

Optimal control in NMR spectroscopy: Numerical implementation in SIMPSON

Zdeněk Tošner^{a,b,*}, Thomas Vosegaard^a, Cindie Kehlet^a, Navin Khaneja^c, Steffen J. Glaser^d, Niels Chr. Nielsen^{a,*}

^a Center for Insoluble Protein Structures (inSPIN), Interdisciplinary Nanoscience Center (iNANO) and Department of Chemistry, University of Aarhus, Langelandsgade 140, DK-8000 Aarhus C, Denmark

^b Department of Chemistry, Faculty of Science, Charles University in Prague, Hlavova 8, CZ-128 43, Czech Republic

^c Division of Applied Sciences, Harvard University, Cambridge, MA 02138, USA

^d Department of Chemistry, Technische Universität München, 85747 Garching, Germany

ARTICLE INFO

Article history:

Received 13 October 2008

Revised 19 November 2008

Available online 8 December 2008

Keywords:

Optimal control

NMR spectroscopy

Rf inhomogeneity

Resonance offsets

Effective Hamiltonian

Composite pulses

Quantum information processing

Strongly modulating pulses

ABSTRACT

We present the implementation of optimal control into the open source simulation package SIMPSON for development and optimization of nuclear magnetic resonance experiments for a wide range of applications, including liquid- and solid-state NMR, magnetic resonance imaging, quantum computation, and combinations between NMR and other spectroscopies. Optimal control enables efficient optimization of NMR experiments in terms of amplitudes, phases, offsets etc. for hundreds-to-thousands of pulses to fully exploit the experimentally available high degree of freedom in pulse sequences to combat variations/limitations in experimental or spin system parameters or design experiments with specific properties typically not covered as easily by standard design procedures. This facilitates straightforward optimization of experiments under consideration of rf and static field inhomogeneities, limitations in available or desired rf field strengths (e.g., for reduction of sample heating), spread in resonance offsets or coupling parameters, variations in spin systems etc. to meet the actual experimental conditions as close as possible. The paper provides a brief account on the relevant theory and in particular the computational interface relevant for optimization of state-to-state transfer (on the density operator level) and the effective Hamiltonian on the level of propagators along with several representative examples within liquid- and solid-state NMR spectroscopy.

© 2008 Elsevier Inc. All rights reserved.

1. Introduction

In the recent few years optimal control [1–3] has found its way into nuclear magnetic resonance as a powerful tool for efficient design and optimization of experiments for applications in imaging [4–7], liquid-state NMR [8–13], solid-state NMR [14–20], quantum computation [21–26], and dynamic nuclear polarization/electron-nuclear interactions [27–31]. This method, originally being introduced for optimizations in engineering and economy, is very attractive for optimization problems dealing with complex internal Hamiltonians and a large number of external manipulation parameters. Opposed to the small space of variables typically associated with standard non-linear optimization methods [32], optimal control is capable of handling hundreds-to-thousands of free variables within an optimization and provides an optimal solution much fas-

* Corresponding authors. Address: Center for Insoluble Protein Structures (inSPIN), Interdisciplinary Nanoscience Center (iNANO) and Department of Chemistry, University of Aarhus, Langelandsgade 140, DK-8000 Aarhus C, Denmark. Fax: +45 8 6196199 (N. C. Nielsen).

E-mail addresses: tosner@natur.cuni.cz (Z. Tošner), ncn@chem.au.dk (N.C. Nielsen).

ter than standard optimization procedures in such cases. This opens up interesting new possibilities for performing experiment design which takes into account all typically available information on (i) the relevant nuclear spin systems (size, typical nuclear spin interaction parameters, and variations in these), (ii) the available experimental manipulations (external magnetic field, rf fields, limitations and inhomogeneities in these), and (iii) relevant experimental conditions (sample spinning, dynamics).

Using optimal control, it is possible to extend the numerical design and optimization of NMR methods [33,34] to fully exploit the available experimental degrees of freedom to reach the optimal experiment, as for example expressed in terms of universal bounds on spin dynamics [35–39]. The outcome of numerical optimizations with optimal control may not only provide optimum experiments for direct applications, but may also provide new insight into, e.g., the maximal possible transfer efficiencies, which then may be used (i) as an evaluation of whether there is room for improvements of state-of-the-art experiments and (ii) to provide new inspiration to alternative design strategies, for example based on average (or effective) Hamiltonian theory [40–43]. As an example of the latter, it became clear from optimal control design of heteronuclear coherence transfer schemes in solid-state NMR [14]

that optimal control can increase the efficiency of so-called γ -encoded dipolar recoupling experiments [44] by compensating more efficiently for spread in another crystallite orientation angle (β) leading to the concept of COMpensation for Beta (COMB) composite refocusing [45]. This led to an analytical single-spin optimization strategy for improved dipolar recoupling, which in a subsequent phase was feeded back to fast optimal control single-spin numerical optimizations of two-spin dipolar recoupling experiments [20]. Another option is to use optimal control directly as a means to analytically design optimal experiments [46–53].

In this paper, we address software implementation of optimal control for easy conduction of experiment optimizations for specific applications, to improve robustness towards the parameters that are essential for the intended experiments or simply to obtain a test on the maximum possible efficiency of an experiment to compare with existing methods. The paper serves as a description of (and manual for) an extension of the powerful and widely used simulation package SIMPSON [54] to include optimal control applications [14]. The combined software is released as open source software in a new version of SIMPSON. The paper is organized as follows. First the relevant theory is discussed briefly with reference to the original literature for a more detailed account. Subsequently, we introduce the novel features relevant for SIMPSON optimal control calculations, and finally provide a few relevant examples illustrating these features and demonstrating some of the flexibility of an optimal control numerical tool for NMR experiment design and optimization.

2. Theory

The spin dynamics of NMR experiments is typically expressed in terms of the evolution of density operators as expressed by the Liouville-von-Neuman equation

$$\dot{\rho}(t) = -i[H(t), \rho(t)] \quad (1)$$

addressing cases without relaxation. In general terms, the Zeeman frame Hamiltonian may be described as

$$H(t) = H_{\text{int}}(t) + H_{\text{rf}}(t), \quad (2)$$

$$H_{\text{rf}}(t) = \sum_i \omega_{\text{rf}}^{ix}(t) I_{ix} + \omega_{\text{rf}}^{iy}(t) I_{iy}, \quad (3)$$

$$H_{\text{int}}(t) = H_{\text{CS}}(t) + H_J(t) + H_D(t) + H_Q(t). \quad (4)$$

$H_{\text{rf}}(t)$ expresses external manipulations in terms of rf irradiation, here expressed as a summation over x - and y -phase contributions for the involved spins (or spin species) through continuous functions in time t . In numerical calculations the pulses are for practical reasons represented in digitized form as a train of pulses appropriately reflecting the time variation. $H_{\text{int}}(t)$ collects contributions from internal nuclear spin interactions, including chemical shifts, scalar (J) couplings, dipole–dipole couplings, and quadrupolar couplings. The internal Hamiltonian for the various interactions may typically be cast in the form of a Fourier series

$$H_\lambda(t) = \sum_{m=-2}^2 \omega_\lambda^{(m)} e^{im\omega_r t} O_\lambda, \quad (5)$$

where time dependencies, e.g., due to sample spinning with angular frequency ω_r , is expressed through the exponentials. Expressions for the angular frequencies of the isotropic and anisotropic (i.e., orientation dependent) internal nuclear spin interactions $\omega_\lambda^{(m)}$ as well as the associated spin operators O_λ for typical solid- and liquid-state NMR applications can be found in Ref. [54].

For a generalized pulse sequence consisting of a train of N pulses acting consecutively on the initial spin density operator $\rho(0)$ from time 0 to time $t_N = T$, the density operator at the end of the pulse sequence takes the form

$$\rho(T) = U_N(t_N, t_{N-1}) \dots U_2(t_2, t_1) U_1(t_1, 0) \rho(0) U_1^\dagger(t_1, 0) U_2^\dagger(t_2, t_1) \dots U_N^\dagger(t_N, t_{N-1}) \quad (6)$$

with each propagator defined as $U(t_{k+1}, t_k) = \hat{T} \exp\{-i \int_{t_k}^{t_{k+1}} H(t) dt\}$ using the Hamiltonian in Eq. (2) and the Dyson time-ordering operator \hat{T} . The aim of the optimization is to adjust the external manipulation parameters, which are the instantaneous rf field amplitudes $\omega_{\text{rf}}^{ix}(t_k)$ and $\omega_{\text{rf}}^{iy}(t_k)$ in Eq. (3), to provide the best performance of the experiment. This amounts to systematic variation of the amplitudes in Fig. 1 to obtain the optimal experiment.

The performance may be evaluated as the highest transfer efficiency transferring a given initial state operator, e.g., $\rho(0) = A$ to a desired destination spin operator C as expressed by the density operator at time T , $\rho(T) = c_{\text{max}} C + B$, where the coefficient c_{max} is maximized and B contains all residual operator terms. Henceforth, we will refer to this situation as *state-to-state transfer*. In cases of transfer between *Hermitian operators* this amounts to optimization of the overlap between C and the transformed operator $\rho(T) = U\rho(0)U^\dagger$ as expressed in terms of a standard inner product between the operators

$$\Phi_{\text{fin}} = \langle C | \rho(T) \rangle = \text{Tr}\{C^\dagger U\rho(0)U^\dagger\}. \quad (7)$$

In cases of transfer between *non-Hermitian operators*, the transfer function defined in Eq. (7) is complex valued in general and can not be used directly. In this case there are several possible replacements for Φ_{fin} , for example just evaluating the real part or the norm squared of the transfer function. For implementation within SIMPSON, we have chosen the latter case:

$$\Phi_{\text{fin}} = |\langle C | \rho(T) \rangle|^2 = \left| \text{Tr}\{C^\dagger U\rho(0)U^\dagger\} \right|^2. \quad (8)$$

We note that this cost function leads to a multitude of solutions, as discussed previously in the context of unitary bounds on spin dynamics [36,38].

For other applications, it is relevant to optimize the *effective Hamiltonian* H_{eff} (or the effective propagator U_{eff}) to a desired effective Hamiltonian H_D (or the desired propagator U_D) which amounts to optimization of

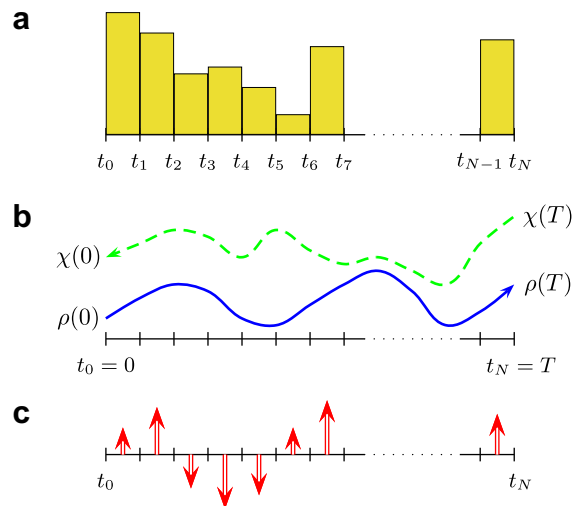


Fig. 1. Schematic illustration of optimal control design of multiple-pulse experiments. (a) The pulse sequence consists of many short pulses with variable amplitude and phase. (b) The effect of the pulse sequence is monitored by forward and backward evolution of initial state $\rho(0)$ and the target state $\chi(T)$, respectively. (c) Utilizing the fact that the two paths are different, corrections to pulse parameters are calculated in terms of a gradient.

$$\text{Tr}\{U_D^\dagger U_{\text{eff}}\} = \text{Tr}\{e^{iH_0 T} e^{-iH_{\text{eff}} T}\}. \quad (9)$$

In practice, the total phase of a propagator is irrelevant and it proves convenient to release the reproduction of the desired propagator to all variations including a phase φ , and optimize

$$\Phi_{\text{fin}} = \left| |e^{-i\varphi} U_D^\dagger U_{\text{eff}}| \right|^2 = \left| \text{Tr}\{U_D^\dagger U_{\text{eff}}\} \right|^2 \quad (10)$$

for all values of φ [8,17].

