



**HAL**  
open science

# Optimizing full 3D SPARKLING trajectories for high-resolution T2\*-weighted Magnetic Resonance Imaging

Chaithya Giliyar Radhakrishna, Pierre Weiss, Guillaume Daval-Fr erot, Aur elien Massire, Alexandre Vignaud, Philippe Ciuciu

► **To cite this version:**

Chaithya Giliyar Radhakrishna, Pierre Weiss, Guillaume Daval-Fr erot, Aur elien Massire, Alexandre Vignaud, et al.. Optimizing full 3D SPARKLING trajectories for high-resolution T2\*-weighted Magnetic Resonance Imaging. IEEE Transactions on Medical Imaging, 2022. hal-03090471v2

**HAL Id: hal-03090471**

**<https://inria.hal.science/hal-03090471v2>**

Submitted on 5 Aug 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destin ee au d ep ot et  a la diffusion de documents scientifiques de niveau recherche, publi es ou non,  emanant des  tablissements d'enseignement et de recherche fran ais ou  trangers, des laboratoires publics ou priv es.

# Optimizing full 3D SPARKLING trajectories for high-resolution $T_2^*$ -weighted Magnetic Resonance Imaging

Chaithya GR, Pierre Weiss, Guillaume Daval-Fr erot, Aur elien Massire, Alexandre Vignaud and Philippe Ciuciu *Senior Member, IEEE*

**Abstract**—The Spreading Projection Algorithm for Rapid K-space sampling, or SPARKLING, is an optimization-driven method that has been recently introduced for accelerated 2D  $T_2^*$ -w MRI using compressed sensing. It has then been extended to address 3D imaging using either stacks of 2D sampling patterns or a local 3D strategy that optimizes a single sampling trajectory at a time. 2D SPARKLING actually performs variable density sampling (VDS) along a prescribed target density while maximizing sampling efficiency and meeting the gradient-based hardware constraints. However, 3D SPARKLING has remained limited in terms of acceleration factors along the third dimension if one wants to preserve a peaky point spread function (PSF) and thus good image quality. In this paper, in order to achieve higher acceleration factors in 3D imaging while preserving image quality, we propose a new efficient algorithm that performs optimization on full 3D SPARKLING. The proposed implementation based on fast multipole methods (FMM) allows us to design sampling patterns with up to  $10^7$  k-space samples, thus opening the door to 3D VDS. We compare multi-CPU and GPU implementations and demonstrate that the latter is optimal for 3D imaging in the high-resolution acquisition regime (600 $\mu$ m isotropic). Finally, we show that this novel optimization for full 3D SPARKLING outperforms stacking strategies or 3D twisted projection imaging through retrospective and prospective studies on NIST phantom and in vivo brain scans at 3 Tesla. Overall the proposed method allows for 2.5-3.75x shorter scan times compared to GRAPPA-4 parallel imaging acquisition at 3 Tesla without compromising image quality.

**Index Terms**—3D MRI, optimization, non-Cartesian, compressed sensing, acceleration.

## I. INTRODUCTION

The quest for efficient sampling strategies has been a major challenge in MRI since its invention. The theory of compressed sensing (CS) [1] boosted this quest by providing significant theoretical insights. It was proved and observed empirically that for under-sampled acquisitions and approximately sparse signals in an orthogonal basis, an efficient implementation relies on trajectories with a variable density in k-space: the lower frequencies (center of k-space) have to be sampled more

densely than the higher at the borders of k-space [2]–[5]. Non-Cartesian k-space trajectories (e.g. spiral, radial, rosette, etc.) [6]–[11] have been proposed for accelerated and robust-to-motion 2D imaging, prior to the existence of theoretical foundations. While being compliant with scanner hardware constraints on the gradients, these trajectories do not sample the k-space according to a well controlled target sampling density. For instance, in spiral imaging, fulfilling these constraints transforms an initially prescribed density into another one [12, p. 97]. Recently, the SPARKLING algorithm [13]–[15] has been shown to automatically generate optimized non-Cartesian sampling patterns compatible with MR hardware constraints on maximum gradient amplitude and slew rate. SPARKLING optimally samples the k-space (see [2], [3]) with a controlled distribution of samples (e.g., variable density) and a locally uniform k-space coverage.

However, for the sake of signal-to-noise ratio (SNR), 3 dimensional (3D) imaging is preferred to reach isotropic high-resolution imaging (e.g. 600 $\mu$ m isotropic). In this regard, multiple approaches have been utilized to efficiently down-sample 3D k-space. Some of these involve a combination of a readout in the z-direction with a 2D under-sampled mask based on Poisson disk sampling [16]. Additional attempts on full 3D readouts were proposed like 3D radial trajectory [17], 3D cones [18], twisted projections [19] and hybrid radial-cones [20]. However, these trajectories were primarily based on parameterizing a k-space sampling curve, and the final sampling pattern was not optimized with respect to the reconstruction quality. Some recent studies explored how to optimize the sampling pattern [21]–[23], but did not include a clear sampling criterion in order to maximize the image reconstruction quality.

Other methodologies in the literature involved stacking a 2D under-sampled trajectory like stack of stars [24], [25], stack of spirals [18], [26] and stack of 2D SPARKLING [27]. Uniform (i.e. cylindrical) stacking is usually implemented even though a spherical strategy, with a number of shots varying as a function of the latitude plan, was shown to be beneficial on image quality for SPARKLING trajectories [27]. Further in [27], a local 3D SPARKLING approach was proposed by designing a single trajectory within a cone obtained from

Chaithya G R, G. Daval-Fr erot and P. Ciuciu are with CEA, NeuroSpin, F-91191 Gif-sur-Yvette, cedex, France and Inria, Parietal, Universit  Paris-Saclay, F-91120 Palaiseau, France, whereas A. Vignaud is only affiliated with CEA, NeuroSpin (Email: philippe.ciuciu@cea.fr). Pierre Weiss is with CNRS, IMT (UMR 5219) and ITAV (USR 3505), F-31106, Toulouse. Aur elien Massire and G. Daval-Fr erot are with Siemens Healthcare SAS, F-93527 Saint-Denis, cedex, France.

a parcellation of the 3D spherical k-space. Then all the cones covering a given elevation plane were filled up using the replication of the resulting trajectory. However, this method did not ensure a locally uniform sampling pattern at the boundaries of cones as the problem was solved locally.

The recent rise of machine and deep learning has impacted the literature on MRI sampling [28]–[35]. These approaches rely on supervised learning techniques, which means that they need a ground truth corresponding to fully sampled data (like fastMRI dataset [36]), to learn an optimal under-sampling pattern, whether it is Cartesian or not. In [28], [29], the authors explore the use of experimental design to choose the best subset of prescribed trajectories. Although there are substantial differences between these two methodologies (deterministic vs Bayesian, offline vs online design, etc.) they share a similar theoretical background with ours in that sparsity is the key underlying hypothesis. In particular, [28], [29] use the Cramér-Rao bound for sparse signals [37] as a tailored optimality criteria. However, such methods are computationally demanding as they try to solve a nonconvex integer programming problem. Hence, in a given time budget this limits the exploration and the potential number of prescribed trajectories.

In [30], [31], the authors proposed to step away from the theoretical consideration in CS and adopted a purely data-driven approach. The authors proposed to find an optimal subset of Cartesian sampling lines by using a greedy algorithm aimed at maximizing the SNR. This algorithm can automatically adapt to different reconstruction algorithms and optimality criteria, but its use is limited to Cartesian imaging. More recent approaches made some advance on learning gridded sampling patterns [32], [33]. Additionally, to the best of our knowledge, the only works that have learned a non-Cartesian trajectory under hardware constraints are PILOT [34] BJORK [38] and 3D FLAT [35] for 2D and 3D imaging, respectively. These works seem very promising despite significant theoretical and numerical challenges with a combinatorial number of local minimizers [39]. In contrast, our work is based on clear theoretical considerations with provable convergence [40] in short computing times [41]. Of interest, let us notice that the sampling patterns generated by these methods resemble the SPARKLING ones very much [34], [38], suggesting that the main ideas behind are now reaching a mature and reliable state. Nonetheless, it is worth noting that none of these approaches has been prospectively validated on real 3D acquisitions. For all these reasons, these works won't be discussed any further in this paper.

In this paper, for the first time, we solve the SPARKLING optimization fully in 3 dimensions. First, in Sec II, we remind the optimization problem to be solved for generating SPARKLING trajectories. Then we focus on major computational bottlenecks that prevented us from scaling the original solution to 3D and provide detail on our main contributions. One key ingredient in

SPARKLING is the setting of the right target sampling density. The latter may vary as a function of the resolution, the acceleration factor and the object to be scanned. For that purpose, we parameterize radially decaying densities by two parameters (cut-off, decay) and find the optimal density through grid search over pairs of parameters. This study can be conducted on retrospective analysis and then the sought optimal density can be used further in prospective acquisitions. In Sec. IV, we present the experimental data sets on which the numerical studies are performed later on for validation and comparison purposes. In this regard, we carry out retrospective and prospective analysis on NIST phantom data collected at 3 Tesla (3T). Then we perform prospective in vivo brain imaging acquisitions on a healthy adult volunteer still at 3T and compare the proposed full 3D SPARKLING with the existing spherical stack of 2D SPARKLING. We do not include any comparison with 3D radial sampling scheme or stack of spirals as this was already done in [27]. However, we do compare our trajectories with improved 3D non-Cartesian trajectories, namely twisted projection imaging (TPI) [19]. TPI trajectories have better k-space coverage as compared to full 3D radial sampling scheme as these trajectories shift to pappus spirals after a fraction of readout.

## II. THEORY

In this section we briefly introduce the SPARKLING algorithm as described in [15]. We detail the particular steps involved in the optimization process. We point to some computational bottlenecks in each of these steps. Later, we describe the methods used to overcome these computational challenges, thereby allowing us to scale the problem to 3 dimensions. Most of the theoretical aspects are directly based on earlier works in [13], [14], [42], which can be consulted for the problem description and derivations of (4).

### A. 3D K-space sampling

A 3D k-space sampling pattern  $\mathbf{K}$  is usually composed of several shots or curves, say  $N_c$ ,  $\mathbf{K} = (\mathbf{k}_i)_{i=1}^{N_c}$ , where each 3D shot  $\mathbf{k}_i(t) = (k_{i,x}(t), k_{i,y}(t), k_{i,z}(t))$ , is controlled by magnetic field gradients  $\mathbf{G}_i(t) = (G_{i,x}(t), G_{i,y}(t), G_{i,z}(t))$  as follows:

$$\mathbf{k}_i(t) = \frac{\gamma}{2\pi} \int_0^t \mathbf{G}_i(\tau) d\tau, \quad (1)$$

with  $\gamma$  the gyro-magnetic ratio ( $\gamma = 42.57\text{MHz/T}$  for proton imaging). In practice, throughout the readout duration  $T_{\text{obs}}$ , we sample each shot  $\mathbf{k}_i(t)$  by a time period  $\Delta t$ , the gradient raster time as the scanner gradient hardware can play gradients at this pace. In the rest of the section, we refer to location of the k-space samples  $\mathbf{K}$  as the samples on gradient raster points. Then the number of gradient time steps is given by  $N_s = \left\lfloor \frac{T_{\text{obs}}}{\Delta t} \right\rfloor$  and the full 3D sampling pattern  $\mathbf{K}$  finally consists of

$p = N_c \times N_s$  points. Additionally, we limit ourselves to a long readout ( $T_{\text{obs}} \simeq 20\text{ms}$ ) for  $T_2^*$ -weighted imaging, as this allows the trajectory to be longer and maximally explore the k-space.

The k-space domain for a 3D MR volume of size  $N_x \times N_y \times N_z$  over a field of view  $\text{FOV}_x \times \text{FOV}_y \times \text{FOV}_z$ , is defined within  $[-K_{\text{max}}^x, K_{\text{max}}^x] \times [-K_{\text{max}}^y, K_{\text{max}}^y] \times [-K_{\text{max}}^z, K_{\text{max}}^z]$ , with  $K_{\text{max}}^\ell = \frac{N_\ell}{2\text{FOV}_\ell}$  and  $\ell = x, y, z$ . For the sake of simplicity, in what follows we assume the same spatial resolution along the three dimensions so  $K_{\text{max}}^x = K_{\text{max}}^y = K_{\text{max}}^z = K_{\text{max}}$  even though we meet different matrix and FOV dimensions ( $N^z \neq (N^x = N^y)$ ) and  $\text{FOV}^z \neq (\text{FOV}^x = \text{FOV}^y)$ . Hereafter, the 3D k-space domain will be normalized to  $\Omega = [-1, 1]^3$ .

