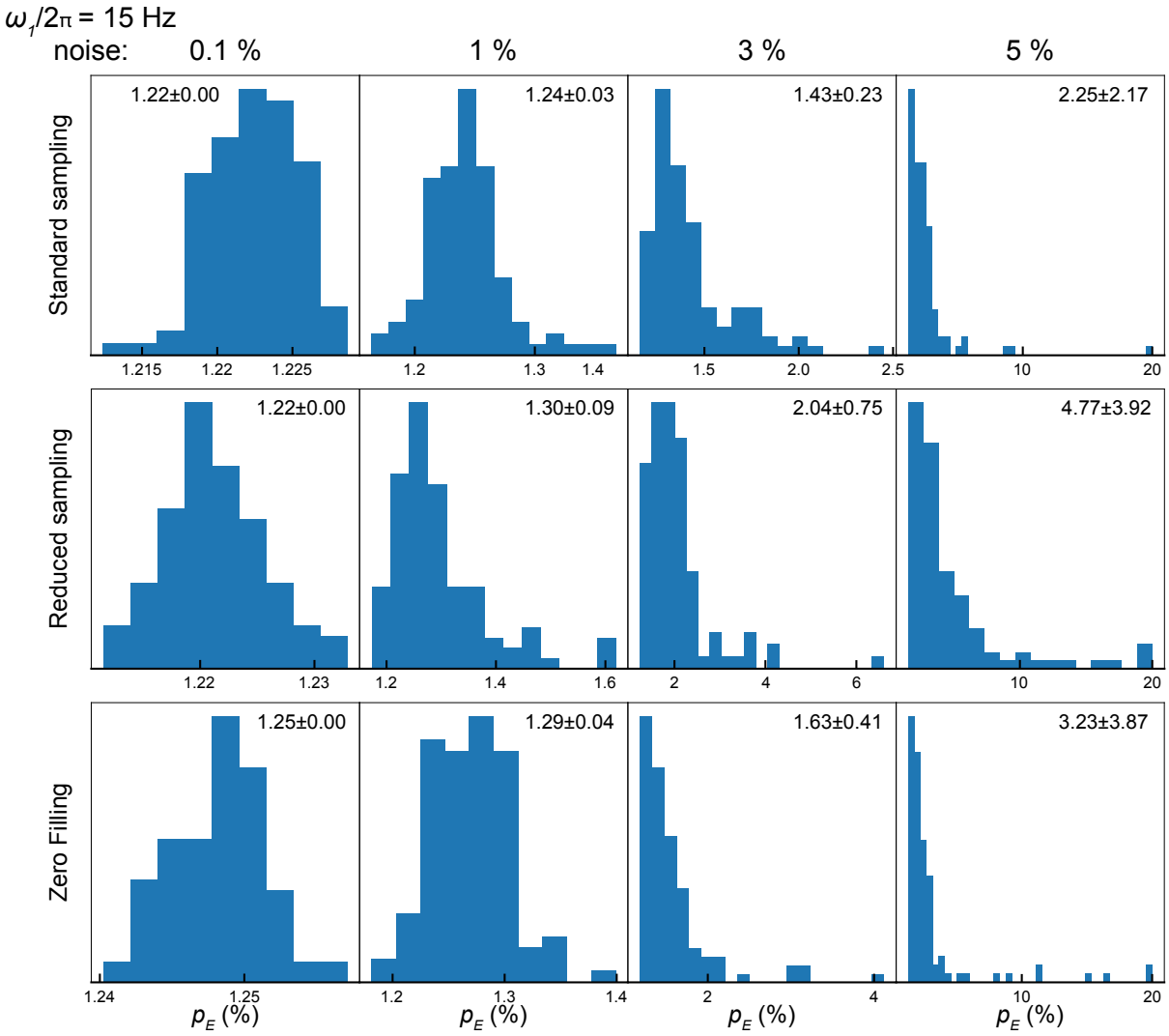

Figure S7. Distribution of p_E values obtained via analysis of datasets generated as described in the legend to Figure S6 from experimental profiles recorded with a B_1 field strength of 15 Hz. The mean and standard deviation of each distribution are indicated on the right-hand side of each panel. The set of panels denoted "Zero Filling" was obtained via analysis of datasets recorded with reduced sampling and subsequently processed in a post-acquisition manner to double the number of points in the CEST frequency domain.