A third frequently encountered situation could be optimization of transfer into a given state while efficiently suppressing transfer to other specified states or optimizing a certain effective Hamiltonian while efficiently suppressing others. A typical example for the first case could be optimization of spin-state selective NMR experiments where only one of two possible spin-states (e.g., $I_z S^+$) is desired while the other should be suppressed (e.g., $I_\beta S^+$) [55–58].

In optimal control, the term, Φ_{fin} in Eqs. (7)–(10) is typically referred to as the “final cost” (explaining the subscript “fin”) in an overall cost function that may contain other terms as well. One of the additional terms can be a “running cost”, which may prevent achieving the optimum because the running cost is too high. In NMR applications this could, for example, be a penalty limiting the overall consumption of rf power (energy) throughout the pulse sequence as expressed through the term

$$\Phi_{\text{rf}} = \lambda \sum_i \sum_{q=x, y} \sum_{k=0}^{N-1} \left(\omega_{\text{rf}}^{iq}(t_k) \right)^2 \Delta t_k, \quad (11)$$

where $\Delta t_k = t_{k+1} - t_k$. We note that the weight of the running cost may be regulated through the factor λ . Obviously, this running cost is only one of many options for penalties on the control fields and their time variation.

In recent NMR applications optimal control optimization have been conducted using the gradient based optimization procedure GRAPE (Gradient Ascent Pulse Engineering) [8]. This method establishes a gradient which in an iterative fashion is used to change the amplitudes of the control fields to improve the pulse sequence to optimum performance.

For the *state-to-state optimization*, the gradient is calculated on the basis of the “forward-transformed” density operator in Eq. (6) and a “back-transformed” operator (cf. Fig. 1b)

$$\chi(t_j) = U_{j+1}^\dagger(t_{j+1}, t_j) \dots U_N^\dagger(t_N, t_{N-1}) C U_N(t_N, t_{N-1}) \dots U_{j+1}(t_{j+1}, t_j), \quad (12)$$

reusing the same propagators as for the “forward-transformed” density operator in Eq. (6). Equipped with Eqs. (6)–(8), it is possible to describe a gradient which in an iterative fashion can be used to modify the rf field amplitudes $\omega_{\text{rf}}^{ix}(t_j)$ and $\omega_{\text{rf}}^{iy}(t_j)$ associated with the spin operators I_{ix} and I_{iy} , respectively, for each pulse in the pulse sequence. For transfer between Hermitian operators, the gradient for the final cost takes the form [8]

$$\frac{\partial \Phi_{\text{fin}}}{\partial \omega_{\text{rf}}^{iq}(t_j)} = \langle \chi(t_j) | -i \Delta t_j [I_{iq}, \rho(t_j)] \rangle, \quad (13)$$

corresponding to an iterative update of the rf fields by

$$\omega_{\text{rf}}^{iq}(t_j) \rightarrow \omega_{\text{rf}}^{iq}(t_j) + \varepsilon \text{Tr}\{\chi^\dagger(t_j) i \Delta t_j [I_{iq}, \rho(t_j)]\}, \quad (14)$$

where $q = x$ or y and ε is a small real number. In the case of non-Hermitian operators, the gradient of the final cost defined in Eq. (8) amounts to

$$\frac{\partial \Phi_{\text{fin}}}{\partial \omega_{\text{rf}}^{iq}(t_j)} = 2 \text{Im}\{\langle \chi(t_j) | \Delta t_j [I_{iq}, \rho(t_j)] \rangle \langle \rho(t_j) | \chi(t_j) \rangle\}. \quad (15)$$

Graphically the origin of the gradient is illustrated in Fig. 1b. Under the influence of an initially guessed pulse sequence, the density operator evolves to some state different from the target state. In

the same time we can back-propagate the target state $\chi(T) = C$ to the beginning of the sequence. In the optimal situation, the two paths would be identical and their difference is used (via the gradient) to generate corrections to all rf pulses (cf. Fig. 1c) at once.

The gradient for the running cost is

$$\frac{\partial \Phi_{\text{rf}}}{\partial \omega_{\text{rf}}^{iq}(t_j)} = 2 \lambda \omega_{\text{rf}}^{iq}(t_j) \Delta t_j, \quad (16)$$

which may be combined directly with the gradient in Eq. (14) (or Eq. (15)) according to formula for the total cost

$$\Phi_{\text{tot}} = \Phi_{\text{fin}} - \Phi_{\text{rf}} \quad (17)$$

if considered relevant.

For optimization of the *effective Hamiltonian*, the gradient expressions take their origin in the cost function in Eq. (10). As it is demonstrated elsewhere (see Refs. [8,17]), the gradient may be derived as

$$\frac{\partial \Phi_{\text{fin}}}{\partial \omega_{\text{rf}}^{iq}(t_j)} = 2 \Delta t_j \text{Im}\{\langle U_D | U_N \dots U_{j+1} I_{iq} U_j \dots U_1 \rangle \langle U | U_D \rangle\} \quad (18)$$

with $U = U_N \dots U_2 U_1$ where we for simplicity did not specify the timing limits of the propagators following the systems in Eq. (6).

We note that normalization is an issue of importance when interpreting the efficiency of transfer between given initial and desired destination operators. Different approaches has been proposed in literature, all with advantages and disadvantages in different contexts. The simplest approach will be to normalize the initial and destination operators, e.g., $C_n = C/||C||$, prior to optimization [38]. We note that we, to avoid confusion with previous releases of SIMPSON and other types of calculations, did not implement this normalization into the software described below.

Optimal control may alternatively be conducted using the so-called Krotov approach [3]. We will not describe this approach in detail in this paper, but refer the reader to a recent paper by Maximov et al. [27]. In most cases the GRAPE and Krotov algorithms behave similarly with respect to results. The Krotov approach is typically faster for the initial part of the optimization and uses fewer operations per iteration, which may prove particularly important for optimizations involving larger spin systems. The Krotov approach may also be recommendable if the running costs are big, and pulse sequences with very low rf power consumption are desired. However, the Krotov algorithm is not at present implemented in SIMPSON, and thereby not covered by the following description.

3. Optimal control procedures in SIMPSON

With the aim of conducting optimal control optimizations of NMR experiments, we have added new functionality to the open-source SIMPSON simulation software package [54] to allow for efficient calculations of GRAPE gradients, and combinations of these with conjugated gradient procedures, required for iterative derivation of optimal pulse sequences. A primary source to improved pulse sequences by optimal control takes its origin in better compensation towards artefacts such as inhomogeneous rf fields, but the added features to evaluate this have more general interest than just optimal control calculations. The new functionality builds on the framework of the original SIMPSON code, which accordingly is not changed and all previous input files and calculations can be conducted unaltered. We note that the present combination between optimal control and SIMPSON does not consider effects from relaxation—as addressed in numerous of our references of optimal control in liquid-state NMR spectroscopy. We are currently developing a version of SIMPSON including relaxation and optimal control.

As the original SIMPSON program, and auxiliary software like SIMMOL [59] and various optimization tools [34,60,61], all interference with the compiled SIMPSON code is effectuated via a Tcl scripting code [62,63]. This script provides a flexible interface to SIMPSON and execution of SIMPSON simulations integrated with external programs. The interface to SIMPSON is a Tcl input file `name.in` (where “name” is the chosen name for the file) which typically consists of four sections, either defined globally as `section {content}` or in the form of standard Tcl procedures, i.e., `proc procedurename {} {content}`, where the `content` may be several lines with relevant instructions and the `procedurename` can be any name with relevance to the content.

The standard SIMPSON input file contains four sections (procedure names): (i) `spinsys {...}` with information on the spin system (spins, interactions etc.), (ii) `par {...}` with information on external parameters (external magnetic field strength, powder angles, initial and destination operators, rf field strengths, spinning speed etc.), (iii) `proc pulseseq {} {...}` with information about the pulse sequence, and (iv) `proc main {} {...}` with instruction on the overall flow of the calculations (conduction of the SIMPSON simulation, data processing, launching of supplementary programs etc.). The optimal control implementation adds new commands to the `par`, `pulseseq`, and `main` procedures, and introduces a couple of additional procedures to provide the necessary flexibility in the program execution. The new commands and procedures will be described below with reference to Table 1 providing a formal description of new commands.

In the following we will address particular attention to a set of novel features facilitating, directly or indirectly, conduction of optimal control experiment design. Many of the commands may also be considered strengthening of standard SIMPSON applications independently of optimal control.

3.1. *Rf inhomogeneity*

Previous applications of optimal control for development of NMR experiments reveal that efficient compensation of rf inhomogeneity is one of the major gains obtained by exploiting the experimental degrees of freedom through complicated rf pulse sequences. Accordingly, and to provide a simple means to analyze the performance of existing pulse sequences under inhomogeneous rf field conditions, without the need for reprogramming, we have included an additional variable in the `par` section:

```
rfprof_file name_of_file.rf
```

where the ASCII (i.e., standard text) file `name_of_file.rf` has the following structure: First line: *(number of elements in the profile)* *(number of channels)*, while all subsequent lines contain *(rf scaling factor for 1st channel)* *(same for 2nd channel)*...*(weight factor)* (for rf scaling factors, 1.0 designates nominal rf field strength). If the `rfprof_file` command is not launched in the `par` section, the calculation assumes homogeneous field, likewise there is no inhomogeneity issued for rf channels that exceed the definition in the rf profile.

3.2. *Pulse shapes*

Shaped pulses become increasingly frequent in advanced NMR experiments and are clearly an integral part of the optimal control sequences. This calls for procedures which in an efficient and transparent way can handle series of pulses with arbitrary amplitude and phase. Within SIMPSON rf shapes can have several representations: (i) internal in SIMPSON, in the same way as the FID, (ii) a list variable in the Tcl environment, or (iii) a standard ASCII file for easy communication with external programs and spectrome-

ters. Addressing the ASCII representation, the file has the form of two columns of which the first determines the rf amplitude (in Hz) and the second the phase (in degrees). The shape can be loaded to a Tcl variable (e.g., `sh`) using

```
set sh [load_shape name_of_file]
```

The shape may also be created/derived from a Tcl list using

```
set sh [list2shape {{1000 0} {1500 341}...}]
set listsh [shape2list $sh]
```

where the `$` in the last line is the standard Tcl way to activate the earlier defined variable `sh` for the shape. Alternatively, the shape may be established with random amplitudes and phases using

```
set sh [rand_shape 30000 100 10]
```

which creates a shape with 100 elements and maximum amplitude 30 kHz based on 10 random points equally distributed over the length of the shape and interlaying points calculated by spline interpolation.

With the aim of manipulating shapes (e.g., from the `main` section of the input file), we introduce several other commands demonstrated below (see Table 1 for details).

```
save_shape $rf_shape file_name
set len [shape_len $rf_shape]
set elem [shape_index $rf_shape $i -ampl]
set newshape [shape_join $rf_shape1 $rf_shape2]
set newshape [shape_dup $rf_shape 60 0.8]
set newshape [shape_create 100 -ampl 10000\
-phase 90]
shape_manipulate $rf_shape -time_reversal
set f [shape2fid $shape]
shape2bruker $rf_shape file_name $maxrf
shape2varian $rf_shape file_name
free_shape $rf_shape
free_all_shapes
```

Their action, respectively, corresponds to: save a shape to a file (`file_name`), extract the length of a shape into a variable `len`, extract the amplitude of a certain element (defined by the number `i`), join several shapes into a concatenated shape `newshape`, duplicate a shape with flexibility to change overall phase (increase by 60° is used) and rf amplitudes scaled by a certain factor (given here specifically as 0.8), create shape (100 elements) with constant amplitude (10,000 Hz) and phase (90°), perform time reversal symmetry operation on a given shape (note that the latter two commands have much wider functionality), save a shape into the format of a SIMPSON FID (e.g., for visualization using SIMPLOT), save the shape in Bruker or Varian format, which allow an optional normalization of the amplitudes invoked by `$maxrf`, and free a single shape or all shapes from the memory. The backslash is the standard Tcl nomenclature to specify that a command continues on the next line.

