J Biomol NMR (2015) 63:9–19
DOI 10.1007/s10858-015-9952-5

ARTICLE

Fast multi-dimensional NMR acquisition and processing using the sparse FFT

Haitham Hassanieh[1] · Maxim Mayzel[2] · Lixin Shi[1] · Dina Katabi[1] · Vladislav Yu Orekhov[2]

Received: 18 December 2014 / Accepted: 28 May 2015 / Published online: 30 June 2015

Abstract Increasing the dimensionality of NMR experiments strongly enhances the spectral resolution and provides invaluable direct information about atomic interactions. However, the price tag is high: long measurement times and heavy requirements on the computation power and data storage. We introduce sparse fast Fourier transform as a new method of NMR signal collection and processing, which is capable of reconstructing high quality spectra of large size and dimensionality with short measurement times, faster computations than the fast Fourier transform, and minimal storage for processing and handling of sparse spectra. The new algorithm is described and demonstrated for a 4D BEST-HNCOCA spectrum.

Keywords Compressed sensing · Reduced dimensionality · Non uniform sampling · Fast NMR

Introduction

Multi-dimensional NMR spectroscopy is an invaluable biophysical tool in chemistry, structural biology, and many other applications. However, from its introduction in 1970s, the technique is impeded by long measurement times, heavy computations and large data storage requirements. These problems stem from the huge number of data points needed for quantifying the frequency domain spectrum with the required resolution.

With the traditional systematic data sampling in the time domain, the duration of an NMR experiment increases exponentially with spectral dimensionality and polynomially with resolution. The rapid development in the field of fast spectroscopy with non-uniform sampling reduces the measurement time by decreasing number of the acquired data points (Billeter and Orekhov 2012; Coggins et al. 2010; Hyberts et al. 2014; Orekhov and Jaravine 2011). Non-uniform sampling (NUS) has enabled the acquisition and analysis of practical high-resolution experiments of dimensionality up to 7D (Hiller et al. 2007; Kazimierczuk et al. 2010; Motáčková et al. 2010). Success of the NUS techniques is explained by the notion that the NMR spectrum is sparse in the frequency domain, i.e. only a small fraction of the spectrum contains signals, while the rest contains only baseline noise. Moreover, typically the higher the spectrum dimensionality and resolution are, the sparser the frequency domain spectrum is. While the numerous NUS spectra reconstruction algorithms differ in their underlying assumptions, the common theme is that all information about the spectral signals can be obtained from a relatively small number of measurements, which is linear to the number of signals and nearly independent on the spectrum dimensionality and resolution.

NMR measurements are performed in the time domain and, in the case of traditional Fourier spectroscopy, the time signal is converted to the frequency spectrum by the Discrete Fourier Transforms (DFT). For a d-dimensional spectrum with N points for each spectral dimension, we need to sample N^d experimental points, perform DFT with

Electronic supplementary material The online version of this article (doi:10.1007/s10858-015-9952-5) contains supplementary material, which is available to authorized users.

✉ Vladislav Yu Orekhov
vladislav.orekhov@nmr.gu.se

[1] Massachusetts Institute of Technology, 32 Vassar Street, 32-G936, Cambridge, MA 02139, USA

[2] Swedish NMR Centre at University of Gothenburg, Box 465, 40530 Gothenburg, Sweden

$O(N^d \log N^d)$ elementary mathematical operations and allocate $O(N^d)$ bytes for spectrum processing, storage, and analysis. For example, a moderate-resolution 5D spectrum with $N = 256$ for all dimensions requires 4 TB of storage. Even if such spectrum can be computed, it cannot be easily handled in the downstream analysis. Algorithms used for reconstructing the complete spectrum from the NUS data, require at least the same and often significantly larger computations and storage than the traditional Fourier based approach. For example, for the Compressed Sensing (CS) (Holland et al. 2011; Kazimierczuk and Orekhov 2011), storage is $O(N^d)$ and the amount of calculations is polynomial on N^d. Moreover, these algorithms are iterative and thus are impractical, when data do not fit into computer operative memory. Modern computers meet the computational and storage requirements for 2D, 3D, and relatively low-resolution 4D spectra. Yet, spectra of higher dimensionality and/or resolution are still beyond reach, unless the analysis is performed in low dimensional projections or is reduced to small regions restricted in several spectral dimensions. In this work, we demonstrate for the first time a new approach allowing reconstruction, storage, and handling of high dimensionality and resolution spectra.

Reconstructing a spectrum with computational complexity and storage, which are sub-linear in respect to the number of points in the full spectrum (N^d) may only work by using NUS in the time domain and by computing a sparse representation of the spectrum, i.e. without producing the complete spectrum at any stage of the procedure. The latter requirement excludes powerful non-parametric NUS processing algorithms designed to reconstruct the full spectrum, such as Maximum Entropy (ME) (Barna et al. 1987; Hoch et al. 2014), Projection Reconstruction (PR) (Freeman and Kupce 2003), Spectroscopy by Integration of Frequency and Time Domain Information (SIFT) (Frey et al. 2013; Matsuki et al. 2009), Compressed Sensing (Holland et al. 2011; Kazimierczuk and Orekhov 2011), and Low Rank reconstruction (Qu et al. 2014). The parametric methods such as Bayesian (Bretthorst 1990), maximum likelihood (Chylla and Markley 1995), and multidimensional decomposition (MDD) (Jaravine et al. 2006) approximate the spectrum using a relatively small number of adjustable parameters, and thus are not limited in spectral dimensionality and resolution. However, due to the intrinsic problems of choosing the right model and convergence, the parametric algorithms cannot guaranty the detection of all significant signals in a large spectrum. Another approach Multidimensional Fourier Transform (MFT) (Kazimierczuk et al. 2006) for large spectra exploits prior knowledge about the signal positions in some or all of the spectral dimensions. MFT reconstructs only small regions around known spectral peaks and thus requires less computations and storage. The Signal Separation Algorithm (SSA) (Stanek et al. 2012), represents a combination of the parametric and MFT methods and to some extent inherits strong and weak points of both. Notably, the SSA also avoids dealing with the full spectrum matrices in time and frequency domains and can deal with large spectra. The method was demonstrated for high-resolution 4D spectra with the corresponding full sizes of tens of gigabytes.

