FULL PAPER

Magnetic Resonance in Medicine

A gradient optimization toolbox for general purpose time-optimal MRI gradient waveform design

Michael Loecher[1,2] 🔾 | Matthew J. Middione[1,2] | Daniel B. Ennis[1,2]

[1]Department of Radiology, Stanford University, Stanford, CA, USA

[2]Department of Radiology, Veterans Affairs Palo Alto Health Care System, Palo Alto, CA, USA

Correspondence
Michael Loecher, Radiological Sciences
Lab, Stanford University, 1201 Welch Road,
Stanford, CA 94305, USA.
Email: mloecher@stanford.edu

Purpose: To introduce and demonstrate a software library for time-optimal gradient waveform optimization with a wide range of applications. The software enables direct on-the-fly gradient waveform design on the scanner hardware for multiple vendors.

Methods: The open-source gradient optimization (GrOpt) toolbox was implemented in C with both Matlab and Python wrappers. The toolbox enables gradient waveforms to be generated based on a set of constraints that define the features and encodings for a given acquisition. The GrOpt optimization routine is based on the alternating direction method of multipliers (ADMM). Additional constraints enable error corrections to be added, or patient comfort and safety to be adressed. A range of applications and compute speed metrics are analyzed. Finally, the method is implemented and tested on scanners from different vendors.

Results: Time-optimal gradient waveforms for different pulse sequences and the constraints that define them are shown. Additionally, the ability to add, arbitrary motion (gradient moment) compensation or limit peripheral nerve stimulation is demonstrated. There exists a trade-off between computation time and gradient raster time, but it was observed that acceptable gradient waveforms could be generated in 1-40 ms. Gradient waveforms generated and run on the different scanners were functionally equivalent, and the images were comparable.

Conclusions: GrOpt is an open source toolbox that enables on-the-fly optimization of gradient waveform design, subject to a set of defined constraints. GrOpt was presented for a range of imaging applications, analyzed in terms of computational complexity, and implemented to run on the scanner for a multi-vendor demonstration.

KEYWORDS
gradient waveform design, magnetic resonance imaging, open source software, optimization

1 | INTRODUCTION

Gradient waveforms for Magnetic Resonance Imaging (MRI) have traditionally been designed using a combination of analytical and *ad hoc* methods. These methods work well for specific subsets of acquisition types, but usually need to be re-derived or re-designed on an application-specific basis. Additionally, as more performance or error

reduction is required for an acquisition, the analytical solution can grow to become very specific, difficult to solve, or impossible to provide a single optimal solution. The determination of time-optimal gradient waveforms can vary widely from application to application, making it difficult to be confident in the selection of a waveform generation algorithm. Additionally, even when an analytically optimal solution is achievable, it may not be used in a sequence due to implementation overhead.

An alternative to this style of analytical or *ad hoc* methodology is to generate gradient waveforms via a numerical optimization algorithm. With this approach to gradient waveform design, the goal is to define the ideal characteristics of the target gradient waveforms and enable algorithmic generation of the rasterized gradient waveforms. Previous works have demonstrated this type of optimization with a range of solvers and applications, mostly showing the ability to generate more time-optimal gradient waveforms than what may have been produced with a standard *ad hoc* implementation. These works include general usage optimizations that are primarily controlled by gradient moment constraints,[1-4] optimizations of flow and motion encoding waveforms,[5-7] and optimizations of diffusion waveforms with different options for the reduction of coil heating,[8] motion based errors,[9] eddy current errors,[10,11] and errors due to concomitant fields.[12-14]

These previous works have shown the capabilities of numerical optimization for gradient waveform design, demonstrating more time-efficient waveforms, or gradients that can reduce artifacts in the resulting images. However, existing work in this field has often been difficult to translate to real-world usage for multiple reasons. A principal reason is that these methods are often not sufficiently fast computationally to generate gradient waveforms on-the-fly (ie, within 10s of milliseconds). For most sequences, the optimal gradient waveform is different for every unique combination of sequence parameters, such as resolution, FOV, bandwidth, and slice orientation. Therefore, on-the-fly optimization enables all protocol options to remain available as the waveforms can be updated as the user is editing the sequence on the scanner. Additionally, many previous implementations were either closed source, or the source code and applications were limited in scope, so their usage was limited to a small subset of MRI problems.

The goal of this work was to design an open source software package that generates time-optimal gradient waveforms for a range of MRI protocols by providing a flexible set of input constraints to define the gradient waveforms. Additionally, constraints are made available that provide a framework to reduce imaging artifacts (eg, mitigate eddy currents) and increase patient comfort (eg, reduce peripheral nerve stimulation, PNS). These methods are also designed to solve the gradient waveform design problem using on-the-fly protocol realization in a clinical setting. The software is provided as the open source Gradient Optimization (GrOpt) toolbox available at www.github.com/cmr-group/gropt.

2 | METHODS

GrOpt is a software package that takes as input a set of constraints that define a pulse sequence, then seeks to find a feasible set of gradient waveforms that satisfies those constraints. Additional objective functions can be added to the procedure, to maximize or minimize features of the output gradient waveforms, or to find the gradient waveforms that meet the given constraints in the shortest amount of time possible. Figure 1 provides an outline of the toolbox operations, where input constraints are given, and the optimization routine seeks a gradient waveform subject to the constraints within a specified tolerance. The optimization routines are written in C for maximum compatibility on multiple scanner systems, easily compiling to C or C++ directly into sequences for vendor specific platforms. Additionally, Matlab and Python wrappers are provided for prototyping and visualizing gradient waveforms. An example of the usage with the Python wrapper is shown in Figure 2, which shows how a diffusion encoding gradient waveform can be designed with GrOpt.

2.1 | Constraints

The GrOpt toolbox uses a set of user-defined constraints to meet the targeted purpose of the pulse sequence. The most fundamental of these constraints are on gradient hardware and gradient zeroth moment $\left(\vec{M}_0, \left[\frac{\text{mT ms}}{\text{m}} \right] \right)$. Setting a constraint on maximum allowed gradient amplitude $\left(G_{\max}, \left[\frac{\text{mT}}{\text{m}} \right] \right)$ and maximum allowed gradient slew rate $\left(S_{\max}, \left[\frac{\text{mT}}{\text{m ms}} \right] \right)$ ensures that any gradient waveform generated by the optimization are within scanner's hardware limits. The gradient $\vec{M}_0$ represents the gradient area and is used to control the most common sequence needs, such as slice selection rewinding, readout prewinding, phase encoding, spoiling, and/or gradient balancing.

Additional constraints available in the toolbox allow for further control of the output gradient waveforms. Constraints are available for any higher order gradient moment $\left(\vec{M}_n, \left[\frac{\text{mT ms}^n}{\text{m}} \right] \right)$, which allows for full or partial moment nulling to reduce error due to motion (such as in flow-compensated encodings) or for motion encoding (such as in phase-contrast MRI). It is even possible, for example, to design gradient waveforms with non-zero $\vec{M}_1$ and nulled $\vec{M}_2$ for velocity-sensitive and acceleration-insensitive encoding.