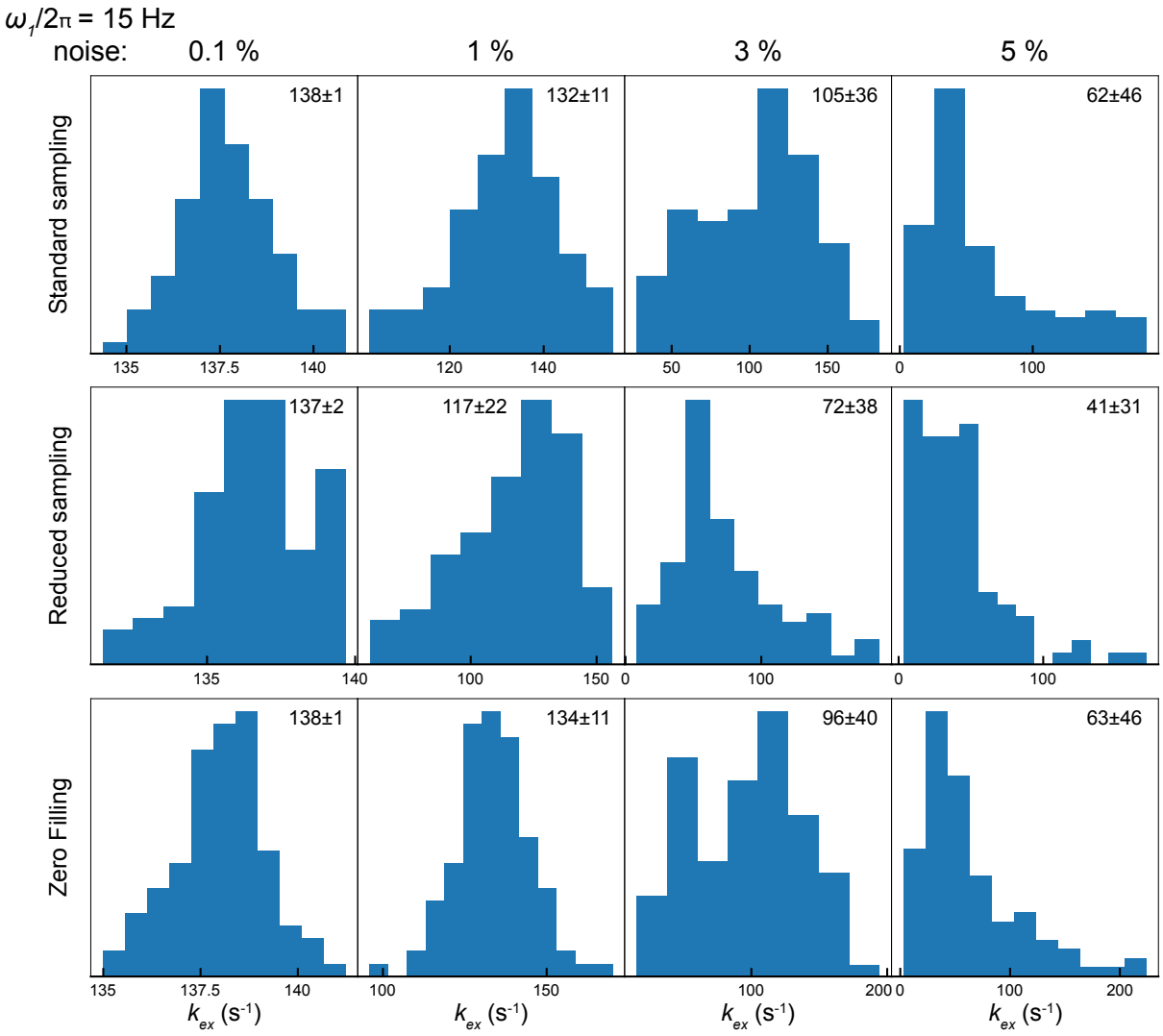

Figure S8. Distribution of k_{ex} values obtained via analysis of datasets generated as described in the legend to Figure S6 from experimental profiles recorded with a B_1 field strength of 15 Hz. Details are described in legends to Figures S6 and S7.