In this paper, we introduce the sparse fast Fourier transform (SFFT) as the first non-parametric non-iterative algorithm capable of producing a high quality sparse representation for high resolution and dimensionality spectra. SFFT (Ghazi et al. 2013; Hassanieh et al. 2012a, b) is a rapidly developing field of signal processing methods, which is designed for reconstruction of a spectrum from a minimal number of sampled data points and with minimal computations. Similar to CS, the task is to use a small number of measurements in the time domain for reconstructing only the essential, i.e. above noise, data points in the frequency domain spectrum. The difference with CS is that the remaining data points are assumed to be exactly zero and thus can be omitted during the calculations and storage. Since for a sparse spectrum the number of both measured and reconstructed points is small, the amount of calculations may be significantly smaller than for the DFT of the complete spectrum. To the best of our knowledge, SFFT is the first method for NMR spectra processing that harness the Fourier projection-slice theorem in its discrete form. The method fills a several decade gap between the projection and non-uniform sampling approaches. By doing so, SFFT combines the benefits of these two worlds. Namely, while offering fast processing and requiring manageable data storage, which are sub-linear to the total number of points in the frequency spectrum, SFFT allows reconstruction of complete high quality ND spectra of any size and dimensionality. The method is most useful for high-resolution spectra of four and more dimensions, where methods like CS require too much computations and storage.

Below, we describe the SFFT algorithm used to process NMR spectra and demonstrate results for the reconstruction of a 4D BEST-HNCOCA spectrum.

The sparse FFT algorithm

The SFFT algorithm operates in two key steps: *bucketization* and *estimation*. In the *bucketization* step, SFFT divides the frequency spectrum into buckets where the value of each bucket represents the sum of the values of frequencies that map to that bucket. Since the spectrum is

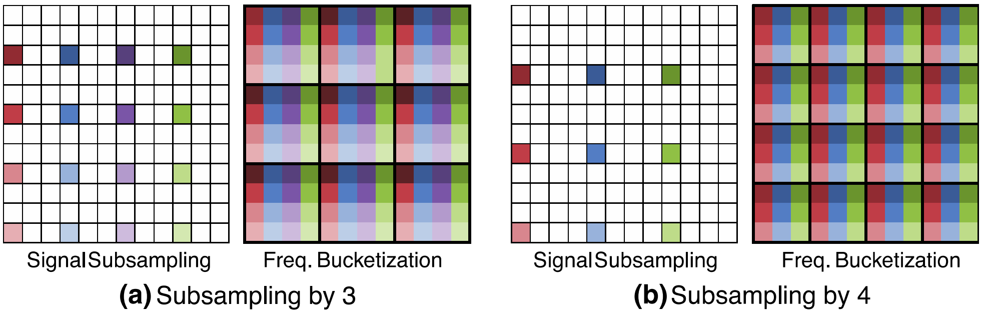


Fig. 1 Bucketization using co-prime aliasing on a 12 × 12 signal. **a** Subsampling by a factor of threefolds (aliases) the spectrum by 3. Frequencies with the same color sum up together in the same bucket. **b** Subsampling by a factor of 4, which is co-prime to 3 ensures that the frequencies will be bucketized differently avoiding collisions

sparse, many buckets will be empty and can be discarded. SFFT focuses on the non-empty buckets and computes the frequencies with large values in those buckets in the *estimation* step. Below we describe in details the *bucketization* and *estimation* techniques we use for NMR.

We will describe the SFFT algorithm for 2D signals of size $N \times N$. However, the algorithm can be easily extended to any dimension. We will use $\mathbf{X}$ to denote the 2D time signal and $\hat{\mathbf{X}}$ to denote its 2D discrete Fourier transform (DFT). We will use (f_1,f_2) to denote a frequency position and $\hat{\mathbf{X}}(f_1,f_2)$ to denote the spectral value at this frequency position. For simplicity, we will refer to frequencies that have no signal energy, i.e. just noise, as the zero frequencies and the frequencies that have signal energy as the non-zero frequencies. For a sparse spectrum, the number of non-zero frequencies is small and the goal of SFFT is to compute the positions and values of these non-zero frequencies.

Frequency bucketization

Bucketization using co-prime aliasing

To map frequencies into buckets, one approach is to use a well known property of the Fourier transform that is frequently used in NMR: *sub-sampling in the time domain causes aliasing in the frequency domain*. Let $\mathbf{B}$ be a sub-sampled version of $\mathbf{X}$, i.e. $\mathbf{B}(t_1,t_2) = \mathbf{X}(p \cdot t_1, p \cdot t_2)$ where p is the sub-sampling factor. Then, $\hat{\mathbf{B}}$, the FFT of $\mathbf{B}$, is an aliased version of $\hat{\mathbf{X}}$, i.e.:

$$\hat{\mathbf{B}}(b_1,b_2) = \sum_{i=0}^{p-1}\sum_{j=0}^{p-1}\hat{\mathbf{X}}(b_1 + i \cdot N/p, b_2 + i \cdot N/p) \quad (1)$$

Thus, aliasing is a form of bucketization in which frequencies equally spaced by an interval *N/p* map to the same bucket, i.e. frequency (f_1,f_2) maps to bucket number (b_1,b_2) such that:

$$(b_1,b_2) = (f_1 \bmod N/p, f_2 \bmod N/p) \quad (2)$$

Further, the value of each bucket is the sum of the values of only the frequencies that map to the bucket as can be seen from Eq. 1. Now that we mapped the frequencies into buckets, we can leverage the fact that the spectrum of interest is sparse and hence most buckets have noise and no signal. SFFT compares the energy (i.e. the magnitude squared) of a bucket with the noise level and considers all buckets whose energy is below a threshold to be empty. It then focuses on the occupied buckets and ignores empty buckets.

For occupied buckets, most of these buckets will have a single non-zero frequency. The value of this frequency can then be immediately evaluated from the value of the bucket since it is the only non-zero frequency in the bucket. However, some of these buckets will have more than one non-zero frequency. We refer to this as a *collision* of non-zero frequencies which prevents us from estimating the frequency. To resolve collisions, we need to repeat the bucketization but with different frequencies sharing the same bucket. In other words, we need to randomize how frequencies map into buckets. This can be done by using different sampling intervals which change how frequencies alias. In this case, we repeat the bucketization with a different sampling factor p.

So how should we choose the different sampling factors to randomize the bucketization? The best choice of is to use co-prime aliasing. Thus, if the first bucketization uses as sampling factor p_1, then the second bucketization should use a sampling factor p_2 such that p_1 and p_2 are co-prime i.e. the greatest common divisor of p_1 and p_2 is equal to 1. Co-prime aliasing guarantees that any two frequencies that collide in the first bucketization will not collide in the second bucketization. Figure 1 shows an example of bucketization using co-prime aliasing, which illustrates which frequencies map to which buckets in each of the bucketizations.