The shaped pulses may be applied in the `pulseseq` section using the command

```
pulse_shaped $duration $shape1 $shape2
```

with the `duration` given in μs and the shapes (all having same number of elements) providing action on different channels as defined in the `spinsys` section, i.e., the command above applies

Table 1
Summary of new features and commands in SIMPSON program.

Parameters for the <code>par</code> section	
<code>string name</code>	Sets a user specific string variable with a value that can be retrieved throughout the input file by declaring <code>\$par(name)</code> .
<code>rfprof_file</code>	Name of the file that defines rf field inhomogeneity distributions for all channels. The file has the structure: First line: (number of elements in the profile) (number of rf channels) Following lines: (scaling for 1st channel) (scaling for 2nd channel) ... (weight factor) Note: (i) The channels are ordered as in the command <code>channel</code> . (ii) If the command, or a file name, is not given no rf inhomogeneity is used, (iii) no rf inhomogeneity is issued for those channels in the simulation that exceed the definition in the rf profile.
<code>oc_tol_cg</code>	Exit tolerance on target function (default 10^{-6}).
<code>oc_tol_ls</code>	Exit tolerance during line-search (default 10^{-3}).
<code>oc_mnbrak_step</code>	Initial step size for bracketing minimum during line-search (default 10).
<code>oc_max_iter</code>	Maximum number of iterations for <code>oc_optimize</code> command (default 1000).
<code>oc_cutoff</code>	Cut off level for unsuccessful optimization trials (default 0.0).
<code>oc_cutoff_iter</code>	Number of iterations before cut off is applied to test for unsuccessful optimization trials (default 1000).
<code>oc_var_save_iter</code>	Number of iterations after which procedure defined in input file with name given in <code>par(oc_var_save_proc)</code> is executed; it is mainly intended to save current rf shape(s) during optimization (default 1001).
<code>oc_var_save_proc</code>	Procedure name executed every <code>par(oc_var_save_iter)</code> iterations; e.g., for storing rf shapes during <code>oc_optimize</code> (max 64 characters, default is none).
<code>oc_cg_min_step</code>	Minimal step taken along the search direction, otherwise optimization finishes (default $1e-3$).
<code>oc_max_brack_eval</code>	Maximum number of target function evaluations for bracketing algorithm during line-search; warning is issued afterwards and optimization finishes with possibly non-ideal solution (default 100).
<code>oc_max_brent_eval</code>	Maximum number of evaluations in Brent's algorithm, then it uses current estimate, line-search is inexact (default 100).
<code>oc_verbose</code>	Causes extensive output during optimization when set to 1 (default 0).
Commands for the <code>pulseseq</code> section	
<code>pulse_shaped dur rf_shape1 rf_shape2 ...</code>	Extends the current propagator to include shaped pulses of duration <code>dur</code> (μ s), and shapes determined by <code>rf_shape1</code> , <code>rf_shape2</code> (etc.) for all channels successively. All shapes within one pulse command need to have equal number of elements. Duration of each element is determined automatically with respect to total duration. If no rf irradiation is needed on any channel, use keyword <code>nothing</code> placed properly on the channel's position.
<code>avgham_static dur ham</code>	Extends the current propagator to include evolution under static Hamiltonian <code>ham</code> for a time period <code>dur</code> (μ s). <code>ham</code> is given in the form of operators similarly to <code>start_operator</code> and <code>detect_operator</code> , and is in frequency units (Hz). Corresponding propagator is created and applied, specified duration is added to internal time controlling variable.
<code>list eulerangles</code>	Returns a list with the three Euler angles describing the orientation of the current crystallite.
<code>store n [from]</code>	Stores the current propagator in memory slot number <code>n</code> (the same functionality as in original SIMPSON). If the second argument is given, then <code>from</code> is used to define the propagator. <code>from</code> can be a number specifying matrix memory slot, the full matrix defined by <code>list {row row...}</code> where each <code>row</code> is a list of elements being either <code>re</code> or <code>im</code> , specific matrix elements (<code>elements {ij}...</code>), or all elements excluding specific matrix elements (<code>notelements {ij}...</code>). Note that internal checking of reusing the stored propagator (relevant for simulations with spinning samples) is disabled when using the new propagator feature.
<code>oc_acq_hermit</code>	Optimal control related commands replacing normal acquisition in pulse sequences. They calculate either target function or gradients (depending what calls the pulse sequence) for a current crystallite. These commands can be used only once in a pulse sequence script, typically at the end. <code>oc_acq_hermit</code> and <code>oc_acq_nonhermit</code> are used when optimizing transfer between Hermitian and non-Hermitian operators, respectively; <code>oc_acq_prop</code> is used for synthesis of desired propagator which is stored (via <code>store</code> command) in the memory slot <code>Nprop</code> .
<code>oc_acq_nonhermit</code>	
<code>oc_acq_prop Nprop</code>	
Commands for the main section	
<code>rf_shape load_shape file_name</code>	Creates internal representation of a shaped pulse from a file <code>file_name</code> . The file contains two columns of numbers, the first specifies a particular rf amplitude (in Hz), the second is a phase (in degrees).
<code>rf_shape list2shape {{ampl phase}\ {ampl phase}...}</code>	Creates internal representation of a shaped pulse from a list of rf amplitudes <code>ampl</code> (in Hz) and phases <code>phase</code> (in degrees).
<code>rf_shape rand_shape max_rf elem rand_elem</code>	Generates internal representation of a shaped pulse created from <code>rand_elem</code> random numbers equally distributed over the total length of a shape consisting of <code>elem</code> elements. The missing elements are calculated by spline interpolation, maximal rf amplitude is limited to <code>max_rf</code> (in Hz).
<code>save_shape rf_shape file_name</code>	Saves the internal representation of a shaped pulse <code>rf_shape</code> to a text file <code>file_name</code> . The file contains two columns of numbers, the first specifies a particular rf amplitude (in Hz), the second is a phase (in degrees).
<code>list shape2list rf_shape</code>	Converts internal representation of a shaped pulse <code>rf_shape</code> into Tcl list variable structured as <code>{ampl phase} {ampl phase} ...</code> , where <code>ampl</code> is rf amplitude (in Hz) and <code>phase</code> is phase (in degrees).
<code>desc shape2fid rf_shape [duration] [-xy -ap -apw]</code>	Converts internal representation of a shaped pulse to a SIMPSON FID file descriptor. This enables, e.g., plotting of the shape using SIMPLOT. The optional duration is the length of the shape in seconds, while the qualifiers <code>-xy</code> , <code>-ap</code> , and <code>-apw</code> direct the FID to contain <code>x/y</code> -phase rf amplitude, amplitude/phase, and amplitude/wrapped-phase (within $0-360^\circ$).
<code>shape2bruker rf_shapefile_name [maxrf]</code>	Stores shape in Bruker shape format. Normalizes maximum rf field strength to the value of 100 in the file.
<code>shape2varian rf_shapefile_name [maxrf]</code>	Stores shape in Varian shape format. Normalizes maximum rf field strength to the value of 1023 in the file.
<code>elem shape_index rf_shape idx [-ampl -phase]</code>	Extracts single element <code>idx</code> from a shaped pulse internally represented by <code>rf_shape</code> . If <code>-ampl</code> or <code>-phase</code> is given, then just amplitude or phase is returned, both values otherwise.
<code>rf_shape shape_join rf_shape1 rf_shape2 ...</code>	Creates a new shape by concatenating previously defined shapes.
<code>rf_shape shape_dup rf_shape_orig phase [scale]</code>	Creates a new shape by duplicating the rf amplitudes in <code>rf_shape_orig</code> and increasing phases by <code>phase</code> . If <code>scale</code> is given the original amplitudes are multiplied by this factor.
<code>rf_shape shape_create Nelem [-ampl expr\ -phase expr]</code>	Creates a new shape with <code>Nelem</code> elements with amplitudes and phases calculated according to given Tcl math expression <code>expr</code> . Within <code>expr</code> , the predefined variable <code>\$i</code> can be used to access current element index, starting from one. Expression example: <code>{1000.0*cos((\$i-1)/100.0)}</code> . If one or both switches with their expression are omitted, then zero values are used.

Table 1 (continued)

<pre>shape_manipulate rf_shape\ [-time_reversal\ -phase_invert\ -cyclic_shift N -ampl\ expr -phase expr] value shape_len rf_shape value shape_ampl rf_shape [-avg\ -min -max -rms] value shape_energy rf_shape dur free_shape rf_shape free_all_shapes value oc_optimize rf_shape1 rf_shape2 ... value oc_optimize rf_shape1 -min min1\ -max max1 -rmsmax rms1 rf_shape2 ... fid oc_evaluate_grad grad_proc Commands for the gradient section oc_grad_shapes rf_shape1 ... oc_grad_add_energy_penalty fid rf_shape1\ lam1 rf_shape2 lam2 ...</pre>	<p>Manipulates an existing shape <i>rf_shape</i> by time reversal operation, by inverting sign of all phases, by performing circular permutation of <i>N</i> elements (<i>N</i> can be both positive or negative, then shift right or left is done), and by performing Tcl math expression <i>expr</i> on current amplitudes and phases. Valid variables for <i>expr</i> are the amplitude $\\$amp1$, the phase $\\$phase$ and element index $\\$i$. Example: $\{\\$amp1/10.0+\\$i*100\}$. Note that operations are performed in the same order as they are specified in the command, from left to right.</p> <p>Gives number of elements in a shaped pulse <i>rf_shape</i>.</p> <p>Gives some statistical data on amplitudes in shaped pulse <i>rf_shape</i>; <i>-avg</i> for a mean value, <i>-min</i> for a minimal and <i>-max</i> for a maximal values, <i>-rms</i> for root-mean-square rf amplitude. These options can be combined in any order.</p> <p>Gives rf energy of a shaped pulse <i>rf_shape</i> with duration <i>dur</i> (in μs).</p> <p>Destroys internal representation of a shaped pulse <i>rf_shape</i> and frees the allocated memory.</p> <p>Destroys internal representations of all shaped pulses and frees the allocated memory.</p> <p>Performs GRAPE optimization on shaped pulses <i>rf_shape1</i>, <i>rf_shape2</i>, ... without or with limitations on minimal and/or maximal rf amplitudes, and/or maximal allowed value of root-mean-squared amplitudes in the corresponding pulse shape(s). It returns the value of target function and variables <i>rf_shape1</i>, <i>rf_shape2</i>, ... are updated with the resulting shaped pulses.</p> <p>For debugging purposes. Calls a Tcl procedure <i>grad_proc</i> that calculates gradients for optimization and returns a <i>fid</i> with the gradient values.</p> <p>Tells the program which shaped pulses (<i>rf_shape1</i>, ...) are optimized; it causes calculation of corresponding gradients. This command should be used in procedure defining calculation of gradients for optimization.</p> <p>Calculates gradients of rf energy of shaped pulses <i>rf_shape1</i>, <i>rf_shape2</i>, ... and adds them to previously calculated gradients stored in <i>fid</i>, multiplied by penalty factors <i>lam1</i>, <i>lam2</i>, ... correspondingly for each shape. This command should be used in procedure defining calculation of gradients for optimization.</p>
--	---

to a spin-system with two channels. If no rf irradiation is needed on any of the channels, write the word `nothing` at the position of the channel in the command above.

3.3. Effective Hamiltonians and propagators

In many cases, it is relevant to address the effective Hamiltonian representing the evolution over a certain time period rather than the effect of the pulse sequence as monitored directly through the evolution of the density operator. In the context of optimal control theory it may, for example, be interesting to optimize a certain Hamiltonian to a particular form, e.g., an isotropic mixing Hamiltonian as discussed by Tošner et al. [17].

In the simple case of a static Hamiltonian, the following command can be used in the `pulseq` section

```
avgham_static $duration\
  $v*(I1x*I2x+I1y*I2y+I1z*I2z)
```

where the Hamiltonian is given in form of operators parsed in the same way as used for the initial and destination operators. In this example, we specifically use terms defining the isotropic mixing, where *v* is the numerical value for the amplitude in frequency units (Hz). If the Hamiltonian depends on the crystallite orientation, the crystallite Euler angles may be accessed in the `pulseq` section by the newly implemented command `set angles [eulerangles]`, which assigns the Tcl variable $\$angles$ to a list containing the alpha, beta, and gamma angles.

