University of Bologna

A numerical method to optimize presaturation sequences on multi-exponential samples

M. Gombia[a], S. Sykora[b], V. Bortolotti[c], E. Vacchelli[a], P. Fantazzini[a]

[a]Department of Physics, University of Bologna, Viale Berti Pichat 6/2, 40127 Bologna, Italy, **www.mrpm.org**
[b]Extra Byte, Via Raffaello Sanzio 22c, 20022 Castano Primo, Italy, **www.ebyte.it**
[c]DICMA, University of Bologna, Via Terracini 28, 40131 Bologna, Italy

DOI: 10.3247/sl2nmr08.006

A mathematical model has been developed to simulate the effects of pulse sequences on composite samples and the algorithm was encoded in Matlab. The object-oriented code accommodates pulse sequences and samples of any complexity, including ones with any distribution of relaxation rates in the presence of the most common instrumental artifacts (**B_0** and **B_1** inhomogeneity). The package permits to simulate the magnetization of a composite sample during the whole sequence by using Bloch equations to separately track the magnetization of each sample component. It allows one to obtain three quality factors related to: (i) the highest relative residual magnetization among all components (**Q_1**); (ii) the square mean of the relative residual magnetizations of all components (**Q_2**); (iii) the modulus of the relative total magnetization vector of the sample (**Q_3**). Low values of **Q_i** (i=1,2,3) indicate good suppressions of the residual sample magnetization. In particular, a low value of **Q_1** indicates good suppression of the residual magnetizations of all sample components. This approach has been applied to the problem of fast and efficient presaturation by a suitable *Sample Magnetization Suppression* (**SMS**) pulse sequence, which led us eventually to ***the Logarithmically-distributed A-Periodic Saturation Recovery*** (**LAPSR**) sequence which comes as close as possible to suppressing the absolute magnetization of all sample components and is, in this respect, much better than the classical *Saturation Recovery* (**SR**) and *A-Periodic Saturation Recovery* (**APSR**) sequences. An important additional insight we have gained is that *no matter which sequence, one should always use pre-saturation sequences composed exclusively of 180º inversion pulses*.

The development of SMS sequences started from these observations:

(i) Classical NMR relaxometry methods, such as **inversion recovery** (IR), are slow because they require reaching the equilibrium magnetization before every scan and they often fail in situations where sample complexity combines with severe instrumental imperfections (ex-situ NMR, large samples, severe **B_1** inhomogeneity, insufficient transmitter power, etc.).

(ii) The alternative is to use the **saturation recovery** (SR) sequence or the **APSR** sequence (train of 90º pulses with linearly decreasing delays), possibly in tandem with gradient pulses. The goal is to achieve zero-magnetization state and do so as fast as possible. Compared to IR, the results are encouraging but usually still far from ideal and full of multiple-echo artifacts.

These observations prompted us to carry out an extensive series of **computer simulations** in order to answer objectively the question of ***how fast and how well can one suppress the magnetization of complex samples using standard pulse sequences***. The interesting results, obtained applying integrated Bloch equations to multi-component virtual samples, were then compared with experiments. It turns out that the best saturation sequences are composed of about **20 inversion pulses** with **geometrically decreasing delays**. Relaxation curves can be measured **about 2.5 times faster** than with IR and remain **meaningful even under very imperfect experimental/instrumental conditions.**

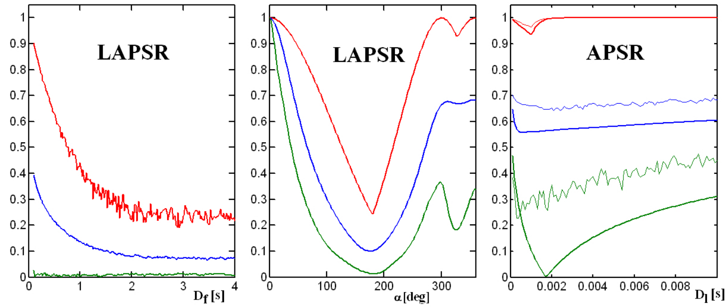
Examples of typical simulation results

The Figure below shows the magnetization suppression factors **Q_1** (**red**), **Q_2** (**blue**) and **Q_3** (**green**) computed for a virtual "rock" phantom with 270 components covering all combinations of nine T_1's (3000, 1000, 300, 100, 30, 10, 3, 1, and 0.3 ms), ten offsets (0-1800 Hz, step 200 Hz) and three B_1 inhomogeneity settings (RF - amplitudes of 0.8, 1.0, and 1.2).

