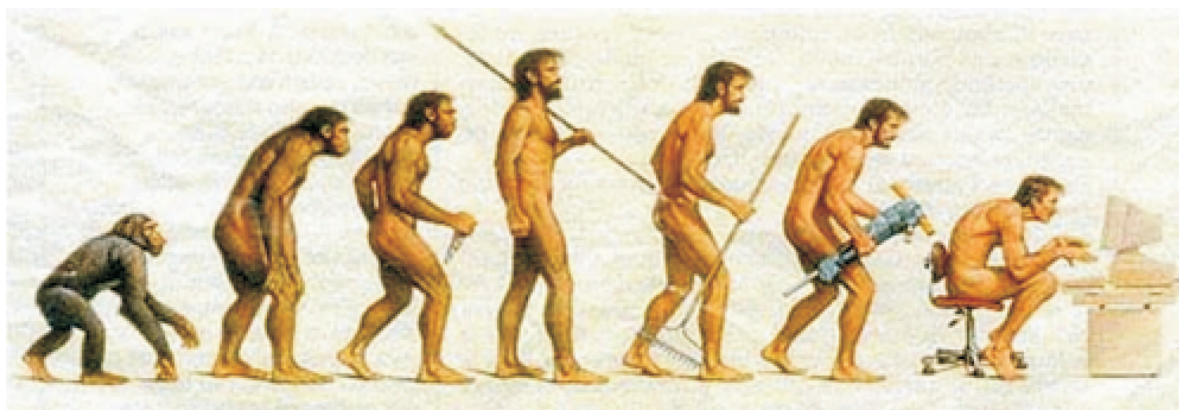


# Description and features of the program EGdeconv: Automated assign and fit programs for powder NMR spectra using Evolutionary Algorithms.

W. Leo Meerts, Arno P.M. Kentgens and Dennis L.A.G. Grimminck  
Radboud University Nijmegen, Institute for Molecules and Materials,  
Heyendaalseweg 135, NL-6525 AJ Nijmegen, The Netherlands\*  
(Dated: July 26, 2013)

**1d powder spectra** A fast and accurate fit program is presented for deconvolution of one-dimensional solid-state quadrupole NMR spectra of powdered materials. Computational costs of the synthesis of theoretical spectra are reduced by the use of libraries containing simulated time/frequency domain data. These libraries are calculated once and with the use of second-party simulation software to ensure a maximum flexibility and accuracy with respect to experimental conditions. *EASY-GOING deconvolution (EGdeconv)* is equipped with evolutionary algorithms that provide robust many-parameter fitting and offers efficient parallelised computing. The program supports quantification of relative chemical site abundances and (dis)order in the solid-state by incorporation of (extended) Czjzek and order parameter models. To illustrate *EGdeconv*'s current capabilities, we provide three case studies. Given the program's simple concept it allows a straightforward extension to include other NMR interactions. The program is available *as is* for 64-bit Linux operating systems.

**MQMAS** The *EASY-GOING deconvolution (EGdeconv)* program is extended to enable fast and automated fitting of MQMAS spectra guided by evolutionary algorithms. In addition to providing an ideal crystallite excitation model for spectrum simulation, we included analytical excitation efficiencies for specific cases. Currently these efficiencies are limited to two-pulse and z-filtered 3QMAS spectra of spin  $3/2$  and  $5/2$  nuclei. The analytical expressions are explained in full to avoid ambiguity and facilitate others to use them. The *EGdeconv* program offers interaction parameter distribution models for chemical shift and quadrupolar interaction, Gaussian and (extended) Czjzek distributions respectively. We provide three case studies to illustrate *EGdeconv*'s capabilities for fitting MQMAS spectra that may have multiple (overlapping) chemical sites.



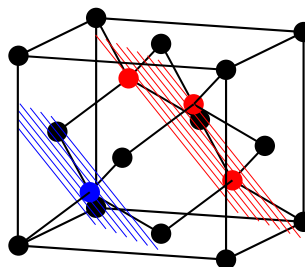
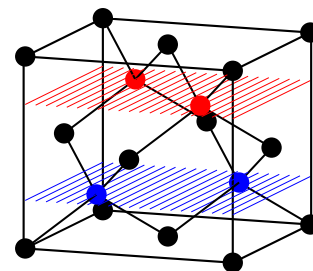
PACS numbers: 07.05.Kf, 33.20.Lg, 33.20.Sn, 33.15.Mt, 36.40.Mr

Keywords: spectrum deconvolution, solid-state NMR, evolutionary algorithms, quadrupole NMR

<b>Contents</b>		<b>I. Introduction</b>	3
<b>Copyright</b>	2	<b>II. Evolutionary Algorithms</b>	4
<b>Preface</b>	3	A. Introduction	4
<b>Acknowledgement</b>	3	1. The genetic algorithm	4
		2. The derandomized-ES DR2 algorithm	5
		3. The CMA Evolution Strategy	6
		B. Definition of the Fitness/Cost functions	7
		1. Definition of the least squares Cost functions	7
		2. Definition of the inproduct fitnessfunction	7

\*Electronic address: [leo.meerts@science.ru.nl](mailto:leo.meerts@science.ru.nl); URL: <http://www.leomeerts.nl>

3. Numerical evaluation of the inproduct fitness function.	7	D. Comparison between the genetic and the derandomized-ES DR2 algorithms.	37
4. Effect of the width ( $\Delta w$ ) of the overlap function $w(r)$ on the convergence of the GA	8	<b>X. Installation of the <i>EGdeconv</i> software</b>	37
C. Selective weight of parts of the spectrum.	8	A. Introduction	37
D. Remove smoothed background.	9	1. Prerequisites	38
<b>III. Theory</b>	9	B. Installation	38
A. Solid State NMR	9	1. First time installation of the EGdeconv package	38
1. The nuclear quadrupole interaction	9	C. Updating the <i>EGdeconv</i> program	38
2. (Extended) Czjzek distribution	10	<b>XI. Miscellaneous program</b>	38
3. The order parameter model	11	A. answer42	38
4. Composing a theoretical Solid-State NMR spectrum	12	<b>XII. Reference to examples of the success of <i>EGdeconv</i> automatic assignments</b>	39
5. Introduction to Multiple Quantum Magic Angle Spinning	15	<b>Appendices</b>	39
6. Composing a MQMAS spectrum	16	A. Conditions for the W-matrix to be positive definite	39
<b>IV. Explanation and definition of the input parameters of <i>EGdeconv</i></b>	20	<b>References</b>	39
A. General	20		
B. The keyword string " <i>parameter</i> "	20		
1. Definition and use of the keyword string " <i>parameter</i> "	20		
2. List of parameters.	21		
3. Parameter constraints.	21		
<b>V. Parameter and option list used by the <i>EGdeconv</i> program</b>	22		
A. Fit parameters	22		
B. Control options of the <i>EGdeconv</i> program	22		
C. Specific definitions of the options of the 1D <i>EGdeconv</i> program	27		
D. Specific definitions of the options of the MQMAS <i>EGdeconv</i> program	28		
<b>VI. Running the program <i>EGdeconv</i></b>	29		
A. Introduction	29		
B. Stopping a EGdeconv-job	29		
C. Resume a EGdeconv-job	29		
D. Output generated by EGdeconv.	29		
<b>VII. 1D-NMR examples</b>	30		
A. $^{27}\text{Al}$ MAS NMR spectrum of Yttrium Aluminium Garnet	30		
B. Oxidation of Sodium Aluminium Hydride ( $\text{NaAlH}_4$ )	30		
C. Static Hahn echo spectrum of Aluminium Gallium Arsenide ( $\text{Al}_x\text{Ga}_{1-x}\text{As}$ )	31		
<b>VIII. MQMAS examples</b>	32		
A. Aluminium Alkoxides	32		
B. Rubidium-nitrate	32		
C. Yttrium-sialon glass	33		
<b>IX. How to optimize the <i>EGdeconv</i> fits</b>	34		
A. Tips for running a succesfull <i>EGdeconv</i> fit.	34		
B. Influence of the width ( $\Delta w$ ) of the overlap function $w(r)$ on the convergence of the EA	35		
C. Meta-optimization of the internal GA parameters	36		



Copyright

Copyright©2002/2013 W. Leo Meerts and Michael Schmitt. The usage of the *EGdeconv* program is free if a reference is made to

Dennis L. A. G. Grimminck, Ben J. W. Polman, Arno P. M. Kentgens, and W. Leo Meerts. **EASY-GOING deconvolution: Combining accurate simulation and evolutionary algorithms for fast deconvolution of solid-state quadrupolar NMR spectra.** *J. Magn. Resonance* 211 (2011) 114-120. doi: <http://dx.doi.org/10.1016/j.jmr.2011.04.009>.

or

Dennis L. A. G. Grimminck, Bas van Meerten, Margriet H. W. Verkuijlen, Ernst R. H. van Eck,

W Leo Meerts, and Arno P. M. Kentgens. **EASY-GOING deconvolution: Automated MQMAS NMR spectrum analysis based on a model with analytical crystallite excitation efficiencies.** *J. Magn.Resonance* **228 (2013) 116-124.** doi: <http://dx.doi.org/10.1016/j.jmr.2012.12.012>.

for fitting MQMAS

The above papers can be found on our website [egdeconv.science.ru.nl](http://egdeconv.science.ru.nl).

Separate license agreements and copyrights are applicable to the genetic algorithm package PgaPack, the multiprocessor software MPI(CH) and the spectra display and manipulation program *answer42*, in case of usage of these packages. We refer here to the particular conditions for use that comes with those packages. By installing and/or using the *EGdeconv* program and accompanying software, you accept the license conditions.

### Preface

This manual is part of a larger manual for the automated deconvolution program [1–3], that includes many more deconvolution models in a range of spectroscopy applications. During compilation of this manual and the *EGdeconv* program we purposely excluded parts that we will not support.

This manual has been prepared with great care and reflects the version 2.70 (time stamp 24-May-2013) and the options of the *EGdeconv* program with which it is included. The program is compiled for 64-bit Linux operating systems and is only available for ubuntu 12.04 LTS and ubuntu 10.04 LTS.

The program is accompanied with an automatic installation as described in section X B. Before starting the installation of the *EGdeconv* program, please read section X A 1.

If you encounter a problem you are requested to send a report to the authors with (i) the type of error, including the full error message, (ii) the operating system, (iii) the program version and its time stamp<sup>1</sup>, (iv) the input files and (v) if present, the output file(s).

Since the *EGdeconv* program is still under development the authors greatly appreciate all suggestions and remarks to improve the program, to simplify/modify the input and requests for additional output(s).

Suggestions to extend or improve this manual will also seriously be considered.

All remarks can be sent to [Leo.Meerts@science.ru.nl](mailto:Leo.Meerts@science.ru.nl).

W. Leo Meerts and Dennis L.A.G. Grimminck  
Nijmegen, The Netherlands, July 26, 2013

### Acknowledgement

The authors thank Paulus J. Knijn, Ernst R.H. van Eck and Margriet W.H. Verkuijlen for providing experimental spectra for analysis using *EGdeconv*. Ben Polman is thanked for his help in making the program as portable as possible and Bert Groenenboom for giving the mathematical prove presented in Appendix A. We thank Gerrit C. Groenenboom for useful discussions on theoretical considerations and testing of the program. The Netherlands Organisation for Scientific Research (NWO) and Radboud University Nijmegen are acknowledged for their financial support of the solid-state NMR facility for advanced material science.

### I. INTRODUCTION

This project started as a collaboration between Arno Kentgens and Paul Knijn of the Solid State Nuclear Magnetic Resonance group and Dennis Grimminck and Leo Meerts of the Molecular and Biophysics group both part of the Institute for Molecules and Materials of the Radboud University Nijmegen. The aim of this project was to apply an automated fitting procedure using evolutionary algorithms to quadrupole interaction governed NMR powder spectra measured at our NMR facility in Nijmegen. The success of such a procedure to interpret high resolution laser induced fluorescence spectra by W. Leo Meerts *et al.* [2, 3] gave the first incentive to start this work.

Quadrupole nuclear magnetic resonance is a valuable tool in materials research. The technique applies to atomic nuclei that possess a total angular momentum  $\geq 1$  and thereby a quadrupole moment [4]. This concerns over two-thirds of the NMR active nuclei, hence the area of application for quadrupole NMR is potentially large. Solid-state NMR techniques help to study the interaction between the quadrupole moment of nuclear charge and the electrical field gradient (*efg*) generated by the chemical surroundings; the microscopic system of nucleus and surroundings is known as a site. The nucleus thereby acts as a probe for local structure, which typically extends over several coordination spheres [5]. This makes quadrupole NMR a particularly sensitive technique for structural information from (partly) disordered solid-state materials, e.g. glasses, semiconductors and polymers. A class of materials that cannot be studied by X-ray techniques due to their requirement for crystalline materials.

Structural information contained in the experimental spectra can be extracted by deconvolution in terms of quadrupole interaction parameters and relative abundances of different sites. This requires a deconvolution model, which accurately describes the experiment and spin system at reasonable computational costs. Fitting programs such as DMFit [6] and QuadFit [7] have their own dedicated model, which is optimised towards low costs, but not versatility in terms of experimental conditions. Conversely, there are general purpose NMR simulation programs, such as SIMPSON [8] and SPINEVO-LUTION [9], that are versatile but miss dedication for spectrum fitting. In practise the quality of simulations

<sup>1</sup> The time-stamp of the currently installed *EGdeconv* program can be obtained by the command: *ga-timestamp*. It is also displayed in the output header of the *EGdeconv* program after running it.

for quadrupole NMR is mostly restrained by computational costs, for example due to orientational averaging of crystallites for powder spectra. With this in mind we created a separation between the computationally intensive simulations and the actual data-fitting, by the use of second-party software to pre-calculate time or frequency domain libraries that are subsequently used during the fitting. This approach allows, as for example in the case of  $\text{Al}_{0.5}\text{Ga}_{0.5}\text{As}$  (discussed below), a  $\sim 700$  ms calculation of a five-site spectrum, all sites with a quadrupole parameter distribution, on a 2.3 GHz core.

A second requirement for spectrum deconvolution is the choice of a suitable algorithm to the fitting of parameters. If the experimental spectrum contains spectral lines of multiple sites that are not well resolved, the deconvolution model becomes more complex and the number of fit parameters increases. To our knowledge, only one NMR fitting program [10] is able to handle this many parameters, by the use of genetic algorithms. In combination with SIMPSON, this program was used to fit spectra of Vanadium nuclei. It, however, is not readily available, and does not separate the fitting from the time consuming simulation step. We introduce the use of evolutionary algorithms [1], of which genetic algorithms are a sub-class, in the *EGdeconv* program. The additional advantage of these algorithms is that they can be efficiently parallelised, leading to an increased computation speed. Typical convergence rates involve 100 iterations with a population of 30 trial-solutions. Assuming the case of a 2.3 GHz core previously mentioned, this amounts to 35 minutes for a spectrum deconvolution. This calculation time is further reduced by the use of the parallelised code on all available cores.

Deconvolution of spectra from (partially) disordered materials requires accounting for variations in interaction angles and/or distances. For the quadrupole interaction this corresponds to retrieval of the distribution of its interaction parameters. Czjzek *et al.* [11] were able to determine the shape of such a distribution for random ordering. By calculating the *efg* tensor components for a large number of random repositionings of surrounding charges, they determined the components become normally distributed. Starting from this multivariate Gaussian distribution they could analytically derive a probability density function for the quadrupole interaction parameters. Le Caër *et al.* [12, 13] extended this model to express the probability density function for a microscopic system with an ordered short-range and disordered long-range surroundings.

Underlying distributions in structure influence the shape of spectral lines. Deconvolution of the experimental spectrum assisted by these models provides a way to indicate and possibly quantify structural disorder. Currently QuadFit allows the use of the Czjzek model for data-fitting, but no fit-programs are readily available that apply the extended model. Complementary to the microscopic Czjzek model, there is a more macroscopic approach to quantify (dis)order that fixes the relative site abundances given a crystal structure. This fixed set of abundances is related to an order parameter [5, 14], of which the value can be varied to range from disordered structure to a targeted crystal struc-

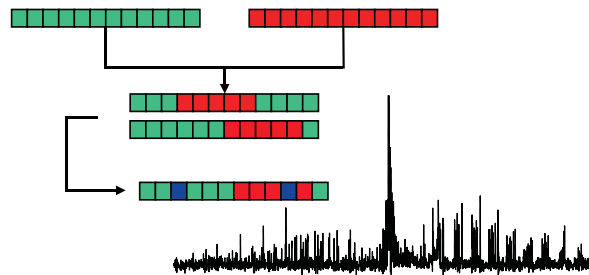


FIG. 1: Schematics of the GA-process

ture. Fitting this parameter provides an indication of the material's structure.

The remainder of this manual will be concerned with explaining the various parts of the program. As mentioned in the preface, this manual is part of a larger manual of the ga-program from which *EGdeconv* was derived; the reader may notice this during reading.

## MQMAS project

The MQMAS project started as a master's research project by Bas van Meerten in november 2010 under direct supervision of Dennis Grimminck. We started by playing with simpson and convincing ourselves that a library approach or directly linking SIMPSON to the *EGdeconv* program was too time-consuming or computer-memory intensive. Then we decided to learn about the behaviour of multiple-quantum-excitation via numerical integration of the Liouville-Von Neumann equation for the full quadrupolar interaction Hamiltonian including rf-field. When we saw the relatively simple structure of the answer, we saw an opportunity to advance analytically. This worked so well we were able to publish the results.

The MQMAS related parts of this manual are part of Bas' Master thesis. That is reason that the text and some of the symbols in the mathematics differ from the paper. We hope the combination of the paper and the master thesis provide slightly different and helpful angles for readers to learn from.

## II. EVOLUTIONARY ALGORITHMS

### A. Introduction

Three different type of Evolutionary Algorithms are implemented in the current program: The genetic algorithm, the derandomized-ES DR2 algorithm and the CMA Evolution Strategy (CMA-ES).

#### 1. The genetic algorithm

A description of the GA used in this investigation can be found in [2, 15]. The genetic algorithm is basically a global optimizer, which uses concepts copied from natural reproduction and selection processes. For a detailed description of the GA the reader is referred to the original literature [16–18]. We shortly introduce

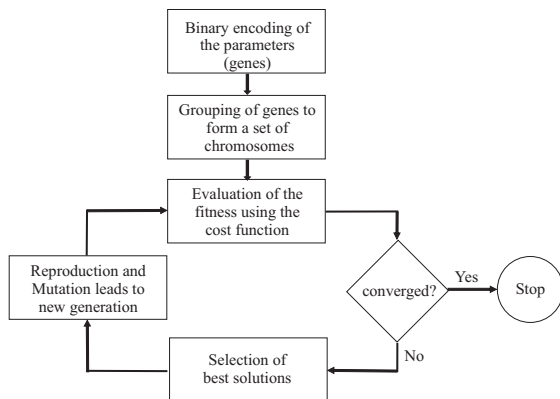


FIG. 2: Schematics of the GA-process

the elements of the GA, which will be used in the following.

- Representation of the parameters: The molecular parameters are encoded binary or as real data type, each parameter representing a gene. A vector of all genes, which contains all molecular parameters is called a chromosome. In an initial step the values for all parameters are set to random values between lower and upper limits which have to be chosen by the user. No prior knowledge of the parameters is necessary. A total of 300 - 500 chromosomes are randomly generated, forming a population.
- The solutions (chromosomes) are evaluated by a fitness function (or cost function), which is a measure for the quality of the individual solution. The fitness function which is used here, is described in Sect. II B.
- One optimization cycle, including evaluation of the cost of all chromosomes is called a generation. Generally convergence of the fit in our case is reached after 300 - 500 generations.
- Pairs of chromosomes are selected for reproduction and their information is combined via a crossover process. This crossover might take place as a one-point, two-point or uniform crossover. As crossover just combines information from the parent generations. It basically explores the error landscape.
- The value of a small number of bits is changed randomly. This process is called mutation. Mutation can be viewed as exploration of the cost surface. The best solutions within a generation are excluded from mutation. This elitism prevents already good solutions from being degraded.

The performance of the GA depends on internal parameters like mutation rate, elitism, crossover probability and population size, which therefore should also be optimized for a given problem. Fortunately this meta-optimization results in similar parameters for

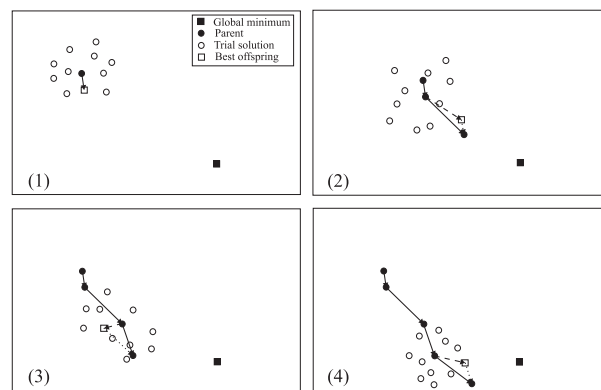


FIG. 3: The first four generations of an Evolutionary Strategy: (1) An initial population is generated, and the best offspring is used as the next parent. (2) The offspring is spread over a larger area in the second generation due to the relatively large step made in the previous generation. The vector from the parent to the best offspring (dashed line) is combined with the (shortened) mutation vector of the last generation (dotted line) to generate the new parent (solid line). (3) Due to the correlation between the past two mutations the search range has been extended again in the general direction of both mutations while it has been limited in the perpendicular direction. The best offspring is now a local minimum. The memory effect of the evolutionary algorithm, which incorporates past mutation vectors into the calculation of the next parent, helps to overcome the local minimum and the next parent is still closer to the global minimum. (4) The barrier between the local and global minima has been overcome, and the optimization is progressing towards the global minimum.

quite different problems of optimization. The meta-optimization for some of the parameters is described in [2].

## 2. The derandomized-ES DR2 algorithm

Mutative step-size control tends to work well for the adaptation of a global step-size, but tends to fail when it comes to the individual step size. This is due to several disruptive effects [19] as well as to the fact that the selection of the strategy parameters setting is indirect, i.e. not the vector of a successful mutation is used to adapt the step size parameters, but the parameters of the distribution that led to this mutation vector. The so-called derandomized mutative step-size control aims to tackle those disruptive effects. The first versions of derandomized evolutionary strategy (ES) algorithms introduced a controlled global step-size in order to monitor the individual step-sizes by decreasing the stochastic effects of a probabilistic sampling. The selection disturbance was completely removed with later versions by omitting the adaptation of strategy parameters by means of probabilistic sampling. This was combined with individual information from the last generation (the successful mutations, i.e. of selected offspring), and then adjusted to correlated mutations. Later on, the concept of adaptation by accumulated information was introduced, aiming to use wisely the past informa-

tion for the purpose of step size adaptation: instead of using the information from the last generation only, it was successfully generalized to a weighted average of the previous generations.

It is important to note that the different variants of *derandomized-ES* hold different numbers of strategy parameters to be adapted, and this is a factor in the speed of the optimization course. The different algorithms hold a number of strategy parameters in either a linear or quadratic order in terms of the dimensionality of the search problem  $n$ , and there seems to be a trade-off between the number of strategy parameters and the time needed for the adaptation/learning process of the step sizes.

We hereby present briefly the derandomized-ES DR2 algorithm that is in use in our program. The DR2 Algorithm [20] is aimed to accumulate information about the correlation or anti-correlation of past mutation vectors in order to adapt the step size: We denote by  $\vec{x}_{E/N}^g$  the parameter vector of generation  $g$  (**E**: parent / **N**: offspring):

$$\vec{x}_N^g = \vec{x}_E^g + \delta^g \delta_{scal}^g \vec{Z}^k \quad (1)$$

$$\vec{Z}^k = \vec{N}(0, \mathbf{I}) \quad (2)$$

$$\vec{Z}^g = c \vec{Z}_{sel} + (1 - c) \vec{Z}^{g-1} \quad (3)$$

$$\delta^{g+1} = \delta^g \cdot \left( \exp \left[ \frac{|\vec{Z}^g|}{\sqrt{n} \sqrt{\frac{c}{2-c}}} - 1 + \frac{1}{5n} \right] \right)^\beta \quad (4)$$

$$\delta_{scal}^{g+1} = \delta_{scal}^g \cdot \left( \frac{|\vec{Z}^g|}{\sqrt{\frac{c}{2-c}}} + 0.35 \right)^{\beta_{scal}} \quad (5)$$

Here  $n$  is the number of parameters to fit and  $\vec{N}(0, \mathbf{I})$  stands for a random number between 0 and 1 for each fitting parameter. The values for  $\beta$  and  $c$  are both set to  $1/\sqrt{n}$ , while  $\beta_{scal} = 1/n$ .

