AIP 2019 Grenoble

APPLIED INVERSE PROBLEMS CONFERENCE

JULY, 08th - 12th, 2019

621 Avenue Centrale, 38400 Saint-Martin-d'Hères, France

Minisymposium "NMR Applications: Inverse Problems and Methods"

Stanislav Sykora, Ester Maria Vasini

Extra Byte snc

Extra Byte
>in spin we trust

Realistic simulated MR data (virtual phantoms) and the development of IP algorithms

DOI of this document: 10.3247/SL7Nmr19.004

Simulated data are essential for the

- **development**,
- **testing**, and
- **comparisons**

of **data evaluation algorithms** for inverse problems of all kinds.

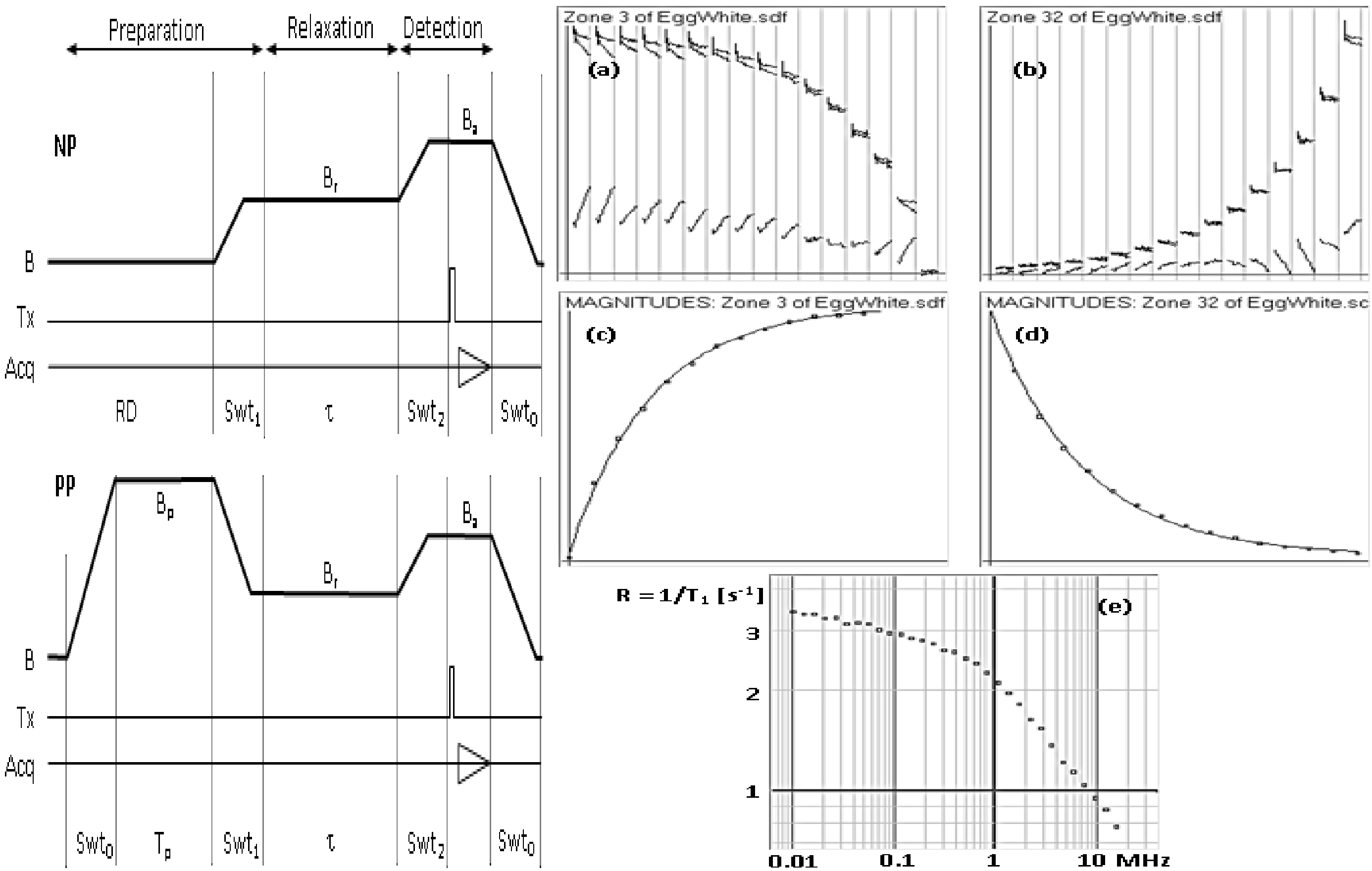
Ok, but WHY do we need to SIMULATE?
Are there not enough real data?