Minimization and maximization terms may also be added to the optimization, allowing for waveforms to be found that are
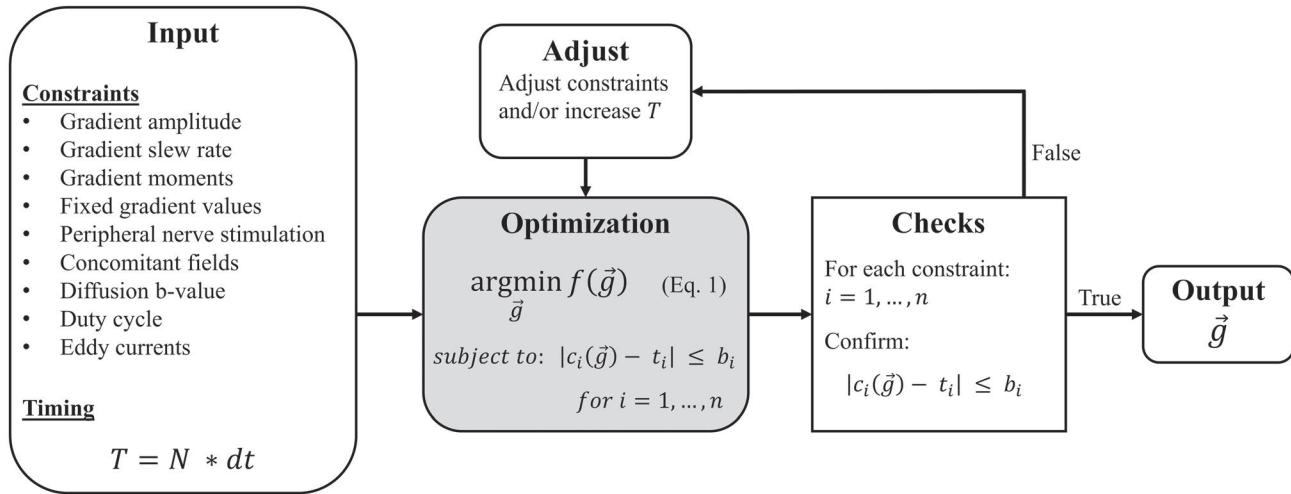
FIGURE 1 Flow chart showing the general operation of the GrOpt library. After getting user inputs, the optimization routine is run to generate acceptable waveforms. If the waveform is not possible, the routine may increase T, or change constraints as allowable to find an acceptable answer. Similarly, if a feasible answer is found, the optimization may try to lower T to find a more time optimal answer, as needed

optimized for these terms. For example, GrOpt implements a maximization of the diffusion b-value, allowing for the highest b-value to be found that fits within a given temporal footprint, while adhering to other constraints. Another example is the minimization of gradient duty cycle, allowing for waveforms to be produced that should result in the minimum possible duty cycle for a given set of constraints and sequence timings.

Another included constraint can be placed on eddy currents, thereby enabling the effects of any number of eddy current time constant responses to be suppressed. This can decrease the negative effects of eddy currents on data acquisition, such as image distortions in echo planar imaging (EPI). Additionally, peripheral nerve stimulation (PNS) can be modeled with various methods such as Schulte et al[15] and Hebrank et al,[16] both of which are implemented in GrOpt as constraints. This PNS constraint allows the maximum gradient slew rate to be used, and derated only as necessary for patient comfort and safety. Traditionally PNS is handled with a global slew rate derating, which can be substantially less time efficient. The GrOpt approach permits more time-optimal gradient waveform design, especially when compared to global slew rate derating. The constraints to the software are written in a modular fashion. This allows additional constraints to be easily added and appended to the optimization functions.

Slice obliquity is handled using either a "rotationally invariant" approach that derates the hardware limitations by $\sqrt{3}$ and defines logical gradient waveforms compatible with all scan plane rotations; or using the "rotationally variant" approach that rotates the relevant constraints and solves for optimized gradient waveforms to define physical gradient waveforms that permit the full hardware limits to be used. An example of this is shown in Supporting Information Figure S1. While most sequence

design problems can be designed on an axis-by-axis basis, the software can also optimize all three waveforms simultaneously. This is necessary for certain constraints such as PNS, which is based on a combination of all three physical gradient axes to accurately predict and constrain the solution.

2.2 | Optimization

The optimization is initialized with a gradient waveform array of ones, with fixed size N and raster time dt determined by the user. The formulation of the problem is of the form:

$$\arg\min_{\vec{g}} \quad f(\vec{g})$$
$$\text{subject to} \quad |c_i(\vec{g}) - t_i| \leq b_i \quad i = 1, \ldots, n \tag{1}$$

where $f(\vec{g})$ is the objective function and may simply be $f(\vec{g}) = 0$ to define the optimization as a feasibility problem. $c_i(\vec{g})$ represents the ith constraint, which can take several forms. In most cases c_i can be represented as a matrix or vector multiplication and may have an offset value if the target value of the constraint is not zero (t_i). b_i represents the ith bound on the inequality constraint, thereby defining the tolerance of the solution. Smaller bounds will give a more precise solution, but may require more iterations to converge, introduce non-time-optimal trade-offs, or result in the inability to minimize the objective function.

The GrOpt routine is based on the alternating direction method of multipliers (ADMM),[17] with modifications made to handle any number of constraints.[18] This allows for quick updates of each constraint sub-problem with proximal mappings, which is particularly efficient for inequality constraints

```
params = {}
# Maximize b-value for diffusion waveforms
params['mode'] = 'diff_bval'

# Hardware constraints
params['gmax'] = 50.0   # Max Gradient Amplitude [mT/m]
params['smax'] = 50.0   # Max Slewrate [mT/m/ms]

# Moment nulling
params['MMT']  = 0

# Sequence TE [ms]
params['TE'] = 60

# Desired output raster time dt [s]
params['dt'] = 200e-6

# Time from end of diffusion waveform to TE [ms]
params['T_readout'] = 16.0
# Time of excitation 90 [ms]
params['T_90'] = 4.0
# Time for 180 flip [ms]
params['T_180'] = 6.0

# Run optimization
G, dd = gropt.gropt(params, verbose=1)
plot_waveform(G, params)
```
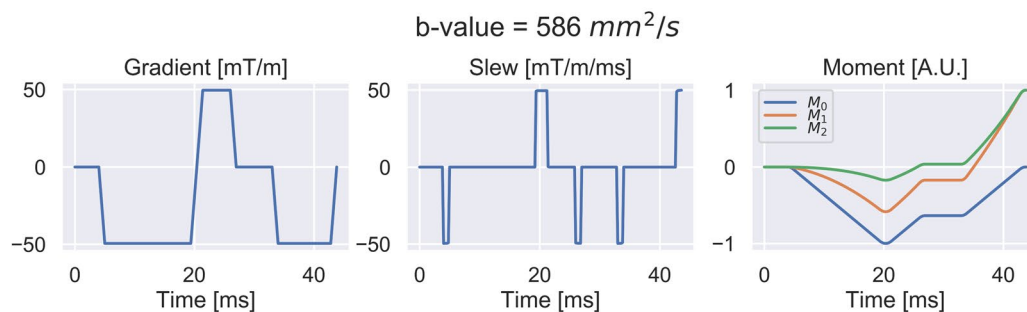