### 3. The CMA Evolution Strategy

The CMA-ES (Covariance Matrix Adaptation Evolution Strategy)<sup>2</sup> is an evolutionary algorithm for difficult non-linear non-convex optimization problems in continuous domain. The CMA-ES is typically applied to unconstrained or bounded constraint optimization problems, and search space dimensions between three and a hundred. The method should be applied, if derivative based methods, e.g. quasi-Newton BFGS or conjugate gradient, (supposedly) fail due to a rugged search landscape (e.g. discontinuities, sharp bends or ridges,

noise, local optima, outliers). If second order derivative based methods are successful, they are usually faster than the CMA-ES: on purely convex-quadratic functions,  $f(x) = \mathbf{x}^T \mathbf{H} \mathbf{x}$ , BFGS (Matlabs function `fminunc`) is typically faster by a factor of about ten (in terms of number of objective function evaluations needed to reach a target function value, assuming that gradients are not available). On the most simple quadratic function  $f(x) = \|\mathbf{x}\|^2 = \mathbf{x}^T \mathbf{x}$  BFGS is faster by a factor of about 30.

Similar to quasi-Newton methods the CMA-ES is a second order approach and estimates the inverse Hessian matrix (more precise: a covariance matrix that is closely related to the inverse Hessian if it exists) within an iterative procedure. This makes the method feasible on non-separable and/or badly conditioned problems. In contrast to quasi-Newton methods the CMA-ES does not use or approximate gradients and does not even presume or require their existence. This makes the method feasible on non-smooth and even non-continuous problems, as well as on multimodal and/or noisy problems. It turns out to be a particularly reliable and highly competitive evolutionary algorithm for local optimization [19] and, surprising at first sight, also for global optimization[21]<sup>3</sup>.

The CMA-ES has several invariance properties. Two of them, inherited from the plain evolution strategy, are (i) invariance to order preserving (i.e. strictly monotonic) transformations of the objective function value (that is, e.g.,  $\|\mathbf{x}\|^2$  and  $3\|\mathbf{x}\|^{0.2} - 100$  are completely equivalent objective functions), and (ii) invariance to angle preserving (rigid) transformations of the search space (including rotation, reflection, and translation), if the initial search point is transformed accordingly. Invariances are highly desirable: they imply uniform behavior on classes of functions and therefore imply the generalization of empirical results.

The CMA-ES does not leave the choice of strategy internal parameters open to the user (arguably with the exception of population size  $\lambda$ ). Finding good (default) strategy parameters is considered as part of the algorithm design, and not part of its application. The aim is to have a well-performing algorithm as is. The default population size  $\lambda$  is comparatively small to allow for fast convergence. Restarts with increasing population size [22] improve the global search performance. For the application of the CMA-ES, the initial solution, an initial standard deviation (step-size, variables should be defined such that the same standard deviations can be reasonably applied to all variables) and eventually the termination criteria (e.g. a function tolerance) need to be set by the user.

<sup>2</sup> This method has been developed by Nikolaus Hansen: <http://www.bionik.tu-berlin.de/user/niko/>

<sup>3</sup> See also <http://www.bionik.tu-berlin.de/user/niko/cec2005.html>

## B. Definition of the Fitness/Cost functions

Currently three Fitness/Cost functions are available in *EGdeconv*. Two of them are least squares based, while the third uses an inproduct with a special metric.

### 1. Definition of the least squares Cost functions

Both least squares cost functions are of the form

$$C = \sum_{i=1}^N \left( \frac{p_i^{exp}}{W^{exp}} - \frac{p_i^{calc}}{W^{calc}} \right)^2 \quad (6)$$

with  $N$  the number of spectrum points and  $p_i$  the  $i^{th}$  spectrum point. The functions differ in the definition of

$W$ , which can either be the maximum intensity or the integral over the absolute values of the corresponding spectrum. Effectively the difference is a comparison between two spectra with the same maximum intensity or the same absolute value integral.

### 2. Definition of the inproduct fitnessfunction

In the following discussion we will use both the terms *Fitness function* ( $F_{fg}$ ) and *Cost function* ( $C_{fg}$ ), where  $C_{fg} = 100(1 - F_{fg})$  to characterize the quality of a solution. The fitness function for the fit of the spectra with  $N$  points using the genetic algorithm has been defined in Eq. (5) of Hageman et al.[15] as:

$$F_{fg} = \frac{\sum_{r=-l}^l w(r) \sum_{i=1}^N f(i)g(i+r)}{\sqrt{\sum_{r=-l}^l w(r) \sum_{i=1}^N f(i)f(i+r)} \sqrt{\sum_{r=-l}^l w(r) \sum_{i=1}^N g(i)g(i+r)}}. \quad (7)$$



FIG. 4: Impression of a complicated error landscape

In ref. [15]  $C_{fg}$  is identical to  $F_{fg}$  in this paper. In this equation  $f$  and  $g$  represent the experimental and calculated spectra, respectively. The overlap function  $w(r)$  determines the sensitivity of the fitness function for a shift of the two spectra relative to each other. This can be rewritten by interchanging the sums and substituting  $i + r = j$  to

$$F_{fg} = \frac{\sum_{i,j} f_i W_{ij} g_j}{\sqrt{\sum_{i,j} f_i W_{ij} f_j} \sqrt{\sum_{i,j} g_i W_{ij} g_j}}, \quad (8)$$

where

$$W_{ij} = w(|j - i|) \quad (9)$$

and  $f_i = f(i)$  and  $g_i = g(i)$ .  $F_{fg}$  in Eq. (8) can be interpreted as the cosine of the "angle" between the experimental and theoretical spec-

trum. With the column vectors:

$$\begin{aligned} \mathbf{f} &= (f_1, f_2, \dots, f_N)^T \\ \mathbf{g} &= (g_1, g_2, \dots, g_N)^T \end{aligned} \quad (10)$$

and the symmetric matrix  $\mathbf{W}$  which has the matrix elements  $W_{ij}$  we can write:

$$F_{fg} = \cos(\alpha) = \frac{(\mathbf{f}, \mathbf{g})}{\|\mathbf{f}\| \|\mathbf{g}\|}. \quad (11)$$

Here the inner product  $(\mathbf{f}, \mathbf{g})$  is defined with the metric  $\mathbf{W}$  as:

$$(\mathbf{f}, \mathbf{g}) = \mathbf{f}^T \mathbf{W} \mathbf{g}, \quad (12)$$

and the norm of  $\mathbf{f}$  as  $\|\mathbf{f}\| = \sqrt{(\mathbf{f}, \mathbf{f})}$ ; similar for  $\mathbf{g}$ . For the overlap function  $w(r)$  we use a triangle function[15] with a width of the base of  $\Delta w$ :

$$w(r) = \begin{cases} 1 - |r| / (\frac{1}{2} \Delta w) & \text{for } |r| \leq \frac{1}{2} \Delta w \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

In order for  $F_{fg}$  to serve as a good fitness function for the quality of the fit, it should have the property that it reaches its maximum value if and only if  $\mathbf{f}$  and  $\mathbf{g}$  are identical (apart from a normalization). This condition is fulfilled provided the matrix  $\mathbf{W}$  is positive definite. In Appendix A it has been shown that this holds for  $w$  defined in Eq. (13).

### 3. Numerical evaluation of the inproduct fitness function.

Let us now consider the numerical evaluation of the fitness function  $F_{fg}$  from Eq. (8) in its relation with the calculated spectrum. The calculated spectrum is

obtained by a convolution of each calculated transition ( $k$ ) with intensity  $s_k$  by the lineshape function  $L$ :

$$g_j = \sum_{k=1}^N L_{jk} s_k \quad (14)$$

with  $L_{jk} = l(|j - k|)$ . In matrix notation this can be rewritten as:

$$\mathbf{g} = \mathbf{L}\mathbf{s}. \quad (15)$$

As it turned out at least 50 % of the computing time in [15] was used to perform a straight forward calculation of the fitness function  $F_{fg}$  from Eq. (8). For a typical GA fit  $F_{fg}$  must be calculated 150,000 times. Hence a considerably reduction in computing time can be obtained by a more effective calculation of  $F_{fg}$ . We start with a rearrangement of the order of the evaluation of Eq. (11) and using the properties of  $\mathbf{W}$  and  $\mathbf{L}$ . The numerator of Eq. (11) is evaluated first:

$$(\mathbf{f}; \mathbf{L}\mathbf{s}) = \mathbf{f}^T \mathbf{W} \mathbf{L} \mathbf{s} = \tilde{\mathbf{f}} \mathbf{s} \quad (16)$$

$$\tilde{\mathbf{f}} = \mathbf{f}^T \mathbf{W} \mathbf{L}. \quad (17)$$

From Eqs. (16) and (17) it is obvious that the effect of  $w(r)$  can be interpreted as an effective line broadening of the experimental<sup>4</sup> spectrum. The use of the overlap function  $w(r)$  results in a smoother error-landscape which allows an easier optimization of the genetic algorithm process. It should be noted that the simple least squares fitness function where all spectral points have the same weight is also described by  $F_{fg}$  for the limiting case that the width of  $w(r)$  is zero and  $\mathbf{W}$  becomes the identity matrix.

The transformed experimental spectrum from Eq. (17) has to be evaluated only once. Formally the sum on the right hand side of Eq. (16) runs over all  $N$  points of the spectrum. However the stick spectrum array is a very sparse one: Typically  $N$  is of the order of 60,000-250,000, while the number of sticks (nonzero values of  $s_k$ ) is only about 1000-3000. Therefore the use of Eq. (16) strongly reduces the necessary computing time. The reduction of computing time is actually more dramatic since the double sum in the numerator of Eq. (8) over  $N$  points is reduced to a single sum over a sparse array in Eq. (16). Furthermore the theoretical spectrum itself  $\{g_i\}$  does not have to be calculated anymore. The first term in the denominator of Eq. (8) also has to be evaluated only once. The second term in this denominator has to be calculated every time a value of  $F_{fg}$  is needed in the process of the genetic algorithm. Fortunately this term can also be expressed in terms of the stick spectrum  $\mathbf{s}(\{s_k\})$ :

$$\|\mathbf{g}\|^2 = \mathbf{s}^\dagger (\mathbf{L}^\dagger \mathbf{W} \mathbf{L}) \mathbf{s} \quad (18)$$

Since  $(\mathbf{L}^\dagger \mathbf{W} \mathbf{L})$  is a banded matrix the evaluation of  $\|\mathbf{g}\|^2$  from Eq. (18) is in practice almost linear in the number of sticks rather than quadratic. Again  $(\mathbf{L}^\dagger \mathbf{W} \mathbf{L})$  has to be evaluated only once. It turned out that the effect of the above discussed modifications of the calculation of  $F_{fg}$  was that its calculation time became negligible to the total computing time.

The use of the stick spectrum, described in this section, is limited to applications in which the lineshape parameters, like Gaussian or Lorentzian width in the Voigt profile, don't have to be fit. Inclusion of the linewidth parameters in the fit requires the reevaluation of Eq. (16) during the GA process, which in practice dramatically increases the computing time. Therefore a linewidth fit should be performed after a determination of all other parameters in a separate fitting procedure. An example for this is given in [2].

#### 4. Effect of the width ( $\Delta w$ ) of the overlap function $w(r)$ on the convergence of the GA

The relative broadening of the spectra, introduced by the overlap function  $w(r)$ , described in section II B 3, critically determines the ability of the GA to converge to the global minimum and also the speed of convergence. The smoothing of the error landscape allows to sense regions far from the minimum. In the first step, the overlap function  $w(r)$  should be chosen relatively broad;  $\Delta w \approx 15 - 20$  times the line widths of an individual rovibronic line in the spectrum ( $\Delta_{lw}$ ). In this way, a first set of molecular parameters is obtained, which still has to be refined. This is done by decreasing  $\Delta w$  and narrowing the limits of the parameter space to be searched in the fit. Decreasing  $\Delta w$  improves the accuracy of the molecular parameters obtained from the fit, while narrowing the parameter space leads to an improved sampling in the region of the minimum. This of course is the critical step in the procedure. Too strong narrowing of the parameter space leads to premature convergence of the fit – with a high probability into a local minimum.

#### C. Selective weight of parts of the spectrum.

In most cases each point (frequency region) in the spectrum has the same weight. However there are cases for which it is necessary that certain parts of the spectrum have a higher weight relative to other parts. For example if one needs to compensate for the Q factor of the NMR probe.

The program is able to apply flexible weights to different parts of the spectra. This is realized with a weight function. For certain parts of the spectrum it is possible to increase or reduce their weight in the cost function evaluation. This is done through a weight function. The first set of functions are of the type:

$$W = 1 - \sum_{i=1}^n A^{(i)} W_{type}^{(i)}(\Delta_w^{(i)}, (\nu - \nu_0^{(i)})) \quad (19)$$

The functions  $W_{type}(\Delta_w, \Delta\nu)$  that are supported at the

<sup>4</sup> A different arrangement for the equations shows that  $w(r)$  also can be interpreted as an effective broadening of the calculated spectrum  $g$ . Actually the experimental and the calculated spectra are broadened relative to each other.



moment are: gaussian, lorentzian, block and triangle. Here  $\Delta_w = \Delta_w^{(i)}$  and  $\Delta\nu = (\nu - \nu_0^{(i)})$  For a gaussian-type function

$$W_G(\Delta_w, \Delta\nu) = \exp[-4\ln(2)(\Delta\nu/\Delta_w)^2], \quad (20)$$

for a lorentzian-type function

$$W_L(\Delta_w, \Delta\nu) = \frac{1}{(\Delta\nu/\frac{1}{2}\Delta_w)^2 + 1}, \quad (21)$$

for a triangle

$$W_T(\Delta_w, \Delta\nu) = \begin{cases} 1 - |\Delta\nu|/(\Delta_w) & \text{for } |\Delta\nu| \leq \Delta_w \\ 0 & \text{otherwise,} \end{cases} \quad (22)$$

and for a block

$$W_B(\Delta_w, \Delta\nu) = \begin{cases} 1 & \text{for } |\Delta\nu| \leq \frac{1}{2}\Delta_w \\ 0 & \text{otherwise.} \end{cases} \quad (23)$$

So  $\Delta_w$  is the full width at half height of the weightfunction.

The control goes through the keyword *weightfunction*:

$$\text{weightfunction} = W_{\text{type}} \nu_L \nu_H \text{Ampl Width Center} \quad (24)$$

In this formula  $W_{\text{type}}$  defines one of the possible functional forms defined above,  $\nu_L$  and  $\nu_H$  are the lowest and highest frequency-value for which the weight function will be applied, respectively. *Ampl* is the amplitude  $A$ , *Width* the width ( $\Delta_w$ ) and *Center* the center ( $\nu_0$ ) of the  $W_{\text{type}}$  function, respectively.

Alternative weightfunctions in terms of a polynomial function can be used:

$$W_P = \sum_{m=0}^M a_m \left( \frac{\nu - \nu_0}{\nu_{\max} - \nu_{\min}} \right)^m \quad (25)$$

Here  $\nu_0$ ,  $\nu_{\max}$  and  $\nu_{\min}$  are the chosen origin, the maximum and minimum frequency in the spectrum, respectively. The control in this case goes through the keyword *weightfunction* with parameters:

$$\text{weightfunction} = \text{Polynomial } \nu_L \nu_H a_0 \dots a_M \text{Origin} \quad (26)$$

$\nu_L$  and  $\nu_H$  are defined as above. The last parameter is the chosen origin ( $\nu_0$ ) for the polynomial. The order of the polynomial is determined by how many values for  $a_M$  are entered (minimal 1).

It is possible to supply more than one keyword *weightfunction*; in fact as much as needed for the problem. The effects of the various weightfunctions simply adds up. For the first group they add up as in Eq. (19). The first group of weightfunctions can be combined with the polynomial functions of Eq. (25). It is also allowed to have more polynomials, for example with different origin. The final weightfunction has to be positive definite. The program checks this and stops if the weightfunction is found not to be positive definite.

As a default the weightfunction is applied to both the experimental and calculated spectrum. An additional option that is controlled by the keyword *apply Weight-*

*functions to Exp spectrum*, determines if the weightfunction is indeed applied to the input experimental spectrum. In some cases, like for example a NMR spectrum observed without corrections due to the Q-factor of the NMR cavity, will have no weightfunction for the experimental spectrum, but a (Lorentzian type of) weight function for the calculated spectrum.

#### D. Remove smoothened background.

A smoothened spectrum can be removed (simultaneously) from the experimental and calculated spectra. The idea behind this option is that a removal of a smoothened spectrum enforces the features in the spectrum which are on top of a *background*. So in the global fit the local features are strengthened. The smoothened spectrum  $\mathbf{f}^S = (f_1^S, f_2^S, \dots, f_N^S)^T$  is defined as:

$$f_i^S = \frac{1}{n_s + 1} \sum_{j=i-n_s/2}^{i+n_s/2} f_j \quad (27)$$

Here  $n_s$  is the number of points over which the smoothening should be performed. Since there is a final number of points in the spectrum the beginning and end parts of the sum should be treated slightly differently: The smoothening at the beginning and end is performed in such a way that discontinuities are avoided. The smoothening over the the calculated spectrum  $\mathbf{g}$  is carried out identically as for  $\mathbf{f}$  in Eq. (27), resulting in  $\mathbf{g}^S = (g_1^S, g_2^S, \dots, g_N^S)^T$ :

$$g_i^S = \frac{1}{n_s + 1} \sum_{j=i-n_s/2}^{i+n_s/2} g_j \quad (28)$$

The actual spectrum which is fitted if this option is activated is  $\mathbf{f}^R = \mathbf{f} - \mathbf{f}^S$  against  $\mathbf{g}^R = \mathbf{g} - \mathbf{g}^S$ . I.e. in Eq. (IIB1)  $\mathbf{f}$  and  $\mathbf{g}$  are replaced by  $\mathbf{f}^R$  and  $\mathbf{g}^R$ , respectively.

This option is controlled with the keyword *remove smoothened background* (Table II). If its value  $> 0$ , it defines the number of points ( $n_s$ ) over which the smoothing takes place. The choice of this value depends on the characteristics of the spectrum.

### III. THEORY

#### A. Solid State NMR

##### 1. The nuclear quadrupole interaction

Atomic nuclei that have total angular momentum  $\geq 1$ , possess a none-spherical charge distribution and consequently an electric quadrupole moment [4]. The Hamiltonian, in units of angular frequency, for the interaction of this moment of nuclear charge with the *efg* of the environment is given by [9, 23]

$$\hat{H}_Q = \sum_i \frac{2\pi C_q^i}{2I_i(2I_i - 1)} \hat{I}_i \cdot \tilde{\mathbf{V}}^i \cdot \hat{I}_i. \quad (29)$$

Here  $i$  is the index of the quadrupole nuclei and  $\hat{I}_i$  and  $I_i$  the nuclear angular momentum vector operator and its quantum number. Furthermore, the reduced  $efg$  tensor  $\tilde{\mathbf{V}}^i$  equals  $\mathbf{V}^i$  divided by its largest principal component, by definition,  $V_{ZZ}$ .  $C_q^i$  is the quadrupole coupling constant. The general form of  $\mathbf{V}$  and  $C_q$  is

$$C_q = \frac{eQV_{ZZ}}{h} \quad \text{and} \quad V_{\alpha,\beta} = \left. \frac{\partial^2 V}{\partial r_\alpha \partial r_\beta} \right|_{r=0}, \quad (30)$$

with  $V$  the electric potential at the position of the nucleus,  $r_\alpha$  its coordinates for  $\alpha=x, y, z$  and  $eQ$  the nuclear quadrupole moment. From the form of equations 29 and 30, it follows that  $C_q$  determines the strength of the coupling and additionally provides limits for the position of transition frequencies and, for powdered materials, the width of the spectral lines. For illustrative purposes we rewrite equation 29 for the case of a single quadrupole nucleus in the principal axis frame (PAF) of the  $efg$  tensor [23]

$$\hat{H}_Q^{PAF} = \frac{2\pi C_q}{4I(2I - 1)} \left\{ 3\hat{I}_Z^2 - \hat{I}^2 + \eta(\hat{I}_X^2 - \hat{I}_Y^2) \right\}. \quad (31)$$

Here the asymmetry parameter  $\eta$  is defined as

$$\eta = \frac{V_{XX} - V_{YY}}{V_{ZZ}}. \quad (32)$$

It provides symmetry information of the  $efg$  at the nucleus. The asymmetry parameter can assume values between zero and one, adhering the convention  $|V_{ZZ}| \geq |V_{YY}| \geq |V_{XX}|$ . Combined with the requirement for a traceless  $efg$  tensor, corresponding to a nucleus in equilibrium, the quadrupole interaction is fully described with  $C_q$  and  $\eta$  as parameters.

The dominating Zeeman interaction in NMR makes the form of equations 29 and 31 impractical. Advanced theory for a convenient perturbation treatment of the quadrupole interaction under magic angle spinning and radio-frequency pulse conditions is extensively treated in literature [8, 9, 23]. To conclude, it should be noted that the explicit angular dependence of the equations in this section is deliberately not shown. In simulations, the response of the powder is assumed equal to the sum of all crystallite responses. This sum will therefore include all crystallite orientations present, leaving only  $C_q$  and  $\eta$  to parameterise the induction decay/spectrum.

## 2. (Extended) Czjzek distribution

One of the subtleties of the use of quadrupole NMR to study disordered materials, is the experimental spectrum effectively ‘summarises’ all local structure variations felt by the type of nucleus under study. The chal-

lenge in the analysis is thus to find an inverse mapping of experimental data in terms of structural information, which probably all cases is non-linear. By starting from the notion that structural variations translate into a distribution of the  $efg$ , a more accessible step is to find the form of this distribution.

Czjzek *et al* [11] investigated the  $efg$  distribution in the case of structural disorder by computer simulations. Tensor components of the  $efg$  at the center of randomly repositioned charges, become normally distributed in the limit towards infinite samplings. The resulting probability distribution, as a function of  $efg$  tensor components, is therefore a multivariate Gaussian. This distribution can be rewritten as a probability density function of quadrupole interaction parameters  $C_q$  and  $\eta$ , which has the closed-form

$$f(C_q, \eta) = \frac{C_q^{d-1} \eta}{\sqrt{2\pi} \sigma^d} \left(1 - \frac{\eta^2}{9}\right) \exp \left\{ -\frac{C_q^2}{2\sigma^2} \left(1 + \frac{\eta^2}{3}\right) \right\}. \quad (33)$$

Here  $\sigma$  is the standard deviation of the multivariate Gaussian probability distribution of the  $efg$  tensor components and  $d$  we introduced to correspond to the number of independent tensor components [5]. Normalisation of the distribution integral to one, provides the probability of finding a given set of  $C_q$  and  $\eta$ . An illustration of the distribution shape is given in Figure 5. Later the Czjzek model was revisited and extended by Le Caër and Brand [12, 13]. They introduced an ordered (not distributed) contribution to the  $efg$ . In the *EGdeconv* program we implemented the most general form of the resulting probability density function, given by

$$\begin{aligned} f(C_q, \eta) \propto & \frac{C_q^{d-1}}{\sigma^d} \eta \left(1 - \frac{\eta^2}{9}\right) \iiint d\alpha \sin(\beta) d\beta d\gamma \\ & \times \exp \left\{ -\frac{1}{2\sigma^2} \left[ C_{q,o}^2 \left(1 + \frac{\eta_o^2}{3}\right) + C_q^2 \left(1 + \frac{\eta^2}{3}\right) \right. \right. \\ & \left. \left. - \frac{2}{\sqrt{3}} C_q C_{q,o} \left( \sqrt{3} a_{11} + \eta_o a_{15} + \eta \left( a_{51} + \frac{\eta_o a_{55}}{\sqrt{3}} \right) \right) \right] \right\}. \quad (34) \end{aligned}$$

Here  $C_{q,o}$  and  $\eta_o$  are quadrupole parameters for the local order contribution. An illustration of the distribution shape is given in Figure 6. The elements  $a_{ij}$  contain the angular dependence of the  $efg$  tensor components, as defined in [12] as

$$\begin{aligned} a_{11} &= \frac{1}{2} (3 \cos^2 \beta - 1) \\ a_{15} &= \frac{\sqrt{3}}{2} \sin^2 \beta \cos 2\alpha \\ a_{51} &= \frac{\sqrt{3}}{2} \sin^2 \beta \cos 2\gamma \\ a_{55} &= \frac{1}{2} \cos 2\alpha (1 + \cos^2 \beta) \cos 2\gamma \\ &\quad - \sin 2\alpha \cos \beta \sin 2\gamma. \end{aligned} \quad (35)$$

Integration over all angles leaves a distribution as function of quadrupole interaction parameters that encompasses short-range order and long-range disorder.

