Simulating Metabolite Basis Sets for *in vivo* MRS Quantification

Incorporating details of the PRESS Pulse Sequence by means of the GAMMA C++ library

J.W. van der Veen, MRS Core Facility, NIMH, NIH, USA
D. van Ormondt and R. de Beer, Applied Physics, TU Delft, NL, r.debeer@tudelft.nl

***Abstract*—In this work we report on generating/using *simulated* metabolite basis sets for the quantification of *in vivo* MRS signals, assuming that they have been acquired by using the PRESS pulse sequence. To that end we have employed the classes and functions of the GAMMA C++ library.**

By using several versions of our PRESS-simulation program, we were able to study the single-voxel selection, required for detecting *in vivo* MRS signals. Furthermore, by introducing in one of the versions a modified spatial summation scheme, that comes down to crusher-gradient averaging, we could realize a decrease in computation time by about a factor of 256.

We have used four different simulated metabolite basis sets in the quantification of a real-world 3T human-brain ^{1}H MRS signal. The best quantification is obtained, when including into the simulation program −as closely as possible− the related details of the PRESS-based single-voxel selection.

***Index Terms*—*In vivo* MRS, quantification, metabolite basis sets, PRESS simulation, GAMMA C++ library, single-voxel selection, crusher gradients, *j*MRUI QUEST.**

I. Introduction

Recently we have reported on GammaPress [1], a custom plug-in for the *j*MRUI signal-analysis package [2] [3]. By using specific C++ classes and functions for carrying out the density matrix formalism, as provided by the GAMMA C++ library [4], the GammaPress plug-in enables the *j*MRUI user − via a Graphical User Interface (GUI) approach − to generate *simulated* metabolite basis sets for the quantification of *in vivo* Magnetic Resonance Spectroscopy (MRS) signals. To that end it is assumed, that the *in vivo* MRS signals, to be quantified, have been measured by means of the PRESS [5] pulse sequence and that the related experimental details are known.

In the work, presented here, we describe the results of generating metabolite basis sets with related *standalone* versions of the GAMMA-based PRESS-simulation program, just mentioned. That is to say, with GAMMA C++ code, not embedded in the Java-based structure of the *j*MRUI plug-in platform [2]. The reason is, that we have applied the simulated metabolite basis sets to the quantification of a real-world *in vivo* MRS signal, of which some specific details of the PRESS experiment, concerned, are not included in the −*standard* PRESS [6] addressing− GUI of GammaPress.

II. Methods

A. *Simulating PRESS-based* in vivo *MRS experiments*

1) Main structure of the PRESS-simulation program: In its most basic form [5] the *in vivo* PRESS pulse sequence amounts to applying a 90^0 selective RF excitation pulse, followed by two 180^0 selective RF refocusing pulses. This scheme can be denoted by [7] 90^0-τ_1-180^0-τ_1-τ_2-180^0-τ_2-Acq, where τ_1 and τ_2 are delay times, $TE = 2(\tau_1+\tau_2)$ is the echo time and Acq indicates MRS data acquisition.

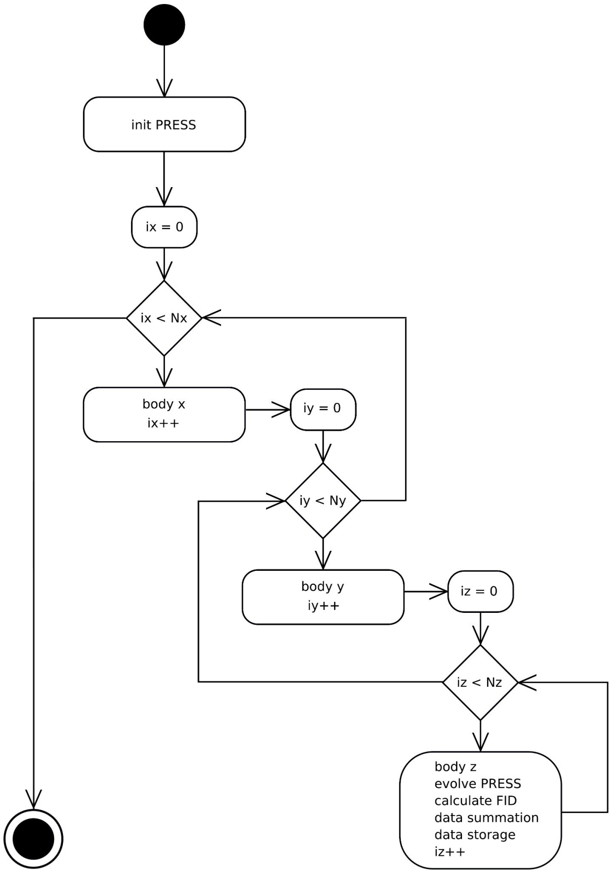

Figure 1. UML activity diagram of the PRESS-simulation program.