For optimal control applications involving effective Hamiltonians, it is relevant to address the handling of propagators, as optimization of the effective Hamiltonian is equivalent to optimizing the overall propagator for the time period. To facilitate the use of propagators for internal calculations or propagation as part of a pulse sequence (using the `store Nprop` command, where *Nprop* is the number for the propagator stored in the internal memory slot as used previously), we have added new facilities to the `store` command demonstrated below:

```
store l0 3
store l0 list {{0.5 0.5 0.5 0.5} {...}...}
store l0 elements {...}
store l0 notelements {...}
```

which stores the propagator defined as a previously assigned matrix slot, as a list of matrix elements, or, if practical, as elements (equal to 1), or excluded elements (equal to 0, 1 otherwise), respectively. It should be noted that when using the new propagator features, the standard timing check (relevant to spinning samples to ensure that a propagator is reused at the same time of a rotor period as it was created) is disabled. Hence these commands should be used with care for spinning samples.

3.4. The target function

To conduct optimal control optimization, we need to specify a target function. This is accomplished in a new procedure in the SIMPSON input file; `proc target_function {} {..}`. Within this procedure, the user has the flexibility to modify the total cost function for the optimization. The target function is activated automatically during the optimization, but as all other SIMPSON Tcl procedures it can also be called directly, e.g., using

```
set tf [target_function]
```

In a typical state-to-state optimization, the target function (i) sets the number of FID points to 1 (by `set par(np) 1`), (ii) calculates a one-point FID (by `set f [fsimpson]`), and (iii) returns the value of the final cost via a variable, e.g.,

```
set Phi [findex $f l -re]
```

The running cost (see Eq. (11)) may be included by calculation of the energy of the shape using

```
set en [shape_energy $shape $duration]
set res [expr $Phi - $lambda*$en]
```

where the last line subtracts the energy with a scaling factor of *lambda* from the final cost (result from previous setting of *Phi*; three lines above) to provide the overall cost of the shape.

3.5. The gradient for the control fields

For optimal control applications, the SIMPSON input file has to be supplemented with a procedure `proc gradient {} {..}` to set

up calculation of the gradients and maintain them updated throughout the optimization in the form of an FID. This construction facilitates automatic updates, for example, adding up contributions from different powder angles or different loops activated through the optimization exactly in the same way as usually encountered for an FID in a SIMPSON calculation. Typically, this procedure only contains a few lines, which in the most basic setup for optimization of the final cost alone includes: (i) definition of the number of FID points to the number of variable elements in the rf shapes (using proper value for `par(np)`), (ii) calculation of the FID (by `set f [fsimpson]`), and (iii) return of the gradient through the FID by `return $f`. In the case where the running cost is included, the following line should be executed right after the `fsimpson` command

```
oc_grad_add_energy_penalty $f $shape -$lambda
```

where `lambda` corresponds to λ in Eq. (11) (or as it was used above in procedure `target_function`), and should be launched with a negative sign as it is a penalty.

The `gradient` procedure enables direct access to the gradients at any point to facilitate mathematical manipulation of them. Using the command

```
oc_grad_shapes $sh1 $sh2 ...
```

it is possible to control which pulse shapes are to be optimized and in what order the gradients are collected in the FID. In that way it is feasible to maintain a series of different gradients for different parts of the pulse sequence (*vide infra*). As an important asset this enables straightforward optimization of several elements in pulse sequences for which part of the pulse sequence is already optimal or not amenable for optimization. In addition this facilitates optimization of pulse sequences going through particular intermediate states, which may be selected using, e.g., the SIMPSON `filter` command. We note that the gradient procedure is activated automatically during the optimization. In contrast to other Tcl procedures it can not be launched directly from the Tcl script. If one for debugging purposes (or the like) wants to have access to the gradients it can be accomplished using

```
set fg [oc_evaluate_grad gradient]
```

where `fg` is a FID containing the gradient. In this particular case, the command `oc_grad_shapes` has to be included in the `gradient` procedure.

3.6. Optimization

Assembly of the overall optimization requires cooperative action in different places of input file, as already addressed above in relation to the `target_function` and `gradient` procedures. Two other parts also needs to be activated: the `pulseseq` and the main sections containing instructions for the specific optimization. Furthermore, more general parameters for the optimization process can be controlled via commands in the `par` section.

The pulse sequence, defined in the `pulseseq` procedure, plays a key role in the optimization: it serves calculation of both the target function and the gradient. This is accomplished using two commands. The first is the `pulse_shaped $duration $shape1 $shape2 ...` command introduced above. During its execution, information on to which variables the gradients should be calculated is gathered. The second command specifies the type of optimization, being state-to-state transfers between *Hermitian* or *non-Hermitian* operators or optimization of a propagator for which

the relevant target function or gradients are calculated. These options are invoked using one of the new acquisition commands (see also Table 1):

```
oc_acq_hermit
oc_acq_nonhermit
oc_acq_prop lo
```

These commands can only be used once in a pulse sequence script (typically in the end), and effectuate calculation of the control field gradients. We note that the command `reset` should be used with great care (should not be used inside an optimized sequence), and the use of `filter` and `evolve` commands are prohibited when optimizing a desired propagator.

In the main section the optimization is activated using the command

```
set tf [oc_optimize $shape1 -min 0 -max 20000\
-rmsmax 10000 $shape2 ...]
```

where `$shape1`, `$shape2` are pulse shapes that are to be optimized, possibly with limitations on their maximal and/or minimal amplitudes and/or maximal root-mean-square amplitudes (any or all of the `-min`, `-max` and `-rmsmax` switches can be omitted). The limits on rf amplitudes are supplemented to facilitate optimization of sequences which can be practically implemented on available experimental equipment, where limitations in peak rf amplitude and overall energy dissemination to the probe is often an issue (another option to limit rf energy is via a running cost).

In the present implementation, the optimal pulse sequences are found using GRAPE [8] in combination with conjugated gradients [32]. As illustrated in Fig. 2, the algorithm starts with evaluating the target function for the initial guess (can be a sequence of random pulses) and calculation of gradient. This gradient determines a direction in the multivariate space along which a line search is performed in order to determine the best value of the “step size” ϵ defined in Eq. (14). The line search is made in two steps, first the optimal value is roughly localized by finding an interval enclosing it, and secondly, the optimum is refined using the Brent method [32]. Subsequently, the new pulse sequence parameters are accepted and a test for convergence is performed. The next iteration starts with the calculation of the new gradient. We use the approach of Polak and Ribiere to determine the new search direction in the conjugated gradients method [32].

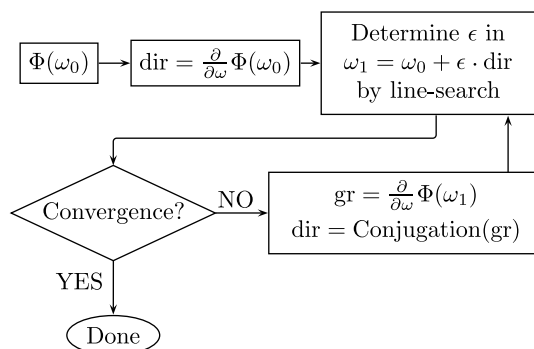


Fig. 2. Block diagram of GRAPE algorithm with conjugated gradients as implemented into the SIMPSON open-source software. The central part is calculation of the step size, ϵ , taken along the gradient/conjugated gradient direction, *dir*, in each iteration (c.f. Eq. (14)).

In the `par` section of the input file, it is possible to control the optimization process described above through the change of a number of variables with predefined values. These parameters are summarized in Table 1.

3.7. Experimental implementation

It is beyond the scope of this paper to demonstrate experimental implementations of the optimized NMR pulse sequences. We refer here to the description in recent papers addressing liquid- and solid-state NMR optimal control experiments (*vide supra*). It should be emphasized, however, that successful implementation requires that: (i) the amplitudes and phases in the rf pulse waveforms are represented correctly at the site of the spins in the probe (i.e., it is important that amplifiers, etc., are setup to provide a linear response to an amplitude list and that the pulse timings are correctly implemented. Both may be tested using an oscilloscope.) and (ii) the pulse sequences optimized to a given overall length is only used with that length experimentally.

4. Examples

In this section, we will demonstrate the power of optimal control in NMR experiment design, its versatility, as well as some pitfalls one can encounter during the optimizations. We present the usage of our optimal control implementation in SIMPSON on several examples. The full codes/input files are collected in [Supplementary material](#) or can be downloaded from the SIMPSON webpages [64].

4.1. Coherence transfer through J coupling

In our first example, we address a heteronuclear J -coupled two-spin- $\frac{1}{2}$ system for which the aim is to optimize transfer between transverse in-phase coherences on the two spins. The transfer is represented in terms of the Hermitian operators $I_x \rightarrow S_x$. It might be considered an academic example since optimal solutions to this problem are very well known. The example, however, illustrates some important features related to pulse sequence optimization.

We select the total time of the shaped pulses to be $T = \frac{1}{J}$ (with $J = 140$ Hz) and digitize it into 150 equally long steps giving a pulse length of $\Delta t = 47.6 \mu\text{s}$ on the two rf channels. We assume that both nuclei are on-resonance (i.e., no chemical shifts are involved) and a priori expect the result to reflect a Hartmann–Hahn matched condition for the rf irradiation on the two rf channels. As illustrated in Fig. 3, this prediction is fulfilled by the optimal sequence.

Before discussing the details of the result, it is relevant to first comment on the input files. For the purpose of illustration we used two variants: case (a), without any constraints, and case (b), with a penalty on the accumulated rf power (or energy). The `spinsys`, `par`, `pulseq`, and `main` sections are common for both simulations:

```
spinsys {
  channels 1H 13C
  nuclei 1H 13C
  jcoupling 1 2 140 0 0 0 0
}
```

where we specifically have chosen proton and carbon nuclei as partners in the heteronuclear spin- $\frac{1}{2}$ system, with the J coupling being 140 Hz.

```
par {
  spin_rate 0
  crystal_file alpha0beta0
  gamma_angles 1
  start_operator I1x
  detect_operator I2x
  variable duration 1.0e6/140.0
  variable NOC 150
  conjugate_fid false
}
```

In this section we set up static sample conditions, mimicking liquid-state NMR experiments, by setting the spin rate to 0. We also define the operators involved in the transfer and the variables `par(duration)` and `par(NOC)` to specify the duration of the experiment and the number of elements in the shaped pulses,

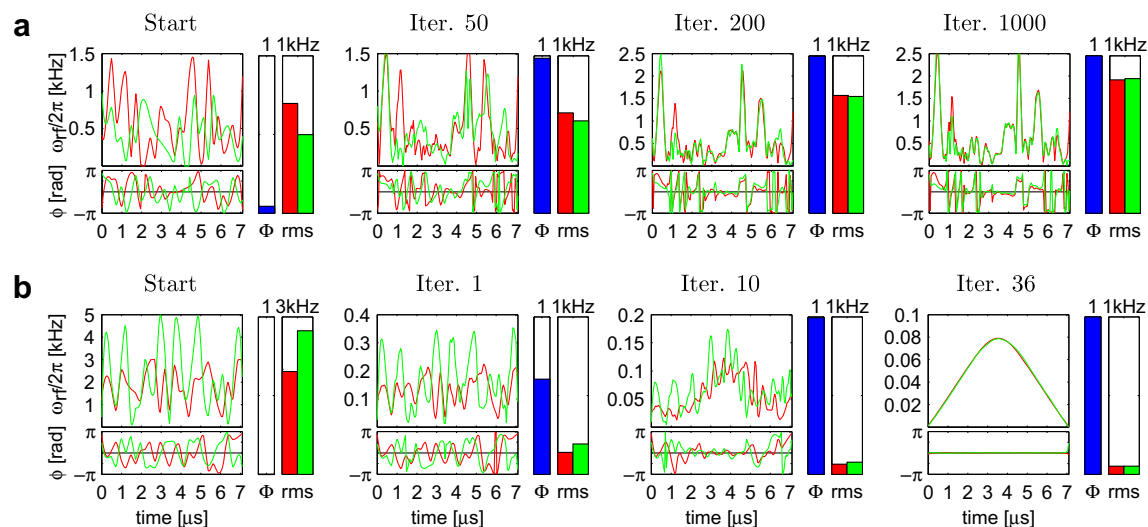


Fig. 3. Pulse sequence changes observed in the course of optimal control optimization a pulse sequence for $I_x \rightarrow S_x$ transfer in heteronuclear two spin- $\frac{1}{2}$ system with J -coupling ($J = 140$ Hz). Amplitudes and phases are displayed in left panels (red and green curves correspond x - and y -phase rf irradiation, respectively), the bar graphs on the right represent the transfer efficiency (normalized final cost Φ_{fin}) and root-mean-squared (rms) amplitude of the shapes on the two rf channels. (a) Unrestricted optimization requires many iterations to reach Hartmann–Hahn match for ideal coherence transfer, with increasing rms pulse amplitudes. (b) Variant with using a penalty on the rf energy converges faster to the unique solution. See text for more details.