In the special case of  $\eta_o = 0$ , equation 34 can be simplified to [12]

$$f(C_q, \eta) \propto \frac{C_q^{d-1}}{\sigma^d} \eta \left(1 - \frac{\eta^2}{9}\right) \times \exp \left\{ -\frac{C_{q,o}^2 + C_q^2 \left(1 + \frac{\eta^2}{3}\right)}{2\sigma^2} \right\} \times \int_0^1 I_0(z) \exp \left\{ \frac{C_q C_{q,o}}{2\sigma^2} (3t^2 - 1) \right\} dt, \quad (36)$$

with  $I_0(z)$  a modified Bessel function, and

$$z = \eta \left| \frac{C_q C_{q,o}}{2\sigma^2} \right| (1 - t^2). \quad (37)$$

It should be noted that we omitted the  $x$  parameter shown in Le Caër's papers, by setting it to a value of 0.5. This means that the relative contribution of local and longrange order and disorder is equal, leaving  $C_{q,o}$  and  $\sigma$  to vary.

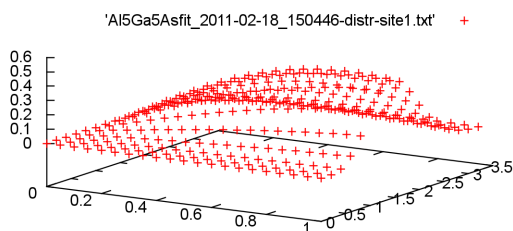


FIG. 5: Czjzek distribution shape for  $\sigma = 0.82$  MHz and  $d=5$ . The axis in the front presents the values of  $\eta$ , the axis on the right are  $C_q$  values. The image is a direct copy of the image shown by gnuplot using the command `splot 'Al5Ga5Asfit.2011-02-18.150446-distr-site1.txt'` for the output file of `EGdeconv`.

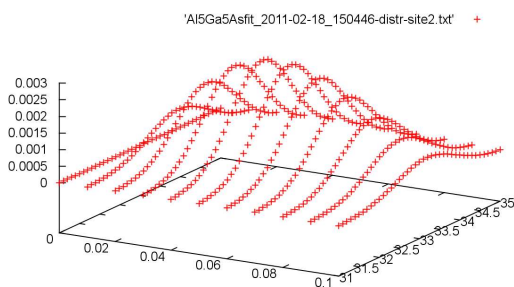


FIG. 6: Extended Czjzek distribution shape for  $\sigma = 0.75$  MHz,  $d=5$ ,  $\eta_o=0.04$  and  $C_{q,o}=33.2$  MHz. The axis in the front presents the values of  $\eta$ , the axis on the right are  $C_q$  values. The image is a direct copy of the image shown by gnuplot using the command `splot 'Al5Ga5Asfit.2011-02-18.150446-distr-site2.txt'` for the output file of `EGdeconv`.

### 3. The order parameter model

Based on theory used to analyze diffraction data, an order parameter  $S$  can be defined to quantify the degree

of long-range order related to the fractional occupancy of lattice sites by their preferred atom [24].  $S = 0$  represents a completely random structure and  $S = 1$  corresponds to a completely ordered lattice:

$$S = |r_A + r_B - 1|, \quad (38)$$

where  $r_A$  and  $r_B$  are the fractions of A and B lattice sites that are occupied by their preferred atom. For binary systems, as the only situation considered in the program yet, this means that  $1 - r_A$  constitutes the fraction of A lattice sites occupied by B atoms and  $1 - r_B$  the fraction of B sites occupied by A atoms. In case of random occupation of the lattice sites, *i.e.* the absence of ordering,  $r_A = x$  and  $r_B = 1 - x$ . For (partially) ordered structures the fractional occupancy of the A and B sites by their preferred atoms can be re-written as:

$$\begin{aligned} r_A &= x + S/2 \\ r_B &= (1 - x) + S/2 \end{aligned} \quad (39)$$

Using this definition it should be realized that the order parameter  $S$  can reach a value of  $2x$  at most. A perfectly ordered structure can only be achieved for  $x = 0.5$ .  $S$  can be scaled, however, so that the maximum possible order for any given composition  $x$  gives an order parameter  $S = 1$  [25].

The presence of (dis-)order in a structure is in most cases manifested in quantifiable NMR parameters. Different structural environments in a material can be distinguished based on the NMR interactions which are influenced by the different coordinating atoms. For example, Tycko *et al.* [14] distinguished five  $^{31}\text{P}$  sites in the  $^{31}\text{P}$  MAS NMR spectrum of GaInP. These five resonances were assigned to the different phosphorus nearest neighbors coordinations  $\text{P}[\text{In}_{4-n}\text{Ga}_n]$  with  $n = 0..4$ . The sites are resolved in this spectrum due to their different isotropic chemical shifts. The presence of order in the sample is reflected in the statistics of the occurrence of the various coordinations. [14, 26] The probabilities  $p_n$  for the occupations of the  $\text{C}[\text{A}_n\text{B}_{4-n}]$  sites are

$$\begin{aligned} p_0 &= p_{\text{C}[\text{B}_4]} \\ p_1 &= p_{\text{C}[\text{A}_1\text{B}_3]} \\ p_2 &= p_{\text{C}[\text{A}_2\text{B}_2]} \\ p_3 &= p_{\text{C}[\text{A}_3\text{B}_1]} \\ p_4 &= p_{\text{C}[\text{A}_4]} \end{aligned} \quad (40)$$

Here, two types of ordering are considered for  $\text{A}_x\text{B}_{1-x}\text{C}$  type materials, see Fig. 7 for an example of  $\text{Al}_x\text{Ga}_{1-x}\text{As}$ . CuPt ordering, with diagonal planes, stacked in the  $\langle 111 \rangle$  direction or a CuAu, with alternating planes in  $\langle 001 \rangle$ . For each configuration, we can quantify the probability of their occurrence. The 16 possible atomic configurations in the first shell contribute each with a different probability. For the

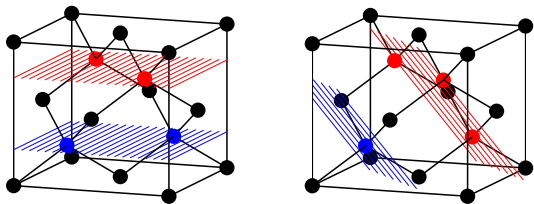


FIG. 7: (Picture originally from [5]) Order in the face centered cubic lattice structure of AlGaAs. The black atoms are Arsenic, the other atoms are the Al and Ga cations. *left* 2-fold symmetry of the As nucleus for a  $\text{AsAl}_2\text{Ga}_2$  configuration, CuAu ordered with planes alternating along the  $\langle 001 \rangle$  direction. *right* 3-fold tetrahedral symmetry of the As nucleus for a  $\text{AsAl}_3\text{Ga}$  and  $\text{AsAlGa}_3$  configuration, also called CuPt ordered lattice with planes extending along the  $\langle 111 \rangle$  direction.

Copper-Gold ordering this results in

$$\begin{aligned}
 p_0^{\text{CuAu}} &= (1 - r_A)^2 r_B^2 \\
 p_1^{\text{CuAu}} &= 2r_A(1 - r_A)r_B^2 + 2(1 - r_B)(1 - r_A)^2 r_B \\
 p_2^{\text{CuAu}} &= r_A^2 r_B^2 + (1 - r_B)^2 (1 - r_A)^2 + \\
 &\quad 4r_A(1 - r_B)r_B(1 - r_A) \\
 p_3^{\text{CuAu}} &= 2r_A^2(1 - r_B)r_B + 2r_A(1 - r_B)^2(1 - r_A) \\
 p_4^{\text{CuAu}} &= (1 - r_B)^2 r_A^2,
 \end{aligned} \tag{41}$$

and for the Copper-Platinum ordering

$$\begin{aligned}
 2p_0^{\text{CuPt}} &= r_B(1 - r_A)^3 + (1 - r_A)r_B^3 \\
 2p_1^{\text{CuPt}} &= (1 - r_B)(1 - r_A)^3 + r_A r_B^3 + \\
 &\quad 3r_A r_B(1 - r_A)^2 + 3(1 - r_B)(1 - r_A)r_B^2 \\
 2p_2^{\text{CuPt}} &= 3(1 - r_B)r_A(1 - r_A)^2 + 3r_A(1 - r_B)r_B^2 + \\
 &\quad 3(1 - r_B)^2 r_B(1 - r_A) + 3r_A^2(1 - r_A)r_B \\
 2p_3^{\text{CuPt}} &= r_A^3 r_B + (1 - r_B)^3(1 - r_A) + \\
 &\quad 3r_A^2(1 - r_B)(1 - r_A) + 3(1 - r_B)^2 r_A r_B \\
 2p_4^{\text{CuPt}} &= (1 - r_B)r_A^3 + r_A(1 - r_B)^3.
 \end{aligned} \tag{42}$$

#### 4. Composing a theoretical Solid-State NMR spectrum

This section covers the practical aspects of *EGdeconv*. Introduction of input for *EGdeconv* adheres the convention that  $\langle \dots \rangle$  input is required and  $[\dots]$  is optional. The words, units and abbreviations replacing the  $\dots$  help to interpret the meaning of the variable as to what should be filled in.

##### a. Introduction

A theoretical spectrum is the sum of subspectra of which each corresponds to a parameter set in the input file and hence a specific chemical site. The program flow to calculate a subspectrum is illustrated in Fig. 8. The figure illustrates in what order the selected library files are processed, and where the various fitting parameters are used. First step is summation of the files

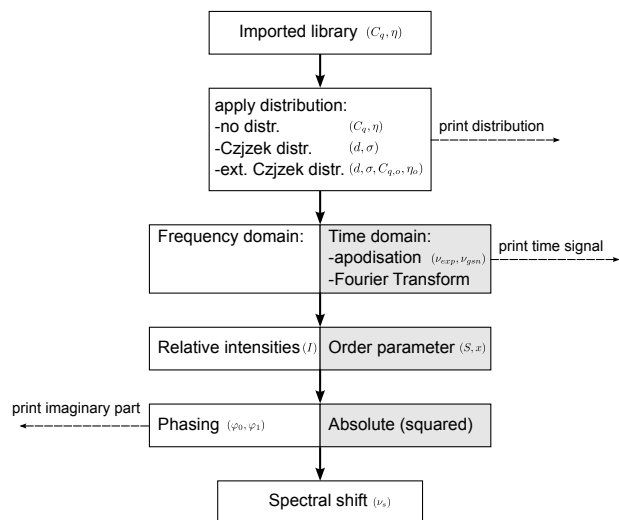


FIG. 8: Program flow of *EGdeconv* for the calculation of a subspectrum. The grey-white blocks indicate the choices the user has in its synthesis. For each step, parameters that can be included in the fit are indicated. The definition of part of the parameters ( $d$ ,  $\sigma$ ,  $\eta(o)$ ,  $C_{q,(o)}$ ,  $x$  and  $S$ ) can be found in the Theory section. The exponential/Gaussian apodisation ( $\nu_{exp}/\nu_{gsn}$ ) and zeroth and first order phasing ( $\varphi_0, \varphi_1$ ) are defined in the manual, leaving only the trivial parameters chemical shift ( $\nu_s$ ) and intensity ( $I$ ).

according to the distribution model chosen. Then depending on the type of data, time or frequency domain, the result requires additional processing in the form of apodisation and a Fourier transform. Subsequently the intensity of the subspectrum is determined either freely or according to an order parameter model. It is possible to apply a zeroth and first order phase correction to the subspectrum or to use the absolute or absolute squared value of the spectrum. The program flow is ended by shifting the subspectrum. Note that the number of fit parameters per subspectrum can vary between one and effectively ten at maximum.

Fig. 8 also illustrates that *EGdeconv* can print extra output. This feature provides the user with more insight in what happens during the different stages in the calculation. The extra output will only be generated for the final fit result.

The rest of this section will treat each of the above mentioned steps in depth.

##### b. Libraries

As figure 8 indicates, the theoretical powder spectra are calculated using a library of precalculated frequency or time domain data. The reason for the library approach is to circumvent large computational costs for the evaluation of the deconvolution model. Till now the *EGdeconv* program is only capable of handling output generated by the NMR simulation program SIMPSON [8, 27].

The library files are a function of  $C_q$  and  $\eta$ , as defined in section III A 1. The program requires at least one value in both parameter dimensions and a constant stepsize if multiple values are provided.

Additionally a constant number of elements is required in both dimensions, in other words, the grid should be rectangular. The filenames have the following format

<sitename>-< $C_q$ (MHz)>-< $\eta$ >.<file extension> ,

<sitename> can be an arbitrary string containing letters and numbers.  $C_q$  needs to be: larger than or equal to zero, in units of MHz and all numbers need to have an equal length. An example of the latter, if *Al4As-10.00-0.00.fid* is available in the library, smaller values of  $C_q$  need to be padded with zeros on the left, e.g. *Al4As-01.00-0.00.fid*. This requirement is related to a sorting step that takes place during the reading of the library. Finally the values of  $\eta$  should be in the domain zero to one and the <file extension> can be any string containing letters and numbers. After *EGdeconv* reads the library filenames based on <file extension>, the file header of the first file, in alphabetical order, will be used to determine: the data type (time or frequency domain), the number of points and where the data starts. Currently the data header should be in the format used by SIMPSON, an example

```
SIMP
NP=8192
SW=2.7
TYPE=FID
DATA
```

For your own reference, a complete summary of everything that has been read into the computer memory, is given in *treefile.txt*, regulated by the keyword *print treefile* (section V C). Any change in <sitename> requires a new library directory.

In the inputfile, libraries are included by using the keyword 'speciesX', also see section V C. To save internal memory or to speed up the calculation it is possible to include only part of the library and/or part of the points in the files, see definition of keyword 'speciesX' and 'speciesX points'. It should be noted that in the case of Hahn echoes the keyword 'echo timing' is important, this will override the keyword 'speciesX points'. It is perfectly fine to use the same library for different *species*. For example, if a large library is available and it is known in advance that two sites need  $C_q$  values that are far apart, two *species* can be assigned using the same library but with different ranges in  $C_q$ .

### c. Distributions

*EGdeconv* is currently capable of applying either no or an (extended) Czjzek distribution, see section III A 2. The type of distribution is regulated by the name of parameter number 198, see section V. Depending on the choice, parameter number one and two ( $\eta$  and  $C_q$  or  $d$  and  $\sigma$ ) or one to four ( $d$ ,  $\sigma$ ,  $C_{q,o}$  and  $\eta_o$ ) are available for fitting.

An important issue when applying a distribution is the choice of an appropriate grid in terms of  $\eta$  and  $C_q$ . For powder spectra this grid is equal to the library grid, see also keyword 'speciesX'. In the case of a single crystal spectrum the ( $C_q, \eta$ ) constraints of keyword

'speciesX' are mandatory input to fix the grid. In particular the value of the  $\sigma$  parameter should be chosen with care. The calculation of the integral over all angles in equation 34 is sensitive to the value of this parameter in combination with the library grid. For example, if the grid stepsize is relatively large in comparison to the value of  $\sigma$ , or the value of  $\sigma$  is so small there is hardly a distribution this may result in a non-converging integration.

*EGdeconv* offers an easy way to visualise the distribution via the keyword 'print distributions'. This option is enabled by default and returns a three-column ( $\eta, C_q, f(\eta, C_q)$ ) ascii file per site, containing the unnormalised Czjzek distribution of the calculated test or final spectrum. A useful program to quickly plot these files is *gnuplot*, see the captions of figures 5 and 6 for the appropriate command.

### d. Apodisation

In case simulated time domain data is supplied to *EGdeconv*, it is possible to fit linewidth. The program allows to apodise the induction decay with either or both an exponential and Gaussian function. This will result in respectively a: Lorentzian, Gaussian or Voigt lineshape. Definitions used for this apodisation are

$$\begin{aligned} f_{exp}(t) &= f(t) \exp\{-\pi\nu_L t\} \\ f_{gauss}(t) &= f(t) \exp\left\{-\frac{\pi^2}{4\ln 2} \nu_G^2 t^2\right\}, \end{aligned} \quad (43)$$

both  $\nu_L$  and  $\nu_G$  frequencies are Full Width Half Height of the resulting spectral lines and have the same units as the frequency axis of the experimental spectrum. Peakwidths from the experimental spectrum can therefore be directly used in the input file for parameter 22 ( $\nu_L$ ) and 23 ( $\nu_G$ ), see section V. In case of Hahn echoes the apodisation will be applied to both the ascending and descending slope, see section III A 4 h for more information.

We conclude with a word of caution that fitting linewidths together with other parameters may easily lead to unsatisfactory fits. This is mainly caused by the fact that most other parameters influence the linewidth as well. The user should at first try to fix the applied linewidth/apodisation at a appropriate value with respect to experimental parameters, e.g. acquisition time which puts an obvious constraint on the spectral linewidths.

### e. Subspectrum intensity or order parameter

If multiple sites are present in the experimental spectrum, it is of interest to quantify their (relative) abundance. *EGdeconv* offers two approaches to obtain this information. The first approach is by allowing every site to have its own independent (relative) intensity<sup>5</sup>. The second approach is by using a model to relate abun-

<sup>5</sup> using the coupling parameter 'R' reduces the number of parameters to fit by one, see section V

dances according to structural considerations. The currently available model is that of the order parameter introduced in section III A 3.

It should be noted that the significance of the fitted intensities strongly depends on the accuracy of the simulated data with respect to the experimental conditions, such as excitation efficiency. Each subspectrum that is generated is multiplied by the intensity factor before adding it to the subspectra of the other sites. Note that if a quadrupole parameter distribution is used, before multiplication with the intensity factor, the subspectrum is weighted such that the integral over the applied distribution is one. An intensity of zero has a special meaning, since it will skip the calculation of the subspectrum reducing computation time.

If an order parameter model is used, the output file will provide extra information by displaying relative abundances of each site with respect to the first site/parameter set. These abundances will be given as the final values of the intensity parameter, currently number 20, for the remaining sites, since in the first parameter set this parameter is reserved for the final value of  $S$ . Any settings of parameter 20 in the input file, in parameter sets other than the first are ignored.

#### f. Spectral phasing or absolute value (squared)

For simulations of for example broadband excitations, it is possible that the theoretical spectra are not phased fully in absorption. Although it is difficult to compensate, EGdeconv does offer two ways to manage this problem. The first approach is a zeroth and first order phase correction according to

$$F_{ph}(\nu) = F(\nu) \exp \{-i(\varphi_0 + \varphi_1(\nu - \nu_0))\} \quad (44)$$

here  $\varphi_0$ ,  $\varphi_1$  and  $\nu_0$  correspond to parameters 23 through 25 in section V. Note, that the choice of value for parameter  $\nu_0$  is independent of the spectral shift, because the latter is applied in subsequent step as shown in figure 8, see also section III A 4 g.

A second approach is to use the absolute (squared) value of the theoretical subspectrum. Make sure to also apply this to the experimental data before using EGdeconv. The keyword 'fit type', see section V C, is used to switch to the appropriate representation.

#### g. Spectral shift

In most cases, it is not necessary to take into account chemical shifts during simulation of the library data. The chemical shift can be fitted with parameter 21, see section V. Note that subspectra are consecutively phased and then shifted, this influences the choice for the  $\nu_0$  parameter mentioned in section III A 4 f.

#### h. Treatment of Hahn echoes

Figure 9 illustrates how Hahn echoes are processed. The position of the maximum intensity of the first echo is indicated with the keyword 'echo timing', which in decimal notation indicates real time and in integer notation means index of the complex datapoint, see section

V C. The two symmetric parts of the echo are switched and the resulting time data is zero filled in the centre to match the amount of points of the experimental data. The remaining part of the simulated signal is not used.

Furthermore, the EGdeconv program will check by the use of the 'echo timing' value if the whole echo is present in the libraries files. If not, only half the echo will be used for processing. The value of the keyword 'speciesX points' is taken into account to determine if the whole echo is present.

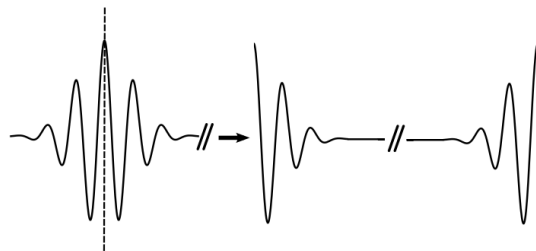


FIG. 9: Procedure to process Hahn echoes. The resulting signal starts with the second half of the echo and finishes with the first half. In the final step the signal is zero filled between the halves to match the amount of experimental data points. Apodisation is applied to the ascending and descending slope.

#### i. Extra NMR related output

As figure 8 suggests, it is possible to generate extra output during several steps of the calculations. First of all a file which summarises what part(s) of the library are read into memory can be generated using the keyword 'print treefile'. This file is printed before an actual calculation of a theoretical spectrum commences. After a regular EGdeconv run the real part or absolute (squared) value of the fit will be given by default. See keyword 'fit type' in section V C to print all complex time and frequency domain data. This output will contain the individual and the sum of the subspectra/sub-induction decays. Furthermore it is possible to output quadrupole parameter distributions per site with the keyword 'print distributions'.

#### j. Checks and error messages

EGdeconv is equipped with an extensive, however, not exhaustive amount of input checks. Caution is especially required when parameter couplings are used, see section IV B. Errors are either aborting the program with an error message or reported in a MESSAGES file, see section VI D. Errors in the MESSAGES file are compensated by EGdeconv according to the printed statements.

If the program stops without an immediately self-explanatory error, there are some basic checks that can be done. In case of a powder spectrum, the *treefile* can be checked to see if the library is read correctly, see also keyword 'print treefile' in section V C. If not, there is a problem with the speciesX keyword, e.g. the library

path, the library filename format or internal file format are incorrect.

If all of the above is correct, the error appears most likely in the input check routine. This is easily verified by looking for an empty MESSAGES file in the log directory. If this is the case, double check the parameter values, ranges and couplings in the EGdeconv inputfile.

If all the above fails, it possible that something during the calculation goes wrong. Figure 8 can be used to devise checks, using extra printouts, to find the module that fails. For example, printing distribution files is a reliable way to see if the grid size for distributions is chosen correctly.

### 5. Introduction to Multiple Quantum Magic Angle Spinning

In this section the Multiple Quantum Magic Angle Spinning experiment is introduced; for a more thorough explanation the reader is referred to [28]. The quadrupole coupling, as described in section III A 1, is a strong interaction. The quadrupole coupling constant  $C_q$  is usually in the order of 0 to 30 MHz. The first order effects of the quadrupole interaction will give rise to a very broad powder pattern. This makes measurements on quadrupole nuclei with sufficient signal to noise potentially difficult. The central transition of half-integer quadrupole nuclei is, however, not broadened to first order. It is therefore easier to measure. Because the quadrupole interaction is strong with respect to the Zeeman interaction, the second order effects should be accounted for. This still gives rise to a quadrupole line-shape of several kHz.