Hardware constraints on the maximum gradient amplitude ( $G_{\text{max}}$ ) and slew rate ( $S_{\text{max}}$ ) induce limitations in trajectory speed and acceleration. These limits can be expressed as box constraints on the amplitude of the discrete derivatives of the k-space trajectory  $(\mathbf{k}_i[n])_{n=0}^{N_s-1}$ . These hardware constraints can be applied on a per dimension basis, giving rotation variant (RV) constraints, whose resulting trajectories cannot be run on the scanner if the FOV is rotated. Due to this limitation, in this work, we focus on rotation invariant speed and acceleration constraints which can be expressed as follows:<sup>1</sup>

$$\mathcal{Q}_{\alpha, \beta}^{N_c} = \left\{ \begin{array}{l} \forall i = 1, \dots, N_c, \quad \mathbf{k}_i \in \mathbb{R}^{3 \times N_s}, \\ \mathbf{A}\mathbf{k}_i = \mathbf{v}, \\ \|\mathbf{k}_i\|_\infty \leq 1, \|\dot{\mathbf{k}}_i\|_{2, \infty} \leq \alpha, \|\ddot{\mathbf{k}}_i\|_{2, \infty} \leq \beta, \end{array} \right\} \quad (2)$$

where

$$\dot{\mathbf{k}}_i[n] = \frac{\mathbf{k}_i[n] - \mathbf{k}_i[n-1]}{\Delta t}$$

$$\ddot{\mathbf{k}}_i[n] = \frac{\mathbf{k}_i[n+1] - 2\mathbf{k}_i[n] + \mathbf{k}_i[n-1]}{\Delta t^2}$$

$$\|\mathbf{c}\|_{2, \infty} = \sup_{0 \leq n \leq N_s-1} \left( |c_x[n]|^2 + |c_y[n]|^2 + |c_z[n]|^2 \right)^{1/2},$$

for all  $\mathbf{c} \in \Omega^{N_s}$  and  $(\alpha, \beta)$  are obtained by normalizing hardware and Nyquist constraints to the sampling domain  $\Omega$  (from [15]):

$$\alpha = \frac{1}{K_{\text{max}}} \min \left( \frac{\gamma G_{\text{max}}}{2\pi}, \frac{1}{\text{FOV} \cdot \delta t} \right) \quad (3a)$$

$$\beta = \frac{\gamma S_{\text{max}}}{2\pi K_{\text{max}}} \quad (3b)$$

The purpose of  $\mathbf{A}$  and  $\mathbf{v}$  are to model linear constraints on the trajectory, like the TE point constraint, which ensures that each trajectory passes through center of k-space at echo time (TE). More sophisticated linear constraints (e.g. gradient moment nulling) can be modeled too, see details in [42]. The normalized constraint  $\alpha \delta t \leq \frac{1}{\text{FOV} \times K_{\text{max}}}$  ensures that the distance between k-space locations associated with two consecutive measurements,

<sup>1</sup>In [42], we have also dealt with the case of RV constraints where the  $\ell_\infty$ -norm replaces the mixed  $\ell_{2, \infty}$ -norm used here.

sampled by the analog-to-digital converter (ADC) at the dwell time period  $\delta t$  (see Subsection IV-A for the relationship between  $\Delta t$  and  $\delta t$ ), is lower than the Nyquist rate, which is essential to discard some undesired filtering effects [43].

### B. 3D SPARKLING formulation

Let  $\rho : \Omega \rightarrow \mathbb{R}$  denote a target sampling density, with  $\rho(x) \geq 0$  for all  $x$  and  $\int \rho(x) dx = 1$ . Following previous works [15], [44]–[46], we obtain  $\mathbf{K} \in \Omega^p$  by solving:

$$\hat{\mathbf{K}} = \arg \min_{\mathbf{K} \in \mathcal{Q}_{\alpha, \beta}^{N_c}} F_p(\mathbf{K}) = [F_p^a(\mathbf{K}) - F_p^r(\mathbf{K})] \quad (4)$$

with  $\mathcal{Q}_{\alpha, \beta}^{N_c}$  being the constraint set for the  $N_c$  shots. Here we remind that  $p$  refers to the total number of k-space samples (or particles), so  $p = N_c \times N_s$ .

The term  $F_p^a(\mathbf{K})$  corresponds to an attraction term which ensures that the final distribution of the k-space sampling points follows the target density  $\rho$  and  $F_p^r(\mathbf{K})$  is the repulsion term to ensure that the sampling is locally uniform and that we don't have any local clusters. These terms are defined as:

$$F_p^a(\mathbf{K}) = \frac{1}{p} \sum_{i=1}^p \int_{\Omega} H(x - \mathbf{K}[i]) \rho(x) dx \quad (5a)$$

$$F_p^r(\mathbf{K}) = \frac{1}{2p^2} \sum_{1 \leq i, j \leq p} H(\mathbf{K}[i] - \mathbf{K}[j]) \quad (5b)$$

where  $\mathbf{K}[i] \in \Omega$  describe the locations of k-space samples in a shot-based lexicographical order  $[\mathbf{k}_1, \dots, \mathbf{k}_{N_c}]$ . The function  $H$  is a well chosen kernel, typically  $H(x) = \|x\|_2$ . Note that alternative choices such as  $H(x) = \log(x)$  have been also investigated in [47]. The minimization problem (4) can be attacked by various nonlinear programming procedures. In this work, we propose to use a projected gradient descent as described below:

$$\mathbf{K}^{(t+1)} = \Pi_{\mathcal{Q}_{\alpha, \beta}^{N_c}} \left( \mathbf{K}^{(t)} - \eta^{(t)} \nabla F_p(\mathbf{K}^{(t)}) \right) \quad (6)$$

The computational bottlenecks in (4) involve the calculation of  $\nabla F_p(\mathbf{K}) = \nabla F_p^a(\mathbf{K}) - \nabla F_p^r(\mathbf{K})$ , and the projection of each shot onto the constraint set  $\mathcal{Q}_{\alpha, \beta}^{N_c}$ .

### C. Gradient Descent Step

In what follows, we provide details about the calculation of  $F_p$  and  $\nabla F_p$ .

1) *Evaluating  $F_p^a$  and its gradient*: To calculate the attraction term and its gradient, we can re-write (5a) as:

$$F_p^a(\mathbf{K}) = \frac{1}{p} \sum_{i=1}^p (H \star \rho)(\mathbf{K}[i]) \quad (7)$$

where  $\star$  denotes the convolution-product in the continuous setting. The main difficulty is thus to quickly evaluate  $(H \star \rho)(x)$  (optional, if we want to compute the cost function) and its derivatives. To this end, we

discretize the target sampling distribution  $\rho$  as follows:

$$\rho[i, j, k] = \rho(i/N, j/N, k/N) \quad (8)$$

where  $i, j, k \in [-N, N]$ , and  $N \in \mathbb{N}$  describes the number of discretization points. We typically take  $N$  twice as large as  $\max(N_x, N_y, N_z)$  to define the density at a better resolution than the image size. Similarly we compute a discrete version of the filter  $H$  as:

$$\mathbf{H}[i, j, k] = H(i/N, j/N, k/N) \quad (9)$$

Letting  $*$  denote the discrete convolution-product, we use the following approximation

$$(H \star \rho)(\mathbf{K}[i]) \simeq \mathcal{I}(\mathbf{H} * \rho)(\mathbf{K}[i]), \quad (10)$$

where  $\mathcal{I} : R^{(2N+1)^3} \rightarrow C^0(\Omega)$  denotes a tri-linear interpolant function. Hence, the computation of  $F_p^a(\mathbf{K})$  requires to precompute  $\mathbf{H} * \rho$  on a discrete grid with fast Fourier transforms once for all. The computation of the sum in (7) then has a complexity  $O(p)$ , which is linear in the number of particles.

Similarly, the computation of  $\nabla F_p^a(\mathbf{K})$  involves the calculation of the partial derivatives  $\partial_{i,\ell} F_p^a(\mathbf{K})$  where  $1 \leq i \leq p$  is the index of a particle and  $1 \leq \ell \leq 3$  the index of a dimension. According to (7), the partial derivative is:

$$\partial_{i,\ell} F_p^a(\mathbf{K}) = \frac{1}{p} (\partial_\ell H \star \rho)(\mathbf{K}[i]) \quad (11)$$

Thus, letting  $\nabla \mathbf{H} \in \mathbb{R}^{(2N+1)^3 \times 3}$  denote a discretization of  $\nabla H$ , we can precompute the discrete vector field  $\nabla \mathbf{H} * \rho$  on a Cartesian grid using fast Fourier transforms and then use a tri-linear interpolant to evaluate it off the grid.

**2) Evaluating  $F^r$  and its gradient:** The problem addressed here is to compute  $F_p^r$  and  $\nabla F_p^r(\mathbf{K})$ . For purposes of simplification, we introduce  $r_{ij} = \|\mathbf{K}[i] - \mathbf{K}[j]\|_2$  and consider  $H$  to be a radial function depending on  $r_{ij}$  only. Letting  $K_\ell$  denote the spatial components of  $\mathbf{K} = [K_1, K_2, K_3]$ , we get:

$$F_p^r(\mathbf{K}) = \sum_{1 \leq i, j \leq p} H(r_{ij}) \quad (12a)$$

$$\partial_{i,\ell} F_p^r(\mathbf{K}) = \frac{1}{p^2} \sum_{j \neq i} \left( \frac{K_\ell[i] - K_\ell[j]}{r_{ij}} \right) \partial_\ell H(r_{ij}) \quad (12b)$$

The evaluation of all the components of the gradient require  $O(p^2)$  computations, where  $p$  can reach  $10^8$  for high resolution imaging. An efficient implementation is therefore critical. In this work, we explored two possibilities.

*a) Brute force calculation using PyKeops:* The computation of (12a) and (12b) can be highly parallelized, which is amenable to efficient GPU implementations. Carrying out such computations on array centric frameworks like PyTorch and Tensorflow would require the use of huge  $p \times p$ -dimensional matrices. This would result in a large memory footprint, much larger than what is typically available on current modern GPUs. For the

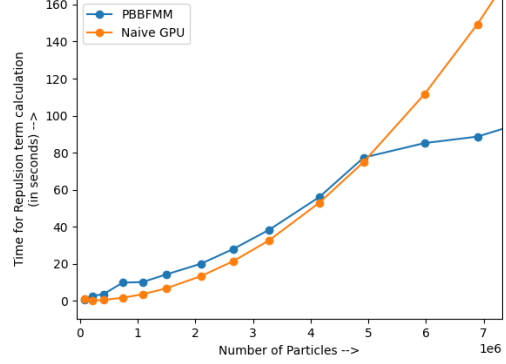


Fig. 1: Computation times for the repulsion term  $F^r$  as a function of the number of particles  $p$ .

sake of efficient memory usage, we used `PyKeops`, a library that permits low cost calculations of large kernel operations [48]. `PyKeops` carries out the naive and direct computations using online map reduce schemes from CUDA routines for summations. Due to this, the whole matrices are not stored in the GPU memory, but rather just the final results.

*b) Fast Multipole Methods:* Sums of the form (12a) and (12b) appear in many n-body problems and can be computed efficiently using Fast Multipole Methods (FMM) [49]. Given a set of positions  $\mathbf{K}[i]$ , a kernel  $\Psi : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$  and a set of weights  $\mathbf{w} \in \mathbb{R}^p$ , the FMM method allows for the efficient computation of vector  $\mathbf{v}$  of the form

$$\mathbf{v}[i] = \sum_{j=1}^p \Psi(\mathbf{K}[i], \mathbf{K}[j]) \mathbf{w}[j].$$

The FMM utilizes a multipole expansion of the kernel  $\Psi$ , which allows for a hierarchical grouping of closely spaced k-space points and treat them as a single source. This results in a massive acceleration of the above computation with a complexity  $O(p \log p / \epsilon)$ , where  $\epsilon$  is a user-prescribed precision. For our implementations, we used the Parallel Black box FMM [50], [51] in 3D (`PBBFMM3D2`), which can be run with any arbitrary kernel  $\Psi$ .

To evaluate the cost function  $F_p^r$ , we only need to set

$$\Psi(\mathbf{K}[i], \mathbf{K}[j]) = H(r_{ij}) \text{ and } \mathbf{w}[j] = 1, \forall j.$$

To evaluate the gradient  $(\partial_{i,\ell} F_p^r(\mathbf{K}))_i$ , we set  $\mathbf{w}[j] = 1$  and

$$\Psi(\mathbf{K}[i], \mathbf{K}[j]) = \left( \frac{K_\ell[i] - K_\ell[j]}{r_{ij}} \right) \partial_\ell H(r_{ij}).$$

*c) Comparisons:* From Fig 1, we see that naive GPU implementations on `PyKeops` outperforms the `PBBFMM3D` implementation for  $p < 5 \times 10^6$ . Beyond this value, `PBBFMM3D` gets faster. It is likely that faster

<sup>2</sup>see <https://github.com/ruoxi-wang/PBBFMM3D>.