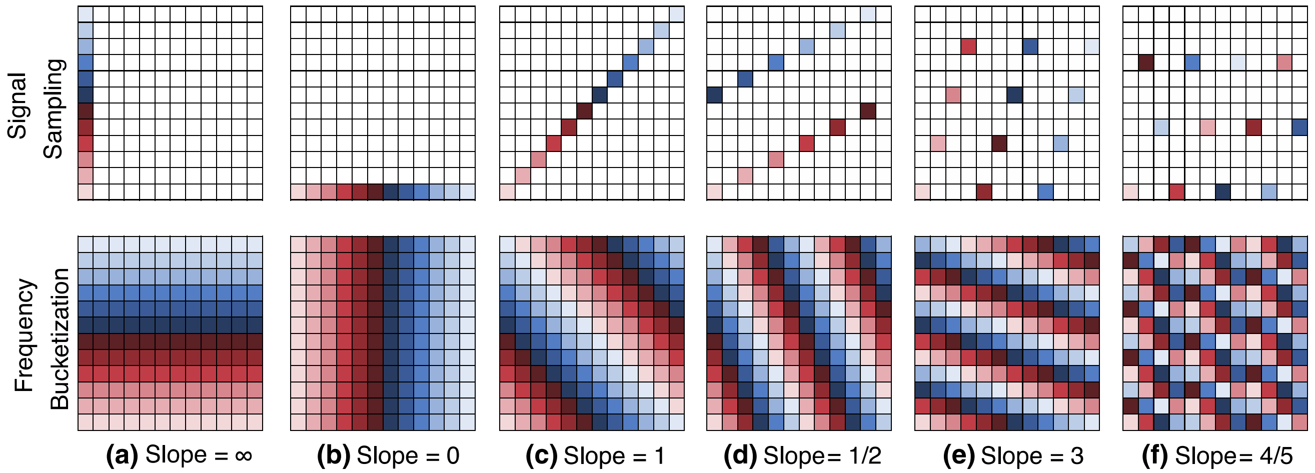


Fig. 2 Bucketization using discrete lines. The *top* shows the discrete line, which was sampled and the *bottom* shows how the frequencies are projected. Frequencies with the same color will sum up together in the same bucket. This is shown for different slope **a**–**f**. Since the lines are discrete, they wrap around and can result in pseudo random sampling and projection patterns as can be seen in **f**

Bucketization using discrete line projections

Another way to bucketize the spectrum is to perform a 1D DFT of discrete lines. This yields the projection of the spectrum onto a corresponding line in the Fourier domain. Specifically, let **y** be the 1D discrete line corresponding to a 2D signal **X**, parameterized by $t \in [0, \cdots, N-1]$:

$$\mathbf{y}(t) = \mathbf{X}(\alpha_1 t \bmod N, \alpha_2 t \bmod N) \tag{3}$$

where α_1, α_2 are integers whose greatest common divisor is invertible modulo N such that $0 \leq \alpha_1, \alpha_2 < N$. α_1/α_2 represents the slope of the line. Then $\hat{\mathbf{y}}$, the DFT of **y**, is a projection of $\hat{\mathbf{X}}$ onto this line. That is each point in $\hat{\mathbf{y}}$ is a summation of the N frequencies that lie on a discrete line orthogonal to y as shown in Fig. 2. Specifically, the frequencies (f_1,f_2) that satisfy $\alpha_1 f_1 + \alpha_2 f_2 = f \bmod N$ will project together into the same bucket f and sum up to $\hat{\mathbf{y}}(f)$. Figure 2 shows some examples of discrete lines and their projections. Note that discrete lines (Eq. 3) wrap around as can be seen in Fig. 2d–f and the bucketization can result in a pseudo random non-uniform sampling as shown in Fig. 2f. Also note that this can be extended to any number of dimensions. In that case, we can take projections of discrete lines, planes or hyper-planes.

The above procedure is based on the Fourier projection-slice theorem (Bracewell 1956) and thus bears resemblance to the reduced dimensionality techniques and radial sampling (Bodenhausen and Ernst 1981; Coggins et al. 2010; Hiller et al. 2005; Szyperski et al. 1993). The important difference, however, is that the sampling defined by Eq. 3 is performed on the Nyquist time domain grid of the full multidimensional experiment, while the traditional radial sampling is off-grid. As it is described in the next section, having all sampled point on the grid allows direct frequency estimation without resorting to the often problematic inverse Radon transform used in the traditional projection reconstruction (Kupce and Freeman 2004).

Further, discrete projections can benefit from the complex virtual echo representation (Mayzel et al. 2014), which improves the sparsity. Once, we sample a discrete line passing through the origin (0, 0), we automatically obtain the samples of another discrete line which is symmetric to the first line with respect to one of the axes i.e. if we sample a line with slope α_1/α_2, we directly get the line with slope $-\alpha_1/\alpha_2$. For higher dimensions the gain is larger. If we sample a discrete line in a d-dimensional signal, we automatically get the samples of $2^d - 1$ other discrete lines. For example, in 3D for a discrete line with slope $(\alpha_1, \alpha_2, \alpha_3)$, we get the samples of three other discrete lines which are $(-\alpha_1, \alpha_2, \alpha_3)$, $(\alpha_1, -\alpha_2, \alpha_3)$, $(\alpha_1, \alpha_2, -\alpha_3)$. Note that $(-\alpha_1, -\alpha_2, \alpha_3)$ and $(\alpha_1, \alpha_2, -\alpha_3)$ define the same projections and thus only one of these is needed.

Choosing the bucketization and number of buckets

The choice of bucketization and number of buckets depends on the sparsity. If the signal has k non-zero frequency peaks, then the number of buckets in each bucketization should be at least $O(k)$ or larger. The discrete projections and aliasing approaches give us a lot of flexibility in choosing the number of buckets. For example, in a 4D signal, if k is very large we can project on 2D discrete planes to get N^2 buckets or 3D discrete hyper-planes to get N^3 buckets. If k is small, we can project on 1D discrete lines to get N buckets. We can also combine discrete projections with aliasing to accommodate almost any value