To achieve a three-dimensional voxel selection, three orthogonal magnetic field gradients are applied in the presence of each of the selective RF pulses. At the end of the pulse sequence, a Free Induction Decay (FID)-like MRS signal is sampled, starting from the time, at which the second spin-echo signal reaches its maximum value.

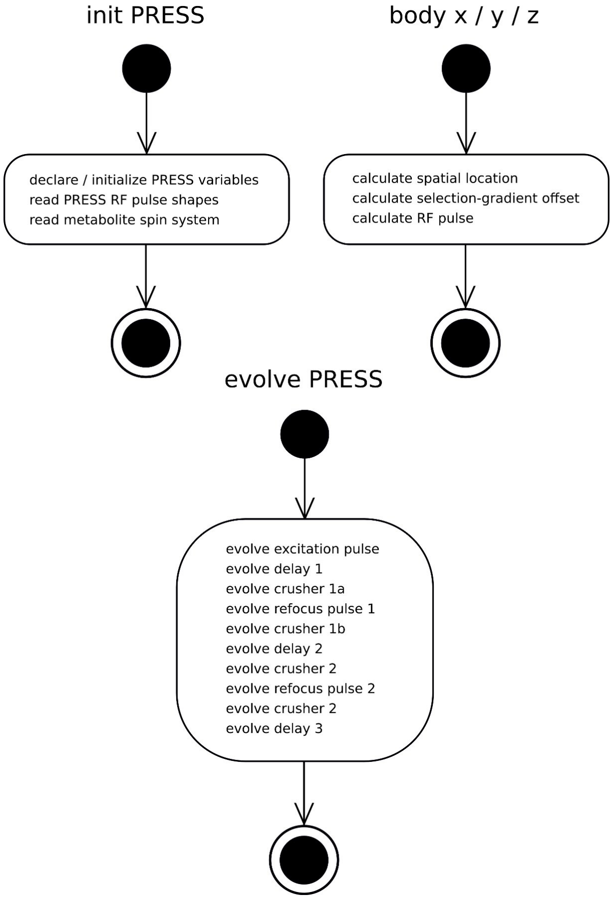

Figure 2. UML activity diagrams of three essential sub-activities of the PRESS-simulation program (see also Figure 1).

In the context of the PRESS experiment, just described, it can be understood that in our Gamma C++ code the simulation of an *in vivo* PRESS-based single-voxel selection is accomplished by carrying out a three-fold summation of the calculated MRS signal over locations inside the Field Of View (FOV) of the MRS experiment, concerned. This main aspect of the computer program is visualized in the UML activity diagrams of Figures 1 and 2 .

2) Crusher gradients: In general, the two selective RF refocusing pulses of an *in vivo* PRESS pulse sequence cannot *adequately* select (on their own) the desired voxel in the FOV of the MRS experiment. In order to achieve improvement, the refocusing pulses must be interspersed between crusher (spoiler) gradients [6]. As a result of these crusher gradients the MRS signals, originating from FOV locations outside of the voxel, are supposed to have phases of the individual NMR components so as to realize *cancellation*.

In the previous sub-subsection (about our GAMMA C++ code) we have shown, that simulating the voxel selection within the FOV is accomplished by stepping through the x-, y- and z-dimensions (by means of a three-fold indexed for-loop) and adding the simulated MRS signals from the FOV locations, included.

In the code we have introduced a *basic* step-size for stepping through the dimensions, which we will denote here by $D_{\rm xyz}$. By basic we mean that, due to the presence of the crusher gradients, the transverse magnetization changes its phase by an amount of 180^0, when applying this step.

If $n_{\rm crush}$ denotes the number of crusher gradients, $\gamma G_{\rm crush}$ their strength (in Hz/cm) and $t_{\rm crush}$ their length (in s), it can be derived [8] [9] that

$$D_{\rm xyz} = \frac{1}{n_{\rm crush}} \times \frac{1}{2\gamma G_{\rm crush} t_{\rm crush}}. \qquad (1)$$

In our GAMMA C++ code the index upper-limit of, say, the for-loop for the z-dimension, is denoted by $N_{\rm z}$. Furthermore, the corresponding step-size in z is chosen to be equal to $\frac{F_{\rm z}^{\rm ov}}{N_{\rm z}}$, where $F_{\rm z}^{\rm ov}$ is the size of the FOV in the z-direction. By requiring, that this step-size is equal to the basic step-size, defined in Equation (1), it follows that the index upper-limit should be

$$N_{\rm z} = 2F_{\rm z}^{\rm ov} n_{\rm crush} \times \gamma G_{\rm crush} t_{\rm crush}. \qquad (2)$$