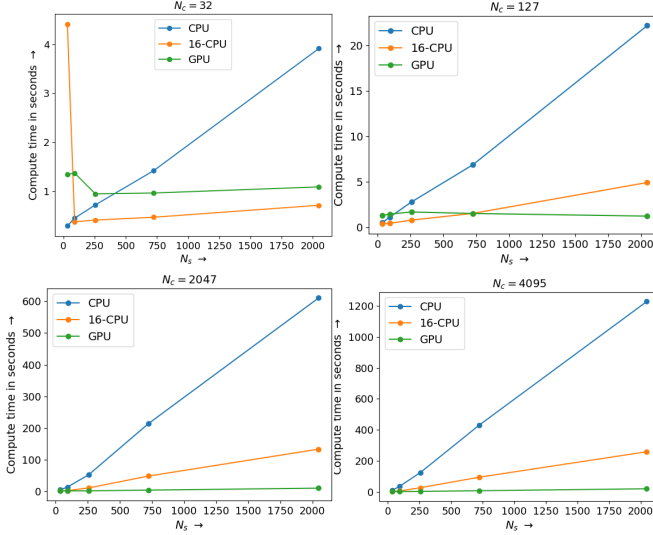


Fig. 2: Computation times for varying  $N_s$  and  $N_c$  for the projection step  $\Pi_{Q_{\alpha,\beta}^{N_c}}(\cdot, n_{pit})$  that was run over  $n_{pit} = 200$  iterations with  $G_{\max} = 40\text{mT/m}$  and  $S_{\max} = 180\text{T/m/s}$ .

computations with the FMM would be obtained with a GPU implementation. Unfortunately, we did not find any robust and efficient GPU implementation of FMM.

3) *Choice of step size*: In our implementation, we use a combination of two step sizes. In the first 20 iterations, we use a fixed step size:  $\eta^{(t)} = \eta$ . As analyzed in [14], this strategy provides a convergence guarantee to a local minimizer of the cost function given that:

- The kernel  $H$  has a  $L$ -Lipschitz continuous gradient.
- The step size is inversely proportional to the Lipschitz constant.

These conditions are satisfied with a regularized norm of the form  $H(r) = \sqrt{r^2 + \epsilon^2}$ . We can then set  $\eta^{(t)}$  proportional to  $\epsilon$  (i.e.  $6.25^{-2}$ ). The value of  $\epsilon$  can be chosen as a fraction of the minimal distance between two points at a stationary point.

A constant step size is too conservative and a faster convergence can be obtained using a second-order dynamics close to the minimizer. This justifies switching to a Barzilai–Borwein [52] after first few iterations. Few theoretical guarantees are available for this technique, but it significantly accelerates the convergence empirically.

#### D. Projection step

The projection step in (6) for a general single k-space shot onto a given constraint set parameterized by  $(\alpha, \beta)$  has been explored in [14], [42]. For our implementations, we needed to extend the single shot iterative procedure called Algorithm 1 in [42] to projecting  $N_c$  shots (see Sec. S1 for details). Note that the actual projection of a k-space shot is independent of other shots, and thus the computation can be done in parallel. Hence, we have implemented this step both on multi-CPU and GPU. To efficiently utilize a GPU, we used the CuPy module [53].

The computation times with different implementations for varying  $N_c$  and  $N_s$  are shown in Fig. 2. We found that the computation times vary linearly with  $N_s$  and are drastically reduced for the GPU implementation compared to the CPU versions (single and multicore). At lower  $N_s$ , we found that m-CPU and GPU implementations are latency bound, giving anomalously higher computation times. However, the speedup obtained for larger  $N_s$  offsets these anomalous cases, giving an overall efficient implementation.

#### E. Multi-resolution strategy for faster convergence

In order to allow for the algorithm to reach faster convergence and lead to a better approximation of the target density, a multi-resolution approach as described in [54] was implemented. Under this methodology, the optimization of the sampling pattern was carried out on down-sampled curves. The interpolated solution was later used as a warm restart for the up-sampled problem. Our implementations involved dyadic scaling and up-scaling through simple linear interpolation of k-space shots. Let the linear interpolator be of the form  $\mathcal{L}_{2^d} : \Omega^d \rightarrow \Omega^{2^d}$ . We define the parameter  $N_d$  as the number of decimation steps in the algorithm. Note that the constraint space needs to be equally scaled with the problem, which results in scaling the  $\alpha$  and  $\beta$  constraints mentioned in (3) to:

$$\alpha = \frac{\gamma G_{\max} 2^{N_d}}{2\pi K_{\max}}, \quad \beta = \frac{\gamma S_{\max} 2^{N_d}}{2\pi K_{\max}} \quad (13)$$

As we move through the dyadic decimation steps and up-sample the curve, these constraints are halved.

#### F. Overall algorithm

Algorithm 1 on the following page summarizes how to concretely compute the SPARKLING solution along with multi-resolution steps described in Sec. II-E. Further, for the sake of completeness, we have briefly summarized the multi-shot projection step  $\Pi_{Q_{\alpha,\beta}^{N_c}}$  in Sec. S1 of Supplementary Material. However, for more details on the iterative procedure involved in the projection step, the reader can refer to [42].

### III. NUMERICAL EXPERIMENTS AND DATA ACQUISITION

The sampling patterns were obtained by carrying out projected gradient descent as described above. With the above described improvements, the SPARKLING Generation time was just 10 minutes for 2D and nearly 6-9 hours for 3D on NVIDIA V100 with 5120 CUDA cores and 16GB DDR5X memory.

#### A. SPARKLING: a Python package

In the ethos of reproducible research and to move forward into better optimized patterns for MRI acquisition, all the implementations as described above is

---

**Algorithm 1: Multi-resolution implementation of SPARKLING**


---

**Inputs:**  $\rho, G_{max}, S_{max}, N_c, N_s, N_d, n_{git}, n_{pit}$ 
**Output:**  $\mathbf{K}$ , the k-space sampling pattern

**Initializations:**  $\mathbf{K}^{(0)} \in \Omega \frac{N_c \times N_s}{2^{N_d}} = \Omega^p$ 
 $\alpha \leftarrow \frac{\gamma G_{max} 2^{N_d}}{2\pi K_{max}}, \beta \leftarrow \frac{\gamma S_{max} 2^{N_d}}{2\pi K_{max}}$ 
**while**  $N_d > 0$  **do**
 $p \leftarrow \frac{N_c \times N_s}{2^{N_d}}$ 
**for**  $t = 1 \dots n_{git}$  **do**
 $\mathbf{K}^{(t-1/2)} = \mathbf{K}^{(t-1)} - \eta^{(t)} \nabla F_p(\mathbf{K}^{(t-1)})$ 
 $\mathbf{K}^{(t)} = \Pi_{Q_{\alpha, \beta}^{N_c}}(\mathbf{K}^{(t-1/2)}, n_{pit})$ 
**end**

```
// Warm restart next decimation
  step with linear interpolation
// The dimension of k is doubled at
  each decimation step
```

**for**  $s = 1 \dots N_c$  **do**
 $\mathbf{k}_s^{(0)} \leftarrow \mathcal{L}_{\frac{N_s}{2^{N_d-1}}}(\mathbf{k}_s^{(n_{git})})$ 
**end**
 $\mathbf{K}^{(0)} \leftarrow [\mathbf{k}_1^{(0)}, \dots, \mathbf{k}_{N_c}^{(0)}]$ 

```
// Scale constraints
```

 $\alpha \leftarrow \frac{\alpha}{2}, \beta \leftarrow \frac{\beta}{2}$ 
 $N_d \leftarrow N_d - 1$ 
**end**


---

present in a Python package at the private repository<sup>3</sup>. All codes in the package scale to 2 and 3 dimensions directly, and most codes are agnostic and can be run on CPU or GPU with some change in run parameters. All the scanner constants and trajectory specification can be provided through a configuration file, and most of the codes are modular in nature. Interested researchers are requested to contact the authors for obtaining access to this package<sup>4</sup>.

### B. Acquisition parameters

With a goal of 600 $\mu\text{m}$  isotropic resolution in 3D MRI acquisitions, we planned to obtain a volume of  $(N_x \times N_y \times N_z) = (384 \times 384 \times 208)$  size in order to cover the whole brain. For the sake of consistency, we used the same matrix size and resolution for our acquisitions on the NIST phantom<sup>5</sup>. The trajectories were generated for a clinical 3T MR system (Magnetom Prisma<sup>FIT</sup>, Siemens Healthcare, Erlangen, Germany) with maximum gradient strength  $G_{max} = 40\text{mT/m}$  and peak slew rate  $S_{max} = 180\text{T/m/s}$ . As the readout time was set to  $T_{obs} = 20.48\text{ms}$  and the gradient raster time is  $\Delta t = 10\mu\text{s}$ , the number of samples per shot  $\mathbf{k}_i$ , was  $N_s = 2048$ . The number of shots  $N_c$  was varied based on the study described

<sup>3</sup>[https://gitlab.com/cea-cosmic/CSMRI\\_sparkling](https://gitlab.com/cea-cosmic/CSMRI_sparkling)
<sup>4</sup>It cannot be made open source given patent application.

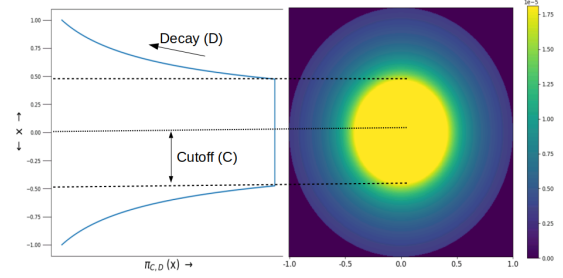
<sup>5</sup><https://www.nist.gov/programs-projects/quantitative-mri>


Fig. 3: Parameterization of variable density with cutoff  $C$  and decay  $D$ .

hereafter. For our in vivo studies, the k-space data was acquired on a Siemens 64 channel Head/Neck coil, while using 44 channels around head during acquisition. The echo time (TE) was 20ms and repetition time (TR) was 37ms. The flip angle was set to  $15^\circ$ , and the slice excitation was slab selective. We also obtained a reference volume collected using a 4-fold accelerated Cartesian acquisition (acquisition time or TA=15min 13sec) based on GRAPPA parallel imaging technique [55] with the same TE, TR and  $T_{obs}$ . The projected gradient descent was carried out with multi-resolution decimation steps  $N_d = 6$  for faster convergence. The algorithm was run for  $n_{git} = 100$  outer gradient descent iterations with  $n_{pit} = 100$  steps in the inner projection loop.

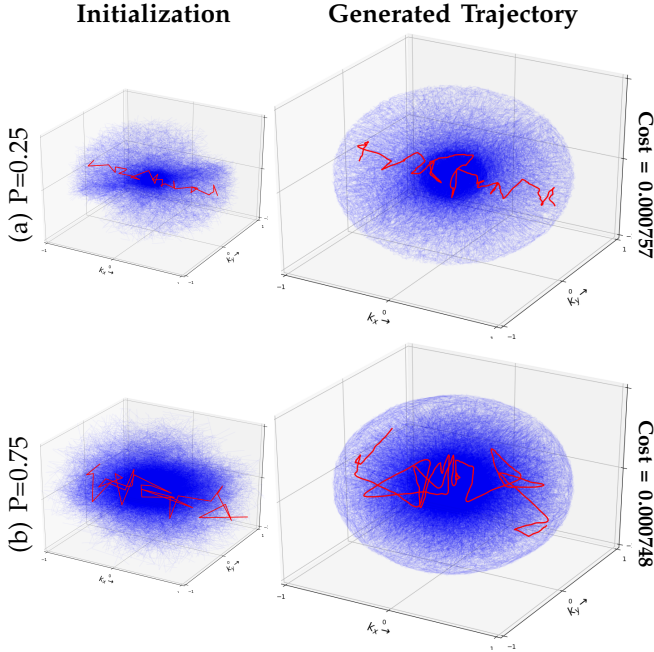
1) *Choice of target sampling density:* The target sampling density was chosen to be radially isotropic, which decays as an inverse polynomial with a constant plateau in the center of k-space. The density was defined with  $C$ , the cutoff frequency in k-space center having a constant density and  $D$ , the rate of decay for higher frequencies. Mathematically, we define the target density  $\pi_{C,D}(x) : \Omega^d \rightarrow \mathbb{R}_+$  as follows:

$$\pi_{C,D}(x) = \begin{cases} \kappa & |x| < C \\ \kappa \left(\frac{C}{|x|}\right)^D & |x| > C \end{cases} \quad (14)$$

where  $\kappa$  is a constant obtained through normalization as  $\kappa = \frac{1-D}{2C(D-1-D)}$ . The resulting density is radially symmetric and is of the form described in Fig. 3. The choice of a radial density was motivated by the wish to provide rotation invariant reconstruction results. Notice that the recent learning based approaches [32], [33] result in non-symmetric densities. This is probably due to the fact that brains or knees databases such as fastMRI [36] used for training have boundaries which are dominantly vertical or horizontal. However, the fine details may be in arbitrary orientations.