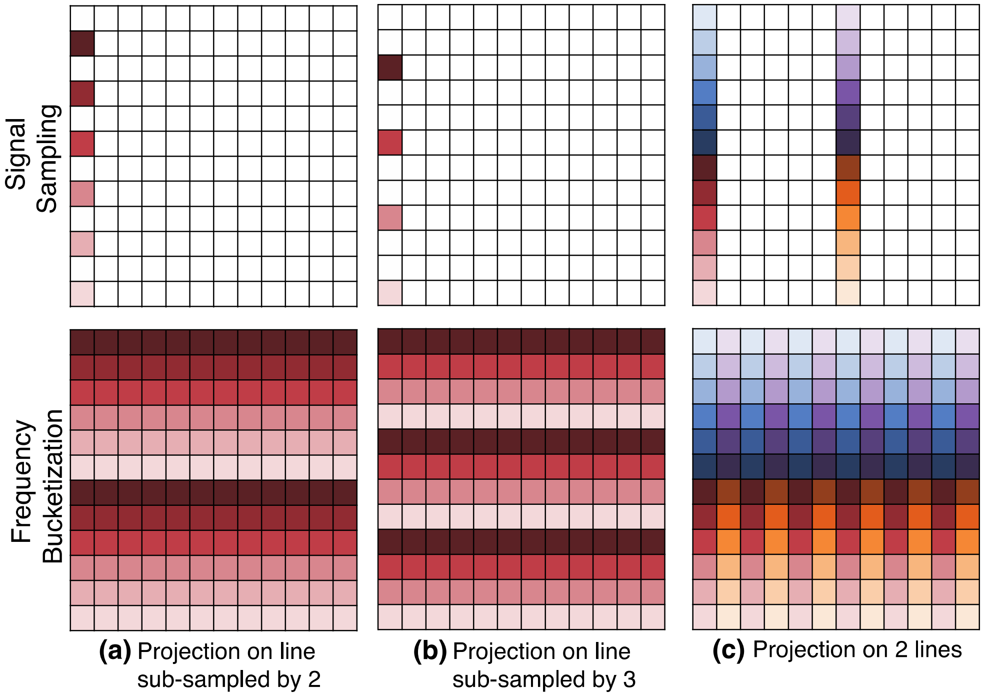

Signal Sampling
Frequency Bucketization
(a) Projection on line sub-sampled by 2
(b) Projection on line sub-sampled by 3
(c) Projection on 2 lines

Fig. 3 Combining discrete projections with aliasing. **a**, **b** Projection on sub-sampled discrete line gives number of buckets less than N. **c** Projection on two lines (i.e. sub-sampled plane) gives number of buckets larger than N. Frequencies with the same color sum up together in the same bucket

of k. For example, we can project on sub-sampled lines as shown in Fig. 3a, b to get $N/2$ or $N/3$ buckets. We can also project on sub-sampled plane as shown in Fig. 3c to get $2N$ buckets. In the next section, we explain how the choice of number of buckets affects the running time of the algorithm.

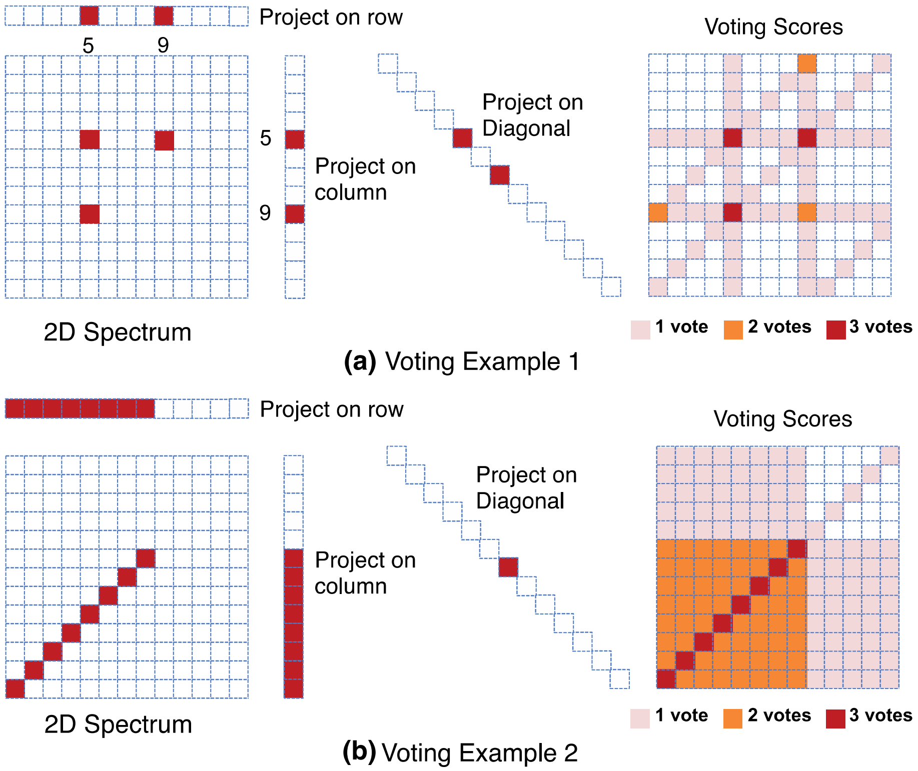
Frequency estimation

In this step, for each of the occupied buckets we want to identify which frequencies created the energy in these buckets, and what are the values of these frequencies. If we can do that, we then have recovered a complete representation of the frequencies with non-zero signal values, i.e. we acquired the full signal in the Fourier domain.

The frequency identification procedure uses a voting based approach where the occupied buckets vote for the frequencies that map to them. Since the spectrum is sparse, most of the buckets are empty and hence only few frequencies get votes each time. Because by definition the non-zero frequencies will end up in occupied buckets, they will get a vote every time we perform a new bucketization. In practice, a non-zero frequency may miss the votes in some of the bucketizations. This may happen when the corresponding spectral peak is very weak and/or is cancelled by superposition with a peak of the opposite sign. Such negative peaks may be present in the spectrum, for example, in case of the peak aliasing when the acquisition is started from half-dwell time. Nevertheless, after performing a few random bucketizations by using co-prime aliasing or discrete lines with different slopes (Eq. 3), the non-zero frequencies will have the largest number of votes, which allows SFFT to identify these frequencies.

To better illustrate this voting technique, consider a simple example shown in Fig. 4a. The 2D spectrum has only 3 nonzero frequencies at (5, 5), (5, 9) and (9, 5). When we perform bucketization by projecting on a vertical line (row), the 5th and 9th buckets will be large and will vote for all frequencies in the 5th and 9th columns. Similarly, when we perform bucketization by projecting onto a horizontal line (column), the projection will vote for all frequencies in the 5th and 9th rows. At this point, only frequencies (5, 5), (5, 9), (9, 5), (9, 9) have two votes. However, when we project on the diagonal, frequency (9, 9) will not get a vote since its projection is not large. After 3 projections only the non-zero frequencies will get 3 votes. Another example is shown in Fig. 4b.