To resolve these broadened spectra Frydman and Harwood [29] introduced the Multiple Quantum Magic Angle Spinning experiment. The pulse sequence for this experiment is shown in Fig. 10. This is a 2D ex-

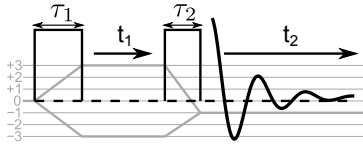


FIG. 10: Schematic overview of the MQMAS experiment and its coherence pathways. The first pulse excites a symmetric MQ coherence, which is allowed to evolve during  $t_1$ . The second pulse then converts the MQ coherence to a detectable  $|1/2\rangle \leftrightarrow |-1/2\rangle$  coherence, which is acquired during  $t_2$ . A phase cycle is used to select the coherence pathway shown at the bottom of the figure.

periment designed to obtain an isotropic spectrum of a half integer quadrupole nucleus. An example MQMAS spectrum is presented in Fig. 11. For a sample spinning infinitely fast at the magic angle the coherence frequen-

cies during  $t_1$  and  $t_2$  are given by [28]

$$\begin{aligned}\nu_1^{-m \rightarrow m} &= 2m\nu_{iso} + C_0^{I,m} \cdot \nu_{QIS} \\ &+ C_4^{I,m} \cdot P_4(\cos(\Theta_M)) \cdot \nu_4(\alpha, \beta) \\ \nu_2^{-\frac{1}{2} \rightarrow \frac{1}{2}} &= \nu_{iso} + C_0^{I, \frac{1}{2}} \cdot \nu_{QIS} \\ &+ C_4^{I, \frac{1}{2}} \cdot P_4(\cos(\Theta_M)) \cdot \nu_4(\alpha, \beta)\end{aligned}\quad (45)$$

here  $I$  is the spin quantum number,  $m$  is the magnetic quantum number,  $\nu_{iso}$  is the isotropic chemical shift frequency,  $\nu_{QIS}$  is the isotropic Quadrupole Induced Shift frequency,  $\nu_4(\alpha, \beta)$  is the angular dependent part of the quadrupole frequency,  $P_4(\cos(\Theta_M))$  is the MAS scaling factor. The frequencies are determined by the quadrupole interaction parameters  $C_q$ ,  $\eta$  and the orientation of the crystallite given by the Euler angles  $\alpha$  and  $\beta$ , where  $\alpha$  is the azimuthal angle and  $\beta$  the polar angle with respect to the MAS rotor frame, see paragraph III A 6 d.  $C_0^{I,m}$  and  $C_4^{I,m}$  are given by

$$\begin{aligned}C_0^{I,m} &= 2m(I(I+1) - 3m^2) \\ C_4^{I,m} &= 2m(18I(I+1) - 34m^2 - 5)\end{aligned}\quad (46)$$

$\nu_{QIS}$  is given by

$$\nu_{QIS} = -\frac{\nu_Q^2}{90\nu_0}(3 + \eta)\quad (47)$$

and  $\nu_4(\alpha, \beta)$  is given by

$$\begin{aligned}\nu_4(\alpha, \beta) &= \frac{\nu_Q^2}{112\nu_0} \cdot \left\{ \frac{7}{18} [3 - \eta \cos(2\alpha)]^2 \sin^4(\beta) \right. \\ &\left. + 2[\eta \cos(2\alpha) - 2 - \frac{\eta^2}{9}] \sin^2(\beta) + \frac{2}{45} \eta^2 + \frac{4}{5} \right\}\end{aligned}\quad (48)$$

The quadrupole frequency is defined as

$$\nu_Q = 3C_q/2I(2I-1)\quad (49)$$

The experiment uses the fact that all symmetric transitions  $|m\rangle \leftrightarrow |-m\rangle$  are not broadened by first order quadrupole effects. The second order quadrupole broadening is scaled by a factor  $C_4^{I,m}/C_4^{I, \frac{1}{2}}$  in the Multiple-Quantum dimension with respect to the Single-Quantum dimension, see Fig 11(a). Due to this scaling the second order quadrupole broadening can be removed from the F1 dimension by a shearing transformation without removing the isotropic shifts such as the chemical shift. Shearing uses the following transformation

$$\begin{aligned}\nu'_1 &= \nu_1 - \lambda\nu_2 \\ \nu'_2 &= \nu_2\end{aligned}\quad (50)$$

If the shearing factor  $\lambda$  is set to  $C_4^{I,m}/C_4^{I, \frac{1}{2}}$  the new frequency in the F1 dimension will be

$$\nu_1^{-m \rightarrow m} = \left(2m - \frac{C_4^{I,m}}{C_4^{I, \frac{1}{2}}}\right)\nu_{iso} + \left(C_0^{I,m} - \frac{C_4^{I,m}}{C_4^{I, \frac{1}{2}}}\right)\nu_{QIS}\quad (51)$$

The shearing transformation removed the angular dependent part from the equation, giving an isotropic spectrum in the F1 dimension and leaves the F2 dimension unchanged, see Fig. 11(b). This makes MQMAS a very powerful method for revealing contributions of different interactions of half-integer quadrupole nuclei.

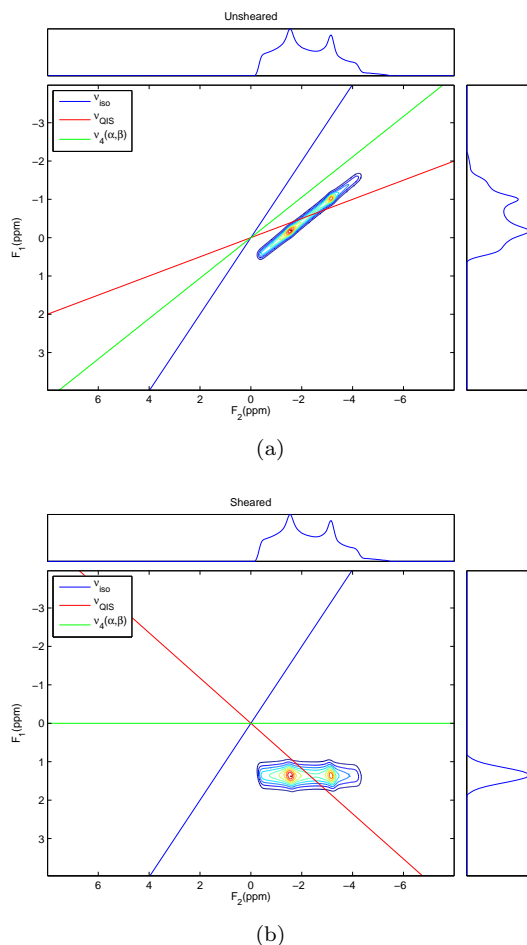


FIG. 11: An unsheared and sheared synthetic MQMAS spectrum ( $C_Q = 3.0$  MHz and  $\eta = 0.39$ ) is shown in Figure (a) and (b) respectively. The F1 axis is scaled to make the slope of the chemical shift ( $\nu_{iso}$ ) equal to one.  $\nu_{QIS}$  is the Quadrupole Induced Shift frequency and  $\nu_4(\alpha, \beta)$  is the angular dependent quadrupole frequency. The colored lines show the direction of the displacement by  $\nu_{iso}$ ,  $\nu_{QIS}$  and  $\nu_4(\alpha, \beta)$ . In the sheared spectrum  $\nu_4(\alpha, \beta)$  in the F1 dimension is zero as shown in equation 51.

### 6. Composing a MQMAS spectrum

The 1D Solid-State NMR spectrum simulation algorithm implemented in the *EGdeconv* program uses a library approach to compose a theoretical spectrum. A “library” of subspectra for a range of  $C_q$  and  $\eta$  values is generated with a simulation program like SIMPSON. Where the weighted sum of these subspectra is used to compose a 1D spectrum, where the weights are determined from the (extended) Czjzek distribution. This library approach for 1D spectra cannot be applied to generate a theoretical MQMAS spectrum. It would

require a library of too high dimensionality. The direct simulation of an MQMAS spectrum using numerical propagation methods in programs such as SIMPSON is too slow for fitting purposes. Therefore an analytic approach is used to generate MQMAS spectra, that includes angular dependent excitation efficiency.

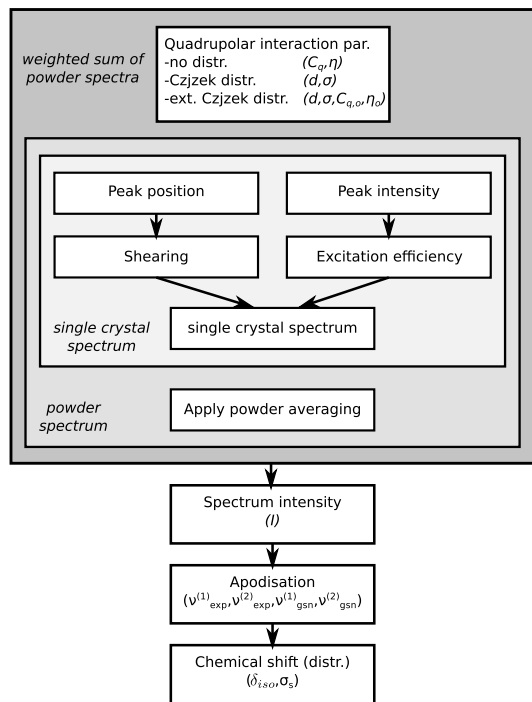


FIG. 12: Program flow of *EGdeconv* for the calculation of a MQMAS subspectrum. A subspectrum is a superposition of single crystal spectra. A superposition of subspectra is used to create a distribution spectrum. The parameters ( $I$ ,  $\lambda$ ,  $Cheng$ ,  $C_q$ ,  $\eta$ ,  $d$ ,  $\sigma$ ,  $C_{q,o}$ ,  $\eta_o$ ) are explained in section III.

#### a. MQMAS subspectrum synthesis

Similar to the theoretical 1D powder NMR spectrum the MQMAS spectrum is the superposition of subspectra of which each corresponds to a parameter-set in the input file. A subspectrum is the weighted sum of single crystal spectra of all possible orientations. A single crystal MQMAS spectrum contains a single peak. To analytically describe a single crystal MQMAS spectrum only the position and the intensity of the peak are required. See Fig. 12 for the program flow of *EGdeconv* for the calculation of a MQMAS subspectrum. The position is given by the frequency in the F1 and F2 dimension given by eqn. 45

The intensity of the peak is determined by the efficiency of the excitation and conversion pulse, see Fig. 10. The default of the program is to assume the excitation/conversion efficiency to be equal for all crystallite orientations; the “ideal excitation” case. This is a very rough assumption, which is only valid if  $\omega_Q \ll \omega_{rf}$  or  $\omega_Q \gg \omega_{rf}$ . For intermediate cases the excitation/conversion efficiency is calculated for each crystallite orientation individually. The expression for the efficiency calculations was obtained by solving the



Liouville von Neumann equation for the density matrix analytically. This method of finding the expressions for the angular dependent efficiency is presented in section III A 6 b. The efficiency calculation is currently implemented for spins  $I = 3/2$  and  $I = 5/2$  measured using 3QMAS. To include efficiencies in the calculation use the keyword 'EFF', see section V C.

*b. Excitation and conversion efficiency calculation*

Analytically solving the Liouville von Neumann equation for the density matrix gives the excitation/conversion efficiency for given  $\Omega_Q$ ,  $\omega_{rf}$  and pulse lengths  $\tau_1$  and  $\tau_2$ . After the first pulse the density operator  $\hat{\rho}$  is given by

$$\hat{\rho}(\tau_1) = e^{-i\hat{H}\tau_1}\hat{\rho}(0)e^{i\hat{H}\tau_1} \quad (52)$$

where  $\hat{H}$  is the Hamiltonian during an rf-pulse in units of angular frequency. In this Hamiltonian only the first order quadrupole interaction is included.

$$\hat{H} = -\omega_{rf}\hat{I}_x + \frac{\Omega_Q(\theta, \phi)}{6}[3\hat{I}_z^2 - \hat{I}^2] \quad (53)$$

and  $\Omega_Q(\theta, \phi)$  is given by

$$\Omega_Q(\theta, \phi) = \frac{\omega_Q}{2}(3\cos^2(\theta) - 1 + \eta\sin^2(\theta)\cos(2\phi)) \quad (54)$$

where  $\phi$  and  $\theta$  are the Euler angles with respect to the main magnetic field, see paragraph III A 6 d.

Equation 52 was first written in a Zeeman basis of spin quantum number  $I=3/2$ . The density matrix at equilibrium  $\hat{\rho}(0)$  was assumed to be equal to  $\hat{I}_z$ . To obtain an analytic expression for  $e^{\pm i\hat{H}\tau_1}$  the Hamiltonian in equation 53 has to be diagonalized. This analytic diagonalization has been performed using *Mathematica* version 8. This results in a density matrix  $\rho(\tau_1)$  as a function of  $\theta$  and  $\phi$  for given  $\Omega_Q$ ,  $\omega_{rf}$  and  $\tau_1$ .

Then all elements of the density matrix  $\rho(\tau_1)$  are set to zero, except for the 3Q coherence terms. This corresponds to the experimental phase-cycle. For spin  $I=3/2$  the 3Q elements are described by

$$\begin{aligned} \rho_{41}(\tau_1) = & -\rho_{14}(\tau_1) = \frac{-3i}{ds} \left( d \cos\left(\frac{d\tau_1}{2}\right) \right. \\ & \times \left[ s \cos\left(\frac{s\tau_1}{2}\right) \sin(\omega_{rf}\tau_1) \right. \\ & - (\Omega_Q + \omega_{rf}) \cos(\omega_{rf}\tau_1) \sin\left(\frac{s\tau_1}{2}\right) \\ & + \sin\left(\frac{d\tau_1}{2}\right) \left[ \Omega_Q^2 \sin(\omega_{rf}\tau_1) \sin\left(\frac{s\tau_1}{2}\right) \right. \\ & \left. \left. + (\Omega_Q - \omega_{rf})s \cos(\omega_{rf}\tau_1) \cos\left(\frac{s\tau_1}{2}\right) \right] \right] \end{aligned} \quad (55)$$

with

$$d = \sqrt{\Omega_Q^2 - 2\Omega_Q\omega_{rf} + 4\omega_{rf}^2} \quad (56)$$

$$s = \sqrt{\Omega_Q^2 + 2\Omega_Q\omega_{rf} + 4\omega_{rf}^2} \quad (57)$$

We also obtained expressions for spin  $I=5/2$  where the *Mathematica*-based Hamiltonian diagonalization of P.Man [30] following Muha [31] and Creel [32] was used. This resulted in the following expressions for the central 3Q elements.

$$\begin{aligned} \rho_{52}(\tau_1) &= \frac{i}{2} \sum_{m=1}^3 \sum_{n=1}^3 \frac{k_{mn}}{Q_{m+}Q_{n-}} \sin((\omega_{m+} - \omega_{n-})\tau_1) \\ \rho_{25}(\tau_1) &= -\rho_{52}(\tau_1) \end{aligned} \quad (58)$$

$$Q_{m\pm} = \left[ 1 + \frac{5\omega_{rf}^2}{4\left(\frac{10}{3}\Omega_Q - \omega_{m\pm}\right)^2} + \frac{2\omega_{rf}^2}{\left(\frac{8}{3}\Omega_Q \pm \frac{3}{2}\omega_{rf} + \omega_{m\pm}\right)^2} \right] \quad (59)$$

$$k_{mn} = \left[ 3 + \frac{25\omega_{rf}}{4\left(\frac{10}{3}\Omega_Q - \omega_{m+}\right)\left(\frac{10}{3}\Omega_Q - \omega_{n-}\right)} + \frac{2\omega_{rf}}{\left(\frac{8}{3}\Omega_Q \pm \frac{3}{2}\omega_{rf} + \omega_{m+}\right)\left(\frac{8}{3}\Omega_Q \pm \frac{3}{2}\omega_{rf} + \omega_{n-}\right)} \right] \quad (60)$$

The eigenvalues  $\omega_{m\pm}$  are given by

$$\begin{aligned} \omega_{1\pm} &= \mp \frac{\omega_{rf}}{2} + 2\sqrt{\frac{s_{\pm}}{3}} \cos\left(\frac{\phi_{\pm}}{3}\right), \\ \omega_{2\pm} &= \mp \frac{\omega_{rf}}{2} - 2\sqrt{\frac{s_{\pm}}{3}} \cos\left(\frac{\pi}{3} - \frac{\phi_{\pm}}{3}\right), \\ \omega_{3\pm} &= \mp \frac{\omega_{rf}}{2} - 2\sqrt{\frac{s_{\pm}}{3}} \cos\left(\frac{\pi}{3} + \frac{\phi_{\pm}}{3}\right), \end{aligned}$$

where

$$\begin{aligned} \cos(\phi_{\pm}) &= \frac{\Omega_Q}{18s_{\pm}} (160\Omega_Q^2 \mp 36\Omega_Q\omega_{rf} - 144\omega_{rf}^2) \sqrt{\frac{3}{s_{\pm}}}, \\ s_{\pm} &= \frac{84}{9}\Omega_Q^2 \pm 4\Omega_Q\omega_{rf} + 4\omega_{rf}^2, \end{aligned} \quad (61)$$

The approach for the first pulse is then applied for the second pulse. This gives the expressions for the efficiency of the excitation and conversion pulses. For spin  $I=3/2$  this gives

$$\begin{aligned} \rho_{23}(\tau_2, \tau_1) &= \\ & \frac{3\omega_{rf}^2(\rho_{14}(\tau_1) - \rho_{41}(\tau_1)) \cos(\omega_{rf}\tau_2) \sin\left(\frac{d\tau_2}{2}\right) \sin\left(\frac{s\tau_2}{2}\right)}{2ds} \end{aligned} \quad (62)$$

The same procedure was applied to spin  $I=5/2$  resulting in the following expression.

$$\begin{aligned} \rho_{34}(\tau_2, \tau_1) &= (\rho_{25}(\tau_1) - \rho_{52}(\tau_1))\omega_{rf}^2 \\ & \times \sum_{m=1}^3 \sum_{n=1}^3 \frac{\cos((\omega_{m+} - \omega_{n-})\tau_2)(Q_{m+}Q_{n-})^{-1}}{\left(\frac{8}{3}\Omega_Q + \frac{3}{2}\omega_{rf} + \omega_{m+}\right)\left(\frac{8}{3}\Omega_Q - \frac{3}{2}\omega_{rf} + \omega_{n-}\right)} \end{aligned} \quad (63)$$

These expressions for 3QMAS of spin  $I=3/2$  and  $I=5/2$  are implemented in the *EGdeconv* program.

*c. Z-filtered MQMAS*

One of the most widely used variations of the two pulse MQMAS is the z-filtered MQMAS experiment [33].

This pulse sequence consists of three pulses, as shown in Fig. 13. The multiple quantum coherence pathways

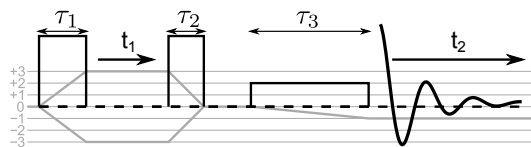


FIG. 13: Schematic overview of the z-filtered MQMAS experiment and its coherence pathways. The first pulse excites a symmetric multiple quantum coherence, which is allowed to evolve during  $t_1$ . The second pulse will convert the MQ coherence back to a population difference or zero quantum coherence. The third pulse is a weak selective pulse which creates the detectable single quantum coherence.

are symmetric during the conversion pulse, which gives less distortions in the final spectrum. Efficiency calculations for the z-filtered MQMAS are implemented in the program. The equations for the excitation efficiency of the z-filtered MQMAS are the same as those given in eqn. 55 and 58. The second pulse creates a population difference between the central-transition energy levels. The third pulse converts this population difference into a measurable single quantum coherence. The analytic expression for the efficiency are defined as the difference between the central two diagonal elements of the density matrix. For spin  $I=3/2$  the formula is given by  $\rho_{22}(\tau_2, \tau_1) - \rho_{33}(\tau_2, \tau_1)$ .

$$\rho_{22}(\tau_2, \tau_1) - \rho_{33}(\tau_2, \tau_1) = \frac{-3i\omega_{rf}^2(\rho_{14}(\tau_1) - \rho_{41}(\tau_1)) \sin(\omega_{rf}\tau_2) \sin(\frac{d\tau_2}{2}) \sin(\frac{s\tau_2}{2})}{ds} \quad (64)$$

For spin  $I=5/2$  the formula is given by  $\rho_{33} - \rho_{44}$

$$\rho_{33}(\tau_2, \tau_1) - \rho_{44}(\tau_2, \tau_1) = -2i(\rho_{25}(\tau_1) - \rho_{52}(\tau_1))\omega_{rf}^2 \times \sum_{m=1}^3 \sum_{n=1}^3 \frac{\sin((\omega_{m+} - \omega_{n-})\tau_2)(Q_{m+}Q_{n-})^{-1}}{(\frac{8}{3}\Omega_Q + \frac{3}{2}\omega_{rf} + \omega_{m+})(\frac{8}{3}\Omega_Q - \frac{3}{2}\omega_{rf} + \omega_{n-})} \quad (65)$$

The third pulse is not included in the calculations, because the effect on the angular dependent efficiency is very small if the rf-field strength is chosen low enough to ensure selective excitation of the central transition. Z-filter efficiency calculations in the program are enabled by default. If the experiment was done using the two-pulse sequence, the z-filter calculation can be disabled by setting 'ZFILTER = 0'.

#### d. Euler angles

The orientation of a tensor with principal axes (X,Y,Z) in a frame (x,y,z) can be described by three Euler angles  $\alpha$ ,  $\beta$  and  $\gamma$ , see Fig. 14.  $\alpha$  is defined as the angle between the x-axis and the line of nodes. The line of nodes is perpendicular to both z-axis and the Z-axis.  $\beta$  is defined as the angle between the z axis and the Z-axis.  $\gamma$  is defined as the angle between the line of nodes and the X-axis. The angles  $\alpha$ ,  $\beta$  and  $\gamma$  are defined with respect to the MAS rotor frame. In the LAB frame the three Euler angles are defined as  $\phi$ ,  $\theta$  and  $\psi$  with respect

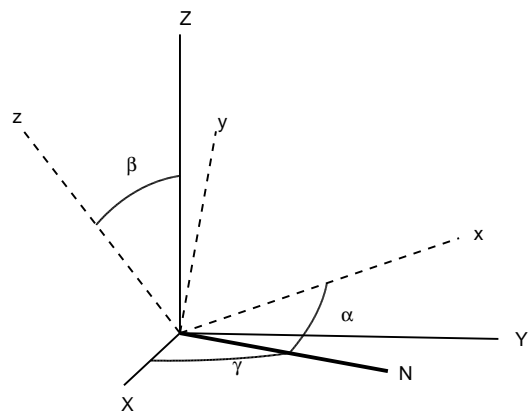


FIG. 14: The principal axes (X,Y,Z) and Euler angles ( $\alpha$ ,  $\beta$ ,  $\gamma$ ) of a tensor with respect to a reference frame (x,y,z). The principal axes are shown in red and the reference frame is shown in blue.

to the main magnetic field. The third angles  $\gamma$  and  $\psi$  are not used in the calculations, because both frames have a cylindrical symmetry. In the LAB frame the quadrupole frequency is invariant to rotation around the main magnetic field. In the MAS frame the sample is assumed to be spinning infinitely fast, which causes the quadrupole frequency to be invariant to rotation around the rotor axis. The angle  $\gamma$  is only needed in the conversion from MAS frame to LAB frame for the efficiency calculation, because the sample is assumed to be static during the rf-pulses. The conversion from MAS to LAB frame is performed as

$$\begin{aligned} \cos(\theta) &= -\sin(\Theta_M) \sin(\beta) \cos(\gamma) + \cos(\Theta_M) \cos(\beta) \\ \cos(\phi) &= \cos(\Theta_M) \sin(\beta) \cos(\alpha) \\ &\quad + \sin(\Theta_M) \cos(\beta) \cos(\alpha) \cos(\gamma) \\ &\quad - \sin(\Theta_M) \sin(\alpha) \sin(\gamma) \end{aligned} \quad (66)$$

where  $\Theta_M$  is the magic angle.

