

# A multi-purpose simulator of coupled spin systems for MR localized spectroscopy and spectroscopic imaging

Z. Starcuk jr.<sup>1</sup>, J. Starcukova<sup>1</sup>, and D. Graveron-Demilly<sup>2</sup>

<sup>1</sup>Magnetic Resonance and Bioinformatics, Institute of Scientific Instruments, Academy of Sciences of the Czech Republic, Brno, Czech Republic, <sup>2</sup>Creatis-LRMN, Université Claude Bernard Lyon 1, France

**Introduction.** In spite of the fact that numerous MR simulators exist, the choice progressively narrows when more functionality for in vivo MR spectroscopy and spectroscopic imaging is required. A practical spectroscopist or a pulse sequence developer alike reach an impasse if the requirements include coupled homo- as well as heteronuclear spin systems undergoing relaxation, cross-relaxation and/or magnetization transfer, which are to be exposed to spatially and/or frequency-selective excitation, potentially involving phase cycles and/or RF and gradient modulation not excluding simultaneous modulation. The wish list further includes simple pulse sequence programming tightly matching the underlying physics, and its combination with protocol-oriented interface appealing to a practical user. The possibility to determine functional dependences and to inspect quantum mechanical features not available in the bulk  $M_{xy}$  data would be favored by a physicist. This new simulator (NMRScope-B), implemented as a part of jMRUI [1], attempts to meet these goals.

**Method:** The substantial components of the simulator are the (a) spin system model and (b) the protocol and pulse sequence editor and interpreter, each of them includes a calculation core and a graphical user interface.

(a) The number of coupled spins is limited only by the available memory and calculation time. Each spin system is defined by the list of nuclei (heteronuclear systems allowed), their chemical shifts and J-couplings, which are treated as weak for heteronuclear and as strong for homonuclear couplings. Additionally, the whole spin system may be characterized by its spatial position. The principal data structures used in calculations are product ket-space operators and the density matrix, but relaxation is nevertheless simulated according to the Redfield relaxation superoperator, including general mixing of populations by  $T_1$  relaxation and cross-relaxation along with  $T_2$  decay of all other coherences. For the definition of all relaxation rates, the user may use the Solomon model parameters for specifying the dynamics of populations, and nuclide-specific additive relaxation rates assigned to individual coherence orders. The latter approach can be justified by the differences in the respective laboratory frame evolution frequencies, resulting in their sensing of perturbations in different frequency bands. Splitting the evolution in a sequence of tiny interleaved hamiltonian-driven and relaxation-driven elements makes it possible to handle relaxation phenomena at negligible computational cost, without introducing noticeable errors. The algorithm recycles precalculated operators as much as possible. To support an inquisitive physicist, the user is given the option to specify a list of observables in a text string. Instead of the standard observation (" $Fx(1H)$ ;  $Fy(1H)$ "), any other set can be asked for (e.g. " $Jx(1)*Jy(2)-Jx(2)*Jy(1)$ ;  $Fx(1H)$ ;  $Fy(1H)$ ;  $Fz(13C)$ "). These observables are collected from all acquisition points into a single data matrix. While standard signals are passed on to jMRUI for processing and visualization, nonstandard observable sets may be exported in Matlab format for customized inspection.

(b) The pulse sequence may be specified at the basic instrumental level as a sequence of amplitude/phase modulated RF pulses, gradient pulses, delays, and observations with their attributes (durations, shapes, amplitudes, phases). A few trick operations (ideal crusher, reestablishment of Boltzmann equilibrium etc.) are available. To support simultaneous application of shaped pulses, even in the same channel, only the starting points are set sequentially on the time axis, and it is the task of the built-in interpreter to arrange all overlapping events in a multichannel stream of control events that the spin system's simulation engine can process. The events are passed in a Matlab binary format, which opens the door for alternative sources of event streams, such as those from scanner-specific simulators; this might help in debugging newly implemented pulse sequences. Phase cycles and simple loops are supported at this level as well. With a few lines of JavaScript code, a basic pulse sequence can be controlled at a protocol layer ensuring the proper interparameter relations. Typical pulse sequences (e.g. STEAM or PRESS) are included by default.