For our PRESS-based *in vivo* MRS experiment the equation parameters have values $F_{\rm z}^{\rm ov}$=8 cm, $n_{\rm crush}$=4, $\gamma G_{\rm crush}$=10644 Hz/cm (fixed percentage of maximum strength) and $t_{\rm crush}$= 0.00352 s. Inserted into Equation (2) this gives $N_{\rm z} \approx 2400$ (64×37.5). Furthermore, with $F_{\rm y}^{\rm ov}$=$F_{\rm z}^{\rm ov}$, this would require an $N_{\rm x}\times 2400 \times 2400$ summation over the FOV. We have estimated for the local computers, concerned, that this would lead to unworkable long computational times, particularly for the metabolites with the larger spin systems. Therefore, we have introduced in the GAMMA C++ code a modified summation scheme, that is described in the next sub-subsection.

```
// .....
// FOVz=8
// Nx=1 Ny=Nz=64
// Nphase=2
// Nsubx=Nsuby=Nsubz=8
// .....
  for (ix=0; ix<Nx; ix++) {
    // .....
    for (iy=0; iy<Ny; iy++) {
      // .....
      for (iz=0; iz<Nz; iz++) {
        // .....
        z=(0.5-(double)iz/(double)Nz)*FOVz;
        // .....
        for (i=0; i<Nphase; i++) U_pulse_2[i]= Soft_U(args); // refocusing pulse 2
        // .....
        for (isubx=0; isubx<Nsubx; isubx++) {
          // .....
          for (isuby=0; isuby<Nsuby; isuby++) {
            // .....
            for (isubz=0; isubz<Nsubz; isubz++) {
              subz=z+(((double)isubz-(double)Nsubz/2.0)+0.5)*Dxyz;
              for (iphase=0; iphase<Nphase; iphase++) {
                // .....
                wFz = complex((subx+suby+subz)*crushStrength)*Fz(sys);
                U_crush = prop(H+wFz,crushLength); // crusher gradient
                // .....
                // water-suppression sequence
                // begin PRESS sequence
                // .....
                evolve_ip(sigma,U_crush);
                evolve_ip(sigma,U_pulse_2[iphase]);
                evolve_ip(sigma,U_crush);
                // .....
                // end PRESS sequence
              } // end phase cycling
            }   // end isubz
          }     // end isuby
        }       // end isubx
        // .....
      }         // end iz
    }           // end iy
  }             // end ix
// .....
```

Figure 3. GAMMA C++ related pseudo-code, showing a summation over locations in the FOV and over sub-locations for crusher-gradient averaging (see text). Also, phase cycling and water suppression is indicated.

3) Crusher-gradient averaging: In order to reduce the $N_{\rm x}{\times}2400{\times}2400$ summation, just mentioned, we have worked with a modified summation scheme, visualized −by means of pseudo-code− in Figure 3 . It amounts to taking $N_{\rm x}=1$ and reducing the $2400{\times}2400$ summation to $64{\times}64$.

The first step can be defended by realizing, that the voxel selection in the x-direction is based on *zero transverse magnetization* outside of the bandwidth of the selective excitation pulse, whereas the voxel selection in the y- and z-direction is effectuated by using selective refocusing pulses, combined with crusher gradients for introducing *phase dispersion*.

The second step (reducing 2400 to 64), however, may lead to non-cancellation effects due to ineffective phase-dispersion. Therefore, we have introduced in the code an additional, inner, three-fold for-loop for summing over x-, y- and z-*sub-locations*. These sub-locations have as step-size the basic step-size, defined in Equation (1).

Since in our PRESS-simulation program the major part of the computational time is used for calculating the RF pulses (see Table 2), the essence of the modified FOV summation scheme, described here, is that the (time-consuming) calculation of the RF pulses is carried out within the outer $1{\times}64{\times}64$ for-loop and the (less time-consuming) calculation of the crusher gradients and PRESS pulse sequence within the inner $N_{\rm subx}{\times}N_{\rm suby}{\times}N_{\rm subz}$ for-loop. This approach comes down to crusher-gradient averaging.

Note in the pseudo-code of Figure 3 , that we have included phase cycling and water suppression.

B. Details of the *in vivo* MRS scans

PRESS-based *in vivo* MRS scans were performed on a 3 Tesla GE whole body scanner (General Electric Medical Systems, Milwaukee, Wisconsin, USA) running the 15M4 software platform. A Medical Advances (Intermagnetics General Corporation, Milwaukee, Wisconsin, USA) RF quadrature transmit/receive coil was used with an inner diameter of 25 cm and a length of 20 cm. A scan session started with a $\rm T_1$ weighted anatomical scan by a three dimensional spoiled gradient echo sequence.