Now that we have a list of non-zero frequencies (f_1,f_2), we want to estimate the values $\hat{\mathbf{X}}(f_1,f_2)$ of these frequencies. We may use a simple approach analogous to those used in the method of projection reconstruction (Kupce and Freeman 2004). It would estimate the value of each non-zero frequency as the median value of the different buckets to which this frequency was mapped across the different



(a) Voting Example 1

(b) Voting Example 2

Fig. 4 Two examples of the voting approach used to recover the discrete positions of the large frequencies from projections on discrete lines. The 2D spectrum is projected on a row, a column and a diagonal. Each large projection votes for the frequencies that map to it. Using only projections on a row and column, many frequencies get two votes. By adding a 3rd projection on the diagonal, only the large frequencies get 3 votes. **a** Frequencies (5, 5), (5, 9), and (9, 5) are large and only they get 3 votes. **b** Some frequencies on the diagonal are large and only these frequencies get 3 votes

bucketizations. However, this approach may yield a poor reconstruction in the presence of noise and significant signal overlap. Instead, we can compute better estimates of the values of the non-zero frequencies by harnessing the fact that all these frequencies are defined at the same Nyquist grid. The values of the occupied buckets can be viewed as linear combinations of the values of the non-zero frequencies. Hence, we can construct a linear system of equations:

$$A\mathbf{x} = \mathbf{b} \tag{4}$$

where the unknown vector $\mathbf{x}$ corresponds to the values of the non-zero frequencies and the known vector $\mathbf{b}$ corresponds to the values of the buckets. The matrix A is a sparse binary matrix that specifies which frequency values contribute to which buckets. In general, this system is over-determined since the number of occupied buckets can be as large as the number of non-zero frequencies times the number of bucketizations. Hence, the solution that minimizes the mean square error of the frequency values is:

$$\mathbf{x}^* = A^\dagger \mathbf{b} \tag{5}$$

where $A^\dagger$ is the is the pseudo inverse of A. This approach of computing the values of non-zero frequencies is more robust to noise and can correct for errors by estimating the falsely presumed non-zero frequencies to near zero values. This comes at the cost of the additional computational complexity associated with computing the pseudo inverse. However, since the number of non-zero frequencies is small, the size of the matrix A is still small.

Running time and sampling complexity

For a signal $\mathbf{X}$ with d dimensions each of which is of size N i.e. total number of samples is N^d, the signal is sparse in the frequency domain if the number of non-zero frequencies is $k \ll N^d$. The running time, storage and sampling complexity depends on the choice of the number of buckets. The discrete line projections, co-prime aliasing algorithms and their combinations allow much freedom for designing sub-sampling schemes and, in particular, for selecting the number of buckets. Thus, one can choose to use one or higher dimensional projections with different degree of aliasing as explained in the end of the bucketization section. To minimize the number of samples and the storage, SFFT would use O(k) buckets. However, this would generate a lot of votes and would take a linear time to process. Alternatively to minimize the running time, SFFT would

Table 1 The sampling and running time complexity of SFFT as compared with CS and FFT

	Samples	Storage	Time
FFT	$O(N^d)$	$O(N^d)$	$O(N^d \log N^d)$
CS	$O(k \log N^d)$	$O(N^d)$	$O(poly(N^d))$
SFFT	$O(k \log N^d)$	$O(k^2 \log N^d)$	$O((N^d + k^3) \log N^d)$
	$O\left(\sqrt{kN^d \log N^d}\right)$	$O\left(\left(\sqrt{kN^d \log N^d} + k^2\right) \log N^d\right)$	$O\left(\left(\sqrt{kN^d \log N^d} + k^3\right) \log N^d\right)$

use $O\left(\sqrt{kN^d / \log N^d}\right)$ buckets, which gives a sub-linear running time faster than FFT as well as sub-linear storage. Table 1 shows the number of samples and running time of SFFT for these two settings of number of buckets. The table also shows the running time and sampling complexity for FFT and compressive sensing (CS). CS uses the minimum number of samples. However, CS has a running time, which is polynomial in the size of the signal.

Materials and methods

The 4D fully sampled BEST-HNCOCA (Lescop et al. 2007) spectrum of 1.7 mM human ubiquitin sample (H2O/D2O 9:1, pH 4.6) was acquired at 25 °C on 800 MHz Bruker AVANCE III HD spectrometer equipped with 5 mm CP-TCI probe with the Nyquist grid of 16 × 16 × 16 complex time points (acquisition times 12, 10 and 4.4 ms) for the ^{15}N, ^{13}CO and ^{13}Cα spectral dimensions, respectively. The amide region of the full reference 4D spectrum (9.7–7.0 ^{1}H ppm, 174 points) was processed using NMRPipe software (Delaglio et al. 1995). For the SFFT processing, only the directly detected dimension was processed in NMRPipe followed by extraction of the same amide region. The hyper-complex time domain data were converted to the complex virtual echo (VE) representation (Mayzel et al. 2014) with dimensions 174 × 32 × 32 × 32. Unlike the original hyper-complex data representation, the VE is directly amenable for the multi-dimensional SFFT processing and improves the result of the reconstruction from the NUS data. However, we should remind that the VE relies on the prior knowledge of the phase and require the linear phase correction in the indirectly detect dimensions of the ND spectrum to be multiple of π (i.e. 0, π, 2π,…). Two independent sampling tables were generated using Discrete Line Projections algorithm described in the theory section (Eq 3). Each of the tables contained 6 orthogonal projections, i.e. (1, 0, 0), (0 1 0), (1 1 0), (1 0 1), (0 1 1), (1 1 1), and 16 projections obtained by random combinations of prime numbers less than 32 (i.e. 0 1 2 3 5 7 11 13 17 23 29 31). As described in the theory, these 6 + 16 unique line projections were augmented by 7 + 48 symmetric projections, respectively, which resulted in total 77 line projections in the SFFT calculations. In total, each NUS data set included 2096 (6.4 %) out of total 32 k complex time domain points in the indirectly detected dimensions. Supplementary Figure S2 shows a few examples of the obtained discrete line projections. The number of used samples or discrete lines depends on the sparsity of the 3D spectra. Since, for all 174 directly detected points, the same indirectly detected points are used, the minimum number of samples needed is bounded by the 3D sub-spectrum with the lowest sparsity, i.e. the largest part occupied by signals. Although, the same discrete lines are used in all 3D sub-spectra, the cut-off threshold for selecting frequencies varies for different directly detected points. SFFT adaptively sets the cut-off threshold by ensuring that the system of linear equations in Eq. 4 is well determined. This allows lowering the cut-off and thus improving sensitivity for regions with small number of signals.

The SFFT calculations were performed in MATLAB with the resulting spectrum exported to NMRPipe format for comparison with the reference spectrum. The code will soon be available from the authors.

Results and discussion

We demonstrate SFFT by reconstructing 4D HNCOCA spectrum using a NUS data set extracted from the complete experiment, which is acquired with 512 × 16 × 16 × 16 complex time points for the ^{1}H, ^{15}N, ^{13}CO and ^{13}Cα spectral dimensions, respectively. After conventional Fourier processing of the directly detected ^{1}H dimension and extraction of the amide region 9.7–7.0 ^{1}H ppm (174 points), the Discrete Line Projections algorithm (Eq 3) selected 262 (6.4 %) hyper complex time domain points in the indirectly detected dimensions. In a real experiment, of course, only these selected data points need to be acquired, thus reducing the measurement time to 6.4 % of the full experiment. The selected hyper-complex data points were converted to the complex virtual echo representation (Frey et al. 2013; Mayzel et al. 2014), which contained 174 × 2096 points out of the full complex array with dimensions 174 × 32 × 32 × 32. Then, in the frequency domain, the SFFT voting algorithm identified 10,588 non-zero points, which correspond to approximately 100 peaks with 100 data points per peak in the 4D spectrum. In the resulting spectrum, only these non-zero intensities were stored, which constitute to <0.2 % of the full reference spectrum.