#### e. Powder average

The powder average is taken using the ZCW scheme [34]. This is a powder average over the two angles  $0 \leq \alpha < 2\pi$  and  $0 \leq \beta < \pi$ . The number of angle pairs used in the powder averaging has to be a Fibonacci number ( $F_A$ ). Fibonacci numbers are generated using the following recursion

$$\begin{aligned} F_{A+2} &= F_{A+1} + F_A, \\ A &= 0, 1, 2, \dots \\ F_0 &= F_1 = 1 \end{aligned} \quad (67)$$

The number of angle pairs  $N_M$  used in the calculation is defined as

$$N_M = F_{M+2} \quad (68)$$

where M is the Cheng number. The Cheng number can be set using the keyword 'CHENG', see section V C. The default Cheng number in the program is 10. The

angles in the  $j$ -th ZCW pair with Cheng number  $M$  are

$$\begin{aligned}\alpha_j^{ZCW} &= 2\pi(jF_M \bmod N_M)/N_M, \\ \beta_j^{ZCW} &= \pi j/N_M, \\ j &= 0, 1, 2, \dots, N_M - 1\end{aligned}\quad (69)$$

These angles are defined with respect to the MAS rotor axis. The efficiency equations are defined in the LAB frame. Conversion between these two frames requires an integration over the third angle  $\gamma$ , see paragraph III A 6 d. The number of integration steps over  $\gamma$  can be set by the keyword 'NGAMMA'. 'NGAMMA' is not used when the efficiency calculation is disabled.

#### f. Input/Output

The format of the input and output spectrum is a one column ASCII file, with the following header

```
n_s 1
np1 1024
np2 1024
sw1 150000.00
sw2 50000.00
DATA
```

`n_s` gives the number of spectra in the file. This should be 1 for the input spectrum. `np1` and `np2` give the number of points in the F1 and F2 dimension respectively. `sw1` and `sw2` give the spectral width in the F1 and F2 dimension respectively. The data is then ordered first by F2 dimension and then F1 dimension. If the fitted spectrum contained multiple sites the output data will be ordered by first the subspectra of the individual sites, followed by the sum of all the subspectra and finally the experimental spectrum.

To save disk space the output can also be printed in a binary format. This is done via the 'BINARY' keyword. The first three values are 32 bit integers representing the number of spectra and the number of points in the F1 and F2 dimension respectively. Followed by two 64 bit double precision floating point numbers giving the spectral widths in the F1 and F2 dimension. The data is all in double precision format and is ordered similarly to the ASCII output.

#### g. Distributions

During the MQMAS fit either no or an (extended) Czjzek distribution can be applied. The type of distribution is set using parameter 198. Depending on the choice, parameter number one and two ( $\eta$  and  $C_q$  or  $d$  and  $\sigma$ ) or one to four ( $d$ ,  $\sigma$ ,  $C_{q,o}$  and  $\eta_o$ ) are available for fitting, see Fig.12. The grid size can be set by the keywords 'NDISTRQ' and 'NDISTRE' for the number of points in the  $C_q$  and  $\eta$  dimension respectively, see section V C. The gridsizes are parameter-set independent. A distribution in chemical shift is described in paragraph III A 6 k

#### h. Apodization

Apodization with an exponential or Gaussian function is applied to the MQMAS spectra. The definitions are

the same as those given in equation 43 for powder NMR. The apodization is set individually for the F1 and F2 dimension. The exponential apodization is set with parameter 23 ( $\nu_{exp}^{(1)}$ ) and 24 ( $\nu_{exp}^{(2)}$ ) for the F1 and F2 dimension respectively. Similarly the Gaussian apodization is set with parameter 25 ( $\nu_{gsn}^{(1)}$ ) and 26 ( $\nu_{gsn}^{(2)}$ ). The apodization frequencies are all given in Hz.

Fitting the apodization parameters together with other parameters is only advisable if there are no distributions present in the spectrum. Distributions will influence the linewidths in the F1 and F2 dimension and this may lead to unsatisfactory fits. If distributions are present the user should try to set the apodization parameters equal to the amount of apodization used on the experimental spectrum.

#### i. Subspectrum intensity

The intensity parameter number 20 is also available for MQMAS fitting. With this parameter it is possible to fit the (relative) abundance of different sites. This is useful in quantifying MQMAS data. The quality of the fit is dependent on the experimental measurement conditions. Large offsets  $\Delta\omega > \omega_Q/2$  can influence the excitation and conversion efficiency and thus the outcome of the intensity fit as shown in Fig. 12

#### j. Shearing

Shearing is a spectral processing method used to aid interpretation of MQMAS spectra, see section III A 5. It also makes spectral editing using *remove parts spectrum* easier, see section V C. Shearing uses the transformation given in equation 50. The effect of shearing is shown in Fig. 11. The value given after the keyword 'SHEAR' will be used as shearing coefficient  $\lambda$ . Shearing is always applied before apodization, see Fig. 12. The simulated spectra will be sheared along the F1 axis and the experimental spectrum is not sheared. By default no shearing will be applied.

#### k. Chemical shift

Chemical shift displaces the spectrum along the chemical shift direction shown in Fig. 11. If the total shift of a peak  $\nu_t$  is larger than half the spectral width  $\nu_t > \frac{1}{2}\nu_{sw}$  the peak will reappear at the opposite side of the spectrum  $\nu_t - \nu_{sw}$ . This effect is called aliasing. Aliasing effects are implemented in the program. Aliasing is only allowed to occur for peaks that fall within  $\nu_t < \frac{3}{2}\nu_{sw}$ , otherwise they are removed.

In MQMAS there can also be a distribution in chemical shift. This original spectrum ( $f$ ) is convoluted with a gaussian function ( $g$ ) along the chemical shift direction ( $\nu_{cs}$ )

$$(f * g)(\nu_{cs}) = \int f(\nu_{cs} - \nu_s)g(\nu_s) d\nu_s \quad (70)$$

$$g(\nu_s) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp\left(-\frac{(\nu_s - \nu_{iso})^2}{2\sigma_s^2}\right) \quad (71)$$

The average value is set by parameter number 21 ( $\nu_{iso}$ )

in ppm). The Full Width at Half Maximum of the Gaussian broadening is set by parameter number 22 ( $\sigma_s$  in ppm). To set the number of points in the chemical shift distribution use the keyword 'NDISTRs'.

#### IV. EXPLANATION AND DEFINITION OF THE INPUT PARAMETERS OF *EGDECONV*

##### A. General

The operation of the program *EGdeconv* is controlled by an input file with a control command on each line. The general structure is:

$$\text{keyword string}=\text{value(s)}. \quad (72)$$

The *value* consists of a *string* or of one or more values; of numerical and/or of character type. The order of appearance of the *keyword strings* in the file is not important with one exception discussed in section IV B. Blanks or TABs at the beginning of a line are removed. A line starting with "#", an unknown *keyword string* or in which no "="-sign is found, is considered as comment and its content is ignored. *EGdeconv* can read Unix/Linux as well as DOS type of files<sup>6</sup>. So irrespectively of the editor you use to create the input file, *EGdeconv* will be able to read the file. The full list of the *keyword strings* and their effects on the program is given in Table II.

##### B. The keyword string "parameter"

###### 1. Definition and use of the keyword string "parameter"

The keyword string *parameter* is different from the other keyword strings. It is used to communicate to the *EGdeconv* program which parameters are used and what their properties are. The keyword string *parameter* has the structure:

$$\text{parameter}=\text{n name start [lower] [upper] [coupling] [nset]} \quad (73)$$

for *coupling* = *E*, *D* or *R*, or

$$\text{parameter}=\text{n name start [lower] [upper] [coupling] [num]} \quad (74)$$

for *coupling* = *I*.

Here *n* is the number of the parameter as used in *EGdeconv* (see Sect. V), *name* a user friendly name to facilitate the recognition of the parameter in the output, *start* the start value of the parameter, *lower* and *upper* are the lower and upper limits of the parameter in the

EA evaluation and *coupling* is the type of coupling between parameters of different sets *nset* or within a set *num* (see below). The following special rules apply for the keyword string *parameter*

- All lines with the keyword string *parameter* should be put together in the input file as a single block.
- The entries behind the "=" sign should be separated by at least one blank; a TAB may also be used as separator. The *name*-field may contain no blanks or TABs.
- The minimum input on a line with keyword *parameter* is *n*, *name* and *start*. The *start*-values are used (i) to simulate the spectrum if no EA-fit is done, (ii) as start values before the EA-procedure starts and (iii) for those parameters that are held fixed during the EA-calculation.
- Parameters for which no line is present in the input are set to zero in *EGdeconv*, are not fit and are not shown in the output lists
- *lower* and *upper* should either both be absent or both be present. If both are present *EGdeconv* uses these values as lower and upper limits in the EA-calculation.
- The numbering *n* should be incremental, except to indicate more spectra (see next point).
- It is possible to have more spectra calculated in the same frequency interval with different parameter values and ranges. In order to indicate this, the program assumes that if during reading of the parameter list a number *n* is found smaller or equal to the last read one, the following parameters to be read are from a next spectrum. This is indicated as a different *sets*.
- The *coupling* flag can be used both within a set as well as between sets. Its function is different in the two cases. if more than one *parameter set* is found. The flag can be upper or lower case, only the first letter is read.

For values of the coupling flag *E*, *D* and *R* it controls the values of the *same* parameter between *different* sets and its effect is defined by (for the definition of *set* see next item):

- *E(qual)*: The value of a *Equal coupled* parameter in the indicated set #*i* is set equal to the value in set #*nset*.  
(parameter set #*i*) = (parameter set #*nset*):

$$p_j^{(i)} = p_j^{(nset)} \quad (75)$$

Since  $p_j^{(i)}$  is fixed to the value  $p_j^{(nset)}$ ,  $p_j^{(i)}$  cannot be varied anymore. If the user gives upper and lower limits for  $p_j^{(i)}$  these limits are ignored by *EGdeconv*

<sup>6</sup> In a Unix/Linux file the end of line is indicated by a line feed character (ASCII 10), while for a DOS file it is the combination of a line feed and a carriage return (ASCII 13) character.

- *D(ifference)*: (parameter set #i) = (parameter set #nset) + (difference):

$$p_j^{(i)} = p_j^{(nset)} + D_j^{(i)} \quad (76)$$

- *R(atio)*: (parameter set #i) = (parameter set #nset) × (ratio):

$$p_j^{(i)} = p_j^{(nset)} R_j^{(i)} \quad (77)$$

The coupling flag *I* controls the value of a *different* parameter within the *same* set and its effect is defined by:

- *I(dentical)*: The value of a *Identical coupled* parameter (i) is set equal to another parameter (j) within the same set.  
(parameter num #i) = (parameter num #j):

$$p_i^{(nset)} = p_j^{(nset)} \quad (78)$$

- The parameter "nset" after the *coupling* flag (*E*, *D*, *R* and *I*) determines with which set the particular parameter is coupled. All combinations of couplings between different sets can be made for each parameter independently. This gives full freedom for the user. If a cyclic relation is found the program stops with an error message<sup>7</sup>. If the *coupling* flag is found a value for *nset* (for *E*, *D* or *R*) or *num* (for *I*) should be present, otherwise the program is stopped with an error message.
- More complicated relations can be imposed on the parameters, both within and between sets by using the *parameter constraint*-option. (See sect. IV B 3)

## 2. List of parameters.

The full list of the *parameters*, their numbers and use is given in Table I. The default dimensions of the parameters are indicated. For most of the parameters the default dimension is Hz.

## 3. Parameter constraints.

The possibility to impose conditions between parameters has implemented in the *EGdeconv* program. Simple mathematical relations like sum/difference and multiply/divide are possible, both between parameters of the same set as well of different sets. It is reasonable flexible and it tests if the parameter numbers are within the allowed ranges. However it does not perform a full check of all errors one can make, like for example cyclic

relations. The user should use this option with some care. The corresponding keyword has the structure:

$$\text{parameter constraint}=(n_i^{par}, n_j^{set}) = \text{expression} \quad (79)$$

Here  $n_i^{par}$  and  $n_j^{set}$  are the values of the parameter and the set for which the relation is imposed on. Examples for use of this option and of the expression are:

$$\begin{aligned} \text{parameter constraint} &= (100, 2) = \\ & (100, 1) + 100 * (105, 1) \\ p_{100}^{(2)} &= p_{100}^{(1)} + 100 p_{105}^{(1)} \quad (80) \\ \text{parameter constraint} &= (100, 2) = \\ & (87, 5) + (107, 2)/100 + \\ & 10 - (103, 4) \\ p_{100}^{(2)} &= p_{87}^{(5)} + p_{107}^{(2)}/100 \\ & + 10 - p_{103}^{(4)} \quad (81) \end{aligned}$$

Here the upper index and lower index of *p* label the parameter set and parameter number, respectively. Note that blanks are ignored, while brackets and commas are required.

<sup>7</sup> It should be noted that for the coupling parameter *I* the check on possible cyclic relations is much more complicated. Hence the program is only able to perform a partial check and the user should take more care in this case.

## V. PARAMETER AND OPTION LIST USED BY THE *EGDECONV* PROGRAM

### A. Fit parameters

TABLE I: List of parameters used by *EGdeconv* for the fit of a powder spectrum. The program uses one set of these parameters per site. See section III A 4. The parameters marked (PN) are used by the 1D powder NMR fitting. The parameters marked (MQ) are used by the MQMAS fitting.

Number	Suggested name	Description of the parameter
1	eta, d	Identity of parameter is related to parameter 198, see also section III A 4 c. Eta should be in the range zero to one and d should be: 1,2,3,4 or 5.
2	Cq, sigma	Identity of parameter is related to parameter 198, see also section III A 4 c. Cq(MHz) and sigma(MHz) can both not be negative and sigma cannot be equal to zero.
3	Cq0	Identity of parameter is related to parameter 198, see also section III A 4 c. Cq0(MHz) cannot be negative.
4	eta0	Identity of parameter is related to parameter 198, see also section III A 4 c. eta0 should be in the range zero to one.
20	int, S	Identity of parameter is determined by keyword <i>fit order parameter</i> , see section V C. Parameter int can be equal to or larger than zero, use <i>R</i> coupling (section IV B) to fit one parameter less. Intensity set to 0 has a special meaning, since a large part of the subspectrum calculation will then be skipped. S can have values in the domain zero to one.
21	shift	Spectral shift of the subspectrum in units of the experimental frequency axis.
22	lor (PN)	Exponential apodisation of the fid, see eqn. 43. Unit of frequency is equal to the experimental frequency axis.
	dsiso (MQ)	The Full Width at Half Maximum of the gaussian chemical shift distribution, see section III A 6 g
23	gauss (PN)	Gaussian apodisation, see eqn. 44 and also parameter 22.
	lor1 (MQ)	Exponential apodisation in the F1 dimension in Hz, see section III A 6 h.
24	phi0 (PN)	Zeroth order phasing of the spectrum, see section III A 4 f. Phase angle is in degrees.
	lor2 (MQ)	Exponential apodisation in the F2 dimension in Hz, see section III A 6 h.
25	phi1 (PN)	First order phasing of the spectrum, see section III A 4 f. Phase angle is in degrees.
	gauss1 (MQ)	Gaussian apodisation in the F1 dimension in Hz, see section III A 6 h.
26	voff (PN)	Frequency offset for phasing, see section III A 4 f. Offset units are equal to experimental frequency axis.
	gauss2 (MQ)	Gaussian apodisation in the F2 dimension in Hz, see section III A 6 h.
45	x (PN)	Parameter is only available if order parameter is fitted, see keyword <i>fit order parameter</i> in section V C. The meaning of the <i>x</i> parameter is defined in section III A 3.
198	no_distr, czj_distr, ext_czj_distr (PN)	The name of this parameter determines the distribution type, see section III A 4 c, the three choices are indicated in the 'suggested name' column. 'ext_czj_distr' involves the numerical integration of Eq. (34) or (36) and is more time consuming. The value of this parameter indicates the type of (sub)spectrum, currently only powder (1) is supported.
198	distr (MQ)	The type of distribution used in the MQMAS. 0 for no distribution, 1 for a Czjzek distribution and 2 for an extended Czjzek distribution, see section III A 6 g,
199	<sitename> (PN)	The name of this parameter will be used in the output to indicate the site name; unless the order parameter is fitted, see keyword 'fit order parameter' in section V C, it can be chosen arbitrarily. In case the order parameter is used the name has to be: A4,A3B,A2B2,AB3 or B4 in accordance with eqn. (40). The value of this parameter is the number <i>X</i> in the 'species <i>X</i> ' keyword that refers to the correct library. It is allowed that not all species are assigned.

### B. Control options of the *EGdeconv* program

TABLE II: Description of the keyword strings that can be used in the input file of *EGdeconv*.

Keyword string <i>EGdeconv</i>	Default	Description
test calculation	0	This parameter determines if the full EA calculation is performed (0) or only a test calculation (1) with a single call to the fitness or cost function with the <i>start</i> parameter values
comment		A string to describe relevant information about the calculation, the spectrum etc. can be entered here. An empty comment-string is allowed.
derandomized DR2 algorithm	0	(0) The genetic algorithm is applied (Sect. II A 1) (1) The ES (Evaluation Strategy) derandomized DR2 will be used. See Sect. II A 2 for more information. See also the next keyword <i>CMA-ES algorithm</i> for default value.
CMA-ES algorithm	0	(0) The genetic algorithm is applied (Sect. II A 1) (1) The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) algorithm will be used. See Sect. II A 3 for more information. If both the derandomizeDR2 and the CMA-ES algorithms are activated, preference is given to the CMA-ES algorithm and the keyword <i>derandomized DR2 algorithm</i> (see above) is neglected.
create CMA-ES resume file	0	(0) No resume file is created (1) A file is created for each generation to allow a resume of the CMA-ES fit after this generation. This is of particular interest if the program is used to optimize an experimental setup. In such a case the optimization can easily be interrupted by all kind of experimental failures. See also sect. VI C.
CMA-ES resume file		If not defined, the option is ignored. Otherwise the CMA-ES fit will resume at the point defined in the resume file created with the option shown above.
population size	300	Population size of the GA or the offspring size for a derandomized DR2 or CMA-ES calculation
number of generations	500	Maximum number of generation in a full calculation
Number of parents	0	This option is only valid for <i>derandomized DR2 algorithm=1</i> or <i>CMA-ES algorithm=1</i> The value of this keyword determines the choice of the first best number of parents of the previous generation from which the next generation is created. For the default value (0): $NumberOfParents = PopulationSize/2$
maximize evaluation function	0	This allows for a choice to minimize the evaluation function (0, Cost function evaluation) or maximize this function (1, Fitness function evaluation).
type of costfunction	1	0: Innerproduct Costfunction defined in eqn. 8 1: Least squares of equal maximum intensity eqn. 6 2: Least squares for equal integral eqn. 6 3: Least squares of the MQMAS projections with maximum intensities.
create CMA-ES output	0	If the value of this keyword is not 0 a MATLAB file and a series of .dat files are created to monitor the convergence of the CMA-ES fit. The prefix and location of these files is controled by the keyword <i>logfile</i> .  The way of plotting the results is determined by the keyword value: (0) No convergence files and MATLAB file are created (1) The convergence characteristics are plotted verses the iteration (generation) (2) The convergence characteristics are plotted verses the number of calls of the evaluation function
parameter constraint <sup>8</sup>		The possibility to impose conditions between parameters is implemented in the <i>EGdeconv</i> -program. Simple mathematical relations like sum/difference and multiply/divide are possible, both between parameters of the same set as well of different sets. See for more details Sec. IV B 3).
restrictions on linewidths	YES	This keyword string has only effect if more than 1 set of parameters are fit. If "YES" the <i>corresponding</i> line widths of all sets obtain the same value. Note: Gaussian line widths are not coupled to Lorentzian widths.

<sup>8</sup> If the expression starts with an operator, the operator is neglected with the exception of "-".

TABLE II: keyword strings (continued)

keyword string	Default	Description
minimal evaluation ratio	1.1	Determines in the final averaging of a series of full EA-calculations are included in the "selected average". Included are fitness function values $< (\text{minimal evaluation ratio}) \times (\text{best fitness function value})$
direct cross correlation	0 <sup>9,10</sup>	(0) Calculation of fitness function and cross sums via stick spectrum (Fast) (1) Full calculation of fitness function and cross sums (Slow)
experimental spectrum		File name of input (experimental) spectrum. If ( <i>test calculation=0</i> ) and the file does not exist the program stops with an error message. The program furthermore assumes the spectrum is linearized, the points are equidistant and the frequency scale is incrementally. It takes the first and last frequency point and the number of points to generate the frequency scale. An error is given if the points are not equidistant or the frequency scale is not incremental. It was chosen on purpose to use only linearized and background free spectra in the <i>EGdeconv</i> -fit. All one can do outside the real fit program without affecting the information, should be done. If ( <i>test calculation=1</i> ) and the keyword string " <i>experimental spectrum</i> " is absent or the string is empty, <i>EGdeconv</i> assumes it has to generate a spectrum based on the <i>start</i> parameters and the key strings " <i>minimum frequency</i> ", " <i>maximum frequency</i> " and " <i>number of points in spectrum</i> ", defined below.
minimum frequency		Minimum frequency for a synthesized spectrum based on the <i>start</i> parameters.
maximum frequency		Maximum frequency for a synthesized spectrum based on the <i>start</i> parameters.
number of points in spectrum		Number of points in a synthesized spectrum based on the <i>start</i> parameters.
remove parts spectrum		The file name that lists the parts of the spectrum that should be removed in the EA fit by setting the intensities of the calculated and experimental spectrum to zero in the removed sections. Each line in the file indicates a part to be removed. It contains two numbers defining the lower and upper boundaries, respectively, of the section to be removed. Overlapping sections are allowed and the order is irrelevant. If the keyword is absent, the <i>EGdeconv</i> -program assumes that the full spectrum should be used. In the MQMAS fit the remove parts spectrum file is used to remove a rectangle from the 2D spectrum. In this case the file should contain four numbers. The first two numbers indicate the vertical index <i>vindex</i> and horizontal index <i>hindex</i> respectively. The last two numbers indicate the vertical width <i>vwidth</i> and horizontal width <i>hwidth</i> respectively. If both widths are positive a rectangle is removed from the spectrum. All points are removed that have a vertical index between $vindex - 1$ and $vindex + vwidth + 1$ and a horizontal index between $hindex - 1$ and $hindex + hwidth + 1$ . If any of the widths is negative all points are removed that have a vertical index outside $vindex$ and $vindex +  vwidth $ or a horizontal index outside $hindex$ and $hindex +  hwidth $ .
remove smoothed background	0	If - 0, No effect If - > 1, A smoothed spectrum is removed (simultaneously) from the experimental and calculated spectra. The idea behind this option is that a removal of a smoothed spectrum enforces the features in the spectrum which are on top of a "background". So in the global fit the local features are strengthened. For more details see Sect. II D. The value of <i>remove smoothed background</i> defines the number of points over which the smoothing takes place. The choice of this value depends on the characteristics of the spectrum.
minimum intensity line	0.005/ 0.15	Minimum intensity of transitions that are included in the calculated spectrum. For a <i>normal</i> spectrum the default value is 0.005; in case of a Stick spectrum (see above) the default value is 0.15.

<sup>9</sup> If the line shape parameters are fit *EGdeconv* automatically chooses 1

<sup>10</sup> For test purposes only: (2) Calculation fitness function and cross sums via full calculation AND stick spectrum + full printout of results to screen.