respectively. For optimal control calculations in SIMPSON, it is crucial to set `conjugate_fid` to `false` value in order not to disturb the gradient calculations. For case (b), we include the rf penalty by adding line `variable lam 1.0e-5`.

```
proc pulseq {} {
  global par rfsh1 rfsh2
  reset
  pulse_shaped $par(duration) $rfsh1 $rfsh2
  oc_acq_hermit
}
```

The pulse sequence consists of two shaped pulses `rfsh1` and `rfsh2` applied simultaneously on the two channels. The actual values in the shapes will be determined in the `main` section and are passed via the `global` statement.

```
proc main {} {
  global par rfsh1 rfsh2
  set rfsh1 [rand_shape 1500 $par(NOC) 30]
  set rfsh2 [rand_shape 1000 $par(NOC) 30]
  set tfopt [oc_optimize $rfsh1 $rfsh2]
  save_shape $rfsh1 $par(name)_sol_H.dat
  save_shape $rfsh2 $par(name)_sol_C.dat
  free_all_shapes
}
```

The `main` section controls the overall flow of the optimization. First, two rf shapes are generated using 30 random rf amplitudes (ranging from 0 up to 1500 or 1000 Hz) and interpolated to give `par(NOC)` elements in each shape. Subsequently, the optimization is activated using the command `oc_optimize` which takes the current rf shapes as the starting point. Upon termination, `rfsh1` and `rfsh2` are holding the resulting pulse shapes. These are stored using the `save_shape` commands. Finally, the rf shapes are released from the memory and the program quits.

Before executing the optimization, we need to define also the `target_function` and `gradient` sections. They differ slightly for case (a) without constraints and case (b) with rf penalty.

```
# case (a)
proc target_function {} {
  global par
  set par(np) 1
  set f [fsimpson]
  set Res [findex $f l -re]
  funload $f
  return [format "%.20f" $Res]
}

# case (b)
proc target_function {} {
  global par rfsh1 rfsh2
  set par(np) 1
  set f [fsimpson]
  set eff [findex $f l -re]
  funload $f
  set en1 [shape_energy $rfsh1 $par(duration)]
  set en2 [shape_energy $rfsh2 $par(duration)]
  set Res [expr $eff - $par(lam)* ($en1 + $en2)]
  return [format "%.20f" $Res]
}
```

Here we set number of points in the FID to 1 and read just the real part of the calculated FID containing the final cost as provided through the `oc_acq_hermit` command in the `pulseq` section.

To complete the code, we need to provide definitions for the calculations of gradients. We note that the length of the new FID with gradient values needs to be set to the total number of pulse elements that we wish to optimize. That is, in our present case, the sum of individual pulse shape elements (or equivalently twice `par(NOC)`).

```
# case (a)
proc gradient {} {
  global par
  set par(np) [expr 2*$par(NOC)]
  set f [fsimpson]
  return $f
}

# case (b)
proc gradient {} {
  global par rfsh1 rfsh2
  set par(np) [expr 2*$par(NOC)]
  set f [fsimpson]
  oc_grad_add_energy_penalty $f $rfsh1 \
    -$par(lam) $rfsh2 -$par(lam)
  return $f
}
```

where the `oc_grad_add_energy_penalty` command in case (b) corrects the gradients to take into account the running cost penalty.

The entire code is stored in a file, e.g., `example.in`, and can be executed by SIMPSON using the command

```
> simpson example.in
```

A typical output from the program is:

```
Argument 1: shape slot 0, min = 0, max = 1e + 06
Argument 2: shape slot 1, min = 0, max = 1e + 06
Number of variables is 300
Global tolerance on target function is 1e-06
Maximal number of iterations is 1000
Optimization is terminated after 1000 iterations if
target function is not higher than 0
Every 100l iterations procedure'none' (for storing
variables) is executed
Minimal step size along conjugated gradient is 0.001
Tolerance for line-search (Brent) is 0.001
Line-search will terminate after reaching 100
evaluations
Initial step for bracketing minimum is 10
Bracketing will fail after conducting 100
evaluations
Initial target function: -0.0027585696
Iter 1: tf = 0.0870565691 (step = 2285.18)
Iter 2: tf = 0.1550790844 (step = 306.365)
...
Iter 643: tf=0.9988834753 (step=90.1747)
Done by reaching target function tolerance limit.
```

This output provides a summary of the parameters controlling the optimization process (in the present case default values are used; see [Table 1](#) for a complete list of parameters and information on how to change them) followed by progress of the target function (`tf`) throughout the iterations (the value in parenthesis indicates the size of the step ϵ taken along the search direction; a more detailed progress report can be obtained by setting the `par(oc_verbose)` flag).

In case (a), the theoretical maximum value for the target function (i.e., the final cost) is 1.0, indicating that the coherence was transferred with 100% efficiency in full agreement with a priori knowledge, e.g., from unitary bounds of spin dynamics. This objective is met after many iterations, depending on initial guess. As it is demonstrated in Fig. 3a, initially different pulse sequences tend to approach each other in order to establish the Hartmann–Hahn match between the rf fields on the two channels. In this particular case, the pulse amplitudes are not limited and can easily rise. The resulting pulse sequences are actually demanding more rf power than the initial ones. If we repeat the calculation with different initial guesses, different solutions will be found. The ensemble of possible solutions is infinite, with the only common feature that those observed all reached Hartmann–Hahn match between the rf fields on the two rf channels.

In case (b), we wish to find among the many solutions one that requires minimal rf power. To accomplish this, a penalty on the accumulated rf energy is included in the cost function. The illustrative result is presented in Fig. 3b. We observe a faster convergence (perhaps an attribute of this specific problem) to a unique solution, independent of the initial guesses (this is an indication of a global optimum, although our approach does not contain a guarantee this is reached). As the iterations proceed, the rf pulse amplitudes represented as root-mean-square (rms) amplitudes are decreasing while the transfer efficiency rises. Some care needs to be taken if one uses rf penalty. A too large scaling factor λ may lead to compromises on the efficiency (i.e., the value of the final cost) while too small λ does not impose real rf restrictions. The presented case, however, is not particularly sensitive to the exact value and all our optimizations converged to the solution presented in Fig. 3b. It should be noted that this pulse shape is superior to an alternative constant-amplitude Hartmann–Hahn match pulse sequence in terms of accumulated rf power. The former sequence is represented by an rms rf field amplitude of 52 Hz and an effective Hamiltonian $H = \pi J(I_z S_z + I_y S_y)$ while the latter would require a pulse rms amplitude of at least 60 Hz to provide transfer through the effective Hamiltonian $H = \pi J[2I_z S_z + \frac{\sqrt{3}}{4}(I_x + S_x)]$.

4.2. Broadband inversion pulses for liquid-state NMR

For successful experimental implementations, it is obviously necessary to design robust pulse sequences that are capable of achieving the goals not only for a single specific set of spin system parameters, as used in our first example. Accordingly, our next example illustrates how one can set up optimization of state-to-state transfers for a range of conditions. We select broadband inversion pulses for liquid-state NMR applications as an example, for which we demonstrate possible problems and efficient solutions. We note that the optimal control problem of broadband excitation and inversion pulses has previously been addressed by Kobzar et al. [65,66], and we take inspiration from their work in the present example.

Our goal is to find a shaped rf pulse that inverts the longitudinal polarization completely ($I_z \rightarrow -I_z$) regardless of the rf offset within a range of ± 50 kHz while keeping the accumulated rf power reasonably low. Extrapolating the results of Kobzar et al. [65] to estimate a minimal pulse length for the given offset range and extending this to a slightly longer pulse $T = 600 \mu\text{s}$, divided into 600 steps $\Delta t = 1 \mu\text{s}$, we provide some flexibility for the pulse to reach an optimum for a broad range of conditions. In this example, the `spinsys` section introduces just one nucleus with zero chemical shift, while the `pulseseq` and `main` sections are identical to the previous example with the exception that we deal with a single-rf-channel experiment. In order to optimize the pulse for many offsets simultaneously, we need to modify target

function and the gradient calculations to include summation loops accommodating the relevant range of chemical shift values (equivalent to offsets):

```
proc target_function {} {
    global par lims
    set par(np) 1
    set Res 0.0
    foreach shift $lims {
        set f [fsimpson [list [list shift_l_iso $shift]]]
        set dum [findex $f l -re]
        set Res [expr $Res+$dum]
        funload $f
    }
    return [format "%.20f" $Res]
}
proc gradient {} {
    global par lims
    set par(np) $par(NOC)
    set f [fcreate -np $par(NOC) -sw $par(sw)]
    foreach shift $lims {
        set g [fsimpson [list [list shift_l_iso $shift]]]
        fadd $f $g
        funload $g
    }
    return $f
}
```

In this implementation, the Tcl list variable `lims` holds a set of offset values spanning the desired range of $\pm(50$ kHz. The command `fsimpson`, which invokes calculation of either the final cost or the gradients, is executed several times with different spin system parameter `shift_l_iso` defining the chemical shift of the nucleus (cf., Ref. [54]; in this way any spin system parameter can be varied. An alternative implementation could make use of the `offset` command instead). The results for the various chemical shifts are added as numbers in case of the target function, or as FIDs in case of the gradients using standard Tcl or original SIMPSON commands [54].

Our first calculation involves 11 offset values separated by 10 kHz. The resulting pulse sequence is presented in Fig. 4a along with its inversion profile. It is evident that although the target function reaches its global extremum the result is not practically useful. This is despite the fact that the optimal control optimization provided us with a perfect and exact answer to our quest: optimized inversion for a specific set of offsets. In between the specified offset points, however, the effect is unpredictable. The obvious remedy is to use a more dense set of offset values. Accordingly, in a second attempt we use 101 offsets separated by 1 kHz which leads to a satisfactory result as illustrated in Fig. 4b. Although it is not a problem in this particular case, we note that optimizations with dense parameter variation may become quite time demanding. In such cases, it is advisable to increase the number of check points gradually, always taking the result of the previous optimization as the initial guess to the next stage. The average inversion efficiency over the whole ± 50 kHz range is excellent (amounts to 0.9997) and rf rms amplitude is about 13 kHz with peak values in the range of 30 kHz. The latter may not be acceptable with respect to hardware or sample heating, implying that we conducted one more optimization with an upper limit of 10 kHz for the rf amplitudes. This requires only a minor modification of the SIMPSON code in the `main` section:

```
set tf [oc_optimize $rfsh -max 10000]
```

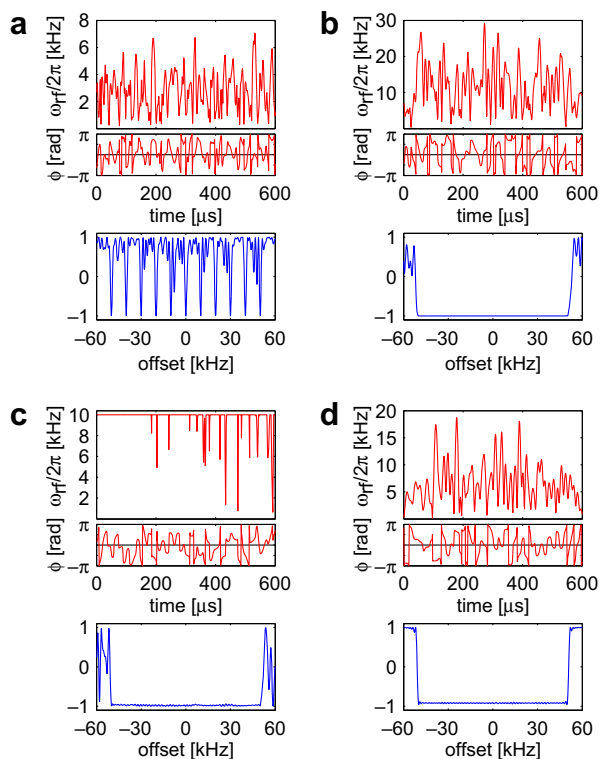


Fig. 4. Optimal control design of broadband inversion pulses with a bandwidth of 100 kHz. The upper panels show the resulting pulse sequences (amplitude and phase), while the lower panels examine their effective I_z inversion profile. (a) Good inversion is achieved only for the limited set of conditions (i.e., the 11 chemical shifts) used during optimization. (b) Perfect results can be obtained when the set of conditions is enlarged. (c) A reasonably good pulse sequence can also be found when the maximal rf amplitude is limited to 10 kHz. (d) Optimization with penalty on the accumulated rf energy provides a pulse with lower rf power demands but may result in lower efficiency.