First group of answers:

Software developer's discomfort with «real» data

- To acquire "real" signals one needs "real" samples which might bring forth the limits of the tested algorithm. Which is often not simple, unless the data are already available.
- One also needs **costly real instruments** to generate real signals. Which is even harder and sometimes completely ruled out.
- Acquisition of "real" signals is often **very slow** compared to the rate with which one can generate simulated data.
- "Real" samples are almost always **poorly characterized**.
- "Real" samples often exhibit **unexpected real-life complications** which **shift the focus from the algorithm** to the sample itself.

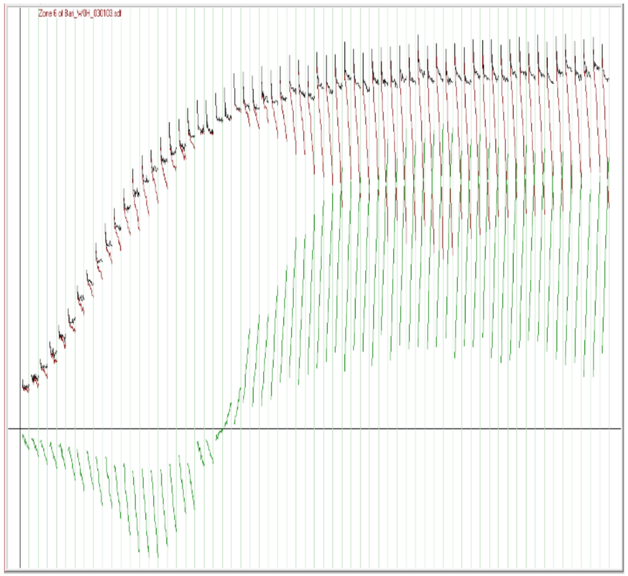
Field-Cycling NMR (**FFC** or **sample-shuffling**)

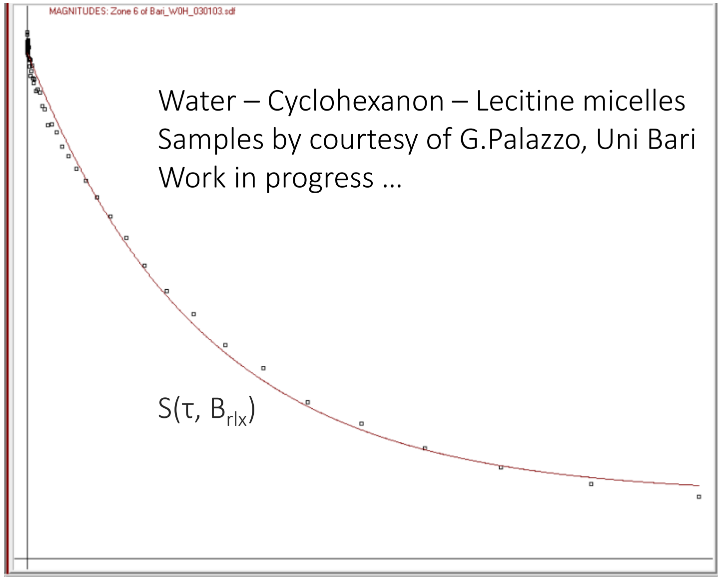


What about FC of <u>multi-component</u> samples?

When we intend only a **small number k of discrete components** that are distinguishable in the decays $S(\tau, B_{rlx})$, then this is not really a new idea.

Yet it is **a bit problematic** and **very time consuming**, so it is **rarely done**.