TABLE II: keyword strings (continued)

keyword string	Default	Description
full width overlap function	1	Full width of overlap function in units of "spectrum points". $\Delta w$ is defined in Eq. (13)
lineshape	G	Lineshape parameter. Allowed values are (G)auss, (L)orentz or (V)oigt. For a <i>Stick spectrum</i> (see above) the value of <i>lineshape</i> is ignored
lineshape derivative	0	Type of lineshape: (0) Normal lineshape, (2) Second derivative of the lineshape is used. (1) is not allowed). For a <i>Stick spectrum</i> only (0) is imposed by the program.
output Costfunction values	0	(0) No output of Cost function values. (1) The Cost function values for each iteration step are written to a $(1 + N_{seeds})$ column ASCII file; $N_{seeds}$ is the number of seeds. The first column contains the number of the generation and the next $N_{seeds}$ column(s) the corresponding Cost function value. This can be used to plot the Cost function value as function of the generation.
create Full spectrum file	1	If (0) No <i>full</i> output spectrum. If (1) A <i>full</i> output spectrum (file name automatically generated) is created which includes the calculated and experimental spectrum and the individual spectra calculated for each set. The parameters of the calculated spectrum are from the last generation. If more than one GA-calculation is performed the unselected average parameter are used. The output file is in the form of a $(3 + nset)$ -column ASCII array with the frequency in the first column, the intensities of the calculated and experimental spectra in the second and third column, respectively. The other <i>nset</i> -columns contain the spectrum for the <i>nset</i> -parameter set. If this option is active no <i>simple</i> output spectrum is generated, i.e. the option <i>output spectrum=1</i> is ignored.
weightfunction		Certain parts of the spectrum can have a higher or lower weight relative to other parts. The program is able to apply flexible weights to different parts of the spectra. This is realized with a <i>weightfunction</i> . For more details see Sect. II C. The currently supported functions are: gaussian, lorentzian, triangle, block and polynomial. The first four types are controlled by: $weightfunction = W_{type} \nu_L \nu_H Ampl Width Center$ $W_{type}$ has one of the possible functional forms defined above, $\nu_L$ and $\nu_H$ are the lowest and highest frequency-value for which the weight function will be applied, respectively. If both $\nu_L = 0$ and $\nu_H = 0$ the default values are applied: $\nu_L = \nu_{min}$ and $\nu_H = \nu_{max}$ with $\nu_{min}$ and $\nu_{max}$ the minimum and maximum frequencies in the experimental spectrum Here <i>Ampl</i> is the amplitude, <i>Width</i> the width and <i>Center</i> the center of the $W_{type}$ function, respectively. For the last one (polynomial) the keyword <i>weightfunction</i> is used with the following parameters: $weightfunction = Polynomial \nu_L \nu_H a_0 \dots a_N Origin$ $\nu_L$ and $\nu_H$ are defined as above. The last parameter is the chosen origin ( $\nu_0$ ) for the polynomial. The number of values for $a_N$ ( $N + 1$ ) determines order ( $N$ ) of the polynomial: At least one value for $a_N$ , $N = 0$ should be supplied. It is possible to have more than a one keyword <i>weightfunction</i> . Whether the weightfunctions are applied on both the experimental and calculated spectra or only on the calculated spectrum is determined by the keyword <i>apply Weightfunctions to Exp spectrum</i> , see below.
apply Weightfunctions to Exp spectrum	1	This keyword is only used in combination with the previous one <i>weightfunction</i> . For the default value (1) it applies the Weightfunction to both the experimental input spectrum as well as the calculated spectrum. If its value is set to 0, the weight function is NOT applied to the experimental spectrum.
data type	B	The way the parameter values are represented in the GA-calculation: Binary (B) or Real (R)
fitness type	1	Fitness type in GA: (1) use raw fitness (evaluation), (2) linear normalization fitness, (3) linear ranking fitness (4) sharing fitness. For the last type a non zero value for the parameter <i>sharing distance</i> (see next parameter) is required, otherwise the program stops.

TABLE II: keyword strings (continued)

keyword string	Default	Description
sharing distance		The user-setable sharing distance ( $D^{shd}$ ). Required if <i>fitness type</i> = 4 (Sharing Fitness). If <i>sharing distance</i> =0 and <i>fitness file</i> =1 fitness sharing is ignored but the file with the distribution of the distances is calculated.
fitness file	0	(0) No output for the fitness differences. (1) A file is created that lists the fitness distances within each generation. The file name is automatically generated. This option is active only in case <i>fitness type</i> = 4 (Sharing Fitness). See Sect. VID. A 3D plot can be created using this file.
project fitness 1-100	1	This parameter determines if in a <i>Sharing Fitness</i> calculation the fitness values are used absolutely (0) or if the range (excluding fitness values of 0) is projected on the interval 1 – 100 (1).
gray length	10	String length per parameter in the Binary parameter representation
random seeds		A list of one or more integer values used to start the seed for the random number generation of a GA evaluation. Each value stands for a full EA-calculation. The numbers are separated by at least one blank. If no values is found the program stops with an error message.
% elitism	50	Percentage elitism in the GA
crossover type	2	Used crossover type in the GA: One point (1), two point (2) or uniform (3)
crossover probability	.85	Crossover probability in the GA
mutation type	2	Mutation type in the GA: (1) Real/Integer: Fixed value, (2) Real/Integer: Uniform range, (3) Real: +- Uniform random no., (4) Real: +- Gaussian random no., (5) Integer: Permutation (swap)
mutation real value	0.1	Mutation real value in the GA
mutation probability	0.05	Mutation probability in the GA: 0.05 $\equiv$ 5 %
select type	1	Selection of solution in the GA: (1) proportional selection, (2) stochastic universal selection, (3) tournament selection, (4) probabilistic tournament selection
duplicates allowed	1	Determines if duplicate solutions are allowed: Yes (1) or No (0)
logfile		Root name for all output files created during the GA calculation. The names of the output files are based on a concatenation of the root name, date and time stamp, numbering and/or labeling and the appropriate extension. The program is stopped with error if no <i>logfile</i> is found.
create Excel file	0	Controls if an ASCII output file is created that contains the final parameter values of the calculation in a format that can be imported in Excel in the so called "comma delimited" format. Yes (1) or No (0) The extension of the file is <i>.csv</i> , which is recognized directly by Excel.
create Ini file	0	Controls if a new input file for the ga-program is created with the final parameter values of the calculation in the proper format, see Sect. IV B. Yes (1) or No (0) The extension of the file is <i>.ini</i> .
output EA evaluation	0	Determines the output of the list of parameter values during the GA evaluation: Default (0) every 20 generation, (1) all values for every generation, any other number > 0 is allowed. Note: If the number of generation is < 70 at least every 10 generations the list of parameters is printed.

### C. Specific definitions of the options of the 1D *EGdeconv* program

Note: when file and directory paths are required, absolute paths start with a '/'. Relative paths should start without '/' and are relative to the directory of the inputfile.

TABLE III: Description of the keyword strings that can be used in the input file of *EGdeconv* for fitting 1D NMR spectra.

Keyword string <i>EGdeconv</i>	Default	Description
echo timing	0	Indicate position of echo maximum. Without dot the number indicates the index of the Hahn echo maximum in the time data. When a dot is included, time in units complementary to the experimental frequency axis (MHz to $\mu$ s, kHz to ms) can be used. See also section III A 4 h.
fit order parameter	0	If set to 'CuPt' or 'CuAu' (no quotes) the order parameter model will be used to determine relative intensities, see also section III A 3. Parameter 20 of the <b>first</b> parameter set will be used as $S$ , while parameter 45 will be used for $x$ . Equivalent parameters in the other parameter sets will not be used. See also description of parameter 199 to link each site to the correct probability.
fit type	real	Fit either the real part of the spectrum or the absolute (squared) value, by setting keyword to 'real', 'abs' or 'power'.
frequency axis direction	neg2pos	The left-to-right direction of the frequency axis for the calculated spectrum, values 'pos2neg' and 'neg2pos'.
libraryfile extension	fid	The extension of the library files. Do not include the dot. Extension does not have to indicate the type of data, fid or spectrum, this will be determined by the file header.
print distributions	1	Print file containing three-column ( $\eta$ , $C_q$ , $f(\eta, C_q)$ ) ascii for every site with an (extended) Czjzek distribution, see also section III A 4 c. File will appear in the log directory with logname and date prefix followed by 'distr', see also section VID. If set to 0 nothing will be printed.
print treefile	1	Print file containing information on all $\eta$ and $C_q$ values present in library or single crystal grid. File will appear in the log directory with logname and date prefix followed by 'treefile', see also section VID. If set to 0 nothing will be printed.
species<X>		<X> is an integer number starting from one. This keyword is used to refer to libraries and the value of X provides the relation to the parameter set via the 'speciesX' parameter. The keyword is used in following format: species<X>= <lib. dir.> [ $\eta_{start}$ $\eta_{end}$ $\eta_{step}$ $C_{qstart}$ $C_{qend}$ $C_{qstep}$ ] $C_q$ needs to be in MHz. In the powder case the parameter ranges indicate the library files that will be read into memory; if no ranges are supplied, the whole library is read. As long as X has a continuous range, it is allowed to have more species defined than are actually used in the parameter sets, excess species will not be loaded.
species<X> points		The number behind this keyword indicates the amount of points that need to be read in from a library file or a single crystal simulation fid. If the number of points is less than the amount of points of the experimental spectrum zero-filling will be used. In the case of Hahn echoes, the value of this parameter will be used to determine if the whole echo or half the echo should be used in the processing, see section III A 4 h.
type of output	0	Type of additional time and frequency domain output. 0 only the regular EA output, 1 imaginary part of the spectrum, 2 real and imaginary part of the fid, 3 gives both 1 and 2. First column is always time/frequency and other columns are sum and than individual subspectra/subfid's.

#### D. Specific definitions of the options of the MQMAS *EGdeconv* program

TABLE IV: Description of the keyword strings that can be used in the input file of *EGdeconv* for fitting MQMAS spectra.

Keyword string <i>EGdeconv</i>	Default	Description
NDISTRS	10	The number of points used in the chemical shift distribution. See also section III A 6 k.
NDISTRQ	10	The number of points of $\sigma$ used in the (extended) Czjzek distribution. See also section III A 6 g.
NDISTRE	10	The number of points of $\eta$ used in the (extended) Czjzek distribution. See also section III A 6 g.
MQ	-	The multiple quantum coherence used in the MQMAS experiment. This value has to be set.
SPINQNUMBERX2	-	Two times the spin quantum number ( $2I$ ). This value has to be set.
B0	-	The rf-frequency (MHz). This value has to be set.
NUREFHZ	B0	The reference frequency (MHz).
NUREFPPM	0	The reference ppm value.
CHENG	10	Cheng number used in the powder average, see section III A 6 e.
NGAMMA	10	The number of integration steps over the $\gamma$ angle, see section III A 6 e.
EFF	0	Set to 1 to enable the efficiency calculation, see section III A 6 b. This is only available for 3QMAS on spin $I=3/2$ and $I=5/2$ .
B1FIELD	-	The rf-field strength (kHz). This value is only needed if the efficiency calculation is enabled.
PW1	-	The first pulse length ( $\mu s$ ). This value is only needed if the efficiency calculation is enabled.
PW2	-	The second pulse length ( $\mu s$ ). This value is only needed if the efficiency calculation is enabled.
ZFILTER	1	Set this values to 0 if a two pulse MQMAS was used, see section III A 6 b. This value is only used if the efficiency calculation is enabled.
BINARY	0	Print the output in a binary format, see section III A 6 f.
SHEAR	0	Shearing factor that is applied to the simulated spectra in the F1 dimension.

## VI. RUNNING THE PROGRAM *EGDECONV*

### A. Introduction

The calculations are performed under a linux operating system. Details on the installation can be found in section X. To speed up the computing time, the *EGdeconv* program allows parallel processing with very high efficiency; so the use of cluster machines looks the ideal solution.

It is recommended to install and run the program on a multi-processor/multi-core machine. For example a machine with 2 dual core processors; this makes 4 cores available. In general this kind of hardware has already enough power to run the *EGdeconv* program with acceptable turn around times. More cores will of course reduce the wall clock time.

The command is:

$$EGdeconv [-n N [-h] [-v]] project.ini \quad (82)$$

The file *project.ini* controls the EA-calculation and contains the appropriate *keyword strings* as defined in Sect. IV B. Here  $N$  is the number of cores to be used. The maximum on the nodes is determined by the available hardware The default value is 2. The flag *-v* displays the current version of the *EGdeconv* program.

*EGdeconv* will run interactively on the background. If the command *EGdeconv* is given without any flags and and *.ini-file* or with the flag *-h* the list of options and flags is given.

The environmental parameter *path* should contain a reference to the directory *.../EGdeconv2.70/bin*.

### B. Stopping a EGdeconv-job

To stop a running *EGdeconv*-job and still create the normal output files<sup>11</sup> (See Sect. VID) the following command can be given from the directory where the *EGdeconv*-job has started:

$$ga-stopEA-dateANDtime \quad (83)$$

Here *dateANDtime* is the date and time stamp of the running *EGdeconv*-job. You can find the latter extensions to *ga-stopEA* by looking to the *logfile name* extension. Usually it will be enough to check the startup directory of the *EGdeconv*-job. In this directory the file *ga-stopEA-dateANDtime* can be found.

Additionally, during run time a file is created from which its name indicates the progress of the EA calculation: *CurrentRunningXX-n<sub>g</sub>-dateANDtime*. Here  $n_g$  is the number of the currently calculated generation and *dateANDtime* is defined above. The string *XX* stands for *GA*, *DR2* or *CMA-ES* for a genetic al-

gorithm, a derandomized DR2 or a Covariance Matrix Adaptation Evolution Strategy fit, respectively.

### C. Resume a EGdeconv-job

The *EGdeconv* program has a *resume* option. This option is **only** available for a CMA-ES fit. It allows to resume the CMA-ES evaluation at the point where it was stopped, either by user interference (*ga-stop*, see VIB), hardware or other problems. It is even possible to resume the CMA-ES at any of the previously completed generations.<sup>12</sup>

This is of particular interest if the program is used to optimize an experimental setup. In such a case the optimization can easily be interrupted by all kind of experimental failures.

This option is controlled by the keyword strings: *create CMA-ES resume file* and *CMA-ES resume file*.

If the option is used a proper *.ini* file is created to resume the CMA-ES fit at the point it was interrupted or stopped. It might be necessary to increase the keyword string *number of generations* in the *.ini* file. This process can be repeated, while each time the next *.ini* file is created.

### D. Output generated by EGdeconv.

The default output by *EGdeconv* are a number of files showing the progress of the genetic algorithm and the final result. The file names are constructed from the *base name* defined by the *keyword string logfile* (see Table II) followed by a unique identification based on the date and time of the start of the program<sup>13</sup>. This defines the *rootname*. Using this *rootname* several file names are constructed to distinguish the various outputs:

- Test output: *rootname-T.txt*. The output file if only a testrun has to be carried out. *keyword string: test calculation=1*
- Normal output: *rootname-n.txt*. The normal output file for the  $n^{th}$ -seed. *keyword string: test calculation=0* and one or more seed values: *random seeds=i j k .....*
- Summarizing text file: *rootname-final.txt*.
- Summarizing Excel type file: *rootname-final.csv*.
- New input file based on the final parameter values: *rootname-n.ini*. for the  $n^{th}$ -seed.
- A  $(1 + N_{seeds})$  column ASCII file containing the costfunction values for the different generations and seeds:*rootname-CostFunction-.dat*.

<sup>11</sup> Preferable this (software)-way of stopping a *EGdeconv*-job is used, rather than using the Unix/Linux command: *kill job-number*. The latter command stops the *EGdeconv*-job in the middle of its execution leaving all outputs unflushed.

<sup>12</sup> This feature is currently undocumented. If you need it, please contact W. Leo Meerts.

<sup>13</sup> In this way all file names are unique and accidental overwriting of previous results is impossible.

$N_{seeds}$  is the number of seeds. The Cost function values for each iteration step are written to this file. The first column contains the the generation number and the next  $N_{seeds}$  column(s) the corresponding Cost function value. This can be used to plot the Cost function value as function of the generation.

- An output file in the form of a  $(3 + \text{nset} + [1/2/3/4])$ -column ASCII array with the frequency in the first column, the intensities of the calculated and experimental spectra in the second and third column, while the next nset-columns contain the sole spectrum of the i-th parameter set. The file name is: *rootname-OutputSpectrum.answ*<sup>14</sup>. This file is created by default. Its creation can be suppressed by the *keyword string: create Full spectrum file=0*.

If the *keyword string: remove smoothed background* is set to a value  $> 0$ , three more columns are present, which represent the calculated and experimental spectra before removal of the backgrounds and the smoothed calculated spectrum (See also Sect.IID).

If the weight function option is activated (Sec. IIC) the weight function is added as last column.

- A three column ASCII file with file name: *rootname-n-fitness.dat* is created if *fitness type=4* (Sharing) and *sharing file=1*. The first column contains the generation number, the second column the distance between population within this generation and the third column the number of populations in the generation for that distance.

A 3D plot can easily be created from this file with proper graphical software, for example Origin.

- CMA-ES summary files: If a CMA Evolutionary Strategy fit is performed (*CMA-ES algorithm=1*). A series of files of the type *rootname-outcmaes\*.dat* and a MATLAB file *rootname\_PlotCMA\_ES.m*<sup>15</sup> are created. By running MATLAB with the file *rootname\_PlotCMA\_ES.m* as input a number of plots are created showing the progress of the CMA-ES fit.
- Summary of library files in memory or gridsize for single crystal calculations, use keyword *print treefile: rootname-treefile.txt*.
- Three-column  $(\eta, C_q, f(\eta, C_q))$  ascii file per site  $N$  with a distribution, use keyword *print distributions: rootname-distr-siteN.txt*.

- Complex part of the calculated spectrum in the format of Normal output, use keyword *type of output: rootname-ImSpectrum.txt*.
- Theoretical fit of the spectrum in the format of Normal output, but with two-column real-complex per fit, use keyword *type of output: rootname-ImSpectrum.txt*.
- Error messages file, reporting compensated errors: *rootname-MESSAGES.txt*.

## VII. 1D-NMR EXAMPLES

In this section three applications of the *EGdeconv* program are presented. They represent the program's versatility, by the use of second-party software, to handle different experimental conditions, for example: static or magic angle spinning (MAS), free induction decay or Hahn echo acquisition etc. Plus lineshape analysis with or without (extended) Czjzek distributions. Input files for these examples are provided with the installation package.

### A. <sup>27</sup>Al MAS NMR spectrum of Yttrium Aluminium Garnet

Fig. 15 shows a hard pulse <sup>27</sup>Al MAS NMR spectrum of YAG (Y<sub>3</sub>Al<sub>5</sub>O<sub>12</sub>) at 15kHz MAS. YAG has a spine type structure with Al occurring in a tetrahedral and an octahedral coordination in 3:2 ratio. The spectrum of powdered YAG was measured and fully characterised by Massiot *et al.* in 1990 [35]. Combination of MAS conditions and the well-defined quadrupolar interactions, indicated by the sharp lineshapes, make this a suitable benchmark study for *EGdeconv*. Massiot *et al.* established interaction constants for the tetrahedrally ( $\sim 6\text{MHz}$ ,  $\eta=0$ ) and octahedrally ( $\sim 0.6\text{MHz}$ ,  $\eta=0$ ) coordinated sites with a relative abundance ( $\sim 3:2$ ). Given the results mentioned in the caption of Fig. 15, *EGdeconv* efficiently reaffirmed these findings.

### B. Oxidation of Sodium Aluminium Hydride (NaAlH<sub>4</sub>)

Fig. 16 shows the results of fitting <sup>27</sup>Al MAS spectra of NaAlH<sub>4</sub>'s oxidation process as a function of time. All information provided here is based on [36] that contains references and more details about the data analysis. The NaAlH<sub>4</sub> sites in the material have a well defined lattice without significant variations. When exposed to air, structurally disordered tetrahedrally and octahedrally coordinated oxide sites will form. From Multiple Quantum MAS experiments it was determined that the quadrupolar interaction of the tetrahedral (Al(IV)) site is distributed. Therefore the lineshape of this site was fitted with a Czjzek distribution that reproduces the asymmetric lineshape, as shown in Fig. 16. The octahedral site (Al(VI)) contained in addition to a distribution in quadrupolar parameters also line broadening

<sup>14</sup> The file extension *.answ* is chosen to make the connection with the spectrum viewing program Answer42 (Sect. XI). Automatic opening (under Windows) of the spectra files can be achieved by coupling the file type *.answ* to Answer42.

<sup>15</sup> In order for MATLAB to be able to process the *.m* file the symbols "-" in *rootname* are replaced by "\_"

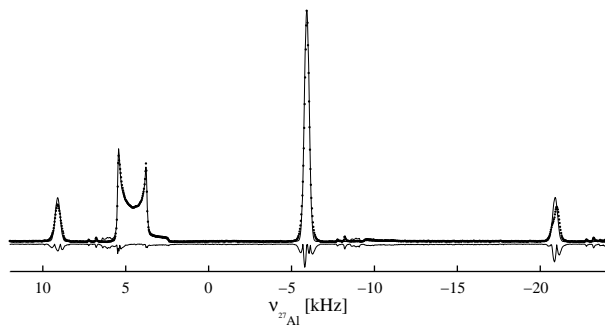


FIG. 15: Experimental (dotted line) and theoretical (continuous line)  $^{27}\text{Al}$  spectrum of YAG together with the negative absolute value of the difference. The spectrum consists of a broad resonance of the tetrahedral site in addition to two spinning side bands and the central transition of the octahedral site. Five independent fits with  $[\eta-C_q]$  grids  $[0.00:0.01:0.10-5.0:0.1:7.0]$  and  $[0.00:0.01:0.10-0.3:0.1:1.0]$  obtained  $C_q$  values of 6.1 and 0.6 MHz and  $\eta$  values of 0.04 and  $0.07 \pm 0.02$ , respectively, only standard deviations larger than the library stepsize are noted. The relative intensity of the respective sites, was determined  $1:(0.69 \pm 0.01)$ .

from dipole-dipole couplings, this influence was mimicked by additional line broadening. As the  $\text{NaAlH}_4$  has a well-defined crystal structure this line was described without a distribution in quadrupole parameters. Additional Gaussian line broadening accounted for the dipolar broadening as the spectra were obtained without proton decoupling.

The aim of the spectral deconvolution here was to quantify the relative intensities of the various lines as a function of time. This allows to monitor the interconversion of the different components during oxidation as shown in Fig. 16, and discussed in detail in [36].

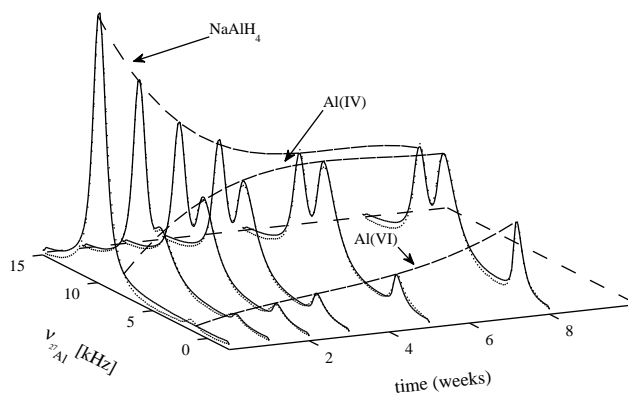


FIG. 16: Oxidation of  $\text{NaAlH}_4$  as a function of time. Experimental (dotted line) and theoretical spectra (continuous line) are overlaid and shown as a function of time. The dashed lines roughly connecting the peaks are qualitative guides to the eye. They emphasize the conversion of  $\text{NaAlH}_4$  in tetrahedral  $\text{Al(IV)}$  and octahedral  $\text{Al(VI)}$  oxide sites. For the fits a  $[\eta-C_q]$  grid of  $[0.00:0.05:1.00-0.5:0.5:15.0]$  was used.

### C. Static Hahn echo spectrum of Aluminium Gallium Arsenide ( $\text{Al}_x\text{Ga}_{1-x}\text{As}$ )

Fig. 17 shows a  $^{75}\text{As}$  Hahn echo spectrum of powdered  $\text{Al}_{0.489}\text{Ga}_{0.511}\text{As}$ , one form of the widely used and well-characterised  $\text{AlGaAs}$  semi-conductor. Solid-state NMR was used to investigate the possible order or disorder in the occupation of the Al/Ga sites. The analysis of this spectrum is published in [5] and was the main driving force behind the development of *EGdeconv*.