FIGURE 2 Sample usage of the GrOpt library in Python. Parameters for the desired waveform are entered: diffusion b-value maximization, hardware constraints, M_0 nulling, and the timings related to the diffusion waveform. The optimization returns the desired waveform. The figure displays the gradient waveform, the slew rate of the waveform, and M_0, M_1, and M_2 scaled to arbitrary units to compare the moment nulling properties

common to the MRI problems described in this work. To speed up the algorithm further, an adaptive relaxation method is used, which updates each constraints penalty weight and over-relaxation term every two iterations.[19,20] Stopping criteria are based on checks of feasibility of each constraint, as well as a conventional check on the convergence of the primal residual.[20] If no feasible solution exists, or cannot be found by the optimization, the process is stopped after 20,000 iterations. Previous implementations of GrOpt[7] used a similar optimization based on the Chambolle-Pock primal dual

algorithm[21] with a re-weighting scheme used to make sure constraints were met when possible. That solver produced similar results, but is slightly slower (Figure 5A), and remains an optional solver in the software for comparison and backward compatibility with previous works.

The output gradient waveforms from the optimization have a fixed duration ($T = dt*N$). Therefore, to determine time-optimal gradient waveforms an outer optimization is also required. Multiple solutions in the software exist to perform this search efficiently without requiring much additional compute time.

This is done by searching different temporal durations (T) in a multi-threaded fashion and performing the initial search of T with higher dt to solve quickly, and using those high-dt waveforms to warm start the target dt final optimization.

2.3 | Usage

The GrOpt toolbox is run by first specifying the maximum gradient amplitude ($\vec{G}_{max}$) and maximum slew rate ($\vec{S}_{max}$). A fixed gradient duration ($T = N*dt$) is required, so T and either dt or N must be entered as input. In practice, the user defines dt as either the scanner gradient raster time, or an integer multiple of it for easier interpolation. However, in this work N is used to investigate computational complexity more directly. Any number of gradient moment ($\vec{M}_n$) constraints can then be added. The gradient moment constraints can limit the gradient moment of the entire waveform or any portion of the waveform. The latter is applicable when a discrete time-optimal gradient is needed within a larger waveform, for example, when designing a spoiler gradient after readout. Fixed gradient waveform intervals can also be pre-defined, such as with slice select gradients and readout gradients which are typically known *a priori*. Importantly, in these examples it is only necessary to define the gradient flat top (not the ramps). The optimization will find the time-optimal way to connect these flat top gradients to the adjacent gradients. This avoids ramping gradient waveforms to zero unnecessarily, which costs time. These constraints are designed primarily for the gradients outside of the slice select and readout gradients, though constraints could be written to better design these gradients as well. Figure 2 shows an example use case of designing asymmetric diffusion gradients, where the code is shown to create constraints and generate a waveform. Figure 3 gives a conceptual overview of how waveforms can be defined purely based on hardware and moment constraints. Figure 4 shows an example of a gradient duty cycle minimizing waveform with a resultant gradient waveform shape that may not be obvious from conventional design approaches. The design target was a flow compensated phase encode gradient with a minimization term on gradient duty cycle $\left(\text{Duty Cycle} = \frac{\sum_{i=0}^{N} G_i^2}{NG_{max}} \right)$. Compared to simply derating gradient amplitude, GrOpt generates a waveform with the same temporal footprint and moments, but less gradient duty cycle.

2.4 | Computational complexity analysis

Compute times were tested for a range of pulse sequence applications using a range of gradient waveform constraints. Execution times were measured while running the optimization routines in C, on a standard desktop computer (Intel Core i7-7700k) by averaging the duration of 100 individual calls to the optimization function. The operation is single threaded and uses minimal memory (<10 Mb).

2.4.1 | Diffusion

Diffusion encoding gradient waveforms were optimized to eliminate gradient dead time (as can occur in spin echo EPI Diffusion Weighted Imaging (DWI) and reduce TE for higher SNR. The compute time was investigated relative to the choice of specified waveform discretization (N) and the type of gradient moment nulling (M_2, M_1 and/or M_0). Compute times for designing time-optimal diffusion encoding gradient waveforms for a range of constraints on eddy current nulling were investigated. The eddy current constraints included no nulling and nulling of eddy current time constant $\lambda = 80$ ms and nulling multiple time constants $\lambda = 40, 80$ ms.

2.4.2 | Phase contrast

Compute times for designing time-optimal phase-contrast MRI slice-selection and flow encoding gradients were also calculated. This gradient waveform design problem used a fixed and flat slice-select gradient (excluding intermediate gradient ramps), and all other gradients were designed with GrOpt. The effect of including a PNS constraint was also evaluated.[15]