The correct choice of density was carried out by grid searching for the optimal parameters  $\hat{C}$  and  $\hat{D}$  on the target sampling distribution as defined in (14). We performed retrospective reconstruction on complex Cartesian reference (with phase to account for off-resonance artifacts by phase accrue) in vivo brain data. For this

study, the number of shots was set to  $N_c = 4096$ . We found that  $\hat{C} = 25\%$  and  $\hat{D} = 2$  provided the highest and most robust SSIM score of 0.97. The SSIMs for varying values of  $C$  and  $D$  are presented in Supplementary Material (see Tab. S1).



**Fig. 4:** Effect of adding a perturbation ( $P$ ) to the initial k-space trajectory in  $\Omega = [-1, 1]^3$  as zero mean uniform random noise at each trajectory sample. Trajectories are generated with maximum displacement of k-space point to (a) 0.25 and (b) 0.75 in the initialization. The left side of the figure is the initialization to SPARKLING algorithm and the right is the output of the algorithm. We also present the values of the cost obtained with (4).

**2) Initialization and Perturbation:** As the problem being solved is non-convex, different choices of initialization would lead to different solutions. In [15] for 2D imaging we observed that radial initialization performed the best for exploring the k-space. Hence, here for 3D imaging we stick to 3D radial initialization too. For the sake of simplicity, and also to ensure radially symmetric initialization, we set up the trajectories with  $\sqrt{N_c}$  shots in x-y plane and then rotate each shot  $\sqrt{N_c}$  times along an in-plane axis orthogonal to the shot. More generic solution can be obtained by solving for the minimum electrostatic potential energy configuration of  $N_c$  electrons over the surface of a unit sphere, however this approach was not pursued in this work.

As the optimization problem in (4) is non-convex, the resulting solution is heavily dependent on the initialization. For best reconstructed image results, we would want each k-space shot to maximally explore the k-space. The 3D radial initialization is too structured with each k-space shot traveling only from end to end of k-space. To enable a broader k-space exploration and obtain a better minimizer of the original problem, we

added a perturbation to each initial shot. To achieve this, we perturbed each trajectory sample point in k-space by adding zero mean uniform random noise along each dimension. Particularly, we compared the resulting optimized trajectory obtained after a perturbation as a random motion of each k-space point with maximum amplitude set at 0.1 and 0.75 (we remind that the sampling domain is normalized to  $\Omega = [-1, 1]^3$ ). The optimized trajectory patterns are presented in Fig. 4. We clearly show that with more perturbation, the k-space trajectory tends to explore a broader part of k-space giving a better coverage overall. We also notice quantitatively that with more perturbation the value of the cost function converges to a lower local minimum. Further, we would like to emphasize that these trajectories are particularly useful in cases of high receiver sampling rates, as they would then sample more of the k-space per shot and would overall prevent the presence of any hole in the sampling pattern.

## IV. RESULTS

### A. From trajectories to actual k-space data

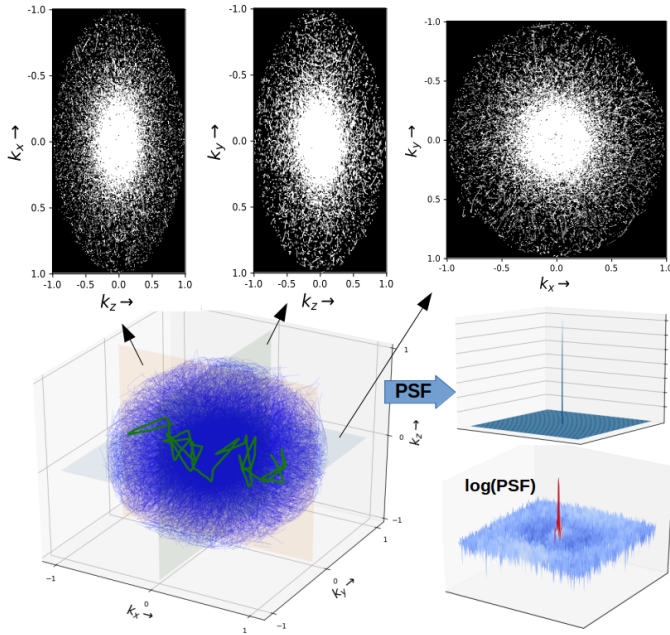
The k-space data  $\mathbf{Y} = (\mathbf{y}_i)_{i=1}^{N_c}$  is sampled by the ADC at the dwell time period  $\delta t$ . In practice, the dwell-time  $\delta t$  is a fraction of the raster time  $\Delta t$  and was set to  $\delta t = 2\mu\text{s}$ . This means that  $\mathbf{y}_i \in \mathbb{C}^m$  with  $m = N_s \lfloor \frac{\Delta t}{\delta t} \rfloor$  the number of measurements per shot. Overall, we collect  $M = N_c m$  k-space data points in  $\mathbf{Y}$ . Consequently, during the image reconstruction process, we obtain the k-space locations of  $\mathbf{Y}$  by linearly interpolating the optimized trajectory  $\hat{\mathbf{K}}$  originally sampled at  $\Delta t$ , to the  $\delta t$  period.

### B. Assessment of point spread function

We present the full 3D SPARKLING, obtained with  $N_c = 4096$  in Fig. 5. We visualize the trajectory along the mid-planes of 3 orthogonal orientations and provide an approximate sampling mask in these planes. Further, to understand why these trajectories are expected to yield good image reconstructions, we measure the 3D point spread function (PSF). Each point spread function was computed by taking a density compensated NUFFT adjoint of k-space measurements set to 1 ( $y_i[n] = 1, \forall i = 1, \dots, N_c, \forall n = 0, \dots, m - 1$ ) as described in [56]. In Fig. 6 we compare the PSFs with respect to earlier generated spherical stack of 2D SPARKLING (SpSOS) trajectories. Particularly, we emphasize the reduction in sidelobes along the  $z$  axis. Further, for the purpose of numerical comparison, we computed the full width at half maximum (FWHM), peak-to-sidelobe level (PSL) and peak-to-noise level (PNL) in Tab. I. The FWHM is calculated as the width of the peak of PSF at half of the maximum value and the PSL and PNL are calculated in dB as presented in Fig. 6(d).

As shown in Fig. 6, we see that the full 3D pattern provides us with much higher PSL (4.15 dB more) and PNL (1.81 dB more), two quantitative indices that





**Fig. 5:** Full 3D SPARKLING Trajectory for  $N_c = 4096, N_s = 2048$  and the point spread function (PSF) along the mid z-plane computed from the sampling mask (measurements sampled at the dwell-time period  $\delta t$ ).

**TABLE I:** Comparing metrics of PSF with FWHM (lower is better), PSL and PNL (higher is better).

Trajectory	FWHM (in voxel units)			PSL (in dB)	PNL (in dB)
	x	y	z		
Full 3D	2.3	2.4	2.5	<b>35.80</b>	<b>67.25</b>
SpSOS	2.4	2.4	2.5	31.65	65.44

demonstrate the full 3D SPARKLING methodology outperforms the spherically stacked version. In contrast, we observe that the FWHM is nearly the same for both methods, even though the FWHM along the  $x$  axis is slightly lower for the full 3D pattern. However, this minor difference in FWHMs and the slight anisotropy in FWHM can be explained by the fact that the full 3D initialization was severely perturbed (0.75) as described in Sec. III-B.2.

### C. Non-Cartesian MR image reconstruction

All MR images that rely on non-Cartesian  $k$ -space data in this paper were reconstructed using a self-calibrated synthesis-based CS reconstruction algorithm [57]–[60] whose details are provided in Supplementary Material (cf. Sec. S3). For the sake of reproducibility, the code for MR image reconstruction is made open source in `pysap-mri`<sup>6</sup>, a plugin of the PySAP software [61]. Of course, future work will combine deep-learning based image reconstruction with full 3D SPARKLING.

<sup>6</sup><https://github.com/CEA-COSMIC/pysap-mri>

In this work, we did not carry out off-resonance artifact corrections using [62], as it is beyond the scope of this manuscript. However, note that this does not require any supplementary scan for obtaining  $\Delta B0$  map as the latter can be directly estimated from phase information using [63]. For the sake of completeness, we show in Supplementary Material the performance of our trajectory with off-resonance corrections for AF=10 in Fig. S6.

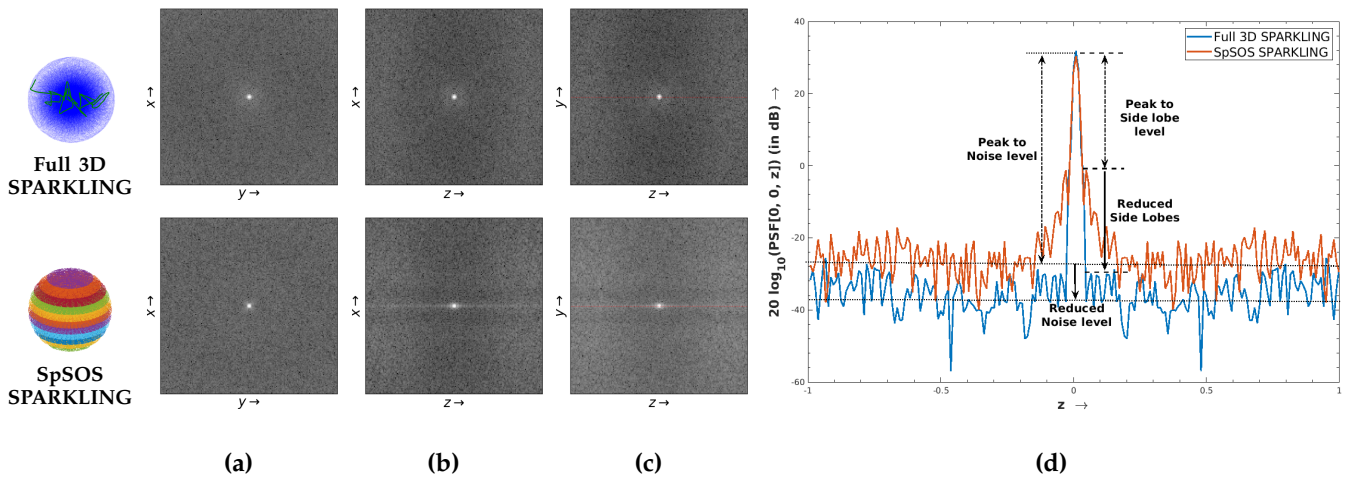
### D. Phantom

1) *Retrospective studies:* We proceed by carrying out a retrospective study to assess the quality of reconstructed images. We varied the acceleration factor ( $AF = \frac{N_y \times N_z}{N_c}$  for 3D MR imaging, i.e. computed with respect to fully sampled data) from 10 (TA=4min 58sec) to 40 (TA=1min 16sec) compared to a fully sampled scenario or equivalently from 2.5 to 10 compared to the reference Cartesian p4 (i.e. AF=4) under-sampled acquisition, reconstructed using the GRAPPA algorithm [55]. Our motivation was to understand the degradation in image quality while decreasing the number of collected spokes. Further, a study was also carried out with the TPI [19], as a comparison with a non-Cartesian reference from the literature. The results are presented in Tab. II. They clearly show that the optimized full 3D SPARKLING strategy is robust to high acceleration factors in terms of image quality as reflected by the higher SSIM scores. In contrast, the performances of the SpSOS approach start to get worse already for AF=20. Finally, the SSIM score for TPI for AF=10 is already significantly lower than that of SpSOS. The reconstructed images are presented in the supplementary material (see Fig. S4).