With a spectral width of 2.5 MHz and sites with small and large quadrupolar coupling constants present, this example is particularly dependent on accurately simulated excitation efficiencies. As the figure shows, the spectrum can be decomposed in terms of tetrahedral aluminium-gallium coordinations of the arsenic atoms. The centered narrow lines correspond to the  $[\text{Al}_4]$  and  $[\text{Ga}_4]$  coordinations of the arsenic atoms, these sites have such a high symmetry that the *efg* due to the first atomic coordination shell is absent, leading to a small quadrupolar interaction strength caused by disorder in the higher coordination spheres. With respect to excitation efficiencies it should be noted that for these sites both central and satellite transitions are of importance. Due to a negligible *efg* contribution of the first atom coordination shell, these sites are specific probes of disorder in higher order coordination spheres. This justifies the use of a Czjzek distribution to calculate the lineshapes. The symmetry of the remaining three sites causes a large *efg* contribution from the first coordination shell. The excitation efficiency of these sites is mainly selective to the central transition. To fit these sites an extended Czjzek distribution was applied, for which the  $\sigma$  parameter was set to the average of the  $[\text{Al}_4]$  and  $[\text{Ga}_4]$  Czjzek distribution, since the latter are probes for long range disorder. The last step in the analysis is the use of an order parameter to relate the site intensities. The resulting fit, shown in [5] and available in the installation package, matches less than that of Fig. 17, but provides a value of  $S$  close to zero, indicating the expected structural disorder. An in-depth study of this disorder supported by DFT calculations reproducing the fitted distributions is described in [5].

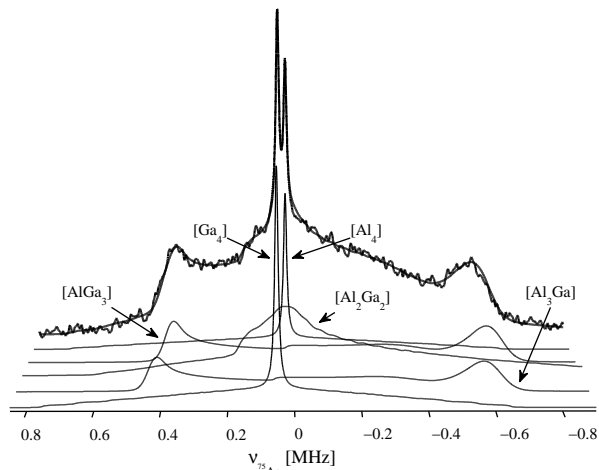


FIG. 17: Experimental (thick dotted line) and theoretical (thick continuous line) Hahn echo spectrum of  $\text{Al}_{0.489}\text{Ga}_{0.511}\text{As}$ , deconvoluted in terms of subspectra corresponding to tetrahedral aluminium-gallium sites. The  $[\text{Al}_4]$  and  $[\text{Ga}_4]$  sites were fitted with a Czjzek distribution, the remaining sites required an extended Czjzek distribution to account for the ordered *efg* contribution in the first coordination sphere. Subspectrum intensities were fitted with respect to one subspectrum, but independently. Three libraries with  $[\eta-C_q]$  grids of  $[0.00:0.05:1.00-0.0:0.2:3.2]$  ( $[\text{Al}_4]$  and  $[\text{Ga}_4]$ ),  $[0.00:0.01:0.10-31.0:0.1:35.0]$  ( $[\text{Al}_3\text{Ga}]$  and  $[\text{AlGa}_3]$ ) and  $[0.85:0.01:1.00-28.0:0.1:35.0]$  ( $[\text{Al}_2\text{Ga}_2]$ ) were used.

## VIII. MQMAS EXAMPLES

In this section some applications of the MQMAS *EGdeconv* program are presented. Examples of fitting MQMAS with and without Czjzek distribution are given. Input files for these examples are provided with the installation package.

### A. Aluminium Alkoxides

In Fig. 18 the  $^{27}\text{Al}$  sheared 3QMAS data fits of three different alkoxides coordinated to aluminium are pre-

sented. All spectra were measured and analysed in ref. [37] on the basis of one-dimensional and 3QMAS data. Table V summarises our resulting parameter values from the 3QMAS fit, alongside their values from the previous study [37]. In all spectra the sum-projections in both direct and indirect dimensions of the two-dimensional experimental and simulated data are shown. A least-squares [38] between these projections was used as a measure of fit-quality for the CMA-ES algorithm. The obvious benefit of the sum-projections is that the intensity of the spinning side bands is added to that of the central transition in the  $\delta_2$  dimension, which can be more readily simulated. The  $\delta_1$  dimension sum-projection provides in these spectra an extra feature to determine the isotropic position of the central transition during the fitting.

The spectrum of aluminium ethoxide, Fig. 18(a), that corresponds to a five-coordinated aluminium, shows a good match between the sum-projections in the ( $\delta_2$ ) dimension. The main difference at high  $\delta_2$  values in the figure is caused by an  $\text{Al}_2\text{O}_3$  impurity. For the case of aluminium butoxide, Fig. 18(b), with a tetrahedral aluminium surroundings the fitted spectrum also closely resembles the sum-projection of the experimental data. The line shape at  $\delta_1 \approx 250$  ppm is an aliased spinning side band. Table V shows that we obtained interaction parameters that agree with the findings of [37] on the basis of MQMAS and one-dimensional data for both spectra.

The spectrum of aluminium isopropoxide in Fig. 18(c) reveals a tetrahedral (IV) and octahedral (VI) site that posed the opportunity to determine relative site abundances. From the fit of these relative intensities we obtained the ratio (IV):(VI)=1:2.7 which was determined 1:3 by [37] on the basis of one-dimensional data. It should be noted that there is a negligible difference between the fitted relative intensities, whether the analytical excitation efficiencies are accounted for or not. This means that with the current pulse widths both sites are excited with similar efficiency. The difference for  $C_q$  of the octahedral site (VI) is due to a lack of quadrupolar features in the sharp line.

### B. Rubidium-nitrate

Rubidium-nitrate ( $\text{RbNO}_3$ ), as the guinea pig for MQMAS experiments, has three well-defined sites with: equal abundance, similar quadrupolar coupling constants, but different asymmetry parameters as given in the most-right column in table VI.

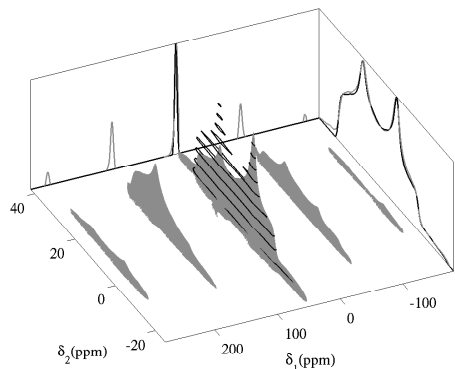
To further test the analytical excitation efficiency model and distinguish it from the ideal excitation model, two  $^{87}\text{Rb}$  3QMAS spectra of  $\text{RbNO}_3$  were measured. One spectrum was obtained using relatively short pulses and strong rf-field strength and represents the ideal excitation case (Fig. 19(a)). The second spectrum was obtained with complementary settings to evoke

non-ideal excitation (Fig. 19(b) and 19(c)). By measuring the spectrum at a magnetic field of 300 MHz proton frequency we also encountered overlapping lines as an additional challenge for the fitting.

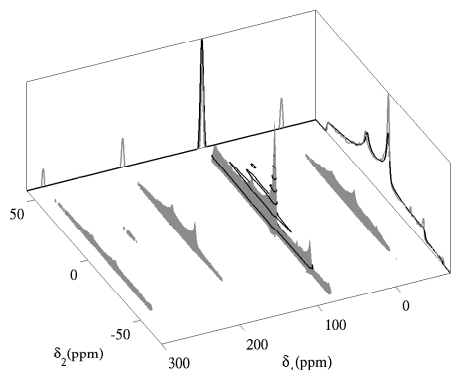
A fit of the ideal excitation spectrum using either model results in Fig. 19(a). The least squares difference between the two-dimensional experimental and simulated spectra was used [38] as the quality measure for the CMA-ES algorithm. Table VI shows that the fitted parameter values are nearly equal to those from literature [39]. This is a confirmation that the assumption of ideal excitation is valid in this case.

Fitting the second spectrum resulted in a different fit for each model as shown in Figs 19(b) and 19(c).

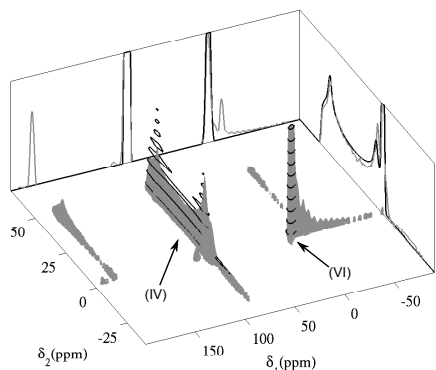




(a)aluminium ethoxide



(b)aluminium butoxide



(c)aluminium isopropoxide

FIG. 18:  $^{27}\text{Al}$  z-filtered 3QMAS spectra ( $\omega_1/2\pi=310$  kHz,  $\tau_1=1.3$   $\mu\text{s}$ ,  $\tau_2=0.45$   $\mu\text{s}$ , and  $\tau_3=4$   $\mu\text{s}$  with  $\omega_1/2\pi=21$  kHz), experiment (grey) and fit (black), for three aluminium alkoxides [37]. The spectra are normalised based on their integrals, and their sum-projections are scaled with their integrals to be visible on the intensity scale of the spectrum. The isopropoxide spectrum is zoomed in on its tetrahedral (IV) site, this is the reason for the cut of the octahedral (VI) site peak. Data was sheared after the fit.

Incorporation of the (analytical) excitation efficiencies provides a better description of especially line shapes I and II. This is reflected in a difference in  $\eta$  parameter for both fits as shown in table VI. Particularly line shape II is different from that in Fig. 19(a). The intensity of the left peak appears to move into the right shoulder of

	$\delta_{iso}$ [ppm]		$C_q$ [MHz]		$\eta$	
ethoxide	35.5	<i>35.5</i>	9.58	<i>9.65</i>	0.37	<i>0.39</i>
butoxide	48.5	<i>48.5</i>	13.03	<i>13.14</i>	0.64	<i>0.61</i>
isoprop. (IV)	60.0	<i>61.5</i>	12.20	<i>12.37</i>	0.14	<i>0.14</i>
(VI)	0.8	<i>2.5</i>	0.6	<i>1.9</i>	0.0	<i>0.0</i>

TABLE V: Fitted interaction parameters for the alkoxide spectra in Fig. 18, values from [37] are in italics. In the fit of isopropoxide,  $\eta$  of the octahedral site was fixed at 0.

site	pars	(Fig. 19(a)) ideal	(Fig. 19(b)) eff. off	(Fig. 19(c)) eff. on	[39]
I	$C_q$ [MHz]	1.75	1.77	1.74	1.77
	$\eta$	0.54	0.60	0.55	0.54
II	$C_q$ [MHz]	1.72	1.75	1.73	1.72
	$\eta$	0.19	0.27	0.16	0.19
	$I$	0.97	0.90	1.06	1.00
	integral	1.14	0.93	0.93	–
III	$C_q$ [MHz]	1.99	1.99	1.99	2.01
	$\eta$	0.89	0.93	0.91	0.90
	$I$	1.00	0.85	0.99	1.00
	integral	0.93	0.87	0.87	–

TABLE VI: Fitted parameter values of the  $^{87}\text{Rb}$  3QMAS spectra of rubidium-nitrate in Fig. 19.  $I$ , as indicated in Fig. 12, stands for relative intensity, in this case with respect to site I. For comparison the parameter values obtained by [39] at 29.3°C are given in the most-right column.

the line.

With respect to the prediction of relative site abundances, it follows from table VI that the excitation model fit comes closest to the actual ratio. Especially if we compare the sum of the intensities (due to the overlap of the line shapes), we obtain the ratios 1:2.05 and 1:1.75. The ideal excitation model appears to follow the trend of the line shape integrals, which leads to an underestimation of the intensities of lines II and III.

### C. Yttrium-sialon glass

Sialon glasses doped with rare-earth metals are interesting for photo-electric applications. Glasses are disordered systems which makes quantification in terms of interaction constants tedious. The benefit of having MQMAS data is that it aids in effectively separating chemical shift and quadrupolar interaction contributions to the spectrum, that would otherwise not be visible in the one-dimensional data.

In practice this means that, after shearing, a distribution in chemical shift will become visible as a broadened line in the F1 dimension. For a distribution in quadrupolar interaction parameters, the line shape is influenced in both dimensions by the (isotropic) quadrupolar induced shift, however, only the F2 dimension is influenced by the anisotropic part of the interaction. The two-dimensional spectrum thereby provides enough features to distinguish and fit interaction parameter distributions, and potentially relative site

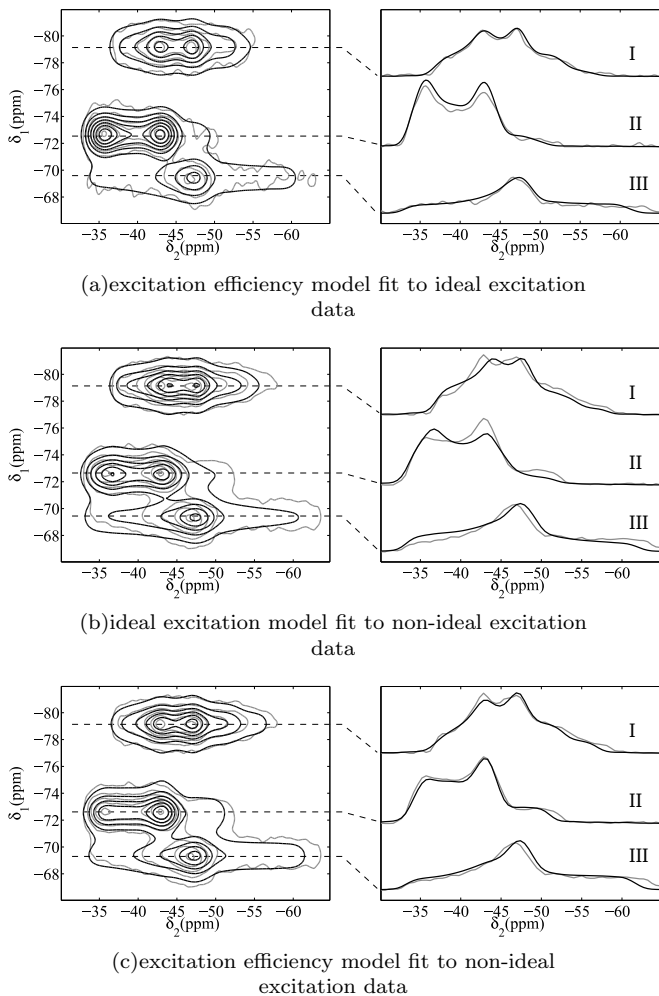


FIG. 19: Simulations (black) of the z-filtered  $^{87}\text{Rb}$  3QMAS spectrum of rubidium-nitrate (grey). Integrals of the spectra are set equal, contour lines are drawn at the same intensities. (a) ideal excitation spectrum ( $\omega_1/2\pi=211$  kHz,  $\tau_1=2.6$   $\mu\text{s}$ ,  $\tau_2=0.9$   $\mu\text{s}$ ) fitted with excitation efficiency model. (b) and (c) the non-ideal excitation spectrum ( $\omega_1/2\pi=108$  kHz,  $\tau_1=5.2$   $\mu\text{s}$ ,  $\tau_2=2.6$   $\mu\text{s}$ ) fitted with and without analytical excitation efficiencies.

abundances as discussed in the rubidium-nitrate case study.

Fig. 20(a) shows the fitted  $^{27}\text{Al}$  3QMAS spectrum of an yttrium-sialon glass [40] (named after the atomic constituents: silicium, aluminium, oxygen and nitrogen), using the inner product cost function [38] between the two-dimensional spectra as the quality measure. In both dimensions of the spectrum we see broadened lines which indicates both chemical shift and quadrupolar interactions are distributed. In the two-dimensional plane, the lines are separated which is useful to avoid ambiguity in the fit. Note the one-pulse spectrum of the sialon material in Fig. 20(b) does not provide these insights.

The three distinct lines of the chemical sites in the spectrum are probably broadened due to structural variations in bond angles and lengths deviating from: tetrahedral, trigonal bipyramidal and octahedral coordinations in the material [41]. Fig. 20(a) shows

how a combination of the Czjzek distribution for the quadrupolar interaction parameters and a Gaussian distribution for the chemical shift can approximate the line shapes in the spectrum. Only at the basis of the largest peak a mismatch is visible between the more triangular basis of the experimental data and the more rounded simulated spectrum. Table VII presents the fitted parameter values. It should be noted that a fit with either ideal or excitation efficiency model did not change the results. Together with the fact that the intensities match with the line integrals, we may conclude ideal excitation of the material.

The fitted interaction parameters of the 3QMAS were used to fit the one-pulse spectrum of the yttrium-sialon glass. This led to a good and stable fit as shown in Fig. 20(b). Note that the asymmetric line shape of the sub-spectra is caused by the Czjzek distribution. Table VII shows the fitted relative intensities of the lines. There is a significant difference for the MQMAS and 1D fit with respect to the tetrahedral and trigonal bipyramidal site intensity ratios. This difference we explain by the different effective  $T_1 > 2s$  of the two sites in combination with the cycle delay of 4 seconds that was used to limit the time needed for the 3QMAS experiment. The one-pulse data should be quantitative since only a short pulse, and therefore a small flip-angle, is used.

coordination	$\delta_{iso}$ [ppm]	$\sigma_s$ [ppm]	$\sigma$ [MHz]	2D integral	2D $I$	1D $I$
tetrahedral	62.6	17.0	4.03	1.00	1.00	1.00
trigon. bip.	27.7	13.6	4.04	0.21	0.20	0.30
octahedral	-2.4	11.4	3.65	0.03	0.04	0.05

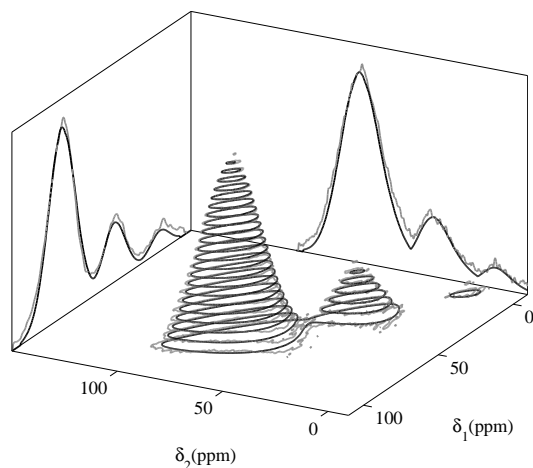
TABLE VII: The parameter values from the fit of the yttrium-sialon glass (Fig. 20). Parameters  $\delta_{iso}$ ,  $\sigma_s$  are the mean and width of the Gaussian chemical shift distribution,  $\sigma$  the width of the Czjzek distribution (power factor  $d=5$ ), 2D integrals of the lines in Fig. 20(a),  $I$  the relative intensities of Figs 20(a) and 20(b).

## IX. HOW TO OPTIMIZE THE *EGDECONV* FITS

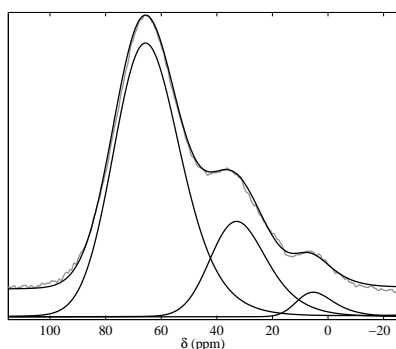
### A. Tips for running a succesfull *EGdeconv* fit.

To perform a succesfull fit using the EA-algorithms used in this program the following tips might be useful.

- Although it is possible to fit a constant background signal which might be present in the experimental spectrum, it is stongly recommended to process the experimental spectrum in such a way that the constant background equals zero. A fit of the parameter that described the background **should be prevented**.
- From experience is is found that it is not a good idea to fit the linewidth together with the other spectroscopic parameters. The reason is that the



(a) yttrium-sialon glass 3QMAS



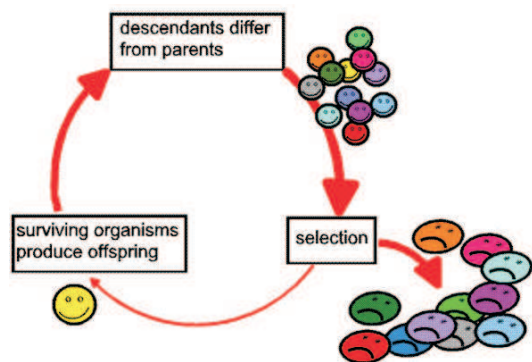
(b) yttrium-sialon glass one-pulse spectrum

FIG. 20: Experimental data is shown in grey, simulated results are in black. (a) Fitted  $^{27}\text{Al}$  3QMAS spectrum of an yttrium-sialon glass with skyline projections. Three chemically distinct sites are shown that correspond to variations in, from bottom to top: tetrahedral, trigonal bipyramidal and octahedral surroundings. A Czjzek and Gaussian distribution is used for the quadrupolar interaction and chemical shift, respectively. Fitted parameter values are given in table VII. (b) Fit of the one-pulse spectrum. Czjzek distribution parameter values from the 3QMAS fit are used (table VII). The subspectra that constitute the simulated spectrum are plotted below the base line.

fit tends to yield too large linewidths, in particular with not fully resolved spectra.

The best procedure is as follows:

1. Estimate the linewidth as good as possible from your experiment. Take for the Gaussian contribution the value determined by the experiment (remaining Doppler width) and add a Lorentzian contribution so that the total linewidth matches that of the observed spectrum. Don't forget to choose Voigt. You can also take only Gaussian in this step, with a larger proper value to match the observed spectrum
2. Fit all other parameters and keep the linewidths fixed
3. If you want to fit the linewidth: Keep all



other parameters fixed to the best values found above, set the Gaussian component to the experimental value (see a) and fit the Lorentzian component only.

- The relative broadening of the spectra, introduced by the overlap function  $w(r)$ , described in Sect. , critically determines the ability of the GA to converge to the global minimum and also the speed of convergence. The smoothing of the error landscape allows to sense regions far from the minimum. In the first step, the overlap function  $w(r)$  should be chosen relatively broad;  $\Delta w \approx 15 - 20$  times the line widths of an individual rovibronic line in the spectrum ( $\Delta_{lw}$ ). In this way, a first set of molecular parameters is obtained, which still has to be refined. This is done by decreasing  $\Delta w$  and narrowing the limits of the parameter space to be searched in the fit. Decreasing  $\Delta w$  improves the accuracy of the molecular parameters obtained from the fit, while narrowing the parameter space leads to an improved sampling in the region of the minimum. This of course is the critical step in the procedure. Too strong narrowing of the parameter space leads to premature convergence of the fit – with a high probability into a local minimum. In Sect. IX B a discussion of the *best* procedure is discussed.
- The default values of the parameters in the *EGdeconv* program that control the GA-process (Table II) are obtained from the GA-meta optimization discussed in Sect. IX C. It is suggested to start your *EGdeconv* fit with these default values i.e. placing none of them in your *ini*-file, except the few ones that are required.

### B. Influence of the width ( $\Delta w$ ) of the overlap function $w(r)$ on the convergence of the EA

We performed a fit [2] of a synthetic spectrum which consists of two overlapping sub-spectra. A single temperature model for the calculation of the intensities has been used. The maximum  $J$ -value used in the computation of the cost function for this spectrum is 22. Five GA calculations were performed stopped after 500 generations, each started with a different randomly generated initial set of parameters for different values of

$\Delta w/\Delta l_w$ . The convergence of the fit using different  $\Delta w/\Delta l_w$  is shown in Figure 21.