Two spectroscopy voxels (2 l/r × 4.5 a/p × 2 s/i cm) were scanned in one session. One voxel was placed immediately superior to the ventricles, straddling the midline, in order to contain the largest proportion of GM possible, with the anterior edge of the voxel never exceeding the anterior portion of the genu of the corpus callosum. This resulted in inclusion mainly of the corpus callosum and ACC.

Another voxel was placed directly adjacent to the first, in the rFWM, adjusting its position in order to minimize the GM and CSF contribution and always remaining dorsal to the caudate nucleus.

III. Results

A. Generating simulated metabolite basis sets

In order to be used in the quantification of a real-world *in vivo* MRS signal (see below), we have generated −with several versions of our PRESS-simulation program− four different metabolite basis sets.

Each basis set contains the contributions from the same metabolites, being choline (cho), creatine (cr), glutamate (glu), glutamine (gln), myo-inositol (myo), N-acetyl aspartate (naa) and scyllo-inositol (scyllo). They differ, however, in the shapes used for the RF pulses (General Electric (GE) or standard Sinc) and the crusher-gradients setup (modified or standard). Also, one basis set was generated, using ideal (hard) rectangular RF pulses. The four basis sets are denoted by GE_mod, GE_stand, Sinc_stand and Ideal, respectively.

Since the computational time of a density-matrix calculation strongly increases, when going to larger spin systems, we also have worked with the concept of *separating moieties* [10]. That is to say, the FIDs of the singlets and multiplets of the metabolites were calculated separately, leading to cho1, cho2, cr1, cr2, naa1 and naa2. In this notation the 1 indicates the singlet and the 2 the multiplet (apart from cr1 and cr2, which are both singlets).

In Table 1 the computational times are presented, that were needed to simulate the various parts of the metabolite basis sets. All calculations were done on the same Linux-based desktop-PC, having an Intel Core 2 Duo E8400 CPU. In order to improve the computational speed, both cores of the CPU were used by employing the `ppss` Linux bash script [11].

Table 1: Computational times (hours:minutes:seconds) as a function of the number of metabolite spins (NS), as required for the simulations of the GE_mod, GE_stand, Sinc_stand and Ideal metabolite basis sets. The calculations were performed with an Intel Core 2 Duo E8400 CPU, using both CPU cores by means of the `ppss` Linux bash script [11].

NS	GE_mod	GE_stand	Sinc_stand	Ideal #	Metabolites
1	18:20	1:30	14	*	cho1 cr1 cr2 naa1 scyllo
3	47:33	5:16	33	*	naa2
4	2:38:06	19:25	1:35	*	cho2
5	11:46:44	1:58:44	8:43	0.03	glu gln
6	87:47:20	16:02:10	1:08:18	0.07	myo

# No `ppss` used   * <= 0.02 seconds

In Table 2 the *details* of the computational times are presented. The were obtained by introducing −at several locations in the code− the `gettimeofday()` C++ function. Note that the time scale in this table is milliseconds whereas in Table 1 it is hours:minutes:seconds.

Table 2: Computational times (milliseconds) as a function of the number of metabolite spins (NS), as required for *details* of the simulations of the GE_mod, GE_stand and Sinc_mod metabolite basis sets. Each row in the tables concerns the calculation for only a *single* location in the FOV (in the center of the voxel). Note in the table for GE_mod the presence of water suppression and an additional factor of 2 due to phase cycling.

GE_mod

NS	Pulses_water	2Pulse_exc	4Pulse_ref	1024(Water+PRESS)	Rest	Total
1	17	20	56	521	3	617
3	65	76	221	1183	4	1549
4	237	288	828	3289	2	4644
5	1489	1833	5209	14762	2	23295
6	11989	14178	40507	81037	7	147718

GE_stand

NS	Pulse_exc	2Pulse_ref	PRESS	Rest	Total
1	12	28	1	14	55
3	39	110	1	13	163
4	144	415	2	15	576
5	884	2522	9	17	3432
6	6953	20004	42	29	27028

Sinc_stand

NS	Pulse_exc	2Pulse_ref	PRESS	Rest	Total
1	1	2	1	14	18
3	4	8	1	13	26
4	15	28	2	16	61
5	80	155	9	16	260
6	611	1222	42	29	1904

B. Viewing realized voxel selection

In Figure 4 the influence of the RF-pulse shapes and crusher-gradients setup on the quality of the voxel selection is demonstrated. This is done by displaying 3 two-dimensional plots of the spectrum of a simulated MRS signal, corresponding to a *surrogate* metabolite spin system with 5 uncoupled proton spins. In the simulations the Larmor frequency was set at 128.0 MHz (1 PPM = 128 Hz), the FID frequency offset at -4.65 PPM and the RF-pulses offset at 2.0 PPM [1] . It can be seen, that the GE_mod setup yields by far the best voxel selection.