Most spin system or pulse sequence parameters may be specified as variables. By specifying their ranges and cycling orders in tables, various dependences may be produced. In the current version, the Java code of jMRUI and the simulator GUI is interfaced through a thin C++ buffer to a compiled Matlab simulation engine.

**Results:** The code has been found to run reliably on a range of PCs – netbooks to workstations. The calculation time depends heavily on the computer, the pulse sequence, spin system, and cycling requirements; 7-8 coupled spins seem to be the current practical limit. Typical simple tasks (individual spectra, excitation profiles,  $B_1$  dependences etc.) can be determined mostly in 0.5-5 min. Typically 70-80% of the total calculation time is spent at 2 code lines performing matrix exponentiation or matrix multiplication. Fig. 1 illustrates the simulator GUI, the sequence in Fig. 2 leading to the spectrum of Fig. 3 illustrates one practical application.

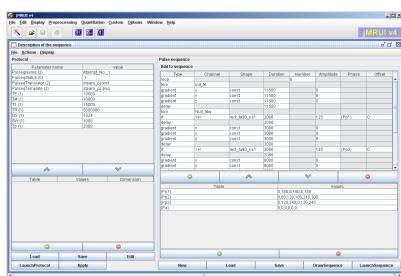


Figure 1. Pulse sequence GUI showing the protocol (left) and the low-level pulse sequence (right).

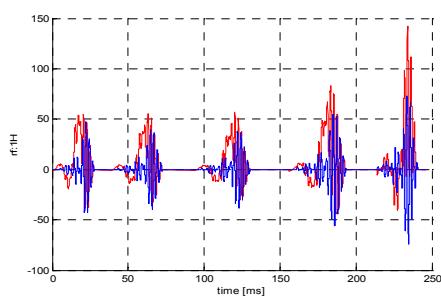


Figure 2. Part of a pulse sequence for dual-band excitation showing overlaid AM RF pulse – superposition is produced automatically.

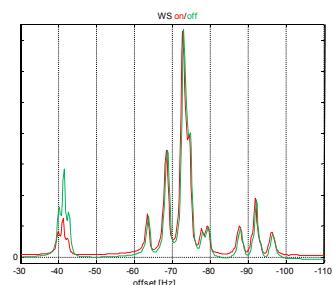


Figure 3. SE spectrum of myoinositol simulated with and without water/fat presaturation ( $B_0=3T$ ,  $T_2=1400ms$ ,  $T_2=200ms$ , 0Hz: 4.7ppm).

**Discussion:** Simulation is an indispensable tool in spectroscopic pulse sequence design and in the preparation of basis sets for spectroscopic quantitation. This simulator gives an MR spectroscopist the power to calculate the necessary signals with the real shaped pulses used in the experiment, to obtain proper spectral patterns including line phases, which may be affected by frequency selective excitation, and to study the effects of various parameters with no or very little traditional programming. Thanks to the Redfield-matrix based treatment of relaxation phenomena, this simulator can be used for deepening the knowledge of relaxation as well as magnetization transfer in spectroscopic measurements. The merger of imaging and spectroscopic approaches may be found useful for NMR education. The current restriction to the Windows platform is only temporary. This simulator is intended to be released with jMRUI by May 1, 2011 and will be available together with this software package.

**Acknowledgements:** The work was supported by the ASCR grant AV0 Z20650511, GACR grant GA102/09/1861, and EU Marie Curie Research Network 'FAST', MRTNCT-2006-035801.

## References.

[1] D. Stefan, F. Di Cesare, A. Andrasecu, E. Popa, A. Lazarev, E. Vescovo, O. Strbak, S. Williams, Z. Starcuk, M. Cabanas, D. van Ormondt and D. Graveron-Demilly, Meas. Sci. Technol. 20 (2009) 104035 (9pp), <http://www.mrui.uab.es/mrui>.