The running time of the SFFT is dominated by the time to compute the projections and perform the pseudo inverse (Eq. 5). For the current experiment, the time to compute all projections in MATLAB is 0.25 ms and the time to perform the pseudoinverse is around 150 ms. CS based algorithms like IST would require between 10 and 100 iteration while performing a full FFT on 3D spectra and hence take between 30 and 300 ms. The computational advantage of the SFFT is expected to increase for higher resolution and dimensions. However, a more thorough analysis of runtime would require implementing the SFFT algorithm in C/C++ and is thus is left for future work.

A few points are worth noting. First, the pseudoinverse matrix is computed separately for each point in the directly detected dimension. Thus, the size of this matrix depends on the number of peaks in each of the 3D spectra of indirectly detected dimensions as opposed to the number of peaks in the entire 4D spectrum. The pseudoinverse of the matrix used in our work (ca 2000 × 250), takes 0.15 s. Hence, calculating the pseudoinverse fits well into a desktop computer memory and scales as the third power of the matrix size. Even for a more demanding case of quadruple matrix size required for a large system or NOESY type spectrum, the calculation will take <40 s per point in the directly detected spectral dimension. Second, the SFFT algorithm can naturally benefit from prior knowledge about the dark regions in the spectrum in a similar manner to the SIFT method. For example, we can compute the pseudoinverse only for the peaks, which we want to estimate. We can also avoid collecting votes for frequencies that we know do not contain energy. However, these extensions of the SFFT algorithm are left for future work.

Figure 5 illustrates the SFFT reconstructed spectrum using two different approaches for the evaluation of the frequency values. Comparison of panels a, b and c, d in Fig. 5 shows that the spectrum obtained using the matrix inversion (Eqs. 4, 5) is very similar to the full reference spectrum. This visual impression is corroborated by the correlation (Fig. 5e) of the cross-peak intensities between the full reference and SFFT spectrum. It can be seen that most of the peaks found in the reference spectrum (red circles) are faithfully reproduced in the SFFT reconstruction. Results of spectral reconstructions from NUS may vary for different sampling schedules (Aoto et al. 2014). In order to check this we calculated SFFT spectrum with an alternative set of randomly selected projections (Supplementary Figure S1). The two independent SFFT spectral reconstructions had comparable quality. Pairwise correlations between the peak intensities in the reference spectrum and in the two independent SFFT reconstructions were very similar. 98 peaks were detected in the reference spectrum using the peak-picker program from NMRPipe software (Delaglio et al. 1995) with the noise level of 0.01 (in the scale used in Fig. 5e, f) and peak detection threshold 0.05. Thanks to the high spectral sensitivity for the 1.7 mM ubiquitin sample, the signal dynamic range in the reference spectrum reached 1:50, which covers the range typically found in the triple resonance experiments for assignment and approaches the dynamic range in 4D NOESY spectra.

Practical sensitivity in a spectrum can be defined as a level of reliable signal detection, i.e. separation of true signals from noise and spectral artefacts. The weakest detected peak in the reference spectrum has intensity 0.12. Out of total 98, five weakest peaks with intensity up to 0.25 were lost in the SFFT reconstruction. No peaks above this level were missing. The observed decrease of the sensitivity seems reasonable considering that duration of the SFFT experiment is only 6.4 % of the reference and thus up to four times drop of sensitivity is expected for the SFFT reconstruction. In the SFFT voting algorithm, the frequency detection is limited by the sensitivity in the individual projections, whose measurement time was 1/77 of the total SFFT experiment time. On the other hand, combined analysis of many projections provides efficient cross-validation of the selected frequencies and allows lowering of the detection threshold in the individual projections (Supplementary Figure S2). Similar to the geometric analysis of projections (GAPRO) algorithm in APSY (Hiller et al. 2005), the SFFT voting procedure, recovers large part of the sensitivity that is lost due to short measurement time of the projections. It should be noted also that in SFFT, purpose of the frequency identification voting algorithm is not to find peaks but to select frequencies, which are worth for the evaluation. The detection limit corresponds to a trade-off between the number of points selected for the evaluation and the requirements for low computational complexity and data storage. Thus, lowering of the detection threshold does not lead to many additional false peaks but only increases the computational complexity and storage. Whatever threshold level is used, the weakest peaks are inevitably lost at the frequency identification step of the SFFT algorithm and consequently have zero intensities in the SFFT reconstruction. Thus, the SFFT, as well as many other NUS processing methods, should be used with caution for spectra with high dynamic range and when detection of the peaks close to the signal-to-nose limit is important, e.g. for NOESY's.

The correlation for the peaks picked in the SFFT spectrum is shown in Fig. 5e, f with blue crosses. The peaks were detected by NMRPipe peak-picker with the same