2.5 | Residual moments

Defining acceptable thresholds for residual gradient moments is important when using optimization approaches. These residual moments need to be specified as a tolerance in the optimization. The optimization ends when the gradient waveform meets the target gradient moments, within a small residual moment tolerance. The choice of residual effects compute time, where a larger residual moment requires fewer iterations to reach an acceptable solution. However, a too large residual moment can cause imaging artifacts. Numerical simulations were performed to analyze the impact of residual gradient moments (due to non-zero convergence criteria) on intravoxel phase dispersion.[22] The simulation evaluated the effect of small M_0, M_1, and M_2 residuals on data accuracy and apparent diffusion coefficient (ADC) in diffusion sequence. Additionally, the computational time in relation to the allowed residual moment was measured for a motion compensated diffusion gradient waveform, with different amounts of allowed residual moments.
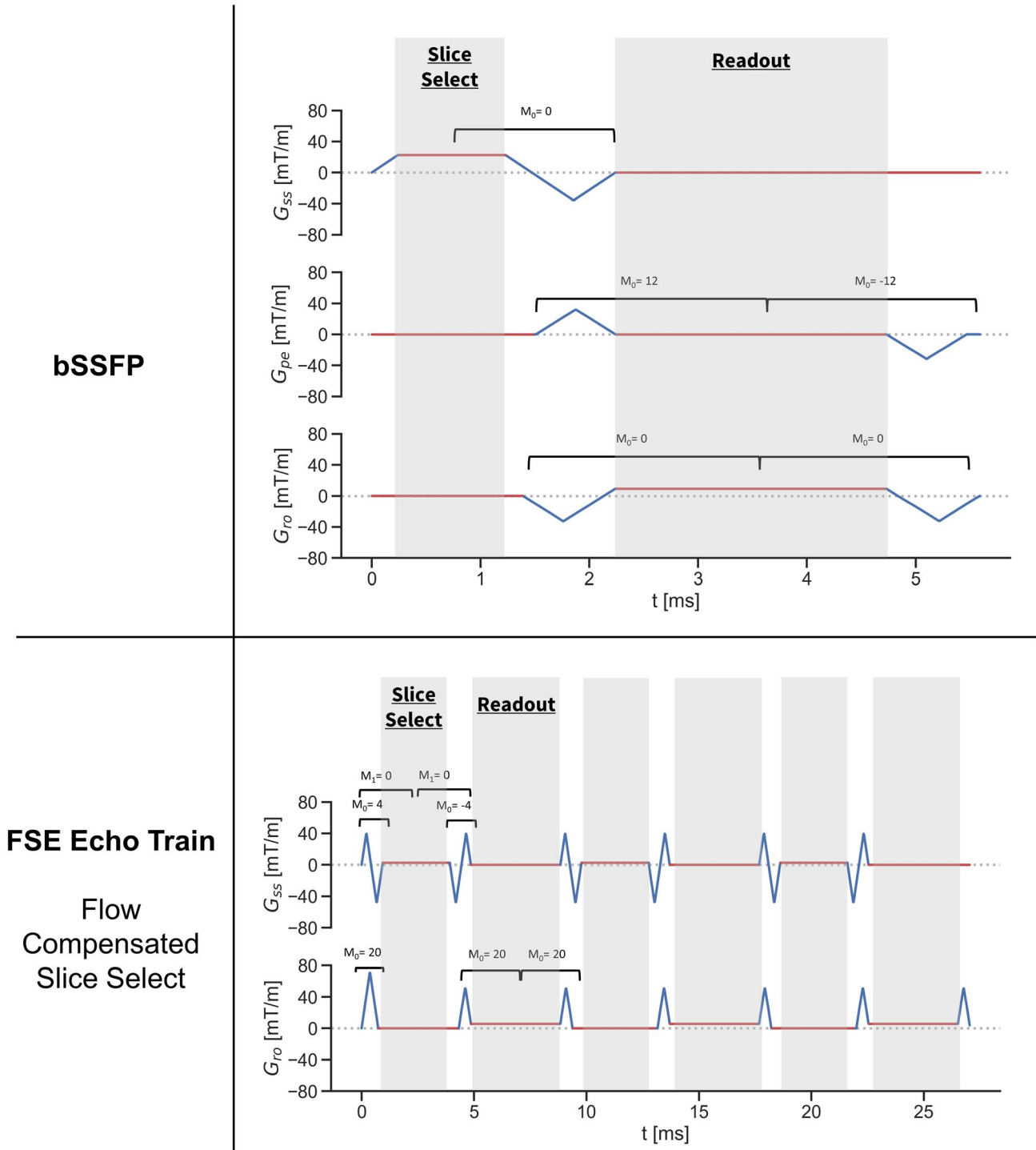
FIGURE 3 A depiction of waveforms can be generated with constraints. Blue line represents gradients that the toolbox is generating via optimization, and red lines represent fixed slice select and readout gradients that were inputted as fixed into the toolbox. The top row shows the design of a simple bSSFP TR and moment constraints required. Bottom row shows the design of a fast spin echo train with flow compensated slice selects. The FSE echo train designed was shorter than a reference product sequence, despite the waveforms being able to be optimized analytically

2.6 | Imaging experiments

In order to demonstrate multi-vendor feasibility, GrOpt was integrated directly into diffusion pulse sequences for two vendor platforms (GE EPIC DV26.0R01 and Siemens IDEA VE11C) using their respective pulse sequence programing environments. The GE pulse sequence was also developed using the KS Foundation abstraction layer (www.ksfoundationepic.org).[23] Both sequences filled in all of the diffusion gradients with waveforms generated by GrOpt, while using
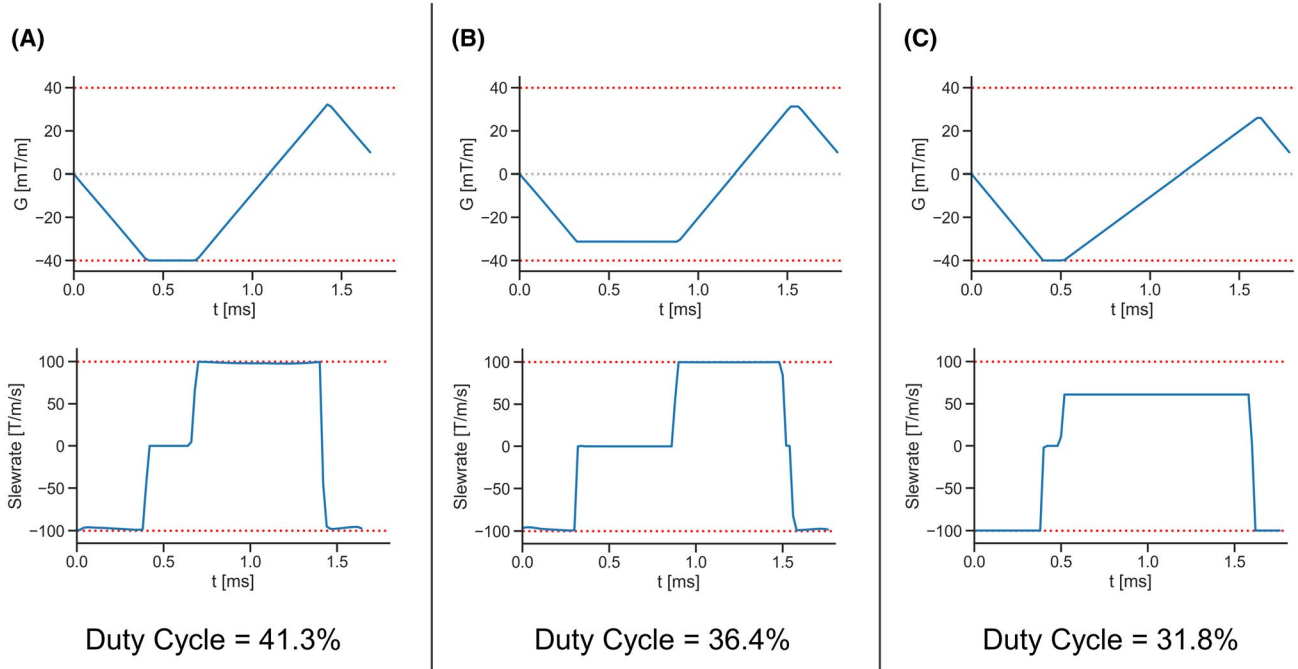
FIGURE 4 A flow compensated readout prewinding gradient with $M_0 = 11.74 \frac{\text{mTms}}{\text{m}}$ and $M_1 = 0 \frac{\text{mTms}^2}{\text{m}}$. A, Shows the fastest possible waveform. B, Shows the same waveform, but with G_{max} derated to control the gradient duty cycle. C, Shows a GrOpt waveform with the same temporal footprint as (B), but created with a duty cycle minimization

their respective conventional RF and EPI readout waveforms. The optimization was able to be run in real-time on the scanner, allowing for protocol parameters to be changed and the waveforms to update in a completely integrated manner. This means that waveforms are able to be regenerated as imaging protcol changes are made that effect the optimization, such as changing resolution. The number of calls to the GrOpt function was recorded within the sequence.