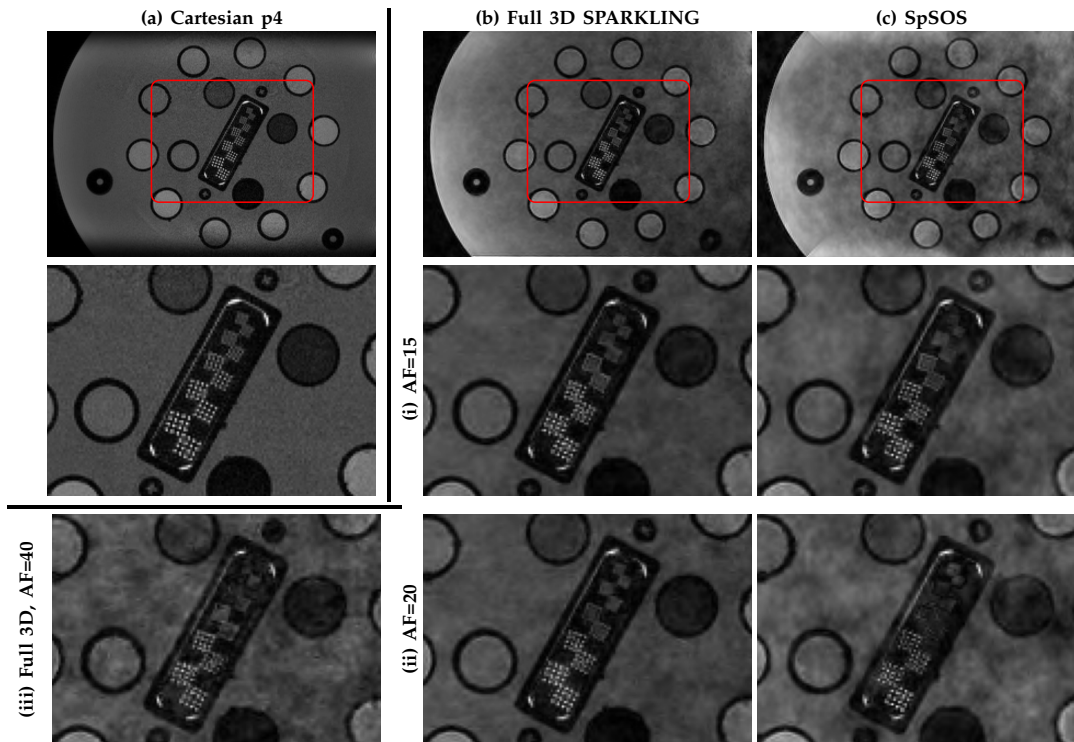
**TABLE II:** Comparing SSIM metrics of retrospective phantom image reconstruction.

Trajectory	AF10	AF15	AF20	AF40
Full 3D	<b>0.964</b>	<b>0.935</b>	<b>0.923</b>	<b>0.816</b>
SpSOS	0.927	0.864	0.737	0.575
TPI	0.63	0.592	0.573	-

2) *Prospective acquisition:* In order to understand how the effective spatial resolution compares to the target resolution (here 0.6mm isotropic), we performed prospective acquisitions on the NIST phantom for full 3D SPARKLING and SpSOS trajectories with varying AF (AF=15 and 20 for both, AF=40 for full 3D). The results are presented in Fig. 7. Particularly, we show a slice that includes the resolution insets present on the NIST phantom (coffin of plate 4). The latter can be used to estimate the effective resolution. This slice consists of 5 resolution insets, each having 2x16 circles. The diameters of these circles vary linearly from 0.8mm down to 0.4mm. The inter-circle space (measured between the centers of the circles) also reduces linearly from 1.6mm down to 0.8mm in steps of 0.2mm.



**Fig. 6:** Comparison of PSFs between full 3D SPARKLING and SpSOS sampling masks (measurements collected at the dwell-time period over the corresponding trajectories). The logarithm of 3D PSFs (in voxel units) are viewed along the mid-slices in (a) axial plane ( $x, y, 0$ ), (b) sagittal plane ( $x, 0, z$ ) and (c) coronal plane ( $0, y, z$ ). (d) The PSFs are compared in logarithmic scale along the  $z$  direction.



**Fig. 7:** Comparison of resolution insets for Full 3D SPARKLING (b) and SpSOS (c) with prospective phantom scans at (i) AF=15, (ii) AF=20 as compared to Cartesian p4 (a). Additionally, we present the results for full 3D SPARKLING trajectory at (iii) AF=40 at the bottom-left.

We see that the intensity profile of our reconstructed MR images does not follow that of Cartesian reference as we did not carry out coil sensitivity normalization in our reconstructions. This can be performed using the rapid pre-scan coil sensitivity measurements done in a few seconds. This point will be addressed in future works. Overall, we observe that full 3D SPARKLING trajectories provide less noisy images compared to SpSOS ones. Further, it is worth noting that at AF=10 and 20, we

can distinguish in between the resolution insets down to 0.7mm, with an increasing noise level over the image for a higher acceleration factor. However, we observe some resolution loss over the images based on SPARKLING trajectories (more pronounced for SpSOS), where we see some blurring for circles of diameter 0.6mm separated by 1.2mm (taken from the center of circle). This helps us understand the expected degradation in image resolution. Therefore, the effective image resolution is estimated to

be 0.6-0.7mm isotropic at AF=10 and 20 and is evolving toward 0.7-0.8mm at AF=40 for full 3D SPARKLING.

### E. In vivo

We collected in vivo data with full 3D SPARKLING and SpSOS trajectories for brain imaging on a healthy volunteer (male, 25 y.o.). This study was approved by a national ethics committee (CPP 100048). The volunteer signed a written informed consent form.

1) *Retrospective studies*: To understand how the trajectories perform for in vivo brain data, we repeat the earlier retrospective study on Cartesian p4 scans acquired on the volunteer. The results of the scans are presented in Fig. 8. We see that full 3D SPARKLING trajectories outperforms the SpSOS trajectories both visually and quantitatively in SSIM metrics with maximum SSIM of 0.964 (AF=10). Moreover we show that both SPARKLING trajectories outperform TPI in Fig. S5 in Supplementary Material. Additionally, the SSIM metrics follow the similar trend as seen for phantom data, with the SSIMs for SpSOS dropping off more rapidly from 0.93 (AF=10) to 0.759 (AF=20). In contrast, the full 3D SPARKLING trajectories tend to preserve the structures (SSIM scores above 0.9 at AF=15 and 20) and show some blurring artifacts only at AF=40 where SSIM drops to 0.792. Particularly, it is interesting to note that full 3D SPARKLING at AF=40 outperforms SpSOS at AF=20.

2) *Prospective acquisition*: Finally, we collected prospectively accelerated in vivo data at 3T on the same individual using the same SPARKLING trajectories. We present the reconstructed images for various accelerations factors in Fig. 9. They clearly show superiority of the full 3D SPARKLING pattern compared to SpSOS. Image quality is well preserved for AF=10 and 15 and slightly noisy at AF=20 in full 3D strategy (Fig. 9, top row), while it tends to get noisy at AF=15 and severely impaired at AF=20 for SpSOS strategy (Fig. 9, bottom row). Moreover, we observed that the quality of AF=15, AF=20 and AF=40 in full 3D strategy is comparable to AF=10, AF=15 and AF=20 in SpSOS pattern respectively, allowing for an additional 2x shorter scan time. Further, we found that the full 3D SPARKLING pattern at AF=15 is comparable to GRAPPA Cartesian p4.

It is important to note that the volunteer slightly moved between some scans, hence prospective image comparisons can only be carried out qualitatively. To better understand reconstruction quality, we present zoomed in visualizations for prospective result at AF=10 in Fig. 10. Further, for the sake of comparison between retrospective simulations and actual prospective scans, we also show the retrospective results for AF=10 with full 3D trajectory. We find that full 3D strategy retains better structures of the brain in the MR image than SpSOS, which is clearly visible in the cerebellum in the sagittal view.

The comparison with retrospective image allows us to directly identify some degradation and loss of small

details in prospective images. Potential explanations for this effect are the  $T_2^*$  blurring and off-resonance artifacts, which drastically drop the effective SNR obtained (see Fig. S2 in supplementary material). This confirms that in vivo acquisitions are more challenging.

## V. DISCUSSIONS

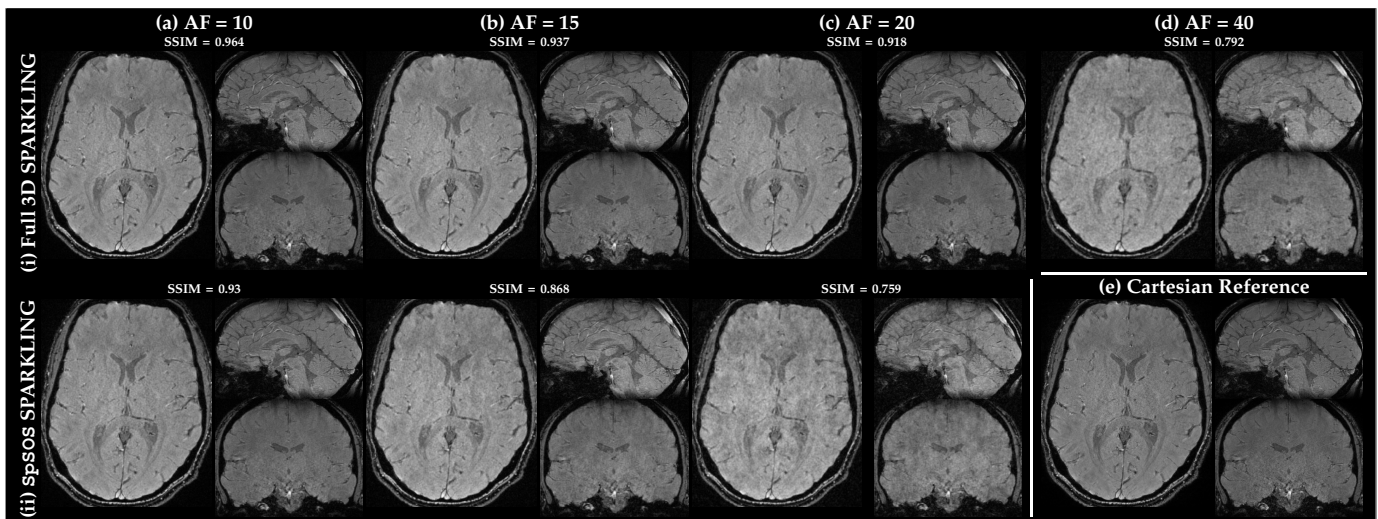
One key aspect of optimized full 3D SPARKLING trajectories is that it results in a sampling pattern that enforces variable density sampling in all the 3 dimensions. We hypothesized that this allows us to efficiently under-sample the k-space acquisitions, thus making it possible to push the acceleration factor to a larger value than what was achieved earlier, while still maintaining a good image quality. The current work actually demonstrates that at fixed acceleration factor, full 3D SPARKLING significantly outperforms the stacking strategy [27] in terms of image quality. Alternatively, we show that this gain can be translated into shorter scan time by a factor of one third (AF=15 for full 3D vs AF=10 for SpSOS) for a given image quality.

Further, the full 3D trajectory is constrained to pass through the center of k-space for each shot at echo time. This ensures that we obtain the lower frequency image content repeatedly, hence we can potentially use these trajectories for motion correction. Also, as the center of k-space is visited repeatedly at different time intervals in scan, this allows for easy adaptability of this trajectory for dynamic imaging like functional MRI. Such a trajectory can also be used for correcting certain artifacts causing off-resonance effects, which are due to static and dynamic B0 inhomogeneities (heart beat, breathing). A preliminary solution has been proposed for static B0 inhomogeneities estimation and correction in [63].

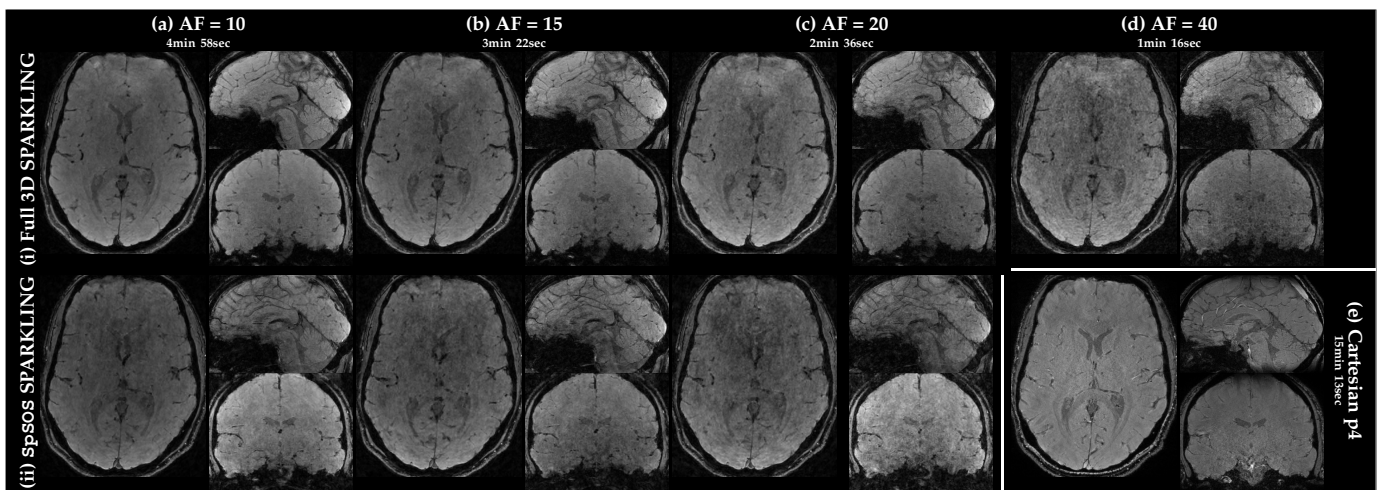
As the developed trajectories exploit the scanner hardware constraints nearly to the maximum, it is worth paying attention to the eddy current effects on the trajectory. To this end, we measured the trajectory with the help of a Skope field camera [64] and observed in Sec. S2-C that the error between the prescribed and actual trajectories is minimal (cf. Fig. S3).