The resulting pulse sequence, shown in Fig. 4c, demonstrates that a fairly good inversion with an average efficiency of 0.977 can be obtained under these limiting conditions. Following this, a natural question would be: How far can we squeeze the rf power demands, while maintaining an acceptable degree of inversion? Some hint is provided in our final demonstration in this section where we include rf penalty in the optimization. The result is shown in Fig. 4d. With a scaling factor of $\lambda = 10^{-5}$ we are forced to make a trade off between efficiency and rf power: the rms amplitude is below 7.5 kHz but the average inversion efficiency is only 0.912. On basis of all optimizations, we estimate that the rf limit is somewhere around 10 kHz for the present offset and pulse length.

4.3. Broadband refocusing pulses

The rf pulses developed in the previous section suffer from an important limitation: they can be used only to invert longitudinal polarization $I_z \rightarrow -I_z$, while the effect on the other components I_x and I_y is undefined. Accordingly a refined goal could be to find a true broadband refocusing pulse that, for example, acts as an ideal x -phase π -pulse. This implies that we wish simultaneously to accomplish the transfers $I_x \rightarrow I_x$, $I_y \rightarrow -I_y$, and $I_z \rightarrow -I_z$. One can readily implement such an optimization by including other summation loops for the two additional transfers to the code of previous example. However, we take another approach involving the synthesis of a π_x -pulse propagator of the form $U_D = \exp\{-i\pi I_x\}$. In this manner, we demonstrate

another important category of optimal control implementations in SIMPSON.

The optimization of a state-to-state transfer may readily be transformed into a propagator optimization simply by modifying the `pulseq` section. In the case it can be coded as follows:

```
proc pulseq {} {
  global par rfsh
  reset
  avgham_static 0.5e6 1lx
  store 10
  reset
  pulse_shaped $par(duration) $rfsh
  oc_acq_prop 10
}
```

The `pulseq` now contains two parts. First, the desired propagator is prepared for a given spin system using the command `avg-ham_static`. Its first argument defines the overall time evolution interval (0.5 s) during which the effective Hamiltonian $H_{\text{eff}} = 2\pi\nu I_x$ (with $\nu = 1$ Hz) is active. The propagator is stored in the memory slot number 10. Second, the propagator representing the effect of the shaped pulse is created and its similarity with the desired one is calculated using the command `oc_acq_prop` (according to Eq. (10)). Please note that the `reset` command is needed to separate the two parts of the code.

It turns out that calculation of broadband refocusing pulses is not straightforward with respect to settings of the offset range, the duration of the pulse, and the maximal rf amplitude. Here we present our preliminary results for a 200 μs long pulse working within the offset range ± 12.5 kHz with the maximum rf amplitude limited to 15 kHz. A detailed study devoted to this problem will be presented elsewhere. The optimizations converged to a shape shown in Fig. 5a. The performance of the pulse was verified by calculating its effect on the three Cartesian operators, as illustrated in Fig. 5b. It is apparent that I_x is left unperturbed with the final value of 0.998 (averaged over the desired offset range), while I_y and I_z are inverted to average values of -0.997 and -0.998 , respectively.

4.4. Application to coherence transfer in solid-state NMR

In the previous examples, we have demonstrated that optimal control can provide proper solutions not only for a single set of spin system parameters but also for a range of offsets. The next challenging task is to use optimal control in NMR of rotating solids, imposing periodical variations to internal Hamiltonians as well as a wide distribution of Hamiltonians due to anisotropic nuclear spin interactions. In solid-state NMR the anisotropy of the Hamiltonian adds complexity to pulse sequence design if the goal is to

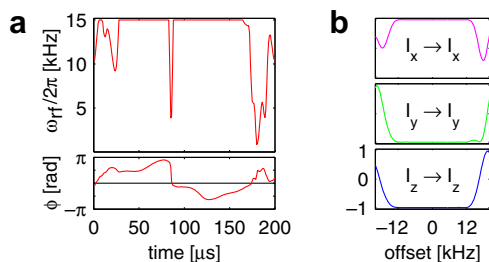


Fig. 5. Optimization of a broadband refocusing pulse providing an effective propagator $\exp(-i\pi I_x)$ extending uniformly over an offset bandwidth of 25 kHz. (a) The pulse shape (amplitude and phase). (b) Effects of the pulse on the Cartesian operators I_x , I_y , and I_z illustrating a performance corresponding to a uniform unitary π rotation around x axis as required.

bring all “crystallites” to the same ending point in terms of a final state or a desired effective Hamiltonian. Unlike liquid-state NMR parameter variations, the spread in sizes for fundamental couplings used for, e.g., coherence transfer through dipole–dipole interactions spans from 0 to a given maximum value determined by the internuclear distance. As so it appears evident that optimal control in combination with the SIMPSON simulation program would be of interest as a new way to design efficient experiments using more experimental degrees of freedom to cope with the challenges of anisotropic nuclear spin interactions and sample rotation. This applies not least in cases where the interest is to design evolution under Hamiltonians that does not match the typical zero- and double-quantum operators typical for the vast majority of dipolar recoupling experiments. In such cases analytical treatment becomes exceedingly difficult [17].

With reference to previous accounts on optimal control in solid-state NMR [14–20], we here illustrate the principles for one simple case: $I_x \rightarrow S_x$ transfer of coherence from one ensemble of spins to another in a powder sample through dipole–dipole coupling under fast sample spinning conditions and in presence of variations in isotropic as well as anisotropic nuclear spin interactions. Our object will be a typical ^{13}C – ^{15}N spin-pair in a peptide bond of a solid protein. This calculation closely follows the work of Kehlet et al. [19] where novel sequences for nitrogen-to-carbon coherence transfer in NCO and NCA experiments were developed using optimal control.

For solid-state NMR applications, we need to introduce anisotropic nuclear spin interactions through the parameters in the `spinsys` section, as exemplified by

```
shift 1 0 99p 0.19 -90 -90 -17
shift 2 0 -76p 0.90 0 0 9
dipole 1 2 1176 0 90 57
```

where the shift parameter uses the entries nucleus, isotropic shift, anisotropic shift, asymmetry parameter, and three Euler angles defining the orientation of the shielding tensor relative to a molecular frame. Correspondingly, the dipole–dipole coupling is defined through the two nuclei, the dipolar coupling constant, and three Euler angles. Details can be found in Ref. [54].

In the `par` section, we add the following entries

```
spin_rate 12000
crystal_file repl144
gamma_angles 10
rfprof_file lorentz5p
```

to specify the sample spinning frequency to 12 kHz, that powder averaging is performed with 144 REPULSION [67] α , β crystallite angles and 10 γ angles. In this specific case a 5% Lorentzian rf inhomogeneity profile is invoked. We note throughout the optimal control sequences the use of fast symmetry-based propagation methods such as γ -COMPUTE [68] should generally be avoided or they should be used with great care due to lack of symmetry in OC pulse sequences. This implies that the `method` command takes its default value `direct` (not invoking special symmetry properties to speed up calculations; and accordingly not listed above) as described in more detail in Ref. [54].

The next example attempts to answer the question what is the maximum efficiency, as a function of pulse sequence duration. The optimizations were conducted repeatedly with different number of elements in shaped pulses applied on the ^{15}N and ^{13}C rf channels and different sequences were obtained for each duration. The overall trend of maximal efficiency is shown in Fig. 6a clearly illustrating that the transfer efficiency may monotonically increased towards 100% (not shown) by increasing the length of the pulse sequence. It may be formulated that optimal control in this sense spans the gap between typical fast γ -encoded coherence transfer [44,69] and the somewhat slower coherence transfer by adiabatic methods [70] while typically keeping the rf field consumption considerably lower than any of these methods [14,19,33].

In practice, it is desirable to conduct coherence transfer applying for specific chemical shift ranges for the initial and target spins rather than produce sequences that are either too narrow-banded or too broad-banded. The latter may cause transfer to undesired spins, thereby reducing the overall transfer efficiency or rendering spectral interpretation ambiguous. Furthermore, excessively broadband pulse sequences are often more demanding with respect to consumed rf power, and accordingly more risky with respect to sample heating for lossy samples. Addressing the popular “double cross polarization experiment” (DCP) from Schaefer and coworkers [69] the offset dependent excitation profile may be altered using ramped rf irradiation [71] or more advanced amplitude and phase-modulated rf irradiation schemes as proposed by Bjerring et al. [72–74]. In addition to the rf inhomogeneity compensation, already introduced above, the optimization over a specific offset ranges corresponding, e.g., to NCA and NCO transfers, may readily be accomplished by introducing the appropriate offset ranges in the `target_function` and `gradient` sections as discussed above in the context of broadband rf pulses and accomplished for the specific case of a NCO transfer in the [Supplementary material](#). The resulting pulse sequence provides an rf inhomogeneity profile as illustrated in Fig. 6b (contour plot calcu-

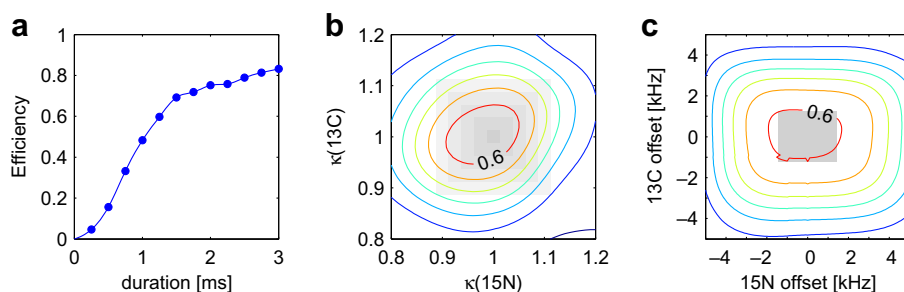


Fig. 6. Results of optimal control design of pulse sequences for $^{15}\text{N} \rightarrow ^{13}\text{C}$ (carbonyl) coherence transfer between directly bonded nuclei in a protein backbone, assuming a solid powder sample subjected to magic-angle-spinning with a frequency of 12 kHz at a 16.5 T static magnetic field. (a) Exploring the limits for the achievable transfer efficiency using ideal experimental conditions; several calculations were started for each pulse sequence duration and the best transfer efficiency is displayed. (b) Inspection of the robustness toward rf inhomogeneity; the optimization assumed a 2.4 ms duration of the pulse sequence and rf profile approximated by Lorentzian shape with 5% half-width, as demonstrated by shaded areas in the graph. (c) The same pulse sequence was also optimized for a given spread in resonance offsets (shaded area) and its sensitivity towards this parameter is presented.

lated without rf inhomogeneity involved using different scaling factors on the rf field strengths used on the two rf channels) and a ^{13}C , ^{15}N offset profile as illustrated in Fig. 6c.