Both sequences were used to acquire brain images in a single volunteer with IRB-approved consent. The images were acquired with a 3T GE (MR750, GE Healthcare, Milwaukee, Wisconsin) and a 3T Siemens (Skyra, Siemens, Erlangen, Germany) using the GrOpt toolbox to design optimized diffusion encoding gradient waveforms for a scan protocol using 2.5 mm³ in-plane resolution, no parallel imaging, a full Fourier acquisition, and b-value $= 1000 \, \text{mm}^2/\text{s}$ for a six-direction tensor diffusion encoding scheme. The waveforms and images were compared qualitatively.

3 | RESULTS

3.1 | Computational complexity analysis

Figure 5 shows compute times for a range of fixed N and various combinations of gradient moment nulling and eddy current time constant nulling. Figure 5B shows the results of a comparison of compute time with N, as well as the impact of motion compensation on the computation times. Similar

compute times are seen for all levels of motion compensation, with small increases for additional motion constraints. Computation complexity ranged from $\mathcal{O}(n^{0.90})$ to $\mathcal{O}(n^{1.32})$. Figure 5C shows example gradient waveforms with different N and demonstrates that using smaller values can lead to <1 ms longer gradient waveforms for the same b-value. Figure 5D expands on this and shows the minimum TE needed to achieve b-value $= 1000 \, \text{mm}^2/\text{s}$. It is seen that for $N = 64$ the TEs were within 1ms of the shortest TE and with $N = 128$ the TEs were within 0.5 ms of the shortest TE. Figure 5E demonstrates the differences in compute times for a range of N and different eddy current nulling constraints for an $M_0 = 0$ diffusion encoding gradient waveform.

Figure 6 shows several results for the GrOpt optimization of a phase-contrast sequence with and without a PNS constraint. Figure 6A shows the compute time in relation to N, where without a PNS constraint the computation complexity was $\mathcal{O}(n^{1.50})$ and with the PNS constraint the complexity was $\mathcal{O}(n^{2.32})$. In all cases the choice of N had no effect on the minimum T required to meet the constraints ($T = 3.42$ ms for PNS constrained and $T = 4.14$ ms for the slew-rate derated). Figure 6B shows the waveforms that were generated, and Figure 6C shows the PNS estimate for each waveform.

3.2 | Residual moments

Figure 7 compares simulated ADC values for a range of residual moments. A $\leq 5\%$ increase in the measured ADC
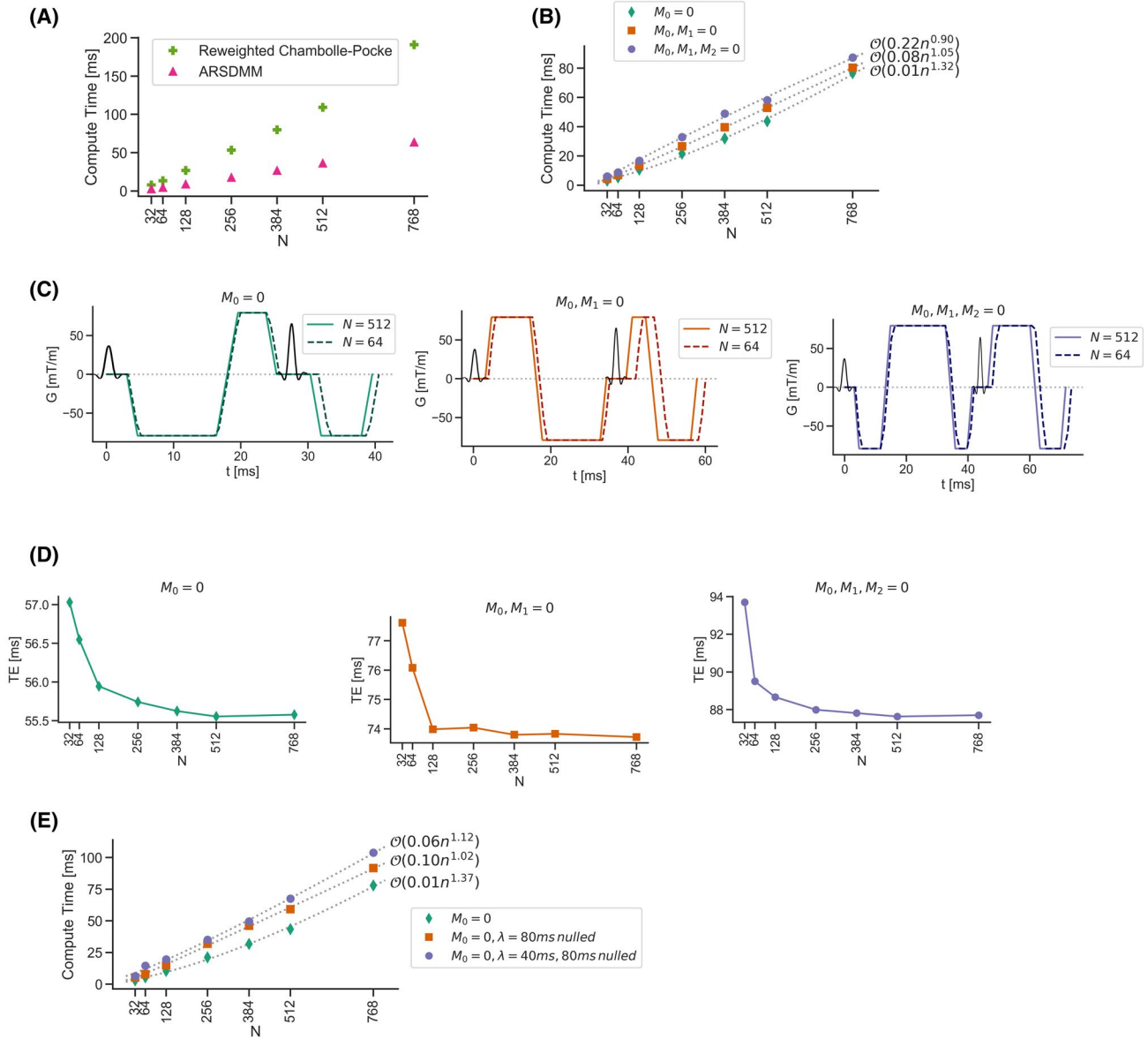
FIGURE 5 A, shows the compute time of diffusion gradient waveforms in relation to the number of computed gradient points. B, Shows the actual output waveforms from $N = 512$ and $N = 64$. C, Shows the TE that would be achieved for a given N. D, Shows the compute times in relation to N with M_0 compensation and different eddy current nulling options

was observed when the residual M_0 was on the order of 10^{-2}mT/m × ms, or the residual M_1 is on the order of 10^{-4} mT/m × ms², or the residual M_2 is on the order of 10^{-5} mT/m × ms³ (isocontour of Figure 7A-C). Figure 7D shows the impact of the allowed residual moment on computation time, where the compute time increased with smaller residual moments. As the moment constraining operation remained the same, this difference in compute time is due to an increased number of iterations required (Figure 7E).