While the current SPARKLING algorithm is generic and can be applied to any imaging contrast a priori, we choose  $T_2^*$ -w imaging as it allows us to keep longer  $T_{\text{obs}}$  hence enabling a full exploration of 3D k-space. In order to understand the effects of  $T_2^*$  blurring and off-resonance, we simulated the PSF under these scenarios in Sec. S2-B. Additionally, as we interfaced a GRE pulse sequence (FLASH in the Siemens taxonomy) with the SPARKLING outputs, the adaptation of this algorithm to other contrasts (e.g.  $T_2$ ) would potentially need the development of a turbo spin echo (TSE) sequence that is able to play arbitrary gradients. These developments are left for future work.

One limitation of SPARKLING is that the original optimization problem (4) is non-convex and the fact we used a locally convergent optimization algorithm to compute



**Fig. 8:** Comparison of retrospective results for (i) fully optimized 3D SPARKLING (top row) and (ii) SpSOS (bottom row) for varying acceleration factors (from left to right, AF=10 (a), 15 (b) and 20 (c)) on in vivo human brain scans. Cartesian Reference (e) is provided for comparison and results for full 3D trajectory at AF 40 (d) is also presented. SSIM scores are reported for each setup.

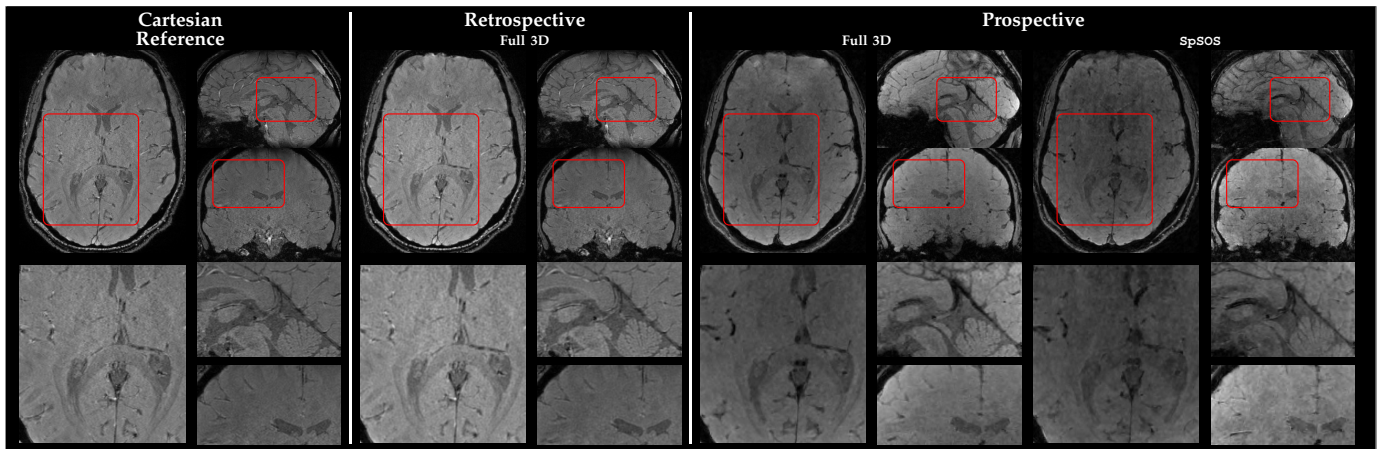


**Fig. 9:** Comparison of prospective results for (i) fully optimized 3D SPARKLING (top row) and (ii) SpSOS (bottom row) for varying acceleration factors (from left to right, AF=10 (a), 15 (b) and 20 (c)) on in vivo human brain scans. Cartesian p4 scan (e) is provided for comparison and results for full 3D trajectory at AF 40 (d) is also presented. The scan times are reported for each AF.

a minimizer. Hence the final solution heavily depends on its initialization. To overcome this issue, we introduced some perturbation (uniform random noise in the k-space locations) and illustrated in Fig. 4 that a larger perturbation results in a much better k-space coverage, allowing us to reach a better minimizer to the original optimization problem. However, there is no theoretical guarantee this approach provides a systematic better solution as the underlying optimization process remains rather disconnected from MR image reconstruction and the maximization of image quality.

In the same vein, another limitation of the resulting reconstructed MR images is that they heavily depend

on the target sampling distribution. We obtained our results by parameterizing this distribution, thereby optimizing for its parameters using a grid search on in vivo brain data. However, these optimal parameters are not generalizable for different contrast and organs. Further, such parametrization can prevent us from using more complex target sampling densities. To overcome this limitation, ongoing work intends to couple SPARKLING with the learning of the target sampling density from the magnitude spectrum of human brain MR images [65]. Akin to this work, we could also jointly optimize for the acquisition (sampling pattern) and reconstruction schemes (regularization parameters) under MR hard-



**Fig. 10:** Comparison of prospective results for fully optimized 3D SPARKLING (right-left) and SpSOS (right most) with Cartesian reference (left) and retrospective full 3D SPARKLING (center) for  $AF=10$  (scan time = 4min 58sec). In each panel of the top row, axial (left), sagittal (top right) and coronal (bottom right) slices are shown and a red frame is delineated in the central part of the brain for zooming purpose. Bottom row shows the magnified views with the same layout (axial, sagittal and coronal slices in the left, top-right and bottom right insets, respectively).

ware and imaging contrast constraints, either in a bilevel optimization [33] or using deep learning approaches [34], [35], [66]. These extensions would help us to take some factors, like the anatomy and the imaging contrast, into account in the design of trajectories with perfectly matched target sampling densities for these cases.

In this work, we did not account for the coil geometry in the design of trajectories. This would require addressing more complex target sampling densities as input which are a function of the coil sensitivities. Optimizing over richer and more adaptive families of densities is left for our future prospects. However, we note that the proposed algorithm can automatically adapt to any future progress made in this direction.

## VI. CONCLUSION

In this paper, we proposed an optimization for full 3D SPARKLING  $k$ -space trajectories for accelerated high resolution 3D  $T_2^*$ -weighted imaging and demonstrated its superiority over the previously proposed stacking strategies on phantom and in vivo human brain data at 3T. We discussed the major computational bottlenecks that prevented us earlier from proceeding towards these full 3D trajectories. We then derived some implementations (GPU and multi-CPU) that helped us massively accelerate the original algorithm. Our results showed that a  $600\mu\text{m}$  isotropic scan on human brain is achievable in 1min 16sec, whereas 3m 22sec is required to reach image quality comparable to GRAPPA-4 parallel imaging. Overall, this is a significant step forward for CS acquisitions in MRI. Future work will be devoted to the extension to 4D imaging, namely for fMRI.

## ACKNOWLEDGEMENTS

Chaithya G R was supported by the CEA NUMERICS PhD program, which received European funding

from Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 800945. We acknowledge the financial support of the Cross-Disciplinary Program on Numerical Simulation of CEA (SILICOSMIC project, PI: P. Ciuciu), the French Alternative Energies and Atomic Energy Commission. This work was granted access to the HPC resources of TGCC in France under the allocation 2019-GCH0424 made by GENCI. Pierre Weiss was supported by the ANR JCJC Optimization on Measures Spaces ANR-17-CE23-0013-01 and the ANR-31A Artificial and Natural Intelligence Toulouse Institute.

## REFERENCES

- [1] M. Lustig *et al.*, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, 2007.
- [2] G. Puy *et al.*, "On variable density compressive sampling," *IEEE Signal Process. Lett.*, vol. 18, no. 10, pp. 595–598, 2011.
- [3] N. Chauffert *et al.*, "Variable density sampling with continuous trajectories. Application to MRI," *SIAM J. Imag. Sci.*, vol. 7, no. 4, pp. 1962–1992, Nov. 2014.
- [4] B. Adcock *et al.*, "Breaking the coherence barrier: A new theory for compressed sensing," in *Forum of Math., Sigma*, vol. 5. Cambridge University Press, 2017.
- [5] C. Boyer *et al.*, "Compressed sensing with structured sparsity and structured acquisition," *Appl. Comput. Harmon. Anal.*, vol. 46, no. 2, pp. 312–350, 2019.
- [6] C. Ahn *et al.*, "High-speed spiral-scan echo planar NMR imaging-I," *IEEE Trans. Med. Imag.*, vol. 5, no. 1, pp. 2–7, 1986.
- [7] C. H. Meyer *et al.*, "Fast spiral coronary artery imaging," *Magn. Reson. Med.*, vol. 28, no. 2, pp. 202–213, 1992.
- [8] J. I. Jackson *et al.*, "Twisting radial lines with application to robust magnetic resonance imaging of irregular flow," *Magn. Reson. Med.*, vol. 25, no. 1, pp. 128–139, 1992.
- [9] D. C. Noll, "Multishot rosette trajectories for spectrally selective MR imaging," *IEEE Trans. Med. Imag.*, vol. 16, no. 4, pp. 372–377, 1997.
- [10] C. S. Law *et al.*, "Interleaved spiral-in/out with application to functional MRI (fMRI)," *Magn. Reson. Med.*, vol. 62, no. 3, pp. 829–834, 2009.