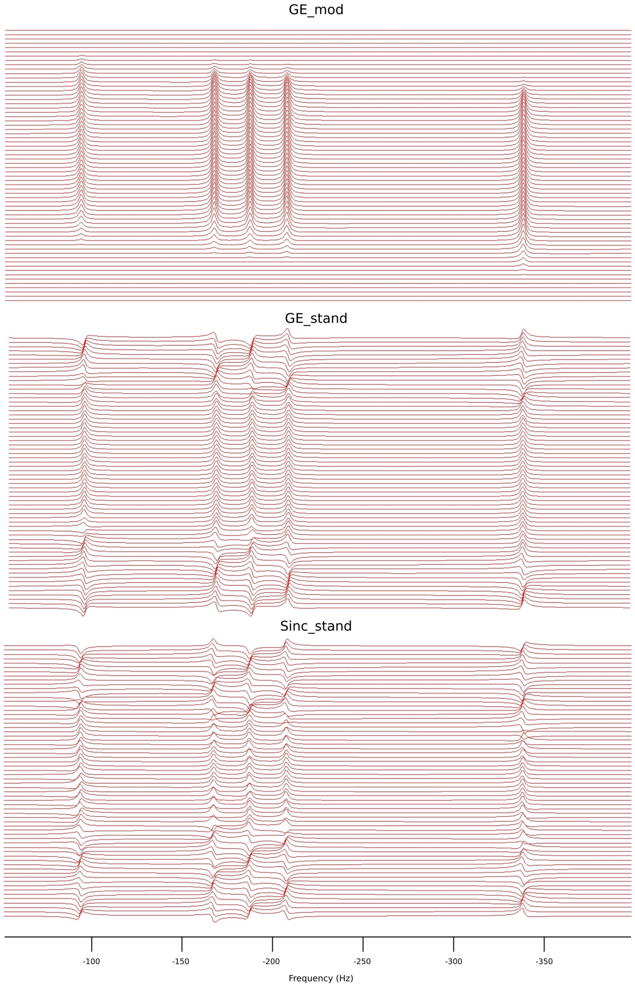

Figure 4. Influence of the RF-pulse shapes and crusher-gradients setup on the voxel selection. The figure displays the spectra of *simulated* FIDs of a *surrogate* metabolite, containing 5 uncoupled proton spins with the chemical shifts (in PPM) of cr2 (3.913), scyllo (3.34), cho1 (3.185), cr1 (3.027) and naa1 (2.008). In the simulations the Larmor frequency was set at 128.0 MHz (1 PPM = 128 Hz), the FID frequency offset at -4.65 PPM and the RF-pulses offset at 2.0 PPM. In vertical direction, the spectra are displayed as a function of the location in the FOV (64 y-locations with $F_y^{\text{ov}} = 8$ cm; the voxel-width in the y-dimension was set to be 4.5 cm). The results are shown for the GE_mod, GE_stand and Sinc_stand metabolite basis set, respectively.

C. Quantifying a real-world in vivo *MRS signal*

We have used the four metabolite basis sets, mentioned in the previous subsections, in the quantification of a 3T human-brain ^{1}H MRS signal. To that end the basis sets were fitted to the MRS signal by means of the *j*MRUI QUEST method [2] [3]. This method is particularly suited for working with basis sets of summed metabolite signals, assuming these signals have a Lorentzian decay (also a fixed Gaussian decay can be added in order to describe the Voigt line-shape).

Another important aspect of QUEST is, that a broad background signal in the *in vivo* MRS signal, to be quantified, can be separated from the metabolite contributions by applying a specific iterative fitting algorithm to the initial data-points.

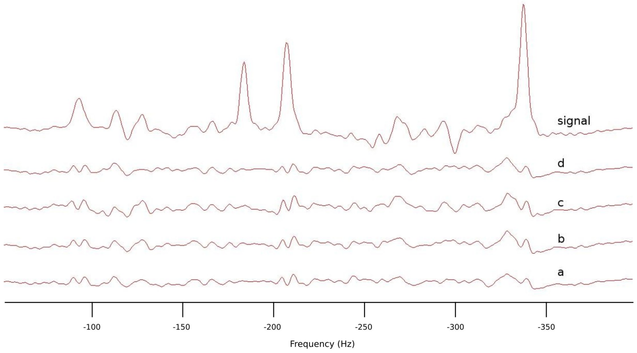

Figure 5. Spectra (real parts) of the residues, obtained by subtracting the fitted simulated metabolite basis sets from a 3T human-brain ^{1}H MRS signal. Residue of the (a) GE_mod, (b) GE_stand, (c) Sinc_stand and (d) Ideal-pulses based basis set.