Second group of answers:
Speeding up the development of a new algorithm

- Usually (though not always), **generation of simulated data is very fast**. This is particularly important when the tested algorithm needs to be numerically validated in terms of its **bias** and/or **noise propagation**, using **Monte Carlo techniques**.
- **Simulated signals are more flexible** in setting sample/acquisition parameters, adding well defined random noise, etc.
- When there exists a good **expert knowledge**, simulated data can be made sophisticated enough to include precisely **controlled amounts of typical artefacts** encountered in experimental data. It is then possible to test to which extent is an algorithm **robust** with respect to such artefacts.

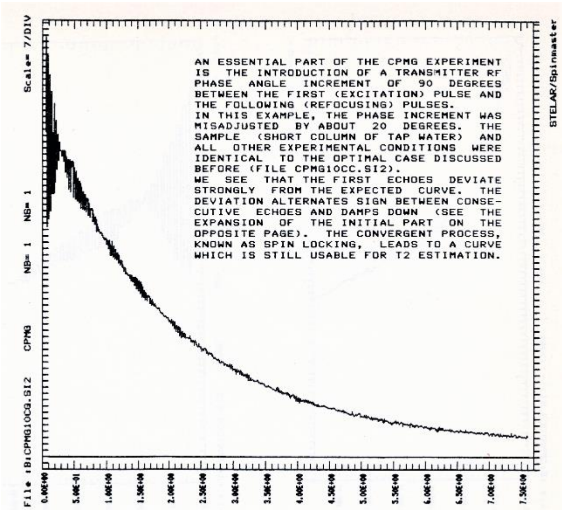
Examples of typical LR-NMR artefacts and problems:

- Noise (low S/N ratio).
- Dead time (receiver "blinded" for a few us after every pulse
- Filter settling distortion of first points. Bruker group delays!
- Imperfect phasing (due to instrument setup or the sample)
- First point(s) of an IR sequence contaminated by spin echo
- Oscillations at the beginning of a CPMG train of echoes
- CPMG baseline drifts (due to several origins)
- Pulse artefacts (phase glitches, amplitude settling)
- Time domain spikes due to environmental interference
- Magnetic field brum (mains pick-up) and other instabilities
- ………… etc.

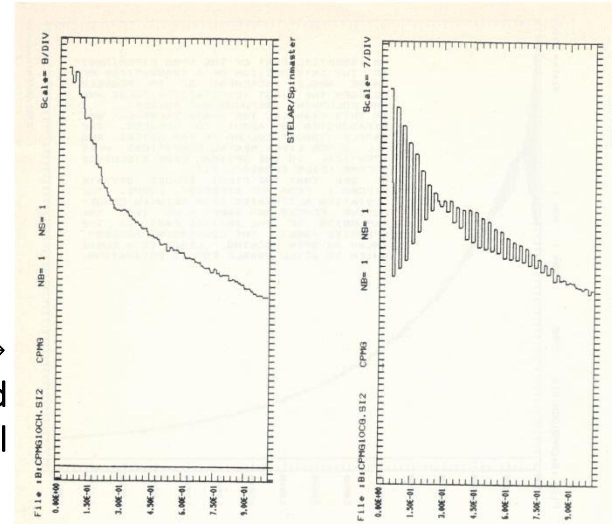
Simulating realistic data is quite an art

But without simulating, how can you trust your algorithms with real data?

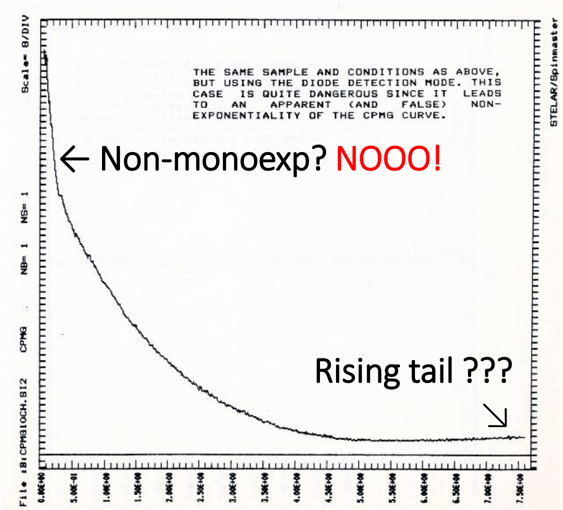
Examples of CPMG artefacts:



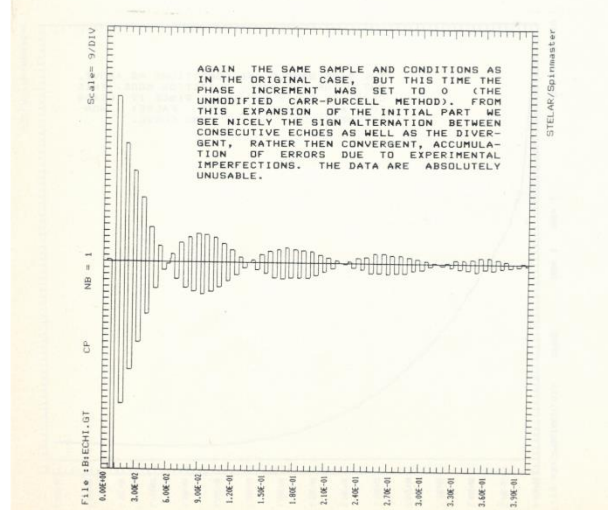
← CPMG
Quad detection,
in-phase channel



CPMG →
Diode vs Quad
detection; detail



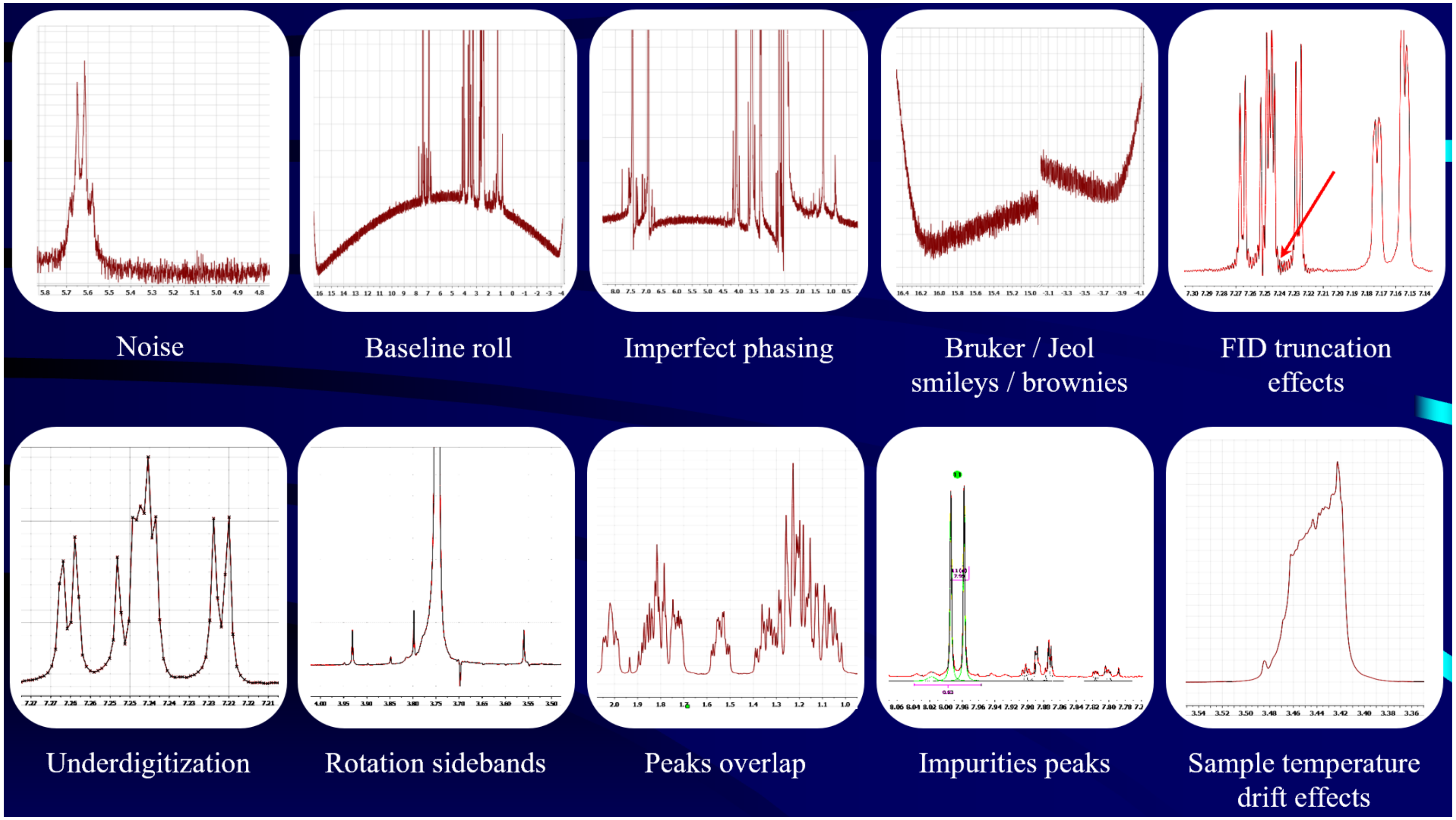
← CPMG
Diode detection



Unmodified CP →
Quad detection

Note: the above are 1982 edu data (before PC’s),
but the «modern» real data are sometimes just as bad!

Examples of typical HR-NMR (spectroscopy) artefacts:



You thought that spectroscopy was better off then relaxometry???

Hehehehehe !!!

Third group of answers

Comparing the relative performance of two or more algorithms

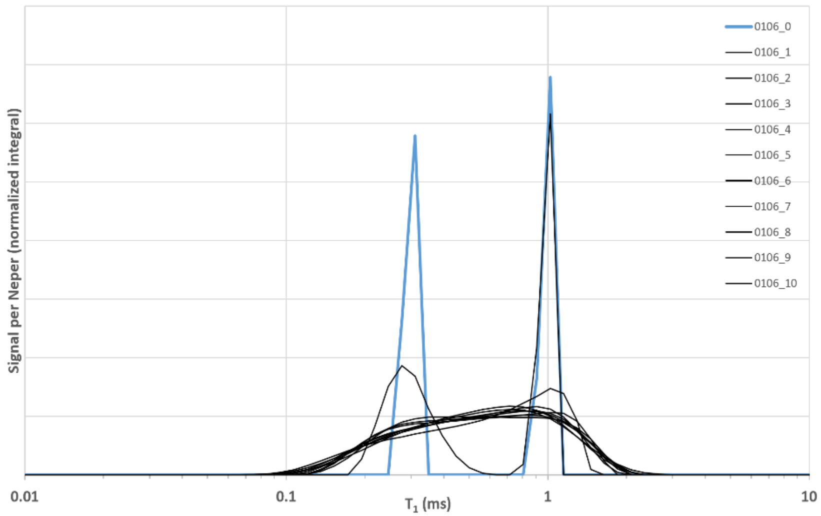
- Real-world algorithms are often aimed at applications of **fuzzy** nature and/or **mathematically ill-defined** ones. In such cases, even the end Users, when presented with several algorithms, are often **unable to tell which one is the best**. More objective methods than User satisfaction (such as **benchmark data sets**) are needed.
- **Training data sets** build from "real" data are often burdened by **errors of human evaluators**. Even infrequent errors in such data sets can have very deleterious effects on long-term development of algorithms.

Hence, **realistically simulated benchmark sets, as well as training sets are often preferable to "real" data**.