- [11] M. Lustig *et al.*, "Faster imaging with randomly perturbed, under-sampled spirals and  $\ell_1$  reconstruction," in *Proc. 13th ISMRM*, Miami Beach, FL, USA, 2005, p. 685.
- [12] N. Chauffert, "Echantillonnage compressé le long de trajectoires physiquement plausibles en irm," Ph.D. dissertation, Université Paris-Sud, Orsay, France, 2015.
- [13] C. Boyer *et al.*, "On the generation of sampling schemes for MRI," *SIAM J. Imag. Sci.*, vol. 9, no. 4, pp. 2039–2072, 2016.
- [14] N. Chauffert *et al.*, "A projection method on measures sets," *Constr. Approx.*, vol. 45, no. 1, pp. 83–111, 2017.
- [15] C. Lazarus *et al.*, "SPARKLING: variable-density k-space filling curves for accelerated T2\*-weighted MRI," *Magn. Reson. Med.*, vol. 81, no. 6, pp. 3643–3661, 2019.
- [16] S. S. Vasawala *et al.*, "Improved pediatric MR imaging with compressed sensing," *Radiology*, vol. 256, no. 2, pp. 607–616, 2010.
- [17] P. E. Larson *et al.*, "Anisotropic field-of-views in radial imaging," *IEEE Trans. Med. Imag.*, vol. 27, no. 1, pp. 47–57, 2008.
- [18] P. Irarrazabal *et al.*, "Fast three dimensional magnetic resonance imaging," *Magn. Reson. Med.*, vol. 33, no. 5, pp. 656–662, 1995.
- [19] F. E. Boada *et al.*, "Fast three dimensional sodium imaging," *Magn. Reson. Med.*, vol. 37, no. 5, pp. 706–715, 1997.
- [20] K. M. Johnson, "Hybrid radial-cones trajectory for accelerated MRI," *Magn. Reson. Med.*, vol. 77, no. 3, pp. 1068–1081, 2017.
- [21] B. M. Dale *et al.*, "Optimal design of k-space trajectories using a multi-objective genetic algorithm," *Magn. Reson. Med.*, vol. 52, no. 4, pp. 831–841, 2004.
- [22] R. Mir *et al.*, "Fast three-dimensional k-space trajectory design using missile guidance ideas," *Magn. Reson. Med.*, vol. 52, no. 2, pp. 329–336, 2004.
- [23] C. Kumar Anand *et al.*, "Durga: A heuristically-optimized data collection strategy for volumetric magnetic resonance imaging," *Eng. Optim.*, vol. 40, no. 2, pp. 117–136, 2008.
- [24] H. K. Song *et al.*, "Dynamic MRI with projection reconstruction and KWIC processing for simultaneous high spatial and temporal resolution," *Magn. Reson. Med.*, vol. 52, no. 4, pp. 815–824, 2004.
- [25] W. Lin and other, "Respiratory motion-compensated radial dynamic contrast-enhanced (DCE)-MRI of chest and abdominal lesions," *Magn. Reson. Med.*, vol. 60, no. 5, pp. 1135–1146, 2008.
- [26] D. R. Thedens *et al.*, "Fast magnetic resonance coronary angiography with a three-dimensional stack of spirals trajectory," *Magn. Reson. Med.*, vol. 41, no. 6, pp. 1170–1179, 1999.
- [27] C. Lazarus *et al.*, "3D variable-density SPARKLING trajectories for high-resolution T2\*-weighted magnetic resonance imaging," *NMR Biomed.*, p. e4349, 2020.
- [28] J. P. Haldar and D. Kim, "Oedipus: An experiment design framework for sparsity-constrained mri," *IEEE Trans. Med. Imag.*, vol. 38, no. 7, pp. 1545–1558, 2019.
- [29] M. Seeger *et al.*, "Optimization of k-space trajectories for compressed sensing by Bayesian experimental design," *Magn. Reson. Med.*, vol. 63, no. 1, pp. 116–126, Jan 2010.
- [30] L. Baldassarre *et al.*, "Learning-based compressive subsampling," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 4, pp. 809–822, 2016.
- [31] B. Gözcü *et al.*, "Learning-based compressive MRI," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1394–1406, 2018.
- [32] C. D. Bahadir *et al.*, "Deep-learning-based optimization of the under-sampling pattern in MRI," *IEEE Trans. Comput. Imaging*, vol. 6, pp. 1139–1152, 2020.
- [33] F. Sherry *et al.*, "Learning the sampling pattern for MRI," *IEEE Trans. Med. Imag.*, 2020.
- [34] T. Weiss *et al.*, "PILOT: Physics-informed learned optimal trajectories for accelerated MRI," *arXiv:1909.05773v4*, Aug. 2020.
- [35] S. Vedula *et al.*, "3D FLAT: Feasible Learned Acquisition Trajectories for Accelerated MRI," in *Machine Learning for Medical Image Reconstruction: 3rd Intern. WS MLMIR 2020, Held in Conjunction with MICCAI 2020*. Lima, Peru: Springer Nature, Oct. 2020, p. 3.
- [36] J. Zbontar *et al.*, "fastMRI: An open dataset and benchmarks for accelerated MRI," *arXiv:1811.08839*, 2018.
- [37] Z. Ben-Haim and Y. C. Eldar, "The cramér-rao bound for estimating a sparse parameter vector," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3384–3389, 2010.
- [38] G. Wang *et al.*, "B-spline parameterized joint optimization of reconstruction and k-space trajectories (BJORK) for accelerated 2d MRI," *arXiv preprint arXiv:2101.11369*, 2021.
- [39] D. Gournay *et al.*, "Spurious minimizers in non uniform Fourier sampling optimization," Jun. 2021, working paper or preprint. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-03212145>
- [40] L. Chizat and F. Bach, "On the Global Convergence of Gradient Descent for Over-parameterized Models using Optimal Transport," in *Advances in Neural Information Processing Systems (NIPS)*, Montréal, Canada, Dec. 2018.
- [41] Q. Merigot, F. Santambrogio, and C. Sarrazin, "Non-asymptotic convergence bounds for Wasserstein approximation using point clouds," Jun. 2021, working paper or preprint. [Online]. Available: <https://hal.archives-ouvertes.fr/hal-03256039>
- [42] N. Chauffert *et al.*, "A projection algorithm for gradient waveforms design in MRI," *IEEE Trans. Med. Imag.*, vol. 35, no. 9, pp. 2026–2039, Sep. 2016.
- [43] C. Lazarus *et al.*, "Correcting the side effects of ADC filtering in MR image reconstruction," *J. Math. Imaging Vis.*, pp. 1–14, 2020.
- [44] M. Gräf *et al.*, "Quadrature errors, discrepancies, and their relations to halftoning on the torus and the sphere," *SIAM J. Sci. Comput.*, vol. 34, no. 5, pp. A2760–A2791, 2012.
- [45] C. Schmaltz *et al.*, "Electrostatic halftoning," in *Computer Graphics Forum*, vol. 29, no. 8. Wiley Online Library, 2010, pp. 2313–2327.
- [46] M. Ehler *et al.*, "Curve based approximation of measures on manifolds by discrepancy minimization," *arXiv:1910.06124*, 2019.
- [47] T. Teuber *et al.*, "Dithering by differences of convex functions," *SIAM J. Imag. Sci.*, vol. 4, no. 1, pp. 79–108, 2011.
- [48] B. Charlier *et al.*, "Kernel operations on the GPU, with autodiff, without memory overflows," *arXiv:2004.11127*, 2020.
- [49] W. Fong *et al.*, "The black-box fast multipole method," *J. Comput. Phys.*, vol. 228, no. 23, pp. 8712–8725, 2009.
- [50] R. Wang *et al.*, "PBBFMM3D: a parallel black-box fast multipole method for non-oscillatory kernels," *CoRR*, vol. abs/1903.02153, 2019. [Online]. Available: <http://arxiv.org/abs/1903.02153>
- [51] W. Fong *et al.*, "The black-box fast multipole method," *J. Comput. Phys.*, no. 228, 2009.
- [52] J. Barzilai *et al.*, "Two-Point Step Size Gradient Methods," *IMA J. Numer. Anal.*, vol. 8, no. 1, pp. 141–148, 1988.
- [53] R. Okuta *et al.*, "Cupy : A numpy-compatible library for nvidia gpu calculations," 2017.
- [54] L. Lebrat *et al.*, "Optimal Transport Approximation of 2-Dimensional Measures," *SIAM J. Imag. Sci.*, vol. 12, no. 2, pp. 762–787, Jan. 2019.
- [55] M. A. Griswold *et al.*, "Generalized autocalibrating partially parallel acquisitions (GRAPPA)," *Magn. Reson. Med.*, vol. 47, no. 6, pp. 1202–1210, 2002.
- [56] "Advrecon-pauly\_read.pdf," [https://users.fmrib.ox.ac.uk/~karla/reading\\_group/lecture\\_notes/AdvRecon\\_Pauly\\_read.pdf](https://users.fmrib.ox.ac.uk/~karla/reading_group/lecture_notes/AdvRecon_Pauly_read.pdf), (Accessed on 05/04/2021).
- [57] J. G. Pipe and P. Menon, "Sampling density compensation in MRI: Rationale and an iterative numerical solution," *Magn. Reson. Med.*, vol. 41, no. 1, pp. 179–186, 1999.
- [58] F. Knoll *et al.*, "gpuNUFFT-An open source GPU library for 3D regridding with direct Matlab interface," in *Proc. 22nd ISMRM*, 2014, p. 4297.
- [59] L. E. Gueddari *et al.*, "PySAP-MRI: a Python Package for MR Image Reconstruction," in *ISMRM WS on Data Sampling and Image Reconstruction*, Sedona, AZ, United States, Jan. 2020. [Online]. Available: <https://hal.inria.fr/hal-02399267>
- [60] L. El Gueddari *et al.*, "Self-calibrating nonlinear reconstruction algorithms for variable density sampling and parallel reception MRI," in *10th IEEE SAM Signal Processing WS*, Sheffield, UK, Jul. 2018, pp. 415–419.
- [61] S. Farnes *et al.*, "PySAP: Python Sparse Data Analysis Package for multidisciplinary image processing," *Astron. Comput.*, vol. 32, p. 100402, 2020.
- [62] B. Sutton *et al.*, "Fast, iterative image reconstruction for mri in the presence of field inhomogeneities," *IEEE Trans. Med. Imag.*, vol. 22, no. 2, pp. 178–188, 2003.
- [63] G. Daval-Frérôt *et al.*, "Off-resonance correction non-Cartesian SWI using internal field map estimation," in *29th Proceedings of the ISMRM society*, no. 3551, virtual, May 2021.
- [64] N. De Zanche *et al.*, "NMR probes for measuring magnetic fields and field dynamics in MR systems," *Magn. Reson. Med.*, vol. 60, pp. 176–186, 2008.
- [65] C. G R *et al.*, "Learning the sampling density in 2D SPARKLING MRI acquisition for optimized image reconstruction," in *29th EUSIPCO*, Dublin, Ireland, Sep. 2021, pp. –.
- [66] Z. Ramzi *et al.*, "Benchmarking MRI Reconstruction Neural Networks on Large Public Datasets," *Applied Sciences*, vol. 10, no. 5, p. 1816, 2020.

# Supplementary Material – Optimizing full 3D SPARKLING trajectories for high-resolution $T_2^*$ -weighted Magnetic Resonance Imaging

Chaithya GR, Pierre Weiss, Guillaume Daval-Fr erot, Aur elien Massire, Alexandre Vignaud, and Philippe Ciuciu *Senior Member, IEEE*

## S1. MULTI-SHOT PROJECTION STEP

We briefly extend the single shot projection step to  $N_c$  shots. Particularly, we would like to project the complete sampling pattern  $\mathbf{K} = [\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_{N_c}]$  such that each shot  $\mathbf{k}_i$  complies with the hardware constraints,  $Q_{\alpha, \beta}$  defined in (2). From [42], the projection step for a single shot  $\mathbf{k}$  is given by:

$$\mathbf{k}^* = \Pi_{Q_{\alpha, \beta}}(\mathbf{k}) = \arg \min_{s \in Q_{\alpha, \beta}} \frac{1}{2} \|\mathbf{s} - \mathbf{k}\|_2^2$$

We can now further generalize above equation to project all shots as:

$$\mathbf{K}^* = \Pi_{Q_{\alpha, \beta}^{N_c}}(\mathbf{K}) = [\Pi_{Q_{\alpha, \beta}}(\mathbf{k}_1), \dots, \Pi_{Q_{\alpha, \beta}}(\mathbf{k}_{N_c})]$$

where, we can use accelerated proximal gradient descent algorithm as described in [42] to carry out individual projections of each shot in parallel.

## S2. TRAJECTORY

### A. Gradients and Slew Rates

We present the gradients and slew rates obtained for a single shot of full 3D SPARKLING trajectory in Fig. S1. We show that the trajectory was mostly slew rate constrained, thereby making the percentage of readouts with gradient magnitude constraint active to be close to 0. The gradient waveform was never saturated but achieved its maximum nearby the center of k-space as shown in Fig. S1(a). This is because all the trajectories pass through the center of k-space, thereby drastically increasing the sampling density of the sampling pattern. Due to this, each trajectory moves at highest velocity to achieve lesser k-space sample points here, thereby achieving the target sampling density. Note that although the scanner slew rate constraint is 200 T/m/s, the trajectories were optimized with a tighter constraint of maximum allowed slew rate of 180 T/m/s, which explains the lower slew rate limits observed in the trajectory in Fig. S1(b).

### B. Off-resonance and $T_2^*$ decay

We carried out off-resonance and  $T_2^*$ -decay simulations on the trajectory and analyzed the point spread functions (PSF) in Fig. S2. We systematically added a  $T_2^*$ -decay by taking a constant value of  $T_2^* = 30\text{ms}$ , and also performed off-resonance simulations by adding a constant off-resonance ( $\Delta B_0$ ) of 25Hz. In Fig. S2 we

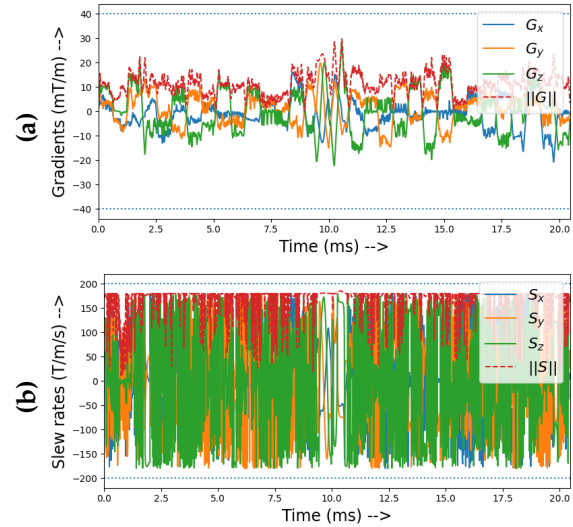
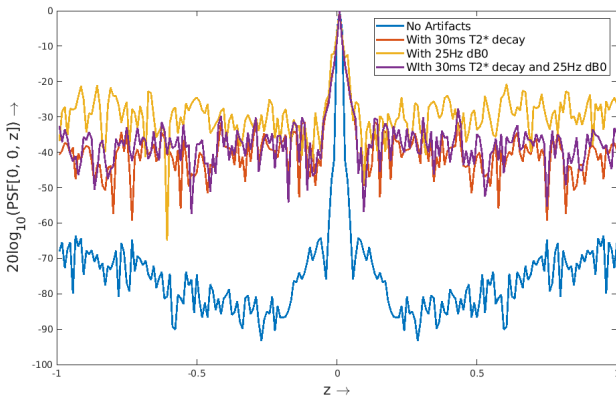


Fig. S1: The (a) gradients and (b) slew rates for a single shot from a Full 3D SPARKLING trajectory with AF=20. We have also marked the Scanner hardware constraints ( $G_{\max} = 40\text{mT/m}$  and  $S_{\max} = 200 \text{ T/m/s}$ ) with black dotted lines.

observe that the noise level in the PSF is significantly increased (by  $\simeq 25 - 30 \text{ dB}$ ) when adding  $\Delta B_0$  and  $T_2^*$  decay. Further, we see an increase in sidelobe level when adding  $\Delta B_0$ , which however is reduced when combined with  $T_2^*$  decay. We observe that the effective PSF is spread under  $T_2^*$  and  $B_0$  inhomogeneities, leading to drop in effective resolution, which is studied in depth in the core paper in Sec. IV-D.2. This study reveals some of the reasons that explain the gap in image quality between retrospective and prospective results, especially for in vivo acquisitions, where we have spatially varying  $T_2^*$  and  $\Delta B_0$  (i.e.  $T_2^*(\mathbf{r})$  and  $\Delta B_0(\mathbf{r})$  maps).