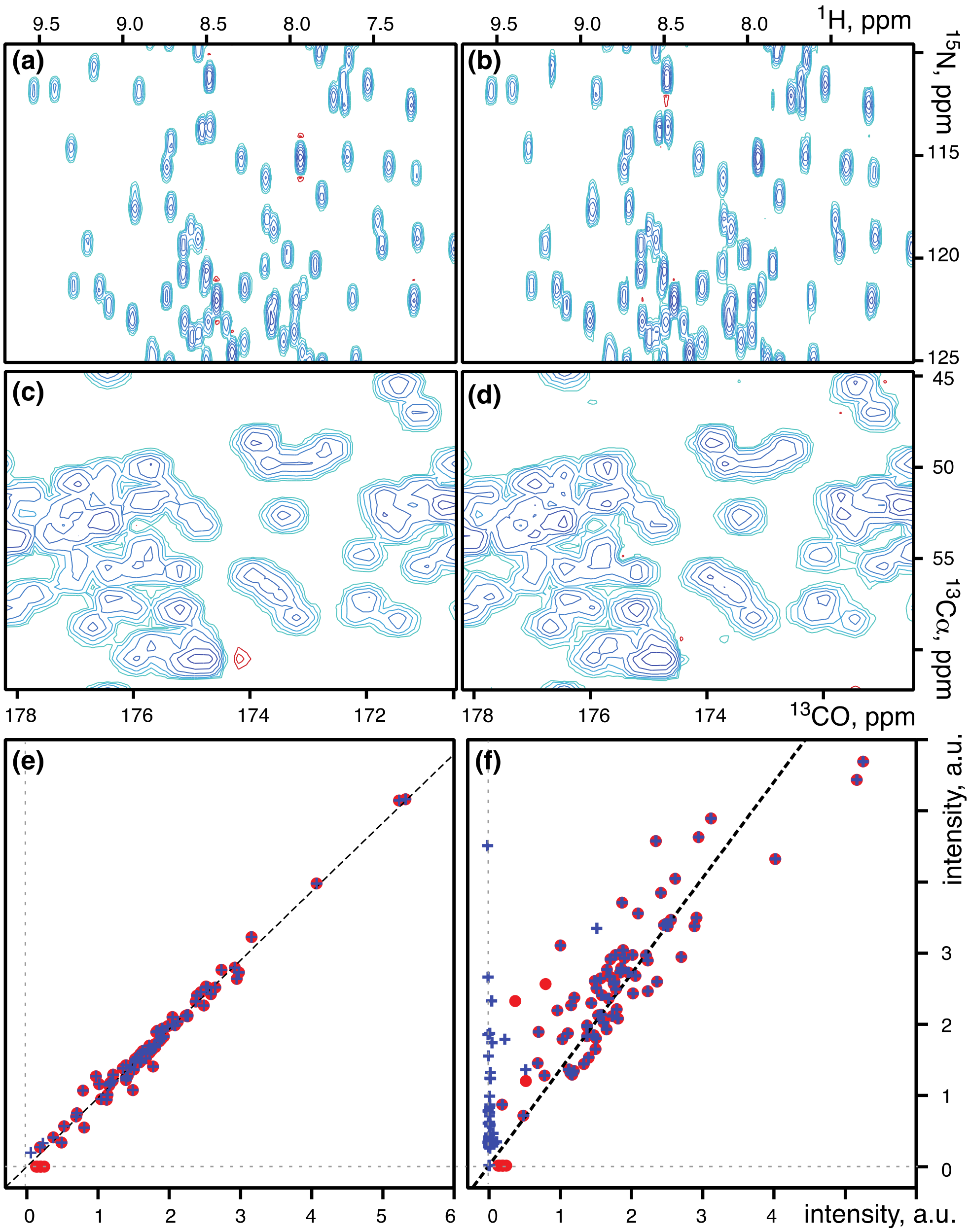


Fig. 5 4D BEST-HNCOCA spectrum of ubiquitin. Orthogonal ^{1}H/^{15}N (**a**, **b**) and **c**, **d** ^{13}CO/^{13}Cα projections of fully sampled and FFT processed (**a**, **c**) and 6.4 % NUS processed with SFFT processed (**b**, **d**). **e**, **f** Correlation of peak intensities measured in the full reference spectrum (abscissa) and the SFFT reconstruction (ordinate) using the matrix inversion (**e**) and median estimation (**f**). *Dark red circles* and *blue crosses* show intensities measured at the positions of peaks picked in the reference and SFFT spectra, respectively

noise and detection threshold parameters as for the reference spectrum. This plot is intended for revealing peaks in the SFFT reconstruction that are not present in the reference spectrum, i.e. false positives. As it is seen in Fig. 5e, f, while the median algorithm for the frequency evaluation resulted in many false peaks, no false peaks were detected when the SFFT reconstruction was obtained using the matrix inversion method. As expected, the median method also provided less accurate peak intensities. Notably, both methods evaluate intensity of the same set of frequencies, which are defined at the common frequency identification step of the SFFT algorithms. Thus, the matrix inversion method effectively supresses the false positive peaks. Apart from the signals, which were identified by the peak-picker as peaks, the SFFT spectrum obtained by the matrix inversion method contained a number of signals with intensities lower than the peak detection cut-off. In addition, there were several relatively low intensity (<0.3) signals, which didn't pass the peak quality checks (see an example in Supplementary Figure S3). In most cases such signals were represented by only one point in two or more spectral dimensions. The reduced dimensionality data collection used by the SFFT may be prone to false peak artefacts that are not, in general, the result of a method used to compute spectra, but are intrinsic for this type of data sampling (Mobli et al. 2006), especially in the case of signals with high dynamic range. Thus, it is unlikely that the SFFT will be able to produce reconstructions for small and medium size spectra that are better than the modern NUS-based techniques, such as CS. On the other hand, the computational and storage efficiency of the SFFT are well suited for the large spectra, i.e. 4Ds and above, where the full spectral reconstructions and computationally demanding algorithms often fail while methods based on the radial sampling (e.g. APSY) are efficiently used. For example we envisage that SFFT will be instrumental in high-dimensional spectra of the Intrinsically Disordered Proteins that often exhibit long transverse relaxation times and heavy peak overlap (Motáčková et al. 2010).

Typically, the number of cross-peaks does not increase with spectrum dimensionality and resolution. Consequently, the number of non-zero frequencies, which is related to the number of cross-peaks, only moderately increases proportionally to dimensionality and resolution of the spectrum. This makes it possible for the SFFT to handle very large spectra.

Another good feature of the technique is that the data sampling using the discrete line projections (Eq. 3) and voting algorithm used by the SFFT for the identification of non-zero frequencies are fully compatible with the optimization by incremental data collection and analysis (Eghbalnia et al. 2005; Jaravine and Orekhov 2006). For example we can envisage an approach where an experiment is continued until the list of identified frequencies stabilizes and reaches a plateau. Thus, the number of projections can be adjusted for every experiment.

Finally, SFFT represents a group of rapidly developing algorithms successfully applied in multitude of technical applications. From the NMR perspective, the method for the first time combines the best features of so far distinctly different approaches known as reduced dimensionality and compressed sensing. The former is robust and computationally very efficient, the later provides highest quality spectral reconstructions. In this work we presented the NMR tailored version of the SFFT algorithm and demonstrated its performance for 4D BEST-HNCOCA spectrum of ubiquitin.

Acknowledgments The work was supported by the Swedish Research Council (Research Grant 2011-5994); Swedish National Infrastructure for Computing (Grant SNIC 001/12-271). The Swedish NMR Centre is acknowledged for spectrometer time.