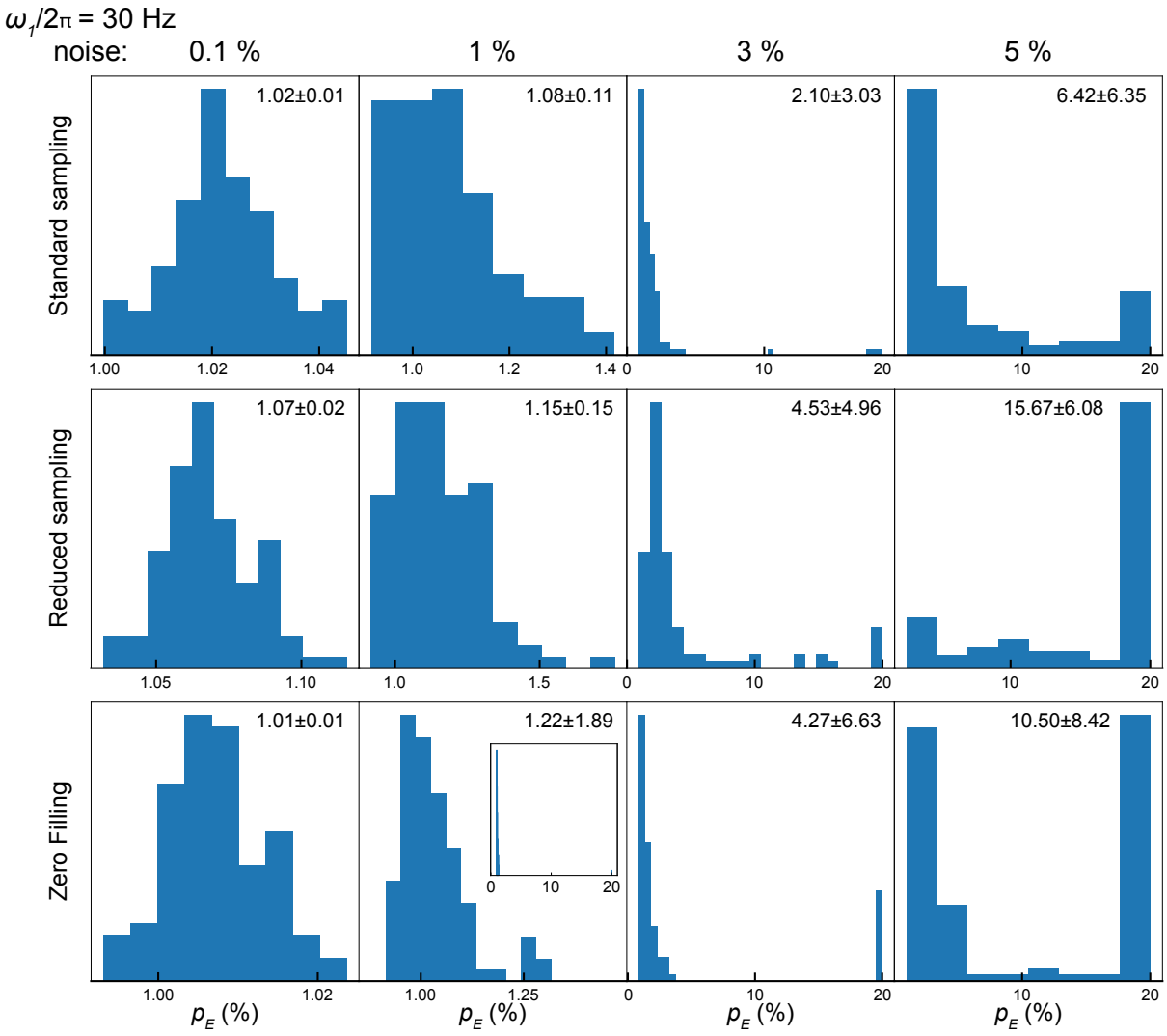

Figure S9. Distribution of p_E values obtained via analysis of datasets generated as described in the legend to Figure S6 from experimental profiles recorded with a B_1 field strength of 30 Hz (see legend to Figure S7).

