

Fast GRAPPA Reconstruction with Random Projection

Jingyuan Lyu,¹ Yuchou Chang,² and Leslie Ying^{1*}

Purpose: To address the issue of computational complexity in generalized autocalibrating partially parallel acquisition (GRAPPA) when several calibration data are used.

Method: GRAPPA requires fully sampled data for accurate calibration with increasing data needed for higher reduction factors to maintain accuracy, which leads to longer computational time, especially in a three-dimensional (3D) setting and with higher channel count coils. Channel reduction methods have been developed to address this issue when massive array coils are used. In this study, the complexity problem was addressed from a different perspective. Instead of compressing to fewer channels, we propose the use of random projections to reduce the dimension of the linear equation in the calibration phase. The equivalence before and after the reduction is supported by the Johnson-Lindenstrauss lemma. The proposed random projection method can be integrated with channel reduction sequentially for even higher computational efficiency.

Results: Experimental results show that GRAPPA with random projection can achieve comparable image quality with much less computational time when compared with conventional GRAPPA without random projection.

Conclusion: The proposed random projection method is able to reduce the computational time of GRAPPA, especially in a 3D setting, without compromising the image quality, or to improve the reconstruction quality by allowing more data for calibration when the computational time is a limiting factor.

Magn Reson Med 000:000–000, 2014. © 2014 Wiley Periodicals, Inc.

Key words: GRAPPA; dimension reduction; random projection; auto-calibration; Johnson-Lindenstrauss lemma

INTRODUCTION

Parallel imaging using phased array coils has been used widely in the clinical setting to accelerate the MR data acquisition speed (1–4). Generalized autocalibrating partially parallel acquisition (GRAPPA) is a popular commercial reconstruction method (3). It

reconstructs the missing k-space data for each channel (known as the target channel) by a linear combination of some acquired data from all channels (source channel), where the coefficients for combination are estimated using some additionally acquired autocalibration signal (ACS) lines. Massive array coils with a large number of channels have been studied and developed for higher signal-to-noise ratios (SNR) and/or higher accelerations (5). As massive array coils become commercially available, the greatly increased computational time has become a concern for GRAPPA reconstruction, because the reconstruction time increases almost quadratically with the number of channels (6). Such a long reconstruction time using massive array coils leads to difficulties in online high-throughput display.

A few works have attempted to address this issue using hardware- or software-based approaches to reduce the effective number of channels. In the hardware-based approach (7), a hardware radiofrequency signal combiner was placed between preamplification and the receiver system to construct an eigencoil array based on the noise covariance of the receiver array. With such a channel reduction method, optimal SNR and similar reconstruction quality can be achieved. However, the requirement of additional hardware can be cumbersome. In contrast, it is more flexible to use the software-based channel reduction methods. The coil compression process generates a new set of fewer virtual channels that can be expressed as linear combinations of the physical channels. These methods aim at reducing the effective number of channels for reconstruction by combining the physically acquired data from a large number of channels before image reconstruction. For example, principal component analysis (PCA) has been used to find the correlation among physical channels and reduce the number of channels to fewer effective ones by linearly combining the data from physical channels (8–15). These fewer combined channels are used for reconstruction, which leads to reduced reconstruction time. With channel reduction, both the numbers of target channels and source channels can be reduced. Because the ultimate goal is the reconstruction of a single-channel image, several studies (16–19) have investigated the synthesis of a single target channel for k-space-based reconstruction techniques. These methods compress multiple channels to a single channel prior to reconstruction so that the convolution-based calibration and synthesis need only be performed once (instead of performing it for each channel), thereby achieving significant computation reduction. A method has also been proposed to reduce the computations for matrix multiplications during the calibration by removing overlapping computations (20).

¹Department of Biomedical Engineering, Department of Electrical Engineering, The State University of New York at Buffalo, Buffalo, New York, USA.

²Neuroimaging Research, Barrow Neurological Institute, Phoenix, Arizona, USA.

Grant sponsor: National Science Foundation; Grant number: CBET-1265612; Grant sponsor: State University of New York at Buffalo.

*Correspondence to: Leslie Ying, PhD, Department of Biomedical Engineering, Department of Electrical Engineering, School of Engineering and Applied Science, University at Buffalo, State University of New York, 223 Davis Hall, Buffalo, NY 14260. leiying@buffalo.edu

Additional Supporting Information may be found in the online version of this article.