4.5. Synthesis of quantum-gate propagators

In a different context, the optimal control procedures implemented in SIMPSON also provides a perfect tool for developing practical pulse sequences for quantum information processors (QIP) based on NMR spectroscopy. QIP requires selective operations traditionally executed using long, soft rf pulses that in turn introduces problems of decoherence and unwanted interference when applied simultaneously on different spins, spoiling the desired action. Recently, in the area of QIP, the so called strongly modulating pulses (SMP), in the context of NMR spectroscopy known for many years as composite pulses, were proposed to tackle these difficulties [75]. This was done by optimizing a few parameters of the composite pulses providing the desired quantum gate propagator, in the full analogy to Eq. (10). Optimal values for amplitudes, phases, frequencies and durations of several element pulses were calculated using the Simplex non-linear optimization method. The approach based on optimal control is more appealing due to its ability to treat hundreds-to-thousands of variables effectively and fast. The straightforward establishment of optimal control SIMPSON calculations opens the possibilities to prepare high fidelity quantum gates tailored precisely for a given molecule and tackling specific experimental issues like rf inhomogeneity. Also, applications of QIP using solid-state NMR are readily and immediately available.

The paper of Fortunato et al. [75] addresses optimization of spin selective pulses, or gates, that allow arbitrary rotations of each spin around independent single-spin axes, while refocusing the internal evolution. The durations of the resulting high-power SMP are decreased by almost an order of magnitude compared to previous soft rf pulses, thereby significantly reducing the effects from relaxation. This work is essentially equivalent to our example with refocusing pulse, just on a more complicated spin system and without consideration of a wide offset range. For our present demonstration, we choose another problem that was treated by Kampermann and Veeman [76] where a single SMP was optimized to provide a whole quantum algorithm at once. The authors consider ^{23}Na nuclei (spin $I = \frac{3}{2}$) in a single crystal of NaNO_3 oriented in such a way that the effective quadrupolar coupling constant was $\omega_Q/2\pi = 84$ kHz. Several SMP's were optimized for different quantum gates and we repeat their calculation just for the quantum Fourier transform (QFT), Grover, and Deutsch operations. The corresponding propagators are explicitly described in the paper by matrix elements and are reproduced here in Fig. 7. We also make an attempt to find shaped pulses with minimal durations.

After providing the relevant spin system parameters in the SIMPSON input file, we need to modify the `pulseseq` section to set up desired propagator. For this purpose, we use the new functionality of the `store` command that enables storing a propagator described by matrix elements. We instruct the program to repeat optimizations with the shape duration decreased in steps of $1 \mu\text{s}$ (and consequently lowered number of element pulses) until the desired propagator is reproduced with a fidelity better than 0.999 (equivalent to the final cost defined in Eq. (10) normalized by matrix dimension). Compressing the total evolution time obviously may lead to an increase in the rf power demands and therefore we set upper limit on rf pulse amplitudes to 20 kHz. This value was chosen to facilitate comparison with the results of Kampermann and Veeman [76], whose QFT pulse sequence used an essentially constant rf amplitude of 19 kHz. Our results, presented in Fig. 7, demonstrates substantially decreased minimal durations of the shaped rf pulses while maintaining the same fidelity as in the original work. Our sequences implement QFT, Grover, and Deutsch operations within 52, 44, and 47 μs , respectively, which compares favourably with the durations of 89.6, 94.8, and 55.1 μs given by Kampermann and Veeman [76]. We obtained these superior timings even without using temporal variables explicitly, which is in direct contrast to a SMP optimization strategy (duration of each element is variable).

5. Conclusions

In conclusion, we have presented the implementation of optimal control procedures into the SIMPSON open source software. The resulting tool has been demonstrated useful for experiment design in various cases of relevance for liquid- and solid-state NMR spectroscopy. Obviously, the range of applications is much wider, including magnetic resonance imaging and more exotic combinations between nuclear magnetic resonance and other spectroscopies. The present paper serves as a manual to the optimal control containing SIMPSON software. We anticipate that this tool will find widespread application in the NMR community as a means to design custom-made pulse sequences for a wide range of applications. This will hopefully lead to a new generation of optimal NMR experiments to be used over the full range of NMR disciplines known so far. In addition, the tool may facilitate studies of more fundamental character, including determination of theoretical maximum values for transformation between spin operators and new inspiration to analytical design strategies reaching these.

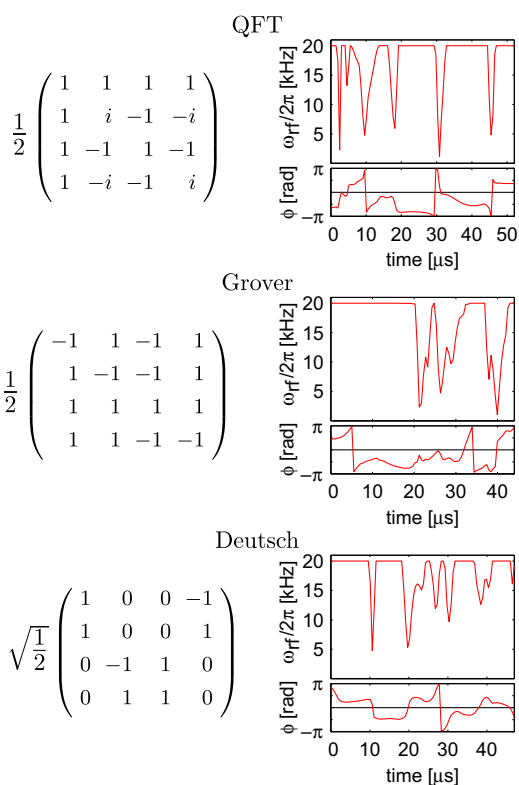


Fig. 7. Optimization of total propagators to implement several quantum operations on ^{23}Na nuclei in a crystal with effective quadrupolar coupling constant $\omega_Q/2\pi = 84$ kHz. The propagators, given in terms of their matrix elements, and corresponding shaped pulses are shown. The durations of the shaped pulses are substantially shorter than those presented in Ref. [76], obtained using a so-called strongly modulating pulses optimization procedure.

Acknowledgments

We acknowledge support from the Danish National Research Foundation, the Danish National Advanced Technology Foundation, the Danish Natural Science Research Foundation, and Carlsbergfondet. Z.T. acknowledges the support from the Czech Science Foundation, Grant No. 202/07/P213. S.J.G. acknowledges support from the European integrated programs Bio-DNP and QAP, from the DFG (GI 203/6-1), SFB 631, and the Fonds der Chemischen Industrie. We thank Dr. T. Reiss for constructive suggestions on the optimal control procedure in earlier stages of the project, as well as M. Bjerring, A.B. Nielsen, J.Ø. Hansen, M.K. Sørensen, and L.A. Straasøe for constructive comments throughout the development of the software and experimental testing of pulse sequences in different applications.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at [doi:10.1016/j.jmr.2008.11.020](https://doi.org/10.1016/j.jmr.2008.11.020).