In Figure 5 we compare the frequency-domain residues for the four metabolite basis sets. Although there are clear differences, it is difficult to judge the results only on the basis of these residues.

In the remaining part of this subsection the quantification results –i.e. the metabolite amplitudes– are presented (in Table 3). It should be noted, that the amplitudes have been normalized with respect to naa or naa1 (the latter being the most significant singlet), because the absolute values of the amplitudes cannot be obtained without a reference method.

Table 3: Metabolite amplitudes (normalized with respect to naa or naa1) of a 3T human-brain ^{1}H MRS signal, as obtained by fitting – with *j*MRUI QUEST [2] [3] – four simulated metabolite basis sets, denoted by GE_mod, GE_stand, Sinc_stand and Ideal, respectively (see text). CRLB stands for Cramér-Rao lower bound. The notations in the second and third row of the tables indicate QUEST options, related to enabling/disabling a *common* exponential decay (comdec) and/or a fitted background (back).

	GE_mod	CRLB	GE_mod	CRLB	[10]
	comdec		comdec		
	noback		back		
cho	0.17	0.00	0.18	0.00	0.14
cr	0.69	0.01	0.69	0.01	0.77
glu	0.97	0.05	0.96	0.03	1.03
gln	0.30	0.04	0.29	0.03	0.21
myo	0.95	0.05	0.53	0.03	0.59
naa	1.00	0.01	1.00	0.01	1.00
scyllo	0.03	0.01	0.03	0.00	0.03
RMSD1	0.16		0.06		

	GE_mod	GE_mod	GE_mod	GE_mod	GE_stand	GE_stand
	comdec	comdec	nocomdec	nocomdec	comdec	comdec
	noback	back	noback	back	noback	back
cho1 #	0.18	0.18	0.16	0.19	0.15	0.16
cho2 #	*	0.18	*	0.21	0.14	0.21
cr1 #	0.71	0.74	0.72	0.79	0.65	0.69
cr2 #	0.65	0.53	0.75	0.57	0.39	0.33
glu	1.01	0.99	0.93	0.94	0.94	0.93
gln	0.35	0.33	0.24	0.29	0.44	0.33
myo	1.02	0.54	1.48	0.43	0.71	0.35
naa1 #	1.00	1.00	1.00	1.00	1.00	1.00
naa2 #	1.44	1.30	1.47	1.23	1.27	1.23
scyllo	0.03	0.03	0.00	0.03	0.03	0.02
RMSD1	0.19	0.09	0.37	0.09	0.15	0.16
RMSD2		0.27		0.26	0.41	0.65

[1] This means, that the NMR components have spectral frequencies equal to $(\sigma_i - 4.65) \times 128$ Hz (σ_i being (in PPM) the i^{th} chemical shift) and that a selected voxel is located symmetrically with respect to the center of the FOV for σ = 2.65 PPM (the region of naa2/glu/gln).

	Sinc_stand	Sinc_stand	Ideal	Ideal
	comdec	comdec	comdec	comdec
	noback	back	noback	back
cho1 #	0.16	0.18	0.18	0.18
cho2 #	*	0.12	0.23	0.31
cr1 #	0.55	0.62	0.71	0.74
cr2 #	0.48	0.49	0.46	0.37
glu	0.88	0.97	0.82	0.84
gln	0.85	0.57	0.26	0.26
myo	0.63	0.38	1.05	0.54
naa1 #	1	1	1	1
naa2 #	2.06	1.75	1.03	0.97
scyllo	0.04	0.02	0.03	0.03
RMSD1	0.29	0.19	0.22	0.12
RMSD2		0.41	0.35	0.62

# Separated moieties * Not found

IV. Discussion

A. Concerning the computational times

1) Using multi-core CPUs: Seen in the light of the long computational times, presented in Table 1, it is clear that employing the multi-core technology (usually available today) of the local desktop-PC is of great importance. In our case we have done that by scheduling the standalone versions of the PRESS-simulation program via the `ppss` Linux bash script [11]. It that way a gain in computational speed by a factor of 2 was achieved.

In this context we like to mention, that we also have employed parallel processing on the desktop-PC by utilizing the `java.util.concurrent` package [12]. This Java package can be combined, when desired, with the Java Native Interface (JNI) approach [13].