3.3 | Imaging experiments

Figure 8 shows the GrOpt optimized diffusion encoding gradient waveforms as designed on each vendor's scanner and the resultant diffusion encoded magnitude image for the same

volunteer. The gradient waveforms show some differences due to differences in the RF pulse durations and differences in the EPI readouts implemented by each vendor. GrOpt accounts for these differences, to generate the time-optimal gradient waveforms in both cases. The resultant images are well matched qualitatively. For reference, during typical modification of protocol parameters (resolution, FOV, and undersampling factor), the GrOpt library was called about 100 times. When designing a sequence from scratch and changing most protocol parameters, the library was called about 400 times.

4 | DISCUSSION

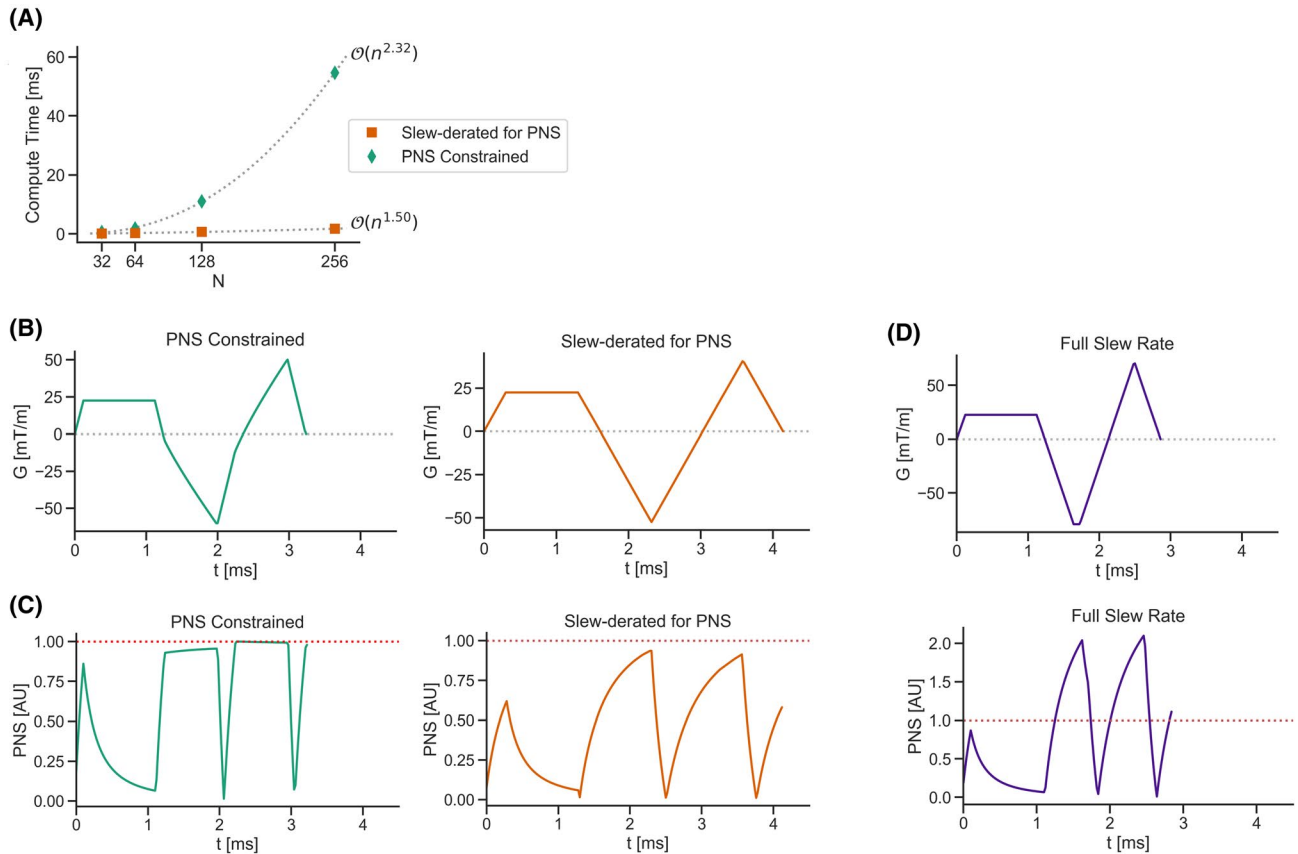The GrOpt toolbox can generate waveforms for a variety of applications with computational speeds that allow for

FIGURE 6 Flow gradient waveforms with $M_1 = 11.74 \frac{\text{mTms}^2}{\text{m}}$ corresponding to Venc $= 100 \frac{\text{cm}}{\text{s}}$, $G_{\text{max}} = 80 \frac{\text{mT}}{\text{m}}$, and $S_{\text{max}} = 200 \frac{T}{\text{ms}}$. A, shows the compute time of a flow gradient waveform in relation to the number of computed gradient points, with and without a PNS constraint. B, Shows the waveforms with and without the PNS constraint. C, shows plots of the PNS of both output waveforms

on-the-fly usage on the scanner. The toolbox can lead to rapid development of gradient waveform designs, and allows for cross-vendor use. Computational times for various real-world scenarios were investigated and it was seen that nearly time-optimal waveforms could be generated in 1-50 ms depending on the application. This represents a 100-1000x increase in compute time compared to values reported in the literature.

The analysis of diffusion encoding waveform compute time in relation to N, the number of moment constraints, and the number of eddy current constraints showed that diffusion gradient waveforms can be computed within 1 ms of optimal timings with $N > 64$. With N in the range of 64-128 this corresponds to gradient raster times of $dt = 400\text{-}800$ ms. The numerical simulations provided insight into the allowable moment residual that should be used to balance image accuracy and compute times. These settings allowed for diffusion gradient waveforms to be generated in approximately 10-50 ms.

The analysis of PC-MRI encoding gradient waveforms demonstrated that the choice of gradient raster time had very little effect on the minimum achievable gradient duration with $N > 32$. For typical encoding gradients of duration

1-10 ms, this corresponds to a gradient raster time $dt = 10\text{-}300$ ms. These parameters allowed for the GrOpt toolbox to compute gradient waveforms within about 1-10 ms. This choice of gradient size (N) and raster time (dt) may need to be adjusted for longer gradients waveforms that encompass multiple sets of encodings, such a multiecho turbo-spin echo sequence.