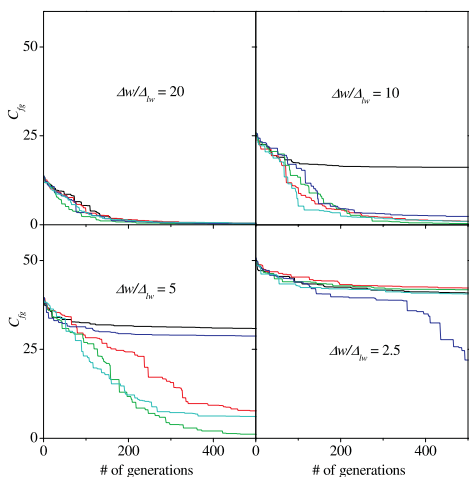


FIG. 21: Dependence of the convergence of five fits of a synthetic spectrum which consists of two overlapping sub-spectra with different starting populations on the ratio  $\Delta w/\Delta l_w$ .

A ratio  $\Delta w/\Delta l_w = 20$  leads to convergence for all five initial seeds into the same minimum. The minimum found is the global one. Nevertheless the deviations of the fitted parameters from the best values were still quite large, due to the large value of  $\Delta w$ , which leads to broad minima at the cost surface. As seen in Figure 21 smaller values of  $\Delta w$  may lead to convergence into other (local) minima of the cost surface. With  $\Delta w/\Delta l_w$  of 10 still three of the fits converge to the global minimum, with a ratio of 5 only one fit converges to the global minimum, and with 2.5 none of the fits converges to the global minimum.

In the next step the fit has to be refined with a limited parameter space centered around the best fit. and with successively smaller values of  $\Delta w$ . We could not establish a hard criterion for the reduction of the parameter space. A successful reduction depends critically on the quality of the first series of fits. On the other hand the parameter space cannot always be reduced by the same factor for each of the parameters. As a rule of thumb we found that the parameter limits can be reduced to one tenth of the initial range. This reduction depends on the quality (signal-to-noise) of the spectrum, and has to be checked carefully after each successive step.

It was found that the fit with the reduced line width ratio and the reduced parameter space is already quite close to the "real" values. It could further be improved by decreasing  $\Delta w/\Delta l_w$  and the parameter search space in an iterative manner until convergence for the molecular parameters is reached.

To generate a more realistic spectrum we added a randomly Gaussian distributed noise to the synthetic spectrum, resulting in a S/N of 10:1 for the strongest lines. The GA algorithm performed equally well in this case. The deviations of the parameters from the true values is similarly small as for the "perfect" spectrum without noise.

### C. Meta-optimization of the internal GA parameters

The need to optimize the internal parameters of the GA (meta-optimization) for alliven problem has been discussed to be a major drawback of this method [42]. The speed and convergence of GA algorithms depend on the data representation (binary or real), the crossover type (one-point, two-point, uniform), the size of the starting population, the rate of elitism and the mutation probability. Several other factors, which also influence the performance of the GA fit have been kept fixed. They will be discussed shortly. The crossover probability is chosen as 85 %. Selection of the best solutions is performed via a tournament selection, which means that a random subset of the chromosomes is taken and within each subset the chromosomes are selected by their cost. Duplicates within one generation are allowed for.

The "natural" choice for a genetic algorithm is a binary representation of the data. It has been discussed, that a direct representation of the parameters as real type data is advantageous if the type of data to be fitted is real. All tests performed by us [2] show that the binary encoding of the parameters leads to a smaller cost and converges more rapidly, compared to real type representation. For the real type data encoding a number of runs even don't reach the global minimum. A change of the encryption depth for the binary representation from 10 bit to 20 bit virtually does not change the performance of the GA. A Gray code [43] is used throughout the present investigations in order to ensure a Hamming distance of one [44]. Comparing uniform and two-point crossover a clear advantage of the two point cross-over regarding speed of convergence and fitness of the solutions was found. Just in the case of real type encoding the uniform crossover forces more runs of the GA into the global minimum than the two-point.

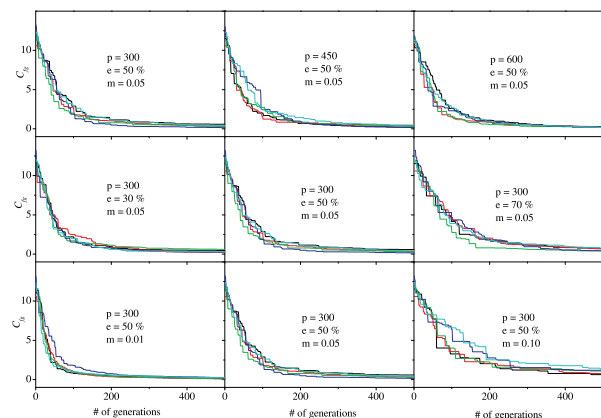


FIG. 22: Convergence of the GA as a function of population size ( $p$ ), rate of elitism ( $e$ ) and mutation probability ( $m$ ). The cost function  $C_{fg}$  is plotted on the vertical axis.

The first row of Figure 22 shows the convergence of five fits with different starting populations as function of the number of generations for different sizes of the starting population using  $\Delta w/\Delta l_w = 20$  (cf. Sect. The elitism was kept at 50 % and mutation probability at

0.05 in these calculations. For a population of 300, the best value of the cost function is 0.15, the mean of five runs using different starting populations is 0.35, and the cost function drops below 0.5 after 372 generations. The larger population of 450 has a slightly better mean of 0.30, and drops below 0.5 after 324 generations. The largest population we investigated contained 600 chromosomes and resulted in a mean of the cost function of 0.21 and dropped below 0.5 after 278 generations. The better performance regarding the convergence as function of the number of generations for the larger populations is more than compensated by the increasing CPU-time for increasing populations. One run for a population of 300 takes 7 minutes, for a population of 450 takes 14 minutes, and for 600 already 17 minutes using a dual processor PC with two Pentium 2.8 GHz processors. All subsequent computing times are for this configuration.

The variation of elitism between 30 % and 70 % is shown in the second row of Figure 22. The size of the population is kept fixed at 300 for these fits. An elitism of 30 % means, that the best 30 % of one generation are passed unchanged to the next generation. Elitism helps to prevent good solutions to be lost from one generation to the next. As can be inferred from Figure 22 a fit with an elitism of 30 % converges more rapidly than with 50 % and with 70 %. Nevertheless, the mean cost function of five runs for an elitism of 50% is slightly better. For an elitism of 70 % the mean cost function never drops below 0.5, because too many bad solutions are kept. The performance regarding CPU-time is accordingly. An elitism of 30 % is the most time consuming (12 minutes) compared to 50 % (7 minutes) and 70 % (4 minutes).

The third row of Figure 22 shows the variation of convergence with the mutation probability. Population size is kept fixed at 300 and elitism at 50%. Mutation is introduced to allow for a thorough exploration of the whole cost surface and prevents the algorithm from prematurely converging into a local minimum. With a mutation probability of 1 % the mean value of the cost function is 0.15, and the cost function drops below 0.5 after 204 generations. With a 5% mutation rate the mean of the cost function increases to 0.35 (drops below 0.5 after 372 generations), while for 10 % mutation rate the mean cost function value is 1.00. Although all five runs converged into the same global minimum for 1, 5, and 10 % mutation probability, the mutation probability of 1 % performed best, considering mean cost function and speed of convergence. However, one has to be careful not to underestimate the risk of converging to a local minimum due to bad exploration of the cost surface. The CPU-time as function of the mutation probability virtually doesn't change (7 minutes for rates of 0.01, 0.05, and 0.1).

To conclude, the population size should not exceed 300, due to the bad time performance of larger populations, which is not compensated by a much better convergence of the algorithm. An elitism of 30 % is advantageous regarding the convergence, but also very time consuming. The best trade-off between time- and convergence performance is found at 50 % elitism. A mutation probability of only 1 % leads to a very fast

convergence, with very exactly determined parameters. Nevertheless, in cases where many local minima at the cost surface are present such a low mutation rate might lead into a local minimum. Reduction of the value of this parameter has therefore to be performed with great care.

#### D. Comparison between the genetic and the derandomized-ES DR2 algorithms.

The number of optimizing parameters for the derandomized-ES DR2 algorithm (DA) are quite a bit smaller than of the genetic algorithm (GA). A comparison has been made between the two algorithms using the same synthetic spectrum as in Sect. IX B.

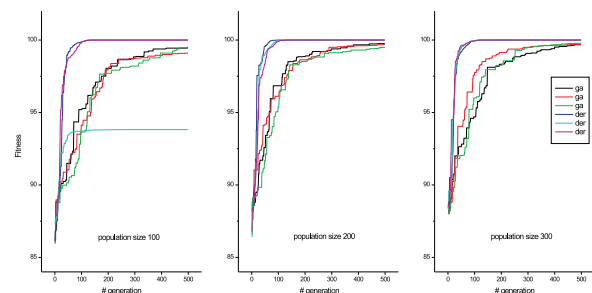


FIG. 23: Comparison of three randomly started fits comparing the genetic (ga) and derandomized-ES DR2 (der) algorithms as function of the size of the population (ga) or offspring (der). In this figure the *fitness* function Eq. ( 8 ) is shown, which has a maximum values of 100.

The result of this comparison<sup>16</sup> is shown in Fig. 23. This figure clearly shows that for this particular kind of spectrum the ES is faster converging and yields parameter values that are closer to the *real* ones.

## X. INSTALLATION OF THE *EGDECONV* SOFTWARE

### A. Introduction

Copyright©2002/2013 W. Leo Meerts and Michael Schmitt. The usage of the *EGdeconv* program is free if a reference is made to

**Dennis L. A. G. Grimminck, Ben J. W. Polman, Arno P. M. Kentgens, and W. Leo Meerts. EASY-GOING deconvolution: Combining accurate simulation and evolutionary algorithms for fast deconvolution of solid-state quadrupolar NMR spectra. J. Magn. Resonance 211 (2011) 114-120. doi: <http://dx.doi.org/10.1016/j.jmr.2011.04.009>.**

<sup>16</sup> In these calculations the other meta-parameter for the ES is: *Maximum age* = 10, while for the GA these parameters are: *Crossover Probability* = 85%, *Mutation Probability* = 5% and *Elitism* = 50%.

or

Dennis L. A. G. Grimminck, Bas van Meerten, Margriet H. W. Verkuijlen, Ernst R. H. van Eck, W Leo Meerts, and Arno P. M. Kentgens. **EASY-GOING deconvolution: Automated MQMAS NMR spectrum analysis based on a model with analytical crystallite excitation efficiencies.** *J. Magn.Resonance* **228** (2013) 116-124. doi: <http://dx.doi.org/10.1016/j.jmr.2012.12.012>.

for fitting MQMAS

The above papers can be found on our website [egdeconv.science.ru.nl](http://egdeconv.science.ru.nl).

Separate license agreements and copyrights are applicable to the genetic algorithm package PgaPack, the multiprocessor software MPI(CH) and the spectra display and manipulation program *answer42*, in case of usage of these packages. We refer here to the particular conditions for use that comes with those packages. By installing and/or using the *EGdeconv* program and accompanying software, you accept the license conditions.

### 1. Prerequisites

The *EGdeconv* program requires minimal a 64 bit PC with a Linux 64 bit operation system installed. Multi processors and multi cores are supported for paralel computing speeding up the calculations.<sup>17</sup> The current *EGdeconv* program will ONLY be supplied for ubuntu 10.04 LTS and 12.04 LTS, running on a single or multi core processor. No guarantee can be given for other linux versions.

Since the script files that come with the installation of the *EGdeconv* program as well as those that run the EGdeconv-program are based on the use of the linux c-shell (csh), it is needed to have at least the c-shell installed on your system.

Depending on your configuration it might be necessary to install some programs or add/change some links. We have added the necessary fix for ubuntu-linux in *ga-ubuntu-fix-libs*. This will be done automatically at the first installation under ubuntu.

In addition the *EGdeconv* program uses some openmpi-libraries, which you might have to install with:

```
sudo apt-get install libopenmpi1.3
```

<sup>17</sup> The original 1D program *EGdeconv* version 2.60 has been tested on the linux operation systems Feudora 9, 10 and 11, Ubuntu 10.04 (LTS) as well as openSUZE 11.3. Most likely the program will also run under other Linux dialects and/or different versions. However, the authors only support Ubuntu 10.04 LTS.

## B. Installation

### 1. First time installation of the *EGdeconv* package

If you have limited access rights and/or want to perform an easy and fast installation perform the following steps to install the *EGdeconv* program:

1. You need the file *EGdeconvPackage2.70*. Download this file from our website [egdeconv.science.ru.nl](http://egdeconv.science.ru.nl). You should place this file in a chosen directory (for example your HOME-directory) and install it with:
2. `chmod 700 EGdeconvPackage2.70`  
`./EGdeconvPackage2.70`
3. Adjust the *path* of your system. Check *README-EGdeconv.installed* after the installation for details
4. A test for the 1D-option:  
`cd EGdeconv2.70/solidstateNMR/1D/YAG`
5. Run a test with: `"EGdeconv YAG.ini"`
6. A test for the MQMASS-option:  
`cd EGdeconv2.70/solidstateNMR/MQMAS/rbno3`
7. Run a test with: `"EGdeconv rbno3_ideal.ini"`

The installation of future updates is very simple and is described in the next section.

## C. Updating the *EGdeconv* program

If a new version of *EGdeconv* has become available an update can easily be made as follows. Download the latest version of the *EGdeconv* Package and repeat the above installation steps.

## XI. MISCELLANEOUS PROGRAM

### A. *answer42*

The full *EGdeconv* Package also contains the program *answer42*, developed in Düsseldorf<sup>18</sup>. This is a Windows-based program, compatible with the 'wine'<sup>19</sup> application under Linux, that allows fast and flexible viewing of the spectra, in particular of the *rootname-output.spectrum.answ* files discussed in Sect. VID. Besides the viewing option it can perform many other mathematical operations on the data such as Auto- and Crosscorrelation, Fast Fourier transformations and much more.

<sup>18</sup> Copyright 1995-2005 Daniel Spangenberg and Johannes Spangenberg

<sup>19</sup> Use `"wine C:/Program Files/Answer42/Answer42.exe"` in your file associations to link the *answ* files to the Answer42 program.

## XII. REFERENCE TO EXAMPLES OF THE SUCCESS OF *EGDECONV* AUTOMATIC ASSIGNMENTS

A large number of relevant examples of successful applications of the *ga*-program on high resolution rotationally resolved spectra are presented in a review paper on this topic by Meerts and Schmitt [1]. For further reading we refer to this paper. A copy is included in the *EGdeconv2.70/doc* directory of the *EGdeconv*-installation (Meerts\_IntRevPhysChem25\_353.pdf). The first automatic analysis of a NMR liquid crystal spectrum using the *EGdeconv*-program can be found in Ref. [3]. Two contributed chapters to books are also available [45, 46] and they are included in the *EGdeconv2.70/doc* directory (Schmitt\_Handbook\_2009.pdf and Burnell\_WordScientificReview\_2009.pdf).

## APPENDICES

### APPENDIX A: CONDITIONS FOR THE W-MATRIX TO BE POSITIVE DEFINITE

In this appendix we show that the real or complex matrix  $\mathbf{W}$  with matrix elements  $W_{ij} = w(r_i - r_j)$  is positive definite if  $w(r)$  can be written as the inverse

Fourier transform of a positive function  $\tilde{w}(t)$ . Also see Ref. [2]. Let

$$w(r) = \int_{-\infty}^{\infty} \tilde{w}(t) e^{2\pi i r t} dt \quad (\text{A-1})$$

then

$$\begin{aligned} \mathbf{x}^\dagger \mathbf{W} \mathbf{x} &= \sum_{i,j} x_i^* W_{ij} x_j \\ &= \sum_{i,j} x_i^* \int_{-\infty}^{\infty} \tilde{w}(t) \exp[2\pi i(r_i - r_j)t] dt x_j \\ &= \int_{-\infty}^{\infty} \left| \sum_i x_i \exp[-2\pi i r_i t] \right|^2 \tilde{w}(t) dt. \end{aligned} \quad (\text{A-2})$$

Hence, if  $\tilde{w}(t) > 0$  then  $\mathbf{x}^\dagger \mathbf{W} \mathbf{x} > 0$  for any vector  $\mathbf{x}$  and so  $\mathbf{W}$  is positive definite. If  $\tilde{w}(t)$  is zero for certain values of  $t$ , the integral is still positive. Note: One should read in

The Fourier transform of  $w(r)$  defined in Eq. (13) is

$$\tilde{w}(t) = \int_{-\infty}^{\infty} e^{-2\pi i r t} w(r) dr = \frac{\Delta w}{2} \operatorname{sinc}^2\left(\frac{\Delta w}{2} \pi t\right), \quad (\text{A-3})$$

where  $\operatorname{sinc}(x) = \sin(x)/x$  is, so  $\tilde{w}(t) \geq 0$ .

- 
- [1] W. L. Meerts and M. Schmitt, *Int. Rev. Phys. Chem.*, 2006, **25**, 353–406.
- [2] W. L. Meerts, M. Schmitt, and G. Groenenboom, *Can. J. Chem.*, 2004, **82**, 804–819.
- [3] W. L. Meerts, C. A. de Lange, C. J. C. Weber, and E. E. Burnell, *Chem. Phys. Letters*, 2007, **441**, 342–364.
- [4] A. Abragam, *The principles of nuclear magnetism*, The international series of monographs on physics, Oxford University Press, 1967.
- [5] P. J. Knijn, P. J. M. van Bentum, E. R. H. van Eck, C. Fang, D. L. A. G. Grimminck, R. A. de Groot, R. W. A. Havenith, M. Marsman, W. L. Meerts, G. A. de Wijs, and A. P. M. Kentgens, *Phys. Chem. Chem. Phys.*, 2010, **12**, 11517–11535.
- [6] D. Massiot, F. Fayon, M. Capron, I. King, S. Le Calvé, B. Alonso, J. Durand, B. Bujoli, Z. Gan, and G. Hoatson, *Magn. Reson. Chem.*, 2002, **40**(1), 70–76.
- [7] T. F. Kemp and M. E. Smith, *Solid State Nucl. Magn. Reson.*, 2009, **35**(4), 243–252.
- [8] M. Bak, J. Rasmussen, and N. Nielsen, *J. Magn. Reson.*, 2000, **147**, 296–330.
- [9] M. Veshtort and R. G. Griffin, *J. Magn. Reson.*, 2006, **178**(2), 248–282.
- [10] M. Wächtler, A. Schweitzer, T. Gutmann, H. Breitcke, and G. Buntkowsky, *Solid State Nucl. Magn. Reson.*, 2009, **35**(1), 37–48.
- [11] G. Czjzek, J. Fink, F. Götz, H. Schmidt, J. M. D. Coey, J. P. Rebouillat, and A. Liénard, *Phys. Rev. B*, 1981, **23**(6), 2513–2530.
- [12] G. Le Caër and R. A. Brand, *J. Phys.: Condens. Matter*, 1998, **10**, 10715–10774.
- [13] G. Le Caër, B. Bureau, and D. Massiot, *J. Phys.: Condens. Matter*, 2010, **22**(6), 065402.
- [14] R. Tycko, G. Dabbagh, S. R. Kurtz, and J. P. Goral, *Phys. Rev. B*, 1992, **45**(23), 13452–13457.
- [15] J. A. Hageman, R. Wehrens, R. de Gelder, W. L. Meerts, and L. M. C. Buydens, *J. Chem. Phys.*, 2000, **113**, 7955–7962.
- [16] J. H. Holland, *Adaption in Natural and Artificial Systems*, MI: The University of Michigan Press, Ann Arbor, 1975.
- [17] D. E. Goldberg, *Genetic Algorithms in search, optimisation and machine learning*, Addison-Wesley, Reading Massachusetts, 1989.
- [18] I. Rechenberg, *Evolutionstrategie - Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, Frommann-Holzboog, Stuttgart, 1973.
- [19] N. Hansen and A. Ostermeier, *Evolutionary Computation*, 2001, **9**(2), 159–195.
- [20] A. Ostermeier, A. Gawelcyk, and N. Hansen in *Parallel Problem Solving from Nature, PPSN III*, ed. Y. Davidor, H.-P. Schwefel, and R. Männer; Springer, Berlin/Heidelberg, 1994.
- [21] N. Hansen and S. Kern In ed. X. Yao et al., *Parallel Problem Solving from Nature PPSN VIII*, Vol. 3242 of *LNCIS*, pp. 282–291. Springer, Berlin/Heidelberg, 2004.
- [22] A. Auger and N. Hansen In *Proceedings of the IEEE Congress on Evolutionary Computation, CEC 2005*, pp. 1769–1776, 2005.
- [23] M. J. Duer, *Introduction to Solid-State NMR Spectroscopy*, Blackwell, 2004.
- [24] T. S. Kuan, T. F. Kuech, W. I. Wang, and E. L. Wilkie, *Phys. Rev. Lett.*, 1985, **54**(3), 201–204.
- [25] B. D. Cullity, *Elements of X-ray diffraction*, Addison-Wesley, Reading, MA, second ed., 1978.
- [26] C. Degen, M. Tomaselli, B. Meier, M. Voncken, and A. Kentgens, *Phys. Rev. B*, 2004, **69**, 193303(1–4).
- [27] T. Vosegaard, A. Malmendal, and N. C. Nielsen, *Journal Monatshefte für Chemie / Chemical Monthly*, 2002, **133**(12), 1555–1574.

- [28] A. P. M. Kentgens, *Geoderma*, 1997, **80**(3–4), 271–306.
- [29] L. Frydman and J. S. Harwood, *J. Am. Chem. Soc.*, 1995, **117**(19), 5367–5368.
- [30] P. P. Man, *Mol. Phys.*, 1993, **78**(2), 307–318.
- [31] G. M. Muha, *J. Magn. Reson.*, 1983, **53**(1), 85–102.
- [32] R. B. Creel and D. A. Drabold, *J. Mol. Struct.*, 1983, **111**(0), 85–90.
- [33] J. Amoureux, C. Fernandez, and S. Steuernagel, November, 1996, **123**(1), 116–118.
- [34] V. B. Cheng, J. Henry H. Suzukawa, and M. Wolfsberg, *J. Chem. Phys.*, 1973, **59**(8), 3992–3999.
- [35] D. Massiot, C. Bessada, J. P. Coutures, and F. Taulelle, *J. Magn. Reson.*, 1990, **90**, 231–242.
- [36] M. H. W. Verkuijlen, R. de Gelder, P. J. M. van Bentum, and A. P. M. Kentgens, *J. Phys. Chem. C*, 2011.
- [37] A. Abraham, R. Prins, J. A. van Bokhoven, E. R. H. van Eck, and A. P. M. Kentgens, *J. Phys. Chem. B*, 2006, **110**(13), 6553–6560.
- [38] D. L. A. G. Grimminck, B. van Meerten, and W. L. Meerts, Description and features of the program egdeconv: Automated assign and fit programs for powder nmr spectra using evolutionary algorithms <http://egdeconv.science.ru.nl>, 2012.
- [39] J. Skibsted and H. J. Jakobsen, *J. Phys. Chem. A*, 1999, **103**(40), 7958–7971.
- [40] D. de Graaf, S. le Rol, H. Hintzen, L. le Gendre, and G. de With, *Journal of the European Ceramic Society*, 2006, **26**(13), 2497–2501.
- [41] P. Kempgens, R. K. Harris, Z. Yu, and D. P. Thompson, *J. Mater. Chem.*, 2001, **11**, 2507–2512.
- [42] R. Wehrens, E. Pretsch, and L. M. C. Buydens, *Anal. Chim. Acta*, 1999, **388**, 265.
- [43] F. Gray, *Pulse Code Communication*, U. S. Patent 2 632 058, March 17, 1953.
- [44] ed. J. Schaffer; *A study of control parameters affecting online performance of genetic algorithms for function optimization.*, Proceedings of the Third International Conference on Genetic Algorithms, San Mateo, CA, 1989. Morgan Kaufmann.
- [45] E. E. Burnell, C. A. de Lange, and W. L. Meerts, in *Nuclear Magnetic Resonance Spectroscopy of Liquid Crystals*, ed. R. Dong, World Scientific Review, 2009; chapter 1.
- [46] M. Schmitt and W. L. Meerts, in *Handbook of High Resolution Spectroscopy*, ed. M. Quack and F. Merkt, John Wiley and sons, London, 2009.