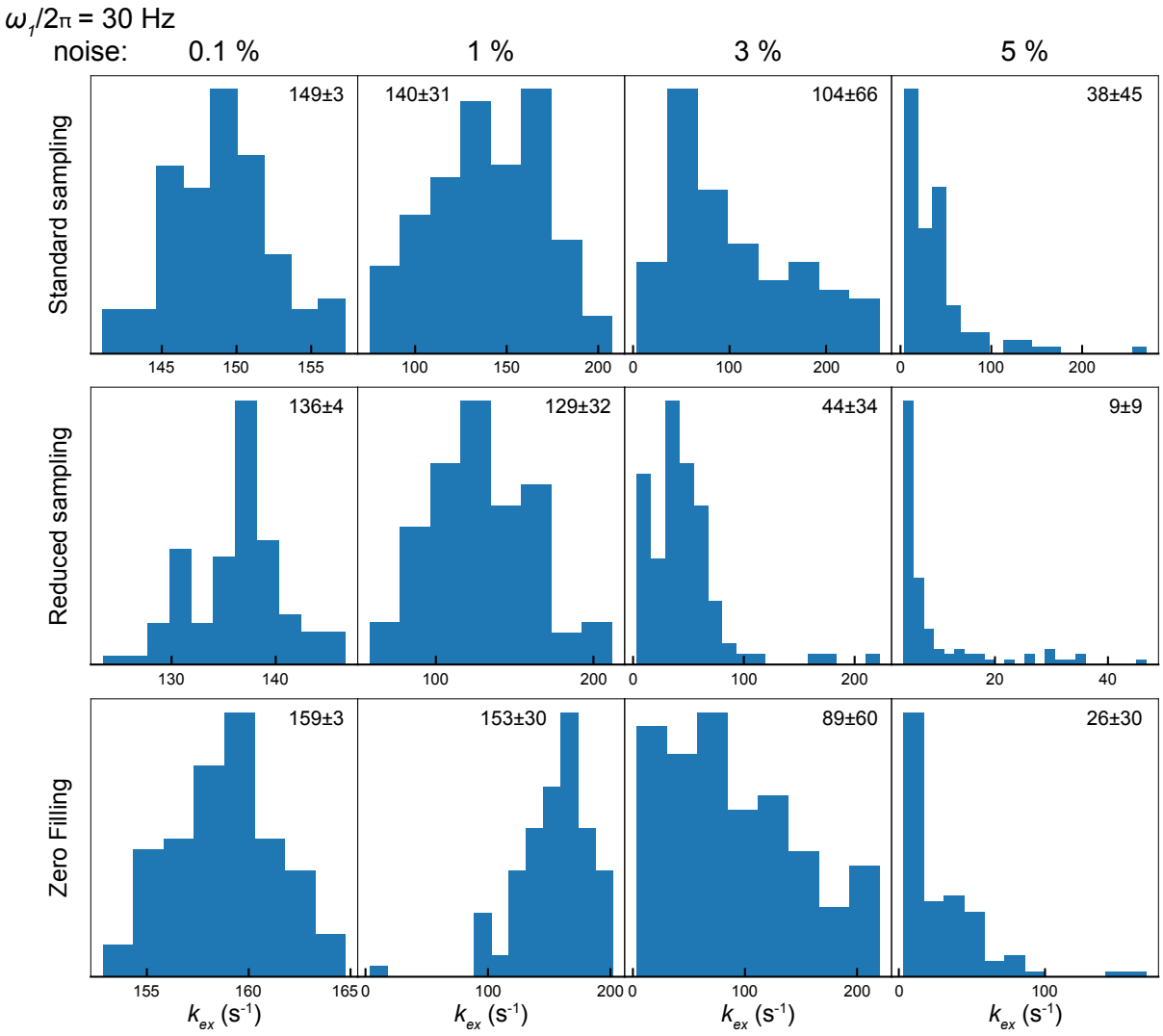

Figure S10. Distribution of k_{ex} values obtained via analysis of datasets generated as described in the legend to Figure S6 from experimental profiles recorded with a B_1 field strength of 30 Hz.