The left graph shows the results for LAPSR sequences composed of 20 pulses with geometrically decreasing delays when the first delay (D_f) is varied, while the last one remains fixed at 50 µs. The central graph regards also LAPSR, but in this case the first and last delays are kept fixed at 2 s and 50 µs, respectively, while the variable parameter is the common nutation angle α of all the pulses.

The graph on the right shows the three suppression factors for the APSR sequences of (again) 20 RF pulses with delays decreasing arithmetically from 19*D_l to D_l, D_l denoting the last value. The thin curves correspond to the setting α = 90º (classical APSR), while the thick ones show the results for α = 180º.

The noise-like, irregular behavior of the curves is due to the discrete number of components and the complexity of each component's evolution. It is this aspect which makes it impossible to use standard optimization algorithms.



Results and discussion

- ➢ It turns out that in all kinds of pre-saturation sequences (including the traditional APSR) it is preferable to use RF pulse trains with the nominal nutation angle of 180º rather than 90º. This maximizes the "true" magnetization suppression (**Q_1**) and also minimizes the formation of small multiple echoes which look as noise but accumulate to a large extent coherently.
- ➢ The best sequence for magnetization suppression we have found is one composed of n = 20 RF pulses P_α, each with nominal nutation angle α=180º, separated by delays which start with a value D and decrease geometrically to the a value of 50 µs. When used to measure T_1 distributions, this saturation preamble is followed by a saturation recovery interval τ and a standard 90º readout pulse P_{90}.

The LAPSR sequence: P_α-Dq^0- P_α-Dq^1 - P_α-Dq^2 -… -P_α-Dq^{n-1} - P_α-- τ --P_{90}-FID(τ), where **Dq^{n-1}** = 50 µs.

- ➢ For unknown samples, or samples with T_1 distributions extending up to free water values (2-3 seconds), we use a "universal" version of the sequence with D = 2 seconds. Otherwise, we set D to about 65% of the maximum T_1 found/expected in the multi-T_1 sample. In all such cases, LAPSR is about 2.5 times faster than a time-optimized (IR) sequence.
- ➢ We have tested a number of similar sequences with delays decreasing according to various rules, such as log-log (LLAPSR) and various powers (XAPSR). In all cases, it was observed that for acceptable results, the delays must decrease over the range of considered values at least as rapidly as in LAPSR. No significantly better sequences than LAPSR were found, though some had a comparable performance.
- ➢ A true suppression of the magnetization of all components turns out to be an extremely difficult task. This is because the effect of RF pulses, short free-evolution delays, and even gradient pulses is just to **rotate** the magnetization vectors of the individual components rather than to reduce their magnitudes. The only way to reduce the magnitudes is through the longitudinal relaxation T_1. The sequence must bring the magnetization vectors of all components close to zero and then forcibly "lock" them there by means of repeated inversion pulses (we might call this effect the **longitudinal spin-lock**, a counterpart of the well known transversal spin-lock known from CPMG and $T_{1\rho}$ sequences). The trick behind LAPSR is that it sweeps the delays in such a way as to lock all the components even in a very complex sample with a broad T_1 distribution and under imperfect conditions.
- ➢ We believe that time-savings bigger than about 2.5 cannot be achieved if what one desires is a true suppression of the magnetization of all components. This is linked to the previous point, but the explanation is a bit lengthy and will be given elsewhere.
- ➢ Phase cycling of LAPSR is very simple. The simplest approach is to keep the phases of all the saturation preamble pulses at 0º and alternate the phase of the readout pulse between 0º and 180º, concurrently with a similar alternation of the "receiver phase" (actually simply addition/subtraction). Quadrature detection schemes are also possible but were not found to be necessary.

The simulation and optimization software:

To simulate the evolution of nuclear magnetization in a complex sample, the latter is decomposed into infinitesimal spatial components and relaxation components. Since we assume that these evolve independently, each elementary component behaves as though it had been subject to a uniform magnetic field and has a unique relaxation rates. The free evolution of its magnetization vector can be therefore described by the explicit solution of Bloch equations. Likewise, the effect of a strong, non-selective RF pulse on a component's magnetization vector can be described by a simple rotation about the effective magnetic field. At any moment, the complex signal at the receiver output is the sum of the transverse components of the magnetizations of all components. Within the software package we have written, all the terms we have mentioned are objects:

- *Sample component* is a root object endowed with the following properties: current magnetization m, equilibrium magnetization m_0 (relative weight), relaxation times T_1, T_2, offset Ω, and a B_1 inhomogeneity factor β (to be described below).
- *Sample* is a composed object containing (n) of *sample components*. The way these parameters vary from component to component defines a particular sample type. For debugging purposes we have used simple samples composed of a single component. For realistic simulations, however, rock samples with typically 81 components were used.
- *RF pulse* is a root object endowed with the properties α (nominal nutation angle) and φ (phase).
- *Delay* is a root object with a single property d denoting its duration.
- *Pulse sequence* is a composed object consisting of a *recycle delay object* (duration d_0), n *RF pulse objects* P_1, …, P_n, each with its characteristic properties, separated by (n-1) *delays* (durations d_1, …, d_{n-1}) and followed by a *detection delay* (duration t). An additional property associated with a pulse sequence is its *phase cycling matrix* indicating the pulse phases to be applied in subsequent scans. To keep track of the phase cycle scans, the pulse sequence uses an internal *phase cycle index*.
- *Receiver* is a root object endowed with properties such as type (phase or diode detection), a vector of *phase cycling settings*, phase cycle index and data accumulation buffers.

A great advantage of object-oriented programming (OOP) is that it leads to a natural model of the experiments with a realistic correspondence between software constructs on one side and physical entities and technical apparatus on the other side.

Naturally, to model the evolution of sample magnetization during an experiment, the OOP objects must be endowed also with suitable functions which confer them functionality. Of these the most important are:

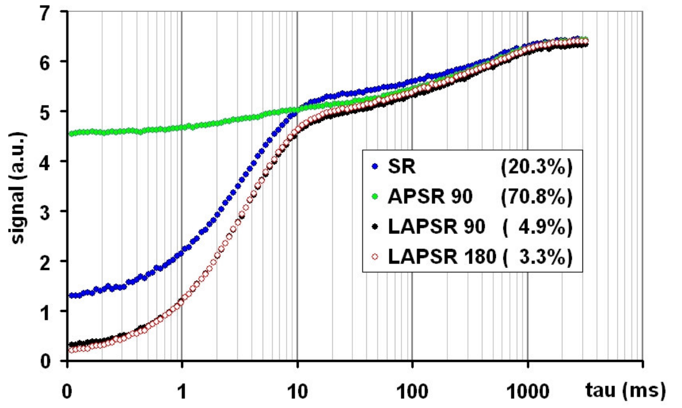
- *Evolution during a delay* which applies either to a *sample component* or to a *sample* and combines their properties with those of a *delay*. In the case of a *sample component*, it changes its current magnetization m according to Bloch equations. In the case of a *sample*, it applies separately to each of its components.
- *Evolution during an RF pulse* also applies either to a *sample component* or to a *sample*, but it combines their properties with those of an *RF pulse*. In the case of a *sample component*, it again changes its current magnetization m to Bloch equations, while in the case of a *sample*, it applies individually to each of its components.
- *Evolution of sample magnetization under the influence of a pulse sequence* applies the preceding evolution functions in the correct order indicated by the *pulse sequence* object to all *sample* components.
- *Sample magnetization evaluation functions* apply to a *sample* and return the various measures of current sample magnetization discussed in the preceding Section.
- In *measurement sequences*, a *receiver object* can be applied either to a *sample component* or to a *sample* and returns the value of the detected signal(s). For a phase detector and a simple component, the signals (u,v) are set equal to the m_x and m_y magnetizations, respectively. For a sample, the signal is a simple arithmetic sum of the signals due to each component.
- Finally, the package contains also functions permitting to simulate repeated applications of a measurement sequence and the *accumulation of the signals* acquired in each scan, keeping in mind any specified phase cycling.

From the above it is evident that the OOP structure of the software package makes it extremely flexible. To get a full advantage of the OOP features, one should use a native OOP language, such as C++. However, while we are currently working on such a package, we have started with a more simple implementation in Matlab, exploiting the capability of the latter to accommodate User-defined structures. This is not optimal when it comes to efficiency (in fact, the Matlab package is very slow) but it permits to obtain almost immediate qualitative results. Without this step we would have ended up with a very fast-executing package but a very poor understanding of the exploratory results and features which represent the core of this article.

Experimental tests

Tests were carried out on a large relaxation phantom with three-components (3.6, 60 and 480 ms) using different saturation sequences and a rather low, inhomogeneous B_1. Under these conditions the traditional APSR sequence, suitable for narrow T_1 distributions, was found even more inefficient than the textbook SR sequence (however, the performance of the latter turns out to be strongly offset-dependent).

As expected from the simulations, LAPSR with 180º pulses gives the least residual magnetization at τ = 0 (% values shown in the parentheses). It also gives by far the the smallest pseudo-noise at small τ–values. These noise-like artifacts are due to undesired echoes contaminating the FID's which, in turn, are linked to the transversal components of the sample magnetization vectors.



9th International Bologna MRPM Conference – MRPM9 – July 13-17, 2008, Cambridge MA, USA