Received 21 March 2014; revised 25 June 2014; accepted 26 June 2014

DOI 10.1002/mrm.25373

Published online 00 Month 2014 in Wiley Online Library (wileyonlinelibrary.com).

In this study, we investigated the use of random projection to reduce the computation in GRAPPA calibration. Random projection has been used for data dimension reduction in machine learning (21,22). The concept of random projection is related to compressed sensing (23,24), a topic that has attracted much attention recently. By projecting the data to lower dimensions using some random matrices with certain properties (e.g., the restricted isometry property), the useful information is still preserved in the reduced data. When applied to GRAPPA, random projection reduces the large number of equations during the calibration phase by projecting the calibration equation onto a much lower dimensional space using random matrices, such that the computational time for solving the calibration equation is reduced without compromising the calibration accuracy. This is quite different from existing methods wherein the number of channels is compressed before reconstruction. Our theoretical calculation and experimental results here demonstrate that in a typical setting, the proposed method can achieve the same reconstruction quality with much less computation time of the conventional GRAPPA. In addition, the method can be complemented by a channel reduction method for further savings in computational time. Several two-dimensional (2D) and three-dimensional (3D) reconstruction experiments with different numbers of physical channels are used to demonstrate the significant improvement in computational efficiency. The preliminary idea of this study was previously presented by Lyu et al. (25,26).

THEORY

Complexity of 2D GRAPPA

There are two phases in GRAPPA reconstruction: the calibration phase and the synthesis phase. During the calibration phase, the acquired ACS data are used to calibrate the GRAPPA reconstruction coefficients in k-space. Specifically, the k-space data point that should be skipped during the accelerated scan is assumed to be approximately equal to a linear combination of the acquired undersampled data in the neighborhood from all coils, which can be represented as

$$S_j(k_y + r\Delta k_y, k_x) = \sum_{c=1}^{N_c} \sum_{b=B_1}^{B_2} \sum_{h=H_1}^{H_2} w_{j,r}(c, b, h) \times S_c(k_y + bR\Delta k_y, k_x + h\Delta k_x), \quad [1]$$

where S_j on the left-hand side denotes the target data that should be skipped but is acquired for the calibration purpose, and S_c on the right-hand side is the source k-space signals that should originally be acquired, both in the ACS region. Here, w denotes the coefficient set; R represents the reduction factor (ORF); j is the target coil; r is the offset; c counts all coils; b and h transverse the acquired neighboring k-space data in k_y and k_x directions, respectively; and the variables k_x and k_y represent the coordinates along the frequency- and phase-encoding

directions, respectively. The GRAPPA calibration phase can be simplified as a matrix equation

$$\mathbf{b}_{m \times l} = \mathbf{A}_{m \times n} \mathbf{x}_{n \times l}, \quad [2]$$

where \mathbf{A} represents the matrix comprised of the source data, $\mathbf{b} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_l]$ denotes the target data for calibration, $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_l]$ represents the linear combination coefficients, and l counts for all coils and all possible offsets and is equal to $(R-1)N_c$. \mathbf{A} has $N_p N_x$ rows and $N_c d_x d_y$ columns, where N_p is the number of phase-encoding lines that are possible fit locations, N_x is the number of points along the frequency encoding direction, N_c is the total number of all channels for the original k-space data, and d_x and d_y are the convolution size of GRAPPA along the frequency-encoding and phase-encoding directions, respectively. The convolution size along the phase-encoding direction is defined as the block size. In general, the coefficients \mathbf{x} depend on the coil sensitivities and are not known a priori. In calibration, the goal is to find the unknown \mathbf{x} based on the matrix in Equation 2. The target data \mathbf{b} and matrix \mathbf{A} include data at all locations of the ACS region to find the best GRAPPA coefficients \mathbf{x} . The formulation in Equation 2 allows different coils and different target offsets to share the same \mathbf{A} such that overlapping computations are avoided in solving the equation, as in Brau et al. (6) and Beatty et al. (20). The least-squares method is commonly used to calculate the coefficients

$$\hat{\mathbf{x}} = \min_{\mathbf{x}} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_F^2, \quad [3]$$

where the subscript F denotes the Frobenius norm. Many ACS data points are usually acquired to set up Equation 2, which means the problem is well overdetermined. In this case, the solution is given by

$$\mathbf{x} = (\mathbf{A}^H \mathbf{A})^{-1} (\mathbf{A}^H \mathbf{b}). \quad [4]$$

It is worth mentioning that there are other methods (e.g., LU decomposition and iterative conjugate gradient) to solve Equation 2. They may be more efficient than the pseudo-inverse method in Equation 4, depending on the scale of the equation. In this study, we focused on the pseudo-inverse method to be consistent with the complexity analysis described by Brau et al. (6).