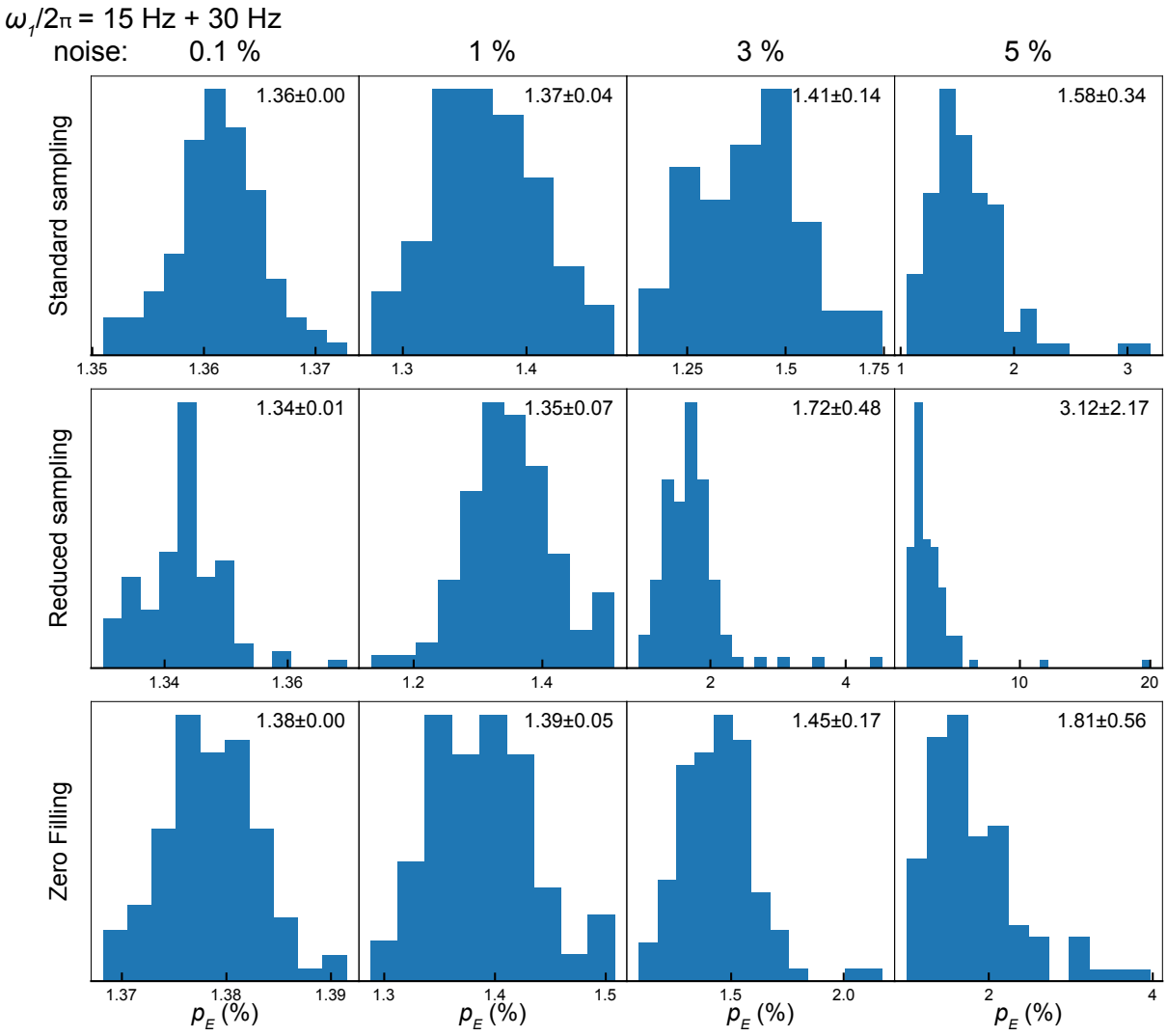

Figure S11. Distribution of p_E values obtained via a combined analysis of datasets recorded with B_1 field strengths of 15 Hz and 30 Hz and generated as described in the legend to Figure S6.