2) Explaining computational times: When analyzing the computational times of the details in Table 2, it can be derived that

$$\frac{t_{\text{GE\_mod}}}{t_{\text{GE\_stand}}} \approx \frac{t_{\text{water}}}{t_{\text{pulses}}} + N_{\text{phase}}\{1 + N_{\text{subloc}}^{3}\, f_{\text{water}}\, \frac{t_{\text{PRESS}}}{t_{\text{pulses}}}\}, \tag{3}$$

where t_{water} is the time for calculating the water-suppression pulses, $t_{\text{pulses}} = t_{\text{pulse}}^{\text{exc}} + 2t_{\text{pulse}}^{\text{ref}}$ the time for calculating the excitation pulse and the two refocusing pulses, N_{phase} the number of phase-cycling values (was set to 2) and N_{subloc} the index upper-limit of the inner for-loops for carrying out the basic step D_{xyz}, defined in Equation (1) (was set to N_{subx}=N_{suby}=N_{subz}=8, as is shown in the pseudo-code of Figure 3).

Furthermore, a factor f_{water} has been introduced in Equation (3), which takes into account the elapsed time for evolving the water-suppression pulse sequence. It either equals to 1, when excluding water suppression, or −for the larger metabolite spin systems− varies from about 1.6 (cho2) to 1.9 (myo).

If we take from Table 2 the times for myo, we find from Equation (3) that $t_{\text{GE\_mod}} \approx 5.5\, t_{\text{GE\_stand}}$. This approximated result is in agreement with the corresponding total computational times, presented in Table 1.

Finally, when using the numerical result, just mentioned above, with the numerical result of Equation (2), we can conclude that the summation scheme for GE_mod has decreased the total computational time for myo by a factor of $\frac{37.5^2}{5.5} \approx 256$, when compared with the time, that would have been required for the (basic-step related) 1×2400×2400 summation.

B. Concerning the quantification results

1) The frequency-domain residues: The frequency-domain residues as a function of the four metabolite basis sets, displayed in Figure 5 , provide not much insight into "which of the basis sets should be preferred".

The residues at the locations of three of the largest singlets indicate, however, that the metabolite line shapes deviate from the fitted Lorentzian line shape. Concerning the residues at other frequency locations it is difficult to say, whether they are due to "incorrect simulation" of the metabolite signals, or to "excluding" other metabolites or background signals from the fitting process.

2) The metabolite amplitudes: When quantifying real-world *in vivo* MRS signals it always is difficult to judge, what are "the best values" for the metabolite amplitudes, concerned?

Also, when having measured only one MRS signal, the statistical errors of the amplitudes cannot be determined. They often are replaced by their (theoretical) Cramér-Rao lower bounds (CRLBs), as is shown −to indicate the order of magnitude− in the first subtable of Table 3 (the CRLBs of the other subtables are not shown, but they are comparable).

When analyzing in Table 3 the metabolite amplitudes as a function of the metabolite basis sets and/or *j*MRUI QUEST options, it is clear that their variations are much larger than the values of the CRLBs, shown in the first subtable. We think this is, because we are dealing with the *semi-parametric* model. By that we mean, that in the field of *in vivo* MRS, usually no parametric model-components are available for exactly describing the metabolite line shapes, concerned (see also the line-shape remark in the previous sub-subsection).

In order to have some idea of "what the best amplitudes are", we have introduced in Table 3 two root-mean-square deviations, called RMSD1 and RMSD2. The first one indicates the deviations from metabolite amplitudes, presented in a recent paper on 4T *in vivo* MRS of the human brain [10] . The second one indicates the deviations from the requirement, that the amplitudes of the separated moieties within a metabolite (i.e. within cho, cr or naa) are equal.

When taking into account both RMSDs, we have concluded, that the GE_mod basis set has yielded the best quantification results.

V. Summarizing remarks

Summarizing we like to make the following remarks:

- By utilizing the GAMMA C++ library, we have developed several versions of a computer program for simulating the *in vivo* PRESS pulse sequence.
- With these programs we have generated four different metabolite basis sets, so as to be used in the quantification of a PRESS-based *in vivo* MRS signal. The basis sets differ by varying the selective RF-pulse shapes and by varying the setup for simulating the effects of the crusher gradients.
- We have demonstrated that the computational times, needed for simulating the various metabolite basis-set

signals, can be reduced considerably by introducing the concept of crusher-gradient averaging.

- We have determined the amplitudes of seven metabolites in the *in vivo* MRS signal, concerned. This was done with the *j*MRUI QUEST quantification program, using the four metabolite basis sets, mentioned above.
- It was found for this single quantification case, that the best results were obtained by working with the basis set, generated by including −as closely as possible− the details of the related *in vivo* PRESS experiment.