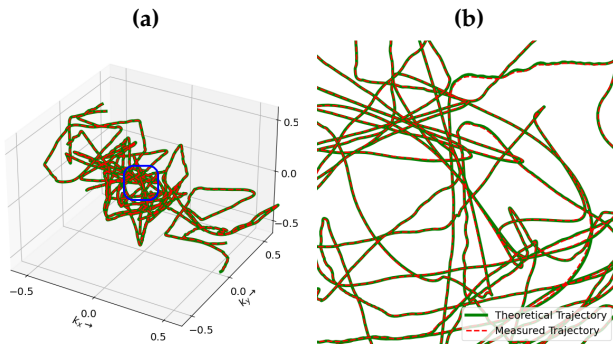
### C. Eddy Current and Trajectory Measurement

As the presented trajectory rapidly explores the k-space, it is vital to ensure that the MRI scanner is able to play the complicated gradient waveforms in Fig. S1 with minimal errors. These errors could be induced by eddy currents and gradient imperfections. To this end, we ran the AF=20 full 3D SPARKLING trajectory on an Investigative 7T MR System (MAGNETOM 7T, Siemens Healthcare, Erlangen, Germany) and measured



**Fig. S2:** Simulated effects of  $T_2^*$  decay (of 30ms) and constant  $\Delta B_0$  (of 25Hz) on the point spread function of the AF=10 full 3D SPARKLING trajectory. All the PSFs are normalized such that the maximum value is 1, for easier visual comparison.

the trajectory with the SKOPE dynamic field camera [64]. We used a 7T scanner for this study due to compatibility issues at 3T. However, we do not expect drastic changes in our results as the gradient system is the same for both scanners. We present the theoretical (i.e. prescribed by the 3D SPARKLING algorithm) and measured trajectories for 3 random k-space shots in Fig. S3. Further, we quantitatively measured the error as to be  $0.0016 \pm 0.0012$  (with the k-space normalized to  $\Omega \in [-1, 1]^D$ ).



**Fig. S3:** (a) Three random shots from theoretical and measured trajectories for AF=20 full 3D SPARKLING trajectory. (b) Zoomed in at the center of k-space.

### S3. MR IMAGE RECONSTRUCTION

The MR image reconstruction of 3D multi-channel data acquired from phased array receiver coils was carried out using a self-calibrating compressed sensing (CS) reconstruction algorithm [60] in the synthesis formulation by solving for the wavelet coefficients  $\mathbf{z}$  in (S1):

$$\hat{\mathbf{z}} = \underset{\mathbf{z} \in \mathbb{C}^{N_x \times N_y \times N_z}}{\operatorname{argmin}} \frac{1}{2} \sum_{\ell=1}^L \|F_\Omega \mathbf{S}_\ell \Psi^* \mathbf{z} - \mathbf{y}_\ell\|_2^2 + \lambda \|\mathbf{z}\|_1 \quad (\text{S1})$$

where the number of channels was  $L = 44$  and  $N_x = N_y = 384$  and  $N_z = 208$ . Here the data fidelity is

enforced with SENSE operators  $(F_\Omega \mathbf{S}_\ell)_{\ell=1}^L$ , where  $F_\Omega$  is the NUFFT operator and  $\mathbf{S}_\ell$  is sensitivity map for  $\ell^{\text{th}}$  channel estimated by density compensated adjoint of the 20 percent of acquired k-space center (see details in [60]).  $\lambda > 0$  is the regularization parameter for promoting sparsity using  $\ell_1$ -norm regularization in the wavelet domain  $\Psi$ . For our reconstructions, we used Symlet 8 wavelet with 4 scales of decomposition for  $\Psi$ . The regularization parameter  $\lambda$  was grid searched between  $(10^{-10}, 10^0)$  while maximizing for the reconstruction quality using SSIM score in retrospective reconstruction. As the sampling operator was 3D non-Cartesian, the reconstruction problem was severely ill-posed with the forward operator  $F_\Omega \mathbf{S}_1 \Psi^*$  having a large condition number, thereby impacting the convergence speed. In order to accelerate convergence, we preconditioned the k-space using density compensation. This translates to adding a preconditioner  $D$  in the classical proximal gradient descent algorithm (here we used FISTA):

$$\mathbf{z}^{(k+1)} = \operatorname{soft}_{\lambda\tau} \left( \mathbf{z}^{(k)} - \tau \sum_{\ell=1}^L \Psi \mathbf{S}_\ell^* F_\Omega^H D \left( F_\Omega \mathbf{S}_\ell \Psi^* \mathbf{z}^{(k)} - \mathbf{y}_\ell \right) \right)$$

where  $\operatorname{soft}_{\lambda\tau}$  is the soft threshold operator and  $\tau$  is the step size. The density compensators  $D$  were obtained by 10 iterations of method described in [57]. The final MR image is given by  $\hat{\mathbf{x}} = \Psi^* \hat{\mathbf{z}} \in \mathbb{C}^{N_x \times N_y \times N_z}$  as  $\Psi$  is a basis.

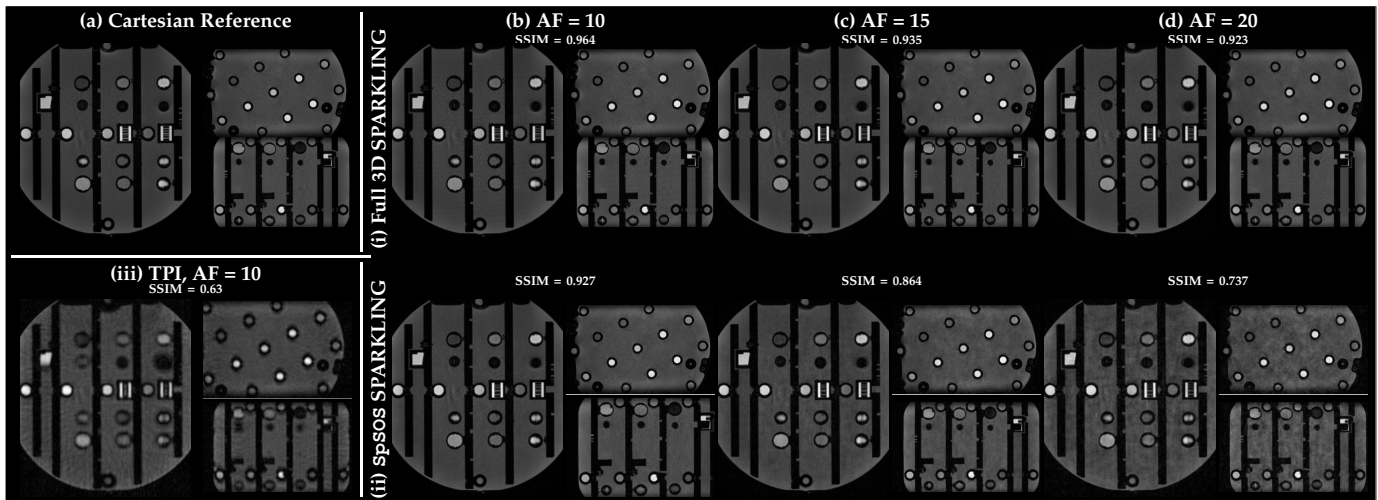
As the raw data was large (for AF=20,  $p = 8, 388, 608$  k-space points), we needed to utilize memory efficient methods to carry out the SENSE operation. For this, we implemented python wrappers for gpuNUFFT [58] which implements the NUFFT operator in CUDA and utilizes cuBLAS and cuFFT libraries to be efficient in speed and memory. The implementation of the reconstruction was completely done using `pysap-mri` [59], the plugin of PySAP [61] dedicated to MR image reconstruction. Despite being a 3D reconstruction problem, the computation time was just 15-30 minutes on a machine with the same hardware specifications as described earlier in Sec. III.

**TABLE S1:** SSIM scores for retrospective reconstructions on in vivo brain MR images as a function of varying density parameterizations (cutoff  $C$  and decay  $D$  are varied across columns and rows, respectively). While ( $C=15, D=1$ ) looks like an optimal setting, we used ( $C=25, D=2$ ) as this configuration is more robust to small changes in density. Best scores appear in bold font.

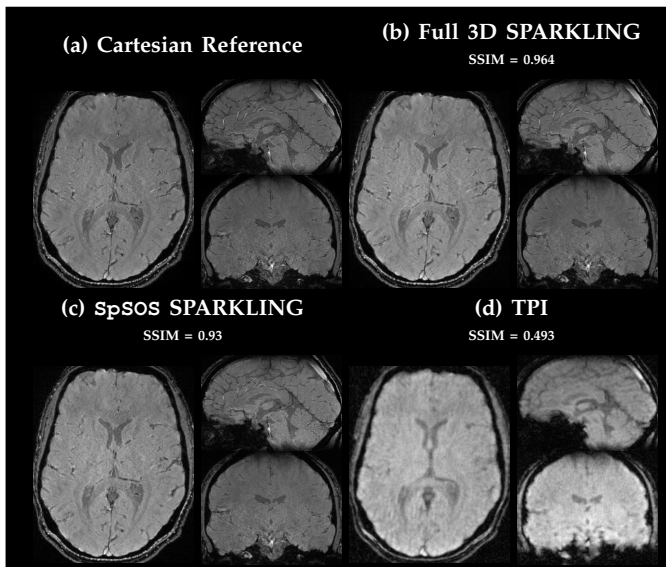
D \ C	C					
	10	15	20	25	30	35
1	0.976	<b>0.982</b>	0.873	0.833	0.766	0.770
2	0.937	0.950	0.967	<b>0.972</b>	<b>0.977</b>	0.961
3	0.836	0.909	0.939	0.961	0.968	0.960

<sup>1</sup><https://github.com/CEA-COSMIC/pysap-mri>

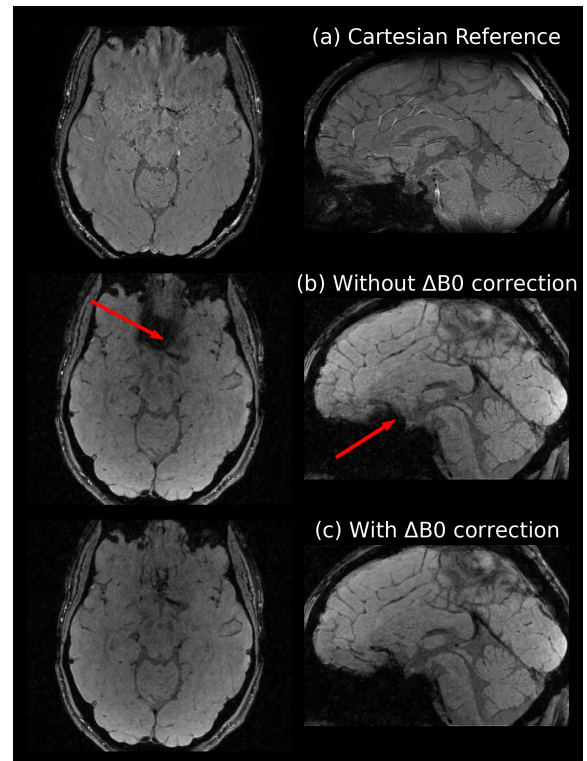




**Fig. S4:** Comparison of retrospective results for (i) fully optimized 3D SPARKLING (right, top row) and (ii) SpSOS (right, bottom row) for varying acceleration factors (from left to right, AF=10 (b), 15 (c) and 20 (d)) on the NIST phantom. Additionally, we present the results of a retrospective study based on (iii) TPI at AF=10 for comparison purposes with the state of the art (left, bottom row). The Cartesian reference image (AF=4, GRAPPA reconstructed) is shown in (a, left top row) [55]. SSIM scores are reported for each setup. Global 3D SPARKLING gives improved results compared to the SpSOS approach which starts to get worse at AF=15 with some blurring and at AF=20 the image gets noisier. On the other hand, TPI images are extremely blurry even at AF=10.



**Fig. S5:** Comparing the performance of full 3D SPARKLING (b) and SpSOS SPARKLING (c) with twisted projection imaging (TPI) (d) using a retrospective study at AF=10 from the Cartesian GRAPPA-4 reference (a).



**Fig. S6:** Prospective reconstruction results (axial and sagittal view only) (b) without  $\Delta B_0(\mathbf{r})$  correction and (c) with  $\Delta B_0(\mathbf{r})$  correction for full 3D SPARKLING trajectory at AF=10. Cartesian reference views (a) are also shown for comparison purpose. Red arrows in (b) refer to the regions of strong  $\Delta B_0$  artifacts. We see that most of the MR signal in these areas is recovered in (c) using the approach proposed in [63].