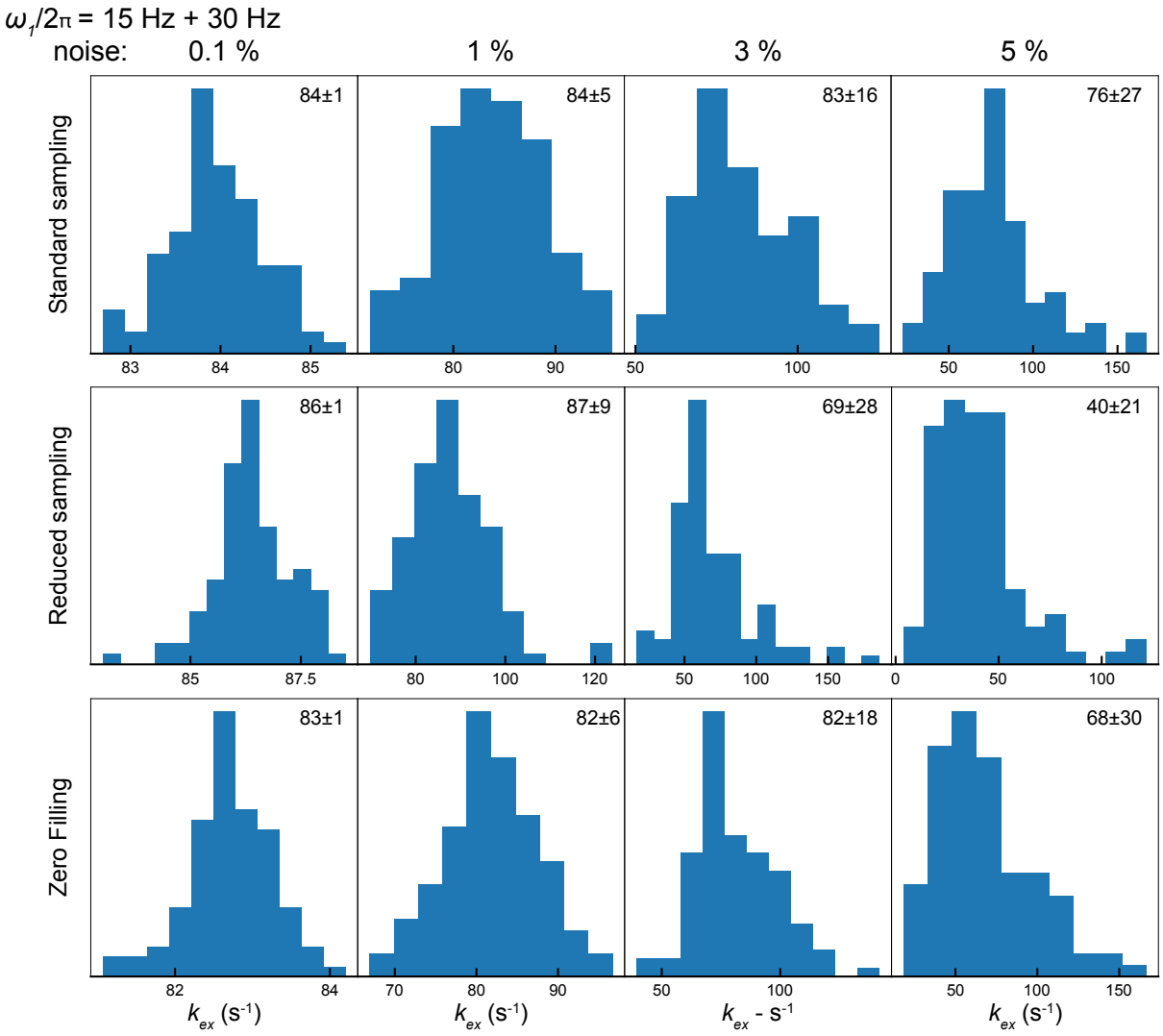

Figure S12. Distribution of k_{ex} values obtained via a combined analysis of datasets recorded with B_1 field strengths of 15 Hz and 30 Hz and generated as described in the legend to Figure S6.