References

Aoto PC, Fenwick RB, Kroon GJA, Wright PE (2014) Accurate scoring of non-uniform sampling schemes for quantitative NMR. J Magn Reson 246:31–35. doi:10.1016/j.jmr.2014.06.020

Barna JCJ, Laue ED, Mayger MR, Skilling J, Worrall SJP (1987) Exponential sampling, an alternative method for sampling in two-dimensional NMR experiments. J Magn Reson 73:69–77

Billeter M, Orekhov VY (2012) Preface: fast NMR methods are here to stay. In: Billeter M, Orekhov VY (eds) Novel sampling approaches in higher dimensional NMR, vol 316. Springer, Heidelberg, pp ix–xiv

Bodenhausen G, Ernst RR (1981) The accordion experiment, a simple approach to 3-dimensional NMR spectroscopy. J Magn Reson 45:367–373. doi:10.1016/0022-2364(81)90137-2

Bracewell RN (1956) Strip integration in radio astronomy. Aust J Phys 9:198–217

Bretthorst GL (1990) Bayesian-analysis. 1. Parameter-estimation using quadrature NMR models. J Magn Reson 88:533–551. doi:10.1016/0022-2364(90)90287-j

Chylla RA, Markley JL (1995) Theory and application of the maximum-likelihood principle to NMR parameter-estimation of multidimensional NMR data. J Biomol NMR 5:245–258

Coggins BE, Venters RA, Zhou P (2010) Radial sampling for fast NMR: concepts and practices over three decades. Prog Nucl Magn Res Spectrosc 57:381–419. doi:10.1016/j.pnmrs.2010.07.001

Delaglio F, Grzesiek S, Vuister GW, Zhu G, Pfeifer J, Bax A (1995) NMRPipe: a multidimensional spectral processing system based on UNIX pipes. J Biomol NMR 6:277–293

Eghbalnia HR, Bahrami A, Tonelli M, Hallenga K, Markley JL (2005) High-resolution iterative frequency identification for NMR as a general strategy for multidimensional data collection. J Am Chem Soc 127:12528–12536. doi:10.1021/ja052120i

Freeman R, Kupce E (2003) New methods for fast multidimensional NMR. J Biomol NMR 27:101–113. doi:10.1023/a:1024960302926

Frey MA, Sethna ZM, Manley GA, Sengupta S, Zilm KW, Loria JP, Barrett SE (2013) Accelerating multidimensional NMR and MRI experiments using iterated maps. J Magn Reson 237:100–109. doi:10.1016/j.jmr.2013.09.005

Ghazi B, Hassanieh H, Indyk P, Katabi D, Price E, Shi L (2013) Sample-optimal average-case sparse Fourier transform in two dimensions. In: 51st annual Allerton conference on communication, control, and computing, October 2013

Hassanieh H, Indyk P, Katabi D, Price E (2012a) Nearly optimal sparse fourier transform. In: The 44th symposium on theory of computing, STOC

Hassanieh H, Indyk P, Katabi D, Price E (2012b) Simple and practical algorithm for sparse fourier transform. In: Twenty-third annual ACM-SIAM symposium on discrete algorithms, SODA

Hiller S, Fiorito F, Wuthrich K, Wider G (2005) Automated projection spectroscopy (APSY). Proc Natl Acad Sci USA 102:10876–10881. doi:10.1073/pnas.0504818102

Hiller S, Wasmer C, Wider G, Wuthrich K (2007) Sequence-specific resonance assignment of soluble nonglobular proteins by 7D APSY-NMR spectroscopy. J Am Chem Soc 129:10823–10828. doi:10.1021/ja072564+

Hoch JC, Maciejewski MW, Mobli M, Schuyler AD, Stern AS (2014) Nonuniform sampling and maximum entropy reconstruction in multidimensional NMR. Acc Chem Res 47:708–717. doi:10.1021/ar400244v

Holland DJ, Bostock MJ, Gladden LF, Nietlispach D (2011) Fast multidimensional NMR spectroscopy using compressed sensing.

Angew Chem Int Ed 50:6548–6551. doi:10.1002/anie.201100440

Hyberts SG, Arthanari H, Robson SA, Wagner G (2014) Perspectives in magnetic resonance: NMR in the post-FFT era. J Magn Reson 241:60–73. doi:10.1016/j.jmr.2013.11.014

Jaravine VA, Orekhov VY (2006) Targeted acquisition for real-time NMR spectroscopy. J Am Chem Soc 128:13421–13426. doi:10.1021/Ja062146p

Jaravine V, Ibraghimov I, Orekhov VY (2006) Removal of a time barrier for high-resolution multidimensional NMR spectroscopy. Nat Methods 3:605–607. doi:10.1038/Nmeth900

Kazimierczuk K, Orekhov VY (2011) Accelerated NMR spectroscopy by using compressed sensing. Angew Chem Int Ed 50:5556–5559. doi:10.1002/anie.201100370

Kazimierczuk K, Kozminski W, Zhukov I (2006) Two-dimensional Fourier transform of arbitrarily sampled NMR data sets. J Magn Reson 179:323–328. doi:10.1016/j.jmr.2006.02.001

Kazimierczuk K, Zawadzka-Kazimierczuk A, Koźmiński W (2010) Non-uniform frequency domain for optimal exploitation of non-uniform sampling. J Magn Reson 205:286–292. doi:10.1016/j.jmr.2010.05.012

Kupce E, Freeman R (2004) Projection-reconstruction technique for speeding up multidimensional NMR spectroscopy. J Am Chem Soc 126:6429–6440. doi:10.1021/ja049432q

Lescop E, Schanda P, Brutscher B (2007) A set of BEST triple-resonance experiments for time-optimized protein resonance assignment. J Magn Reson 187:163–169. doi:10.1016/j.jmr.2007.04.002

Matsuki Y, Eddy MT, Herzfeld J (2009) Spectroscopy by integration of frequency and time domain information for fast acquisition of high-resolution dark spectra. J Am Chem Soc 131:4648–4656. doi:10.1021/ja807893k

Mayzel M, Kazimierczuk K, Orekhov VY (2014) The causality principle in the reconstruction of sparse NMR spectra. Chem Commun (Camb) 50:8947–8950. doi:10.1039/C4CC03047H

Mobli M, Stern AS, Hoch JC (2006) Spectral reconstruction methods in fast NMR: reduced dimensionality, random sampling and maximum entropy. J Magn Reson 182:96–105. doi:10.1016/j.jmr.2006.06.007

Motáčková V, Nováček J, Zawadzka-Kazimierczuk A, Kazimierczuk K, Žídek L, Šanderová H, Krásný L, Koźmiński W, Sklenář V (2010) Strategy for complete NMR assignment of disordered proteins with highly repetitive sequences based on resolution-enhanced 5D experiments. J Biomol NMR 48:169–177. doi:10.1007/s10858-010-9447-3

Orekhov VY, Jaravine VA (2011) Analysis of non–uniformly sampled spectra with multi-dimensional decomposition. Prog Nucl Magn Reson Spectrosc 59:271–292

Qu X, Mayzel M, Cai J-F, Chen Z, Orekhov V (2014) Accelerated NMR spectroscopy with low-rank reconstruction. Angew Chem Int Ed. doi:10.1002/anie.201409291

Stanek J, Augustyniak R, Koźmiński W (2012) Suppression of sampling artefacts in high-resolution four-dimensional NMR spectra using signal separation algorithm. J Magn Reson 214:91–102. doi:10.1016/j.jmr.2011.10.009

Szyperski T, Wider G, Bushweller JH, Wuthrich K (1993) Reduced dimensionality in triple-resonance NMR experiments. J Am Chem Soc 115:9307–9308. doi:10.1021/ja00073a064