During the synthesis phase, based on the shift-invariant assumption, the same Equation 2 is used with the same \mathbf{x} obtained from Equation 4, but to obtain the unknown \mathbf{b} using the acquired k-space data outside the ACS region for \mathbf{A} . By this means, the missing data \mathbf{b} are estimated by the linear combination of the acquired data in \mathbf{A} and the full k-space data can be used to obtain the image of each channel. The final image is reconstructed using the root sum of squares of the images from all channels.

The computational expense of the calibration and synthesis phases can be estimated using matrix multiplication and inversions (6). Several different modes can be calculated individually and then combined using the goodness of fit (3). For each mode, the calibration phase requires $(mn^2 + nl(m+n) + n^3)$ complex-valued multiplications, where $m = N_p N_x$, $n = d_x d_y N_c$, $l = (R-1)N_c$. It is

seen that $\mathbf{A}^H \mathbf{A}$ dominates the calibration cost for $m \gg n$, which requires mn^2 complex-valued multiplications. Furthermore, in the synthesis phase of GRAPPA, $N_u N_x N_c^2 d_x d_y (R-1)$ complex-valued multiplications are needed for each mode, where N_u is the number of phase-encoding lines to be synthesized at a particular offset. The number of modes is commonly chosen to be the same as the number of blocks d_y . The computation of the goodness of fit itself is typically negligible compared with other calculations. Therefore, the total computational expense for GRAPPA reconstruction is approximately on the order of $N_p N_x (N_c d_x)^2 d_y^3 + N_p N_x N_c^2 d_x d_y^2 (R-1) + N_c^3 d_x^2 d_y^3 (R-1) + (N_c d_x)^3 d_y^4 + N_u N_x N_c^2 d_x d_y^2 (R-1)$. The analysis shows that the computational complexity increases almost quadratically with the number of channels. Therefore, channel reduction methods are able to reduce the total reconstruction time significantly. On the other hand, the calibration phase is seen to involve high computational cost due to solving inverse equations. If we adopt the commonly used reconstruction parameters $N_{acs}=30$, $N_x=256$, $d_x=13$, $d_y=4$, $R=3$, $N_u=75$, $N_c=8$, block size=4, and $N_p=24$, the ratio between the calibration and synthesis phases is 9.3. This confirms that the calibration phase dominates the total computational time. It is worthwhile to reduce the computational cost of the calibration phase as done in the proposed method.

PCA-Based Dimension Reduction Method for Channel Compression

Dimensionality reduction has been well studied in the machine learning community. It involves projecting data from a high-dimensional space to a lower-dimensional one without a significant loss of information. The new data with reduced dimensions are expected to maintain most of the important information. Among the existing linear approaches that use linear mapping for dimensionality reduction, PCA is a classical method to remove redundant information from statistically correlated data and reduce the dimensions. It performs the eigen decomposition on the covariance matrix of the data and removes the components that correspond to the smallest eigenvalues. Among all linear approaches, PCA minimizes the projection residuals and maximizes the variance between the combined variables (27–29). A major limitation of PCA is its computational complexity. Performing a PCA requires an order of N^3 multiplications, with N being the original dimension, which becomes prohibitive when dealing with a large dataset.

In parallel imaging, PCA has been successfully used to compress the channels in large array coils (8–11). The resulting fewer combined channels corresponding to the largest eigenvalues are used for reconstruction such that the reconstruction time is reduced. Because the number of channels is not a large number in general, the computational saving from channel reduction overrides the additional computational cost of PCA.

Random Projection for GRAPPA Calibration

Dimensionality reduction is concerned with the problem of projecting a set of n points in \mathbb{R}^m , with m typically large into a lower dimensional Euclidean space \mathbb{R}^k while

approximately preserving the relative distance between any two of these points. We need to know how small k could become and what type of projections would work. The Johnson-Lindenstrauss lemma (30–34) deals with such a problem.