C-code to generate frequency lists:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#define PI 3.141592

#define lw(x, a, b) ( a*x + b )   /*Parameters a and b depend on the
intrinsic R2 of the ground state */
#define time(x, a, b) ( 3.0/(PI * lw(x, a, b)) )
#define Npts(x, y, a, b) ( time(x, a, b) * y + 0.5 ) /*x is the b1 field, y
is the spectral width */

#define max(a, b) \
        ({ __typeof__ (a) _a = (a); \
           __typeof__ (b) _b = (b); \
        _a > _b ? _a: _b; })
#define min(a, b) \
        ({ __typeof__ (a) _a = (a); \
           __typeof__ (b) _b = (b); \
        _a < _b ? _a: _b; })


main(){

        float  fq_max, sw, sw1, sw2, b1_Hz;
        float  fq_incr, fq_incr1, fq_incr2, fq_max1, fq_mid;
        float  fq_max2, fq_max2_th, fq_max2_1, fq_max2_2;
        float  w;
        float  R2G;
        int    i, num_pts, num_pts1, num_pts2;
        FILE   *fi1, *fi2;
        char   ExpType;


        printf("Type of experiment (c for CEST, d for D-CEST): ");
        scanf("%s", &ExpType);


/* CEST case */
        if(ExpType == 'c'){

                printf("b1 field (Hz): ");
                scanf("%f", &b1_Hz);
                printf("First offset frequency (subsequent frequencies will
be higher): ");
                scanf("%f", &fq_max);
                printf("spectral width (Hz): ");
                scanf("%f", &sw);
                printf("Intrinsic R2 of ground state (s-1): ");
```

```
                scanf("%f", &R2G);


                float a = 2.01;
                float b = 5.97;
                if(R2G > 7.5){
                        float a = 2.03;
                        float b = 8.70;
                }
                if(R2G > 12.5){
                        float a = 2.09;
                        float b = 10.24;
                }
                if(R2G > 17.5){
                        float a = 2.15;
                        float b = 11.41;
                }

                num_pts = Npts(b1_Hz, sw, a, b);

                fq_incr = sw / num_pts;

                fi1 = fopen("N_cest_fq", "w");

        /*make file */
                fprintf(fi1, "-12001.0");
                for(i=1;i<=num_pts+1;++i) {
                        fprintf(fi1, "\n");
                        w = fq_max + (float)(i - 1) * fq_incr;
                        fprintf(fi1, "%f", w);
                }
                fclose(fi1);

                printf("N_cest_fq has been created\n");
                exit(1);
        }
/* D-CEST case */

        if(ExpType == 'd'){
                printf("b1 field (Hz): ");
                scanf("%f", &b1_Hz);
                printf("First offset frequency (subsequent frequencies will
be higher) for exp 1: ");
                scanf("%f", &fq_max1);
                printf("spectral width (Hz) for exp 1: ");
                scanf("%f", &sw1);
                printf("spectral width (Hz) for exp 2: ");
                scanf("%f", &sw2);
                printf("Intrinsic R2 of ground state (s-1): ");
                scanf("%f", &R2G);

                float a = 2.01;
```

```
                float b = 5.97;
                if(R2G > 7.5){
                        float a = 2.03;
                        float b = 8.70;
                }
                if(R2G > 12.5) {
                        float a = 2.09;
                        float b = 10.24;
                }
                if(R2G > 17.5) {
                        float a = 2.15;
                        float b = 11.41;
                }

                num_pts1 = Npts(b1_Hz, 0.5 * sw1, a, b);  /* 0.5 here to
divide number of points by two as this is D-CEST */
                num_pts2 = Npts(b1_Hz, 0.5 * sw2, a, b);

                fq_incr1 = sw1 / num_pts1;
                fq_incr2 = sw2 / num_pts2;
                fq_incr = min(fq_incr1, fq_incr2);   /* This is the
frequency increment that will be used for D-CEST */