Acknowledgment

This work was supported by of the Marie-Curie Research Training Network FAST [14].

References

[1] R. de Beer, D. van Ormondt, J.W. van der Veen and D. Graveron-Demilly, "Creating the GammaPress custom plug-in for the *j*MRUI platform. Simulating metabolite basis sets for *in vivo* Magnetic Resonance Spectroscopy," in *Proceedings ICT.OPEN 2011*, Veldhoven, The Netherlands, 14-15 November 2011, pp. 16 – 20, http://www.ictopen2012.nl/nwohome.nsf/pages/NWOP_8UX9KQ#Theme1, organization = NWO/STW. 1

[2] D. Stefan, A. Andrasecu, E. Popa, H. Rabeson, O. Strbak, Z. Starcuk, M. Cabanas, D. van Ormondt, and D. Graveron-Demilly, "jMRUI Version 4 : A Plug-in Platform," in *IEEE International Workshop on Imaging Systems and Techniques, IST 2008*, Chania, Greece, 10-12 September 2008, pp. 346–348. 1, 4

[3] D. Stefan, F. D. Cesare, A. Andrasescu, E. Popa, A. Lazariev, E. Vescovo, O. Strbak, S. Williams, Z. Starcuk, M. Cabanas, D. van Ormondt, and D. Graveron-Demilly, "Quantitation of magnetic resonance spectroscopy signals: the jMRUI software package," *Meas. Sci. Technol.*, vol. 20, p. 104035 (9pp), 2009. 1, 4

[4] S.A. Smith, T.O. Levante, B.H. Meier, and R.R. Ernst, "Computer Simulations in Magnetic Resonance. An Object-Oriented Programming Approach," *J. Magn. Reson. Series A*, vol. 106, pp. 75 – 105, 1994. 1

[5] P. A. Bottomley, "Spatial Localization in NMR Spectroscopy *in Vivo*," *Ann. N.Y. Acad. Sci.*, vol. 508, pp. 333 – 348, 1987. 1

[6] Richard B. Thompson and Peter S. Allen, "Sources of Variability in the Response of Coupled Spins to the PRESS Sequence and Their Potential Impact on Metabolite Quantification," *Magnetic Resonance in Medicine*, vol. 41, pp. 1162 – 1169, 1999. 1, 2

[7] Andrew A. Maudsley, Varanavasi Govindaraju, Karl Young, Zakaria K. Aygula, Pradip M. Pattany, Brian J. Soher, Gerald B. Matson, "Numerical simulation of PRESS localized MR Spectroscopy," *J. Magn. Reson.*, vol. 173, pp. 54 – 63, 2005. 1

[8] Karl Young, Gerald B. Matson, Varanavasi Govindaraju, and Andrew A. Maudsley, "Spectral Simulations Incorporating Gradient Coherence Selection," *J. Magn. Reson.*, vol. 140, pp. 146 – 152, 1999. 2

[9] P.K. Mandal, "Magnetic Resonance Spectroscopy (MRS) and its Application in Alzheimer's Disease," *Concepts in Magn. Reson. Part A*, vol. 30A(1), pp. 40 – 64, 2007. 2

[10] Dinesh Kumar Deelchand, Pierre-Gilles Henry, Kamil Ugurbil, and Malgorzata Marjanska, "Measurement of Transverse Relaxation Times of J-Coupled Metabolites in the Human Visual Cortex at 4 T," *Magn. Reson. in Med.*, vol. 67, pp. 891–897, 2012, . 3, 4, 5

[11] J.W. van der Veen, R. de Beer and D. van Ormondt, "Increasing the computational speed of command line GammaPress. `PPSS`-controlled distribution of parallel processing on multi-core/hyper-threading enabled CPUs of multiple host computers," Report on behalf of the Marie-Curie Research Training Network FAST, 2012, e-mail: r.debeer@tudelft.nl. 3, 5

[12] ——, "Utilizing Java Concurrent Programming for Speeding up External Programs. Fully accessing multi-core CPUs by making use of the `java.util.concurrent` package," Report on behalf of the Marie-Curie Research Training Network FAST, 2012, e-mail: r.debeer@tudelft.nl. 5

[13] ——, "Utilizing Java Concurrent Programming, Multi-Processing and the Java Native Interface. Running Native Code in Separate Parallel Processes," Report on behalf of the Marie-Curie Research Training Network FAST, 2012, e-mail: r.debeer@tudelft.nl. 5

[14] FAST European Union project, "FAST website," http://www.fast-mariecurie-rtn-project.eu/, 2012. 6