A particular study case:

Inverse Laplace Transform of simulated decays

(examples using the UPEN algorithm)



←

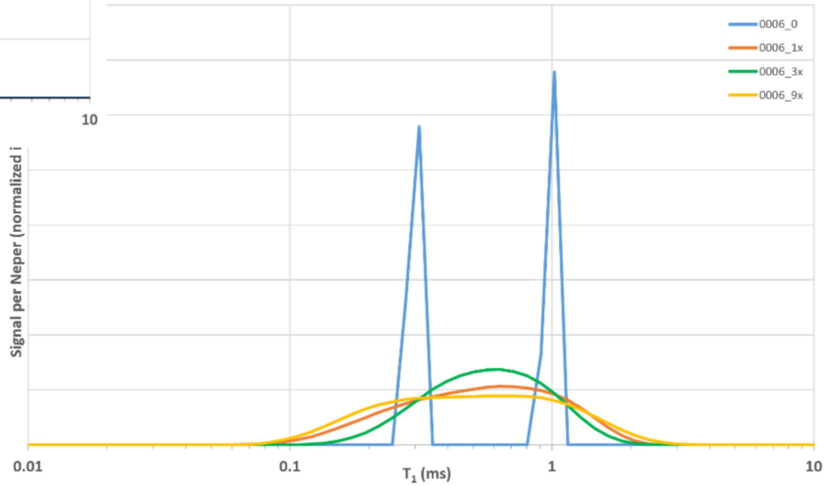
Two decay components

with T_1 values of 1 and 0.3.

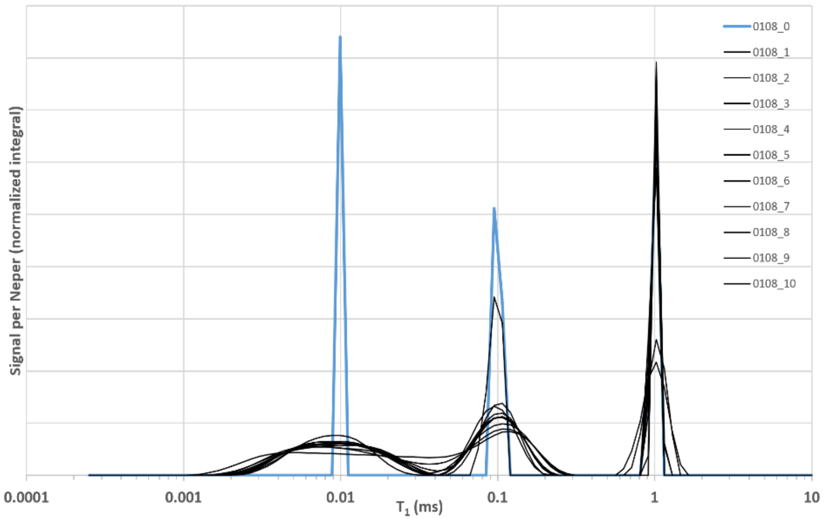
Ten different runs simulated adding always 0.5% noise (just the blue trace has 0% noise)