                fq_mid = fq_max1 + 0.5 * sw1;   /* This is the center
frequency */
                fq_max2_th = fq_mid - 0.5 * sw2;    /* This is the
theoretical maximum frequency for experiment 2 */
                /* Compute starting frequency for experiment 2 */
                if(sw1 < sw2){
                        fq_max2_1 = fq_max1 - 0.5 * fq_incr - fq_incr *
(int) (abs(fq_max2_th - (fq_max1 - 0.5 * fq_incr))/fq_incr + 0.5);
                        fq_max2_2 = fq_max1 + 0.5 * fq_incr - fq_incr *
(int) (abs(fq_max2_th - (fq_max1 + 0.5 * fq_incr))/fq_incr + 0.5);
                }
                else{
                        fq_max2_1 = fq_max1 - 0.5 * fq_incr + fq_incr *
(int) (abs(fq_max2_th - (fq_max1 - 0.5 * fq_incr)) / fq_incr + 0.5);
                        fq_max2_2 = fq_max1 + 0.5 * fq_incr + fq_incr *
(int) (abs(fq_max2_th - (fq_max1 + 0.5 * fq_incr)) / fq_incr + 0.5);
                }

                float diff_1 = abs(fq_max2_1 - fq_max2_th);
                float diff_2 = abs(fq_max2_2 - fq_max2_th);
                if(diff_1 < diff_2){
                        fq_max2 = fq_max2_1;    /* This is the maximum
frequency for experiment 2 that will be used */
                }
                else {
                        fq_max2 = fq_max2_2;
                }
```

```
        /* Make file 1 */
                fi1 = fopen("N_dcest_fq_sw1", "w");

                fprintf(fi1, "-12001.0");

                for(i = 1; i <= num_pts1+1; ++i) {
                        fprintf(fi1, "\n");
                        w = fq_max1 + (float)(i - 1) * fq_incr;
                        fprintf(fi1, "%f", w);
                }

                fclose(fi1);


        /* Make file 2 */
                fi2 = fopen("N_dcest_fq_sw2", "w");

                fprintf(fi2, "-12001.0");

                for(i = 1; i <= num_pts2+1; ++i){
                        fprintf(fi2, "\n");
                        w = fq_max2 + (float)(i - 1) * fq_incr;
                        fprintf(fi2, "%f", w);
                }


                printf("2 files have been created:\n");
                printf(" N_dcest_fq_sw1\n");
                printf(" N_dcest_fq_sw2\n");
                exit(1);

        }



        else{
                exit(1);
        }
```

Python code to zero-fill CEST datasets:

```
import numpy as np

def CalciFT(data): #function to compute the inverse Fourier transform
        iFT = [0. for t in data]
        for t in range(len(data)):
                for w in range(len(data)):
                        iFT[t] += 1./len(data) * data[w] * np.exp(2.*np.pi
* 1j * w * t /len(data))
```

```
        return iFT

def CalcFT(data): #function to compute the Fourier transform
        FT = [0. for w in data]
        for w in range(len(data)):
                for t in range(len(data)):
                        FT[w] += data[t] * np.exp(-2.*np.pi * 1j * w * t /
len(data))

        return FT


#inverse Fourier Transform
data_iFT = CalciFT(CESTdata) #CESTdata is an array containing the peak
intensities

#Zero fill
data_iFT_ZF = np.zeros( int(2.*len(data_iFT)), 'complex64')
data_iFT_ZF[:len(data_iFT)//2] = data_iFT[:len(data_iFT)//2]
data_iFT_ZF[-len(data_iFT)//2:] = data_iFT[len(data_iFT)//2:]

#Fourier transform
data_FT = np.real(CalcFT(data_iFT_ZF))

#Generate the new frequency list
DeltaW_ZF = abs(wList[0] - wList[1])/2. #wList is the experimental
frequency list, DeltaW_ZF is the new frequency increment
wList_ZF = wList[0] + np.asarray(range(len(data_FT))) * DeltaW
```

Supplementary information references

Karunanithy, G., Yuwen, T., Kay, L.E., Hansen, D.F., 2021. Towards autonomous analysis of Chemical Exchange Saturation Transfer experiments using Deep Neural Networks (preprint). Chemistry. https://doi.org/10.26434/chemrxiv-2021-r1cmw