References

- [1] L. Pontryagin, B. Boltyanskii, R. Gamkrelidze, E. Mishchenko, *The mathematical theory of optimal processes*, Wiley-Interscience, New York, 1962.
- [2] A. Bryson Jr., Y.-C. Ho, *Applied optimal control*, Hemisphere, Washington DC, 1975.
- [3] V.F. Krotov, *Global methods in optimal control theory*, Marcel Dekker Inc., New York, 1995.
- [4] S. Conolly, D. Nishimura, A. Macovsky, Optimal control solutions to the magnetic resonance selective excitation problem, *IEEE Trans. Med. Imaging* 5 (1986) 106–115.
- [5] J. Mao, T.H. Mareci, K.N. Scott, E.R. Andrew, Selective inversion radiofrequency pulses by optimal control, *J. Magn. Reson.* 70 (1986) 310–318.
- [6] D. Rosenfeld, Y. Zur, Design of adiabatic selective pulses using optimal control theory, *Magn. Reson. Med.* 36 (1996) 401–409.
- [7] D. Xu, K.F. King, Y. Zhu, G.C. McKinnon, Z.P. Liang, Designing multichannel, multidimensional, arbitrary flip angle RF pulses using an optimal control approach, *Magn. Reson. Med.* 59 (2008) 547–560.
- [8] N. Khaneja, T. Reiss, C. Kehlet, T. Schulte-Herbrüggen, S.J. Glaser, Optimal control of coupled spin dynamics: design of NMR pulse sequences by gradient ascent algorithms, *J. Magn. Reson.* 172 (2005) 296–305.
- [9] T. Reiss, N. Khaneja, S.J. Glaser, Time-optimal coherence-order-selective transfer of in-phase coherence in heteronuclear IS spin systems, *J. Magn. Reson.* 154 (2002) 192–195.
- [10] N. Khaneja, T. Reiss, B. Luy, S.J. Glaser, Optimal control of spin dynamics in the presence of relaxation, *J. Magn. Reson.* 162 (2003) 311–319.
- [11] N. Khaneja, J.-S. Li, C. Kehlet, B. Luy, S.J. Glaser, Broadband relaxation-optimized polarization transfer in magnetic resonance, *Proc. Natl. Acad. Sci. USA* 101 (2004) 14742–14747.
- [12] K. Kobzar, B. Luy, N. Khaneja, S.J. Glaser, Pattern pulses: design of arbitrary excitation profiles as a function of pulse amplitude and offset, *J. Magn. Reson.* 173 (2005) 229–235.
- [13] J.L. Neves, B. Heitmann, T.O. Reiss, H.H.R. Schor, N. Khaneja, S.J. Glaser, Exploring the limits of polarization transfer efficiency in homonuclear three spin systems, *J. Magn. Reson.* 181 (2006) 126–134.
- [14] C.T. Kehlet, A.C. Sivertsen, M. Bjerring, T.O. Reiss, N. Khaneja, S.J. Glaser, N.C. Nielsen, Improving solid-state NMR dipolar recoupling by optimal control, *J. Am. Chem. Soc.* 126 (2004) 10202–10203.
- [15] C. Kehlet, T. Vosegaard, N. Khaneja, S.J. Glaser, N.C. Nielsen, Low-power homonuclear dipolar recoupling in solid-state NMR developed using optimal control theory, *Chem. Phys. Lett.* 414 (2005) 204–209.
- [16] T. Vosegaard, C. Kehlet, N. Khaneja, S.J. Glaser, N.C. Nielsen, Improved excitation schemes for multiple-quantum magic-angle spinning for quadrupolar nuclei designed using optimal control theory, *J. Am. Chem. Soc.* 127 (2005) 13768–13769.
- [17] Z. Tošner, S.J. Glaser, N. Khaneja, N.C. Nielsen, Effective Hamiltonians by optimal control: solid-state NMR double-quantum and isotropic dipolar recoupling, *J. Chem. Phys.* 125 (2006) 184502.
- [18] C. Kehlet, N.C. Nielsen, Solid-state NMR experiments by optimal control, *Bruker Spin Report* 157 (2007) 31–34.
- [19] C. Kehlet, M. Bjerring, A.C. Sivertsen, T. Kristensen, J.J. Enghild, S.J. Glaser, N. Khaneja, N.C. Nielsen, Optimal control based NCO and NCA experiments for spectral assignment in biological solid-state NMR spectroscopy, *J. Magn. Reson.* 188 (2007) 216–230.
- [20] J.Ø. Hansen, C. Kehlet, M. Bjerring, T. Vosegaard, S.J. Glaser, N. Khaneja, N.C. Nielsen, Optimal control based design of composite dipolar recoupling experiments by analogy to single-spin inversion pulses, *Chem. Phys. Lett.* 447 (2007) 154–161.
- [21] N. Khaneja, S.J. Glaser, R. Brockett, Sub-Riemannian geometry and time optimal control of three spin systems: quantum gates and coherence transfer, *Phys. Rev. A* 65 (2002) 032301.
- [22] T. Schulte-Herbrüggen, A. Spörl, N. Khaneja, S.J. Glaser, Optimal control-based efficient synthesis of building blocks of quantum algorithms seen in perspective from network complexity towards time complexity, *Phys. Rev. A* 72 (2005) 042331/1–042331/7.
- [23] T. Schulte-Herbrüggen, A. Spörl, N. Khaneja, S.J. Glaser, From Network Complexity to Time Complexity via Optimal Control, NATO Science Series, III: Computer and System Sciences 199 (2006) 353–357.
- [24] T. Schulte-Herbrüggen, A.K. Spörl, R. Marx, N. Khaneja, J.M. Myers, A.F. Fahmy, S.J. Glaser, Quantum computing implemented via optimal control: theory and application to spin- and pseudo-spin systems, in: D. Bru, G. Leuchs (Eds.), *Lectures on Quantum Information*, Wiley, VCH, 2006, pp. 481–501.
- [25] N. Khaneja, B. Heitmann, A. Spörl, H. Yuan, T. Schulte-Herbrüggen, S.J. Glaser, Shortest paths for efficient control of indirectly coupled qubits, *Phys. Rev. A* 75 (2007) 012322/1–012322/10.
- [26] A. Spörl, T. Schulte-Herbrüggen, S.J. Glaser, V. Bergholm, M.J. Storz, J. Ferber, F.K. Wilhelm, Optimal control of coupled Josephson qubits, *Phys. Rev. A* 75 (2007) 012302/1–012302/9.
- [27] I. Maximov, Z. Tošner, N.C. Nielsen, Optimal control design of NMR and dynamic nuclear polarization experiments using monotonically convergent algorithms, *J. Chem. Phys.* 128 (2008) 184505.
- [28] N. Pomplun, B. Heitmann, N. Khaneja, S.J. Glaser, Optimization of electronuclear polarization transfer, *Applied Magn. Reson.* 34 (2008) 331–346.
- [29] J.S. Hodges, J.C. Yang, C. Ramanathan, D.G. Cory, Universal control of nuclear spins via anisotropic hyperfine interactions, *Phys. Rev. A* 78 (2008) 010303.
- [30] N. Khaneja, Switched control of electron nuclear spin systems, *Phys. Rev. A* 76 (2007) 032326.
- [31] R. Zeier, H.D. Yuan, N. Khaneja, Time-optimal synthesis of unitary transformations in a coupled fast and slow qubit system, *Phys. Rev. A* 77 (2008) 032332.
- [32] W.H. Press, S.A. Teukolsky, W.T. Vetterling, *Numerical recipes in C: the art of scientific computing*, Cambridge University Press, 1993.
- [33] A.C. Sivertsen, M. Bjerring, C.T. Kehlet, T. Vosegaard, N.C. Nielsen, Numerical simulations in biological solid-state NMR spectroscopy, *Ann. Rep. NMR Spectrosc.* 54 (2005) 246–295.
- [34] N.C. Nielsen, Computational aspects of biological solid-state NMR spectroscopy, in: A. Ramamoorthy (Ed.), *Biological solid-state NMR spectroscopy*, C. Dekker Inc, 2005, pp. 261–304.
- [35] O.W. Sørensen, A universal bound on spin dynamics, *J. Magn. Reson.* 86 (1990) 435–440.
- [36] J. Stoustrup, O. Schedletzky, S.J. Glaser, C. Griesinger, N.C. Nielsen, O.W. Sørensen, Generalized bound on quantum dynamics: efficiency of unitary transformations between non-hermitian states, *Phys. Rev. Lett.* 74 (1995) 2921–2924.
- [37] N.C. Nielsen, T. Schulte-Herbrüggen, O.W. Sørensen, Bounds on spin-dynamics tightened by permutation symmetry. Application to coherence transfer in I2S and I3S spin systems, *Mol. Phys.* 85 (1995) 1205–1216.
- [38] S.J. Glaser, T. Schulte-Herbrüggen, M. Sieveking, O. Schedletzky, N.C. Nielsen, O.W. Sørensen, C. Griesinger, Unitary control in quantum ensembles: maximizing signal intensity in coherent spectroscopy, *Science* 280 (1998) 421–424.
- [39] T. Schulte-Herbrüggen, G. Dirr, U. Helmke, S.J. Glaser, The significance of the C-numerical range and the local C-numerical range in quantum control and quantum information, *Lin. Multilin. Alg.* 56 (2008) 3–26.
- [40] U. Haeberlen, J.S. Waugh, Coherent averaging effects in magnetic resonance, *Phys. Rev.* 175 (1968) 453–467.
- [41] M. Hohwy, N.C. Nielsen, Systematic design and evaluation of multiple-pulse experiments in nuclear magnetic resonance spectroscopy using a semi-continuous Baker–Campbell–Hausdorff expansion, *J. Chem. Phys.* 109 (1998) 3780–3791.
- [42] T.S. Untidt, N.C. Nielsen, Closed solution to the Baker–Campbell–Hausdorff problem: exact effective Hamiltonian theory for analysis of nuclear magnetic resonance experiments, *Phys. Rev. E* 65 (2002) 021108.
- [43] D. Siminowitch, T.S. Untidt, N.C. Nielsen, Exact effective Hamiltonian theory II. Expansion of matrix functions and entangled unitary operators, *J. Chem. Phys.* 120 (2004) 51–66.
- [44] N.C. Nielsen, H. Bildsøe, H.J. Jakobsen, M.H. Levitt, Double-quantum homonuclear rotary resonance: efficient dipolar recovery in magic-angle spinning NMR, *J. Chem. Phys.* 101 (1994) 1805–1812.
- [45] N. Khaneja, C. Kehlet, S.J. Glaser, N.C. Nielsen, Composite dipolar recoupling: anisotropy compensated coherence transfer in solid-state NMR, *J. Chem. Phys.* 124 (2006) 114503.
- [46] N. Khaneja, R. Brockett, S.J. Glaser, Time optimal control in spin systems, *Phys. Rev. A* 63 (2001) 032308.
- [47] N. Khaneja, S.J. Glaser, R. Brockett, Sub-Riemannian geometry and time optimal control of three spin systems: quantum gates and coherence transfer, *Phys. Rev. A* 65 (2002) 032301.
- [48] N. Khaneja, T. Reiss, B. Luy, S.J. Glaser, Optimal control of spin dynamics in the presence of relaxation, *J. Magn. Reson.* 162 (2003) 311–319.
- [49] N. Khaneja, B. Luy, S.J. Glaser, Boundary of quantum evolution under decoherence, *Proc. Natl. Acad. Sci. USA* 100 (2003) 13162–13166.

- [50] D. Stefanatos, N. Khaneja, S.J. Glaser, Optimal control of coupled spins in the presence of longitudinal and transverse relaxation, *Phys. Rev. A* 69 (2004) 022319.
- [51] N. Khaneja, F. Kramer, S.J. Glaser, Optimal experiments for maximizing coherence transfer between coupled spins, *J. Magn. Reson.* 173 (2005) 116–124.
- [52] D. Stefanatos, S.J. Glaser, N. Khaneja, Relaxation-optimized transfer of spin order in Ising spin chains, *Phys. Rev. A* 72 (2005) 062320.
- [53] N. Khaneja, B. Heitmann, A. Spörl, H.D. Yuan, T. Schulte-Herbrüggen, S.J. Glaser, Shortest paths for efficient control of indirectly coupled qubits, *Phys. Rev. A* 75 (2007) 012322.
- [54] M. Bak, J.T. Rasmussen, N.C. Nielsen, SIMPSON: a general simulation program for solid-state NMR spectroscopy, *J. Magn. Reson.* 147 (2000) 296–330. Open source software www.bionmr.chem.au.d.
- [55] N.C. Nielsen, H. Thogersen, O.W. Sørensen, Doubling the sensitivity of Inadequate for tracing out the carbon skeleton of molecules by NMR, *J. Am. Chem. Soc.* 117 (1995) 11365–11366.
- [56] N.C. Nielsen, H. Thogersen, O.W. Sørensen, A systematic strategy for design of optimum coherent experiments applied to efficient interconversion of double- and single-quantum coherences in nuclear magnetic resonance, *J. Chem. Phys.* 105 (1996) 3962–3968.
- [57] M. Sattler, J. Schleucher, O. Schedletzy, S.J. Glaser, C. Griesinger, N.C. Nielsen, O.W. Sørensen, α & β HSQC, an HSQC-type experiment with improved resolution for I2S groups, *J. Magn. Reson. A* 119 (1996) 171–179.
- [58] T.S. Untidt, T. Schulte-Herbrüggen, O.W. Sørensen, N.C. Nielsen, Nuclear magnetic resonance coherence-order- and spin-state-selective correlation in I2S spin systems, *J. Phys. Chem. A* 103 (1999) 8921–8926.
- [59] M. Bak, R. Schultz, T. Vosegaard, N.C. Nielsen, Specification and visualization of anisotropic interaction tensors for numerical simulation of solid-state NMR experiments on polypeptide structures, *J. Magn. Reson.* 154 (2002) 28–45. Open source software www.bionmr.chem.au.d.
- [60] T. Vosegaard, A. Malmendal, N.C. Nielsen, The flexibility of SIMPSON and SIMMOL for numerical simulations in solid- and liquid-state NMR spectroscopy, *Monatshefte Fur Chemie* 133 (2002) 1555–1574.
- [61] A.C. Sivertsen, M. Bjerring, C.T. Kehlet, T. Vosegaard, N.C. Nielsen, Numerical simulations in biological solid-state NMR spectroscopy, in: G.A. Webb (Ed.) *Annual Reports on NMR Spectroscopy*, vol. 54, 2005, pp. 243–293.
- [62] B.B. Welch, *Practical Programming in Tcl and Tk*, Prentice Hall, Englewood Cliffs, NJ, 1995.
- [63] J.K. Ousterhout, *Tcl and the Tk Toolkit*, Addison-Wesley, Reading, MA, 1994.
- [64] SIMPSON web-pages: <http://bionmr.chem.au.d/download/simpson/>.
- [65] K. Kobzar, T.E. Skinner, N. Khaneja, S.J. Glaser, B. Luy, Exploring the limits of broadband excitation and inversion pulses, *J. Magn. Reson.* 170 (2004) 236–243.
- [66] K. Kobzar, T.E. Skinner, N. Khaneja, S.J. Glaser, B. Luy, Exploring the limits of broadband excitation and inversion: II. RF-power optimized pulses, *J. Magn. Reson.* 194 (2008) 58–66.
- [67] M. Bak, N.C. Nielsen, REPULSION, a novel approach to efficient powder averaging in solid-state NMR, *J. Magn. Reson.* 125 (1997) 132–139.
- [68] M. Hohwy, H. Bildsøe, H.J. Jakobsen, N.C. Nielsen, Efficient spectral simulations in NMR of rotating solids. The g-COMPUTE algorithm, *J. Magn. Reson.* 136 (1999) 6–14.
- [69] J. Schaefer, E.O. Stejskal, J.R. Garbow, R.A. McKay, Quantitative determination of the concentrations of C-13-N-15 chemical bonds by double cross-polarization NMR, *J. Magn. Reson.* 59 (1984) 150–156.
- [70] M. Baldus, D.G. Geurts, S. Hediger, B.H. Meier, Efficient N-15-C-13 polarization transfer by adiabatic-passage Hartmann–Hahn cross polarization, *J. Magn. Reson. A* 118 (1996) 140–144.
- [71] G. Metz, X. Wu, S.O. Smith, Ramped-amplitude cross polarization in magic-angle-spinning NMR, *J. Magn. Reson. A* 110 (1994) 219–227.
- [72] M. Bjerring, N.C. Nielsen, Solid-state NMR heteronuclear coherence transfer using phase and amplitude modulated rf irradiation at the Hartmann–Hahn condition, *Chem. Phys. Lett.* 382 (2003) 671–678.
- [73] M. Bjerring, J.T. Rasmussen, R.S. Krogshave, N.C. Nielsen, Heteronuclear coherence transfer in solid-state NMR using a γ -encoded transferred echo experiment, *J. Chem. Phys.* 119 (2003) 8916–8926.
- [74] M. Bjerring, N.C. Nielsen, Solid-state NMR heteronuclear dipolar recoupling using off-resonance symmetry-based pulse sequences, *Chem. Phys. Lett.* 370 (2003) 496–503.
- [75] E.M. Fortunato, M.A. Pravia, N. Boulant, G. Teklemariam, T.F. Havel, D.G. Cory, Design of strongly modulating pulses to implement precise effective Hamiltonians for quantum information processing, *J. Chem. Phys.* 116 (2002) 7599–7606.
- [76] H. Kampermann, W.S. Veeman, Characterisation of quantum algorithms by quantum process tomography using quadrupolar spins in solid-state nuclear magnetic resonance, *J. Chem. Phys.* 122 (2005) 214108.