Four runs simulated adding 0%, 1%, 3% and 9% noise

→

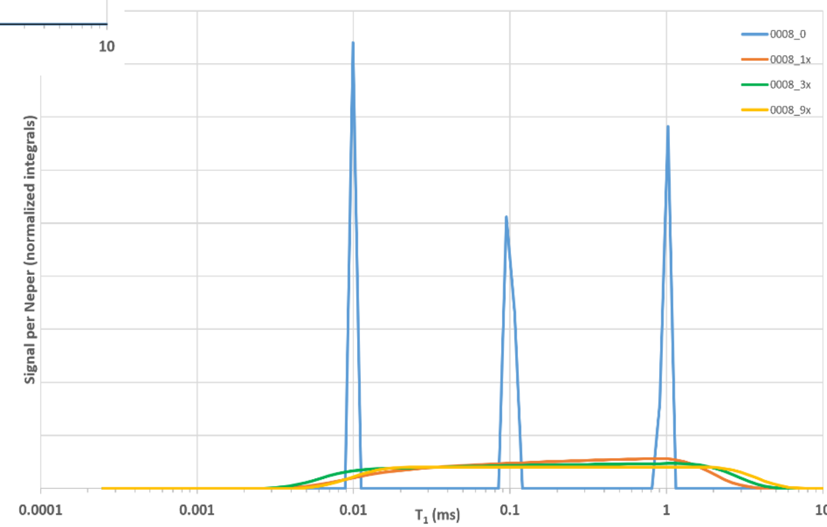


Three relaxation components with equal weights

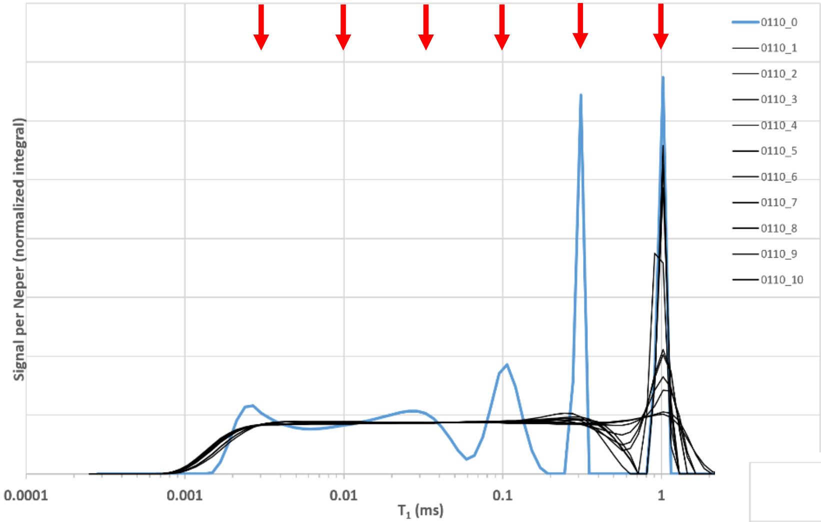


Ten different runs simulated adding always 0.3% of noise
(just the blue trace has 0% noise)

Four runs simulated adding 0%, 1%, 3% and 9% noise



Six relaxation components with the same weights



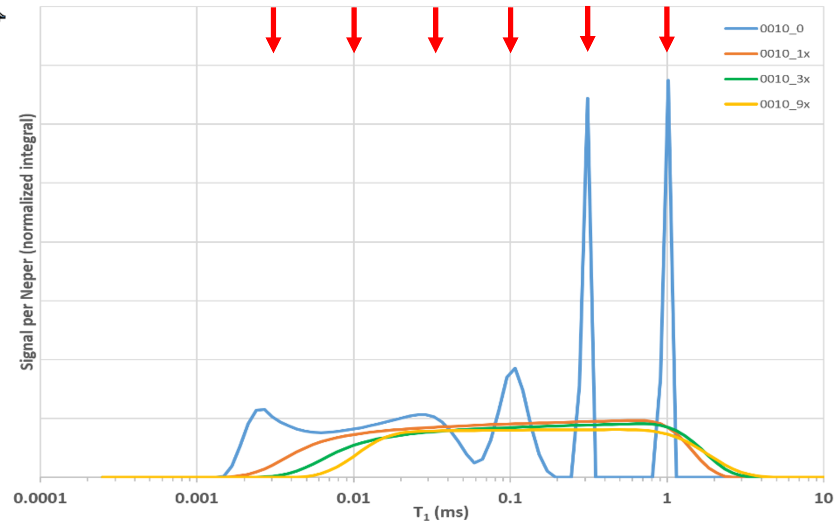
←

Ten different runs simulated adding always 0.16% of noise (just the blue trace has 0% noise)

Better relative resolution for long T_1's, even though tau-values are distributed logarithmically!

Four runs simulated adding 0%, 1%, 3% and 9% noise

→



Algorithms performance testing should:

- Be automated.
- Use well-known and standardized benchmark data sets, preferably simulated, and agreed upon by the whole community.