LEMMA

(Johnson and Lindenstrauss, 1984) Suppose we have an arbitrary matrix $\mathbf{A} \in \mathbb{R}^{n \times m}$. Given any $\varepsilon > 0$, there is a mapping $f: \mathbb{R}^m \rightarrow \mathbb{R}^k$, for any $k \geq 12 \frac{\log n}{\varepsilon^2}$, such that for any two rows $u, v \in \mathbf{A}$, we have

$$(1 - \varepsilon) \|f(u) - f(v)\|^2 \leq \|u - v\|^2 \leq (1 + \varepsilon) \|f(u) - f(v)\|^2. \quad [5]$$

The theorem states that we can find such a projection f for the set of points that preserves pairwise distances up to a factor of $(1 \pm \varepsilon)$, but does not state how to find such an f . More recently, there are simple proofs of the lemma that show that f can be taken as a linear mapping represented by a $k \times m$ matrix \mathbf{R} whose entries are randomly drawn from certain probability distributions (31,35). Apparently, the Johnson-Lindenstrauss lemma shares similarity with the well-known restricted isometry property in compressed sensing (23). In fact, the same proof used for Johnson-Lindenstrauss lemma can be used to prove the restricted isometry property of any random matrices (34,36). Based on the Johnson-Lindenstrauss lemma, random matrices with each element independently drawn from a certain distribution will satisfy the condition for f in Equation 5 with high probability, provided k satisfied the conditions of the lemma.

Where preservation of the relative distance between any two points is concerned, projections with random matrices (i.e., random projections) can be used for dimensionality reduction, which has been well studied in machine learning (21,22). Here, we propose to reduce the dimension of the GRAPPA calibration equation using random projection. In the GRAPPA calibration given by Equations 1 and 2, our objective is to reduce the dimension of $\mathbf{A}\mathbf{x}$ and \mathbf{b} such that their relative distance $\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_F$ is preserved. The random projection is exactly able to serve the purpose based on the Johnson-Lindenstrauss lemma. Specifically, we define a random matrix \mathbf{R} of $k \times m$ to project the original m -dimensional data onto a k -dimensional ($k \ll m$) subspace with a dimension reduction factor m/k . We then multiply the random matrix on both sides of the calibration in Equation 2:

$$\mathbf{R}_{k \times m} \mathbf{b}_{m \times 1} = \mathbf{R}_{k \times m} \mathbf{A}_{m \times n} \mathbf{x}_{n \times 1}, \quad [6]$$

where k is the reduced dimension. Applying the random projection using Equation 6 is equivalent to linearly combining a subset of equations in a random fashion to form a new reduced set of equations. With such a random projection, we have a high probability to maintain the important information in the lower dimensional space after the projection. The solution to the reduced set of equations is approximately the same as the original one because for any $\mathbf{x}_{n \times 1}$,

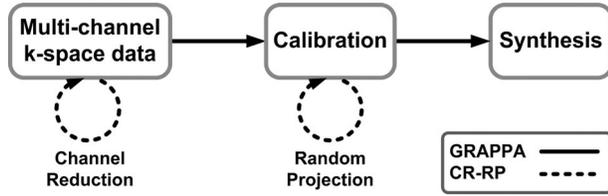


FIG. 1. Diagram of GRAPPA, RP-GRAPPA, CR-GRAPPA, and CR-RP-GRAPPA.

$$\| \mathbf{R}_{k \times m} \mathbf{A}_{m \times n} \mathbf{x}_{n \times l} - \mathbf{R}_{k \times m} \mathbf{b}_{m \times l} \|_F \approx \| \mathbf{A}_{m \times n} \mathbf{x}_{n \times l} - \mathbf{b}_{m \times l} \|_F \quad [7]$$

provided k is not too small.

Choice of \mathbf{R}

In contrast to PCA, the computational complexity of random projection is rather low, requiring km multiplications in general. The complexity can be further reduced by carefully choosing the distribution. Although most random matrices would be applicable, here we use a matrix \mathbf{R} whose elements are drawn independently from identical distribution with the following density (31):

$$R(i, j) = \frac{1}{\sqrt{p}} \begin{cases} 1 & p/2 \\ 0 & \text{with probability of } 1-p \\ -1 & p/2 \end{cases} \quad [8]$$

This random projection has been proved to satisfy the Johnson-Lindenstrauss lemma, where p is usually equal to 1, 1/3 (31), and $1/\sqrt{m}$ (33). Such a simple and sparse construction in Equation 8 incurs a large saving in computational cost, because only a single multiplication and very few additions are needed when computing the projection. As p goes smaller, the matrix \mathbf{R} becomes sparser and thus the computation