With the addition of multiple constraints on the gradient waveform design problem, the convexity of the solution space becomes difficult to prove. Initial tests to investigate this question showed that randomized starts all produced the same waveform,[24] however this is not a complete proof of convexity. Additionally, multiple works have shown that the ADMM style solvers used in this software package can, in many cases, be robust to solving non-convex problems, and will converge to the global minimum of the problem.[25-27] The question of whether a waveform is truly time-optimal is difficult to answer. Brute-force approaches to the problem are computationally very expensive, particularly when arbitrarily shaped gradient waveforms are considered, and when more complicated measures such as eddy current response and PNS are introduced. Nevertheless, the GrOpt toolbox will produced the fastest possible gradient waveforms that it can
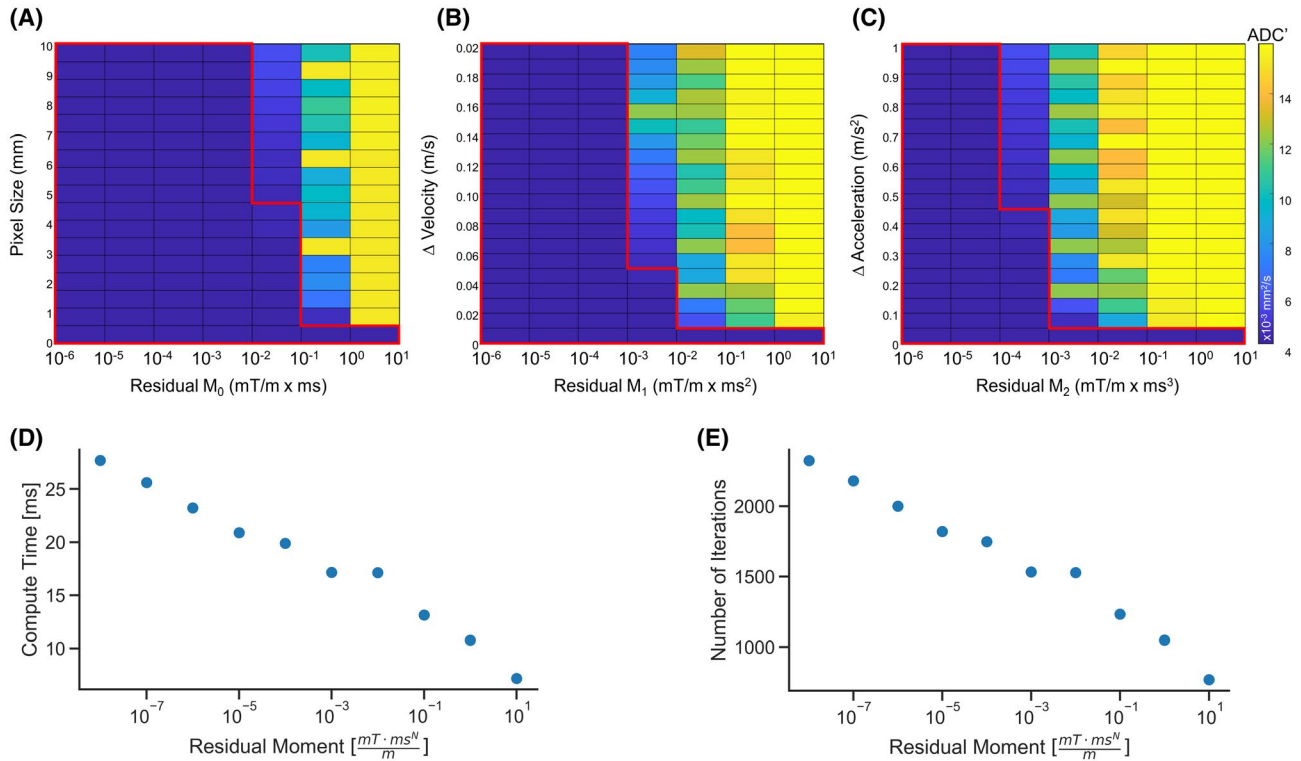
FIGURE 7 Numerical simulations showing the impact on measured ADC arising from intravoxel phase dispersion (signal loss) as a function of residual M_0 and pixel size (A), residual M_1 and intravoxel velocity gradients (B), and residual M_2 and intravoxel acceleration gradients (C). ADC' values that vary ≤5% compared to the simulated ADC (3×10^{-3}mm^2/s) and b-value (1000 mm^2/s), are outlined in red. The compute time (D) and number of iterations (E) both increase with an increasing residual moment tolerance, which arises to due stricter moment convergence criteria
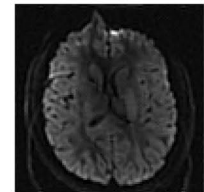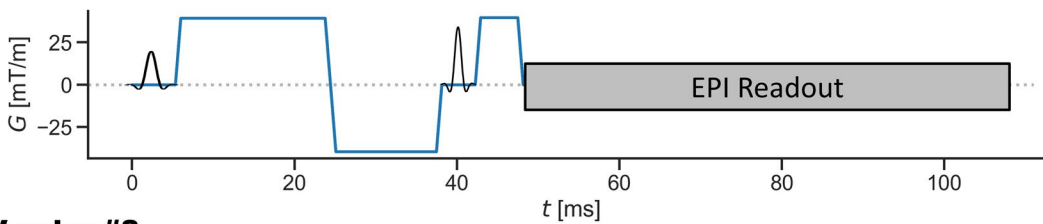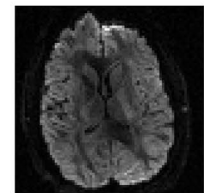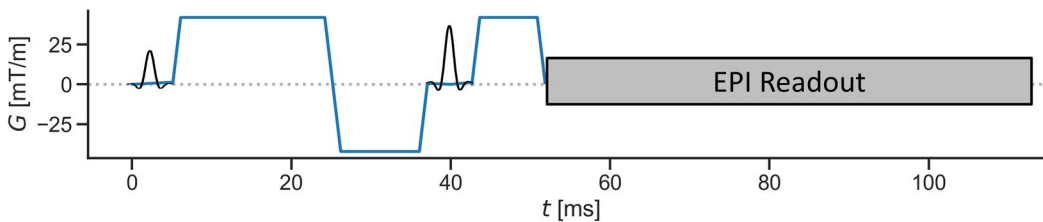

FIGURE 8 Waveforms and magnitude images from the imaging experiments. Actual waveforms are plotted as exported from their respective Programing environments. Non-relevant/proprietary elements of the waveform are blocked out

find in its search, and at this point no waveform was seen that was not time-optimal to our knowledge.

Using GrOpt in individual sequences can involve using the toolbox to create the entire TR, or to generate specific blocks of a sequence interleaved with traditional designs. With the fast compute times, re-designing waveforms on

the scanner as the protocol is edited is feasible, with little addition latency in the user interface. The total time to design the gradient waveforms with GrOpt may include a search for the optimal timings, which typically requires multiple optimizations. GrOpt accelerates the outer-loop search using: (a) a coarse-to-fine dt to quickly refine the

initial gradient waveform timing; and (b) a multi-threaded search for a feasible T. The search time can be further reduced by storing pre-computed waveforms to warm start the optimization, particularly for protocols that are frequently modified within a narrow range of parameters.

Future work for the GrOpt toolbox includes adding more constraint options to the package. This may enable limiting concomitant field terms, attenuating acoustic noise, or further constraints on gradient heating limits. Additionally, the GrOpt toolbox has not been used to optimize readout trajectories or slice-select gradients, which would allow for even more control over the optimization of entire sequence.

5 | CONCLUSION

GrOpt is open source toolbox that enables on-the-fly optimization of gradient waveform design subject to a set of defined constraints. GrOpt was presented for a range of imaging applications, analyzed in terms of computational complexity, and implemented to run on the scanner for a multi-vendor demonstration.