We call those virtual phantoms

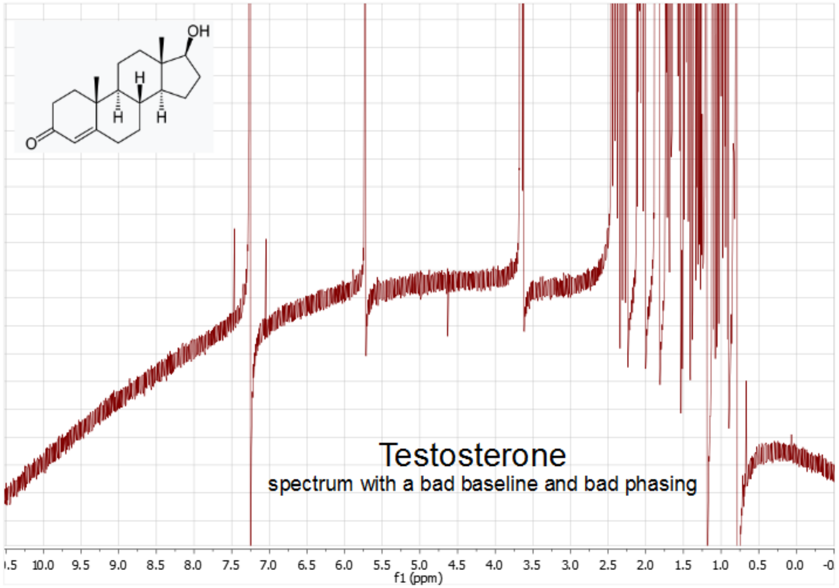
What I did not tell you ☹

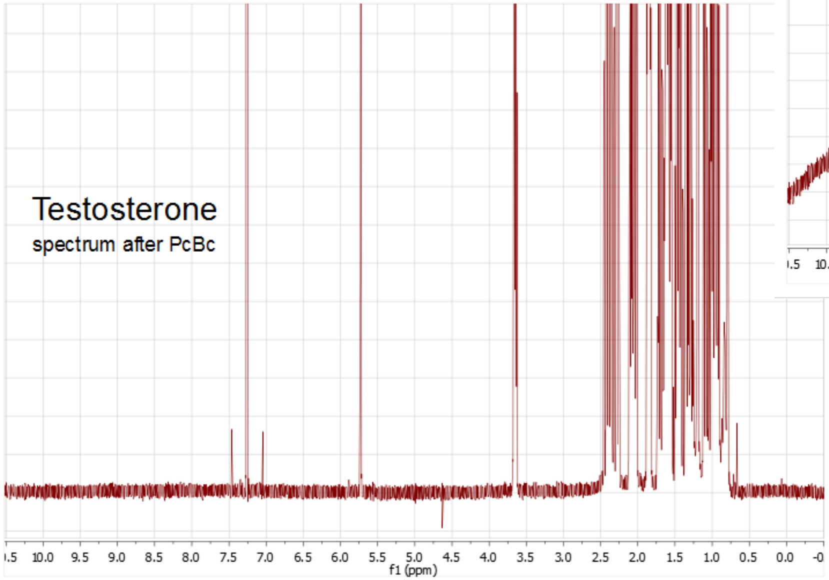
I have originally planned to present much more
but, given the limited time, I could not.

- **Hybrid data**, generated from an experimental data set by adding various artefacts and other features, are also of great interest.
- In particular, iterated evaluations repeated on a hybrid set obtained by adding increasing amounts of noise (**the method of controlled noise addition**) allow in some cases back-extrapolate the results to a zero-noise situation.
- Detailed illustrations of most of the artefact-related concepts had to skipped and will appear in a subsequent publication.

Example of hybrid data used to test a novel PcBc algorithm
(concurrent phase and baseline correction)

Hybrid test data →→→
(distorted experimental)





←←← Corrected by PcBc

Conclusions

- Every software developer needs a **fast generator of simulated data** that is objective and independent of his algorithms.
- It is desirable to be able to **simulate data that include various artefacts**. This requires a considerably deep practical knowledge of the experimental methodology.
- When developing a data-processing algorithm, one should first construct suitable **simulated benchmark data sets** (for **algorithm validation**), and **simulated training data set** (for **algorithm tuning**).
- Fast and realistic generation of simulated data and their various uses is an **emerging, self-standing Science**.

Extra Byte develops commissioned data simulation tools and/or benchmark/training data sets for various NMR Relaxometry and NMR Spectroscopy tasks.

Thank you for your attention!

Extra Byte
>in spin we trust

www.extrabyte.eu