ORCID

Michael Loecher 🔗 https://orcid.org/0000-0002-6533-1113

REFERENCES

1. Simonetti OP, Duerk JL, Chankong V. MRI gradient waveform design by numerical optimization. *Magn Reson Med*. 1993;29:498-504.
2. Hargreaves BA, Nishimura DG, Conolly SM. Time-optimal multidimensional gradient waveform design for rapid imaging. *Magn Reson Med*. 2004;51:81-92.
3. Atalar E, McVeigh ER. Minimization of dead-periods in mri pulse sequences for imaging oblique planes. *Magn Reson Med*. 1994;32:773-777.
4. Derbyshire JA, Herzka DA, McVeigh ER, Lederman RJ. Efficient implementation of hardware-optimized gradient sequences for real-time imaging. *Magn Reson Med*. 2010;64:1814-1820.
5. Bolster BD, Atalar E. Minimizing dead-periods in flow-encoded or -compensated pulse sequences while imaging in oblique planes. *J Magn Reson Imaging*. 1999;10:183-192.
6. Middione MJ, Wu HH, Ennis DB. Convex gradient optimization for increased spatiotemporal resolution and improved accuracy in phase contrast MRI. *Magn Reson Med*. 2014;72:1552-1564.
7. Loecher M, Magrath P, Aliotta E, Ennis DB. Time-optimized 4D phase contrast MRI with real-time convex optimization of gradient waveforms and fast excitation methods. *Magn Reson Med*. 2019;82:213-224.
8. Sjölund J, Szczepankiewicz F, Nilsson M, Topgaard D, Westin C-F, Knutsson H. Constrained optimization of gradient waveforms for generalized diffusion encoding. *J Magn Reson*. 2015;261:157-168.
9. Aliotta E, Wu HH, Ennis DB. Convex optimized diffusion encoding (CODE) gradient waveforms for minimum echo time and bulk

10. Aliotta E, Moulin K, Ennis DB. Eddy current-nulled convex optimized diffusion encoding (EN-CODE) for distortion-free diffusion tensor imaging with short echo times. *Magn Reson Med*. 2018;79:663-672.
11. Shrestha M, Hok P, Nöth U, Lienerth B, Deichmann R. Optimization of diffusion-weighted single-refocused spin-echo EPI by reducing eddy-current artifacts and shortening the echo time. *Magn Reson Mater Phys Biol Med*. 2018;31:585-597. http://dx.doi.org/10.1007/s10334-018-0684-x.
12. Yang G, McNab JA. Eddy current nulled constrained optimization of isotropic diffusion encoding gradient waveforms. *Magn Reson Med*. 2019;81:1818-1832. http://dx.doi.org/10.1002/mrm.27539.
13. Peña-Nogales Ó, Zhang Y, Wang X, et al. Optimized Diffusion-Weighting Gradient Waveform Design (ODGD) formulation for motion compensation and concomitant gradient nulling. *Magn Reson Med*. 2019;81:989-1003. http://dx.doi.org/10.1002/mrm.27462.
14. Szczepankiewicz F, Westin C-F, Nilsson M. Maxwell-compensated waveform design for asymmetric diffusion encoding. *Magn Reson Med*. 2019;82:1424-1437.
15. Schulte RF, Noeske R. Peripheral nerve stimulation-optimal gradient waveform design. *Magn Reson Med*. 2015;74:518-522.
16. Hebrank FX, Gebhardt M. SAFE-model—a new method for predicting peripheral nerve stimulations in MRI. In Proceedings of the 8th Annual Meeting of ISMRM, Denver, CO, 2000. Abstract 2007.
17. Boyd S. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found Trends Mach Learn*. 2010;3:1-122. http://dx.doi.org/10.1561/2200000016.
18. Moolekamp F, Melchior P. Block-simultaneous direction method of multipliers: a proximal primal-dual splitting algorithm for nonconvex problems with multiple constraints. *Optim Eng*. 2018;19:871-885.
19. Xu Z, Figueiredo MAT, Yuan X, Studer C, Goldstein T. Adaptive relaxed ADMM: convergence theory and practical implementation. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017:7234-7243.
20. Peters B, Herrmann FJ. Algorithms and software for projections onto intersections of convex and non-convex sets with applications to inverse problems. arXiv:190209699. 2019. http://arxiv.org/abs/1902.09699. Accessed March 7, 2019.
21. Chambolle A, Pock T. A first-order primal-dual algorithm for convex problems with applications to imaging. *J Math Imaging Vision*. 2011;40:120-145.
22. Middione MJ, Loecher M, Moulin K, Ennis DB. Impact of residual gradient moments on diffusion weighted imaging. In Proceedings of the 27th Annual Meeting of ISMRM, Montreal, QC, Canada; Abstract 5483, 2019.
23. Skare S, Avventi E, Norbeck O, Ryden H. An abstraction layer for simpler epic pulse programming on ge MR systems in a clinical environment. In Proceedings of the 25th Annual Meeting of ISMRM, Honolulu, Hawaii, 2017. Abstract 3813.
24. Loecher M, Middione MJ, Moulin K, Ennis DB. A real-time solver for convex optimization of diffusion encoding gradient waveforms. In ISMRM 8th Scientific Meeting & Exhibition in Denver, Colorado, 2019. Abstract 3495.

25. Wang Y, Yin W, Zeng J. Global convergence of ADMM in nonconvex nonsmooth optimization. *J Sci Comput*. 2019;78:29-63.

26. Li G, Pong TK. Global convergence of splitting methods for nonconvex composite optimization. *SIAM J Optim*. 2015;25:2434-2460.

27. Hong M, Luo ZQ, Razaviyayn M. Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems. *SIAM J Optim*. 2016;26:337-364.

SUPPORTING INFORMATION

Additional Supporting Information may be found online in the Supporting Information section.

FIGURE S1 An overview of the methods for handling slice obliquity in GrOpt. "Logical coordinates" refer to the readout, phase encode, slice select directions. "Physical coordinates" refer to the fixed x, y, z directions of the scanner gradient system. A, Shows the specific (example) constraints used to create a 2D bSSFP sequence. B, Example rotation matrix that transforms from logical coordinates to physical coordinates (in this case a 45° rotation around the y-axis). C, Pulse sequence timing diagram used to define the constraint intervals. D, Gradient waveforms generated with G_{max} and S_{max} derated by a "rotationally invariant" factor of $\sqrt{3}$, which permits all rotations to remain within the full G_{max} and S_{max}. E, An example where the full G_{max} and S_{max} were used, which produces a feasible waveform in logical coordinates, but after rotation G_{max} is exceeded (red arrow), making the waveform infeasible. F, An example where the full G_{max} and S_{max} were used, and the waveform was solved directly in the physical coordinate system by applying the rotation matrix to the constraints. This allows for the full hardware limits to be used. The TRs for D, E, and F, are 6.30 ms (feasible), 5.50 ms (infeasible) and 5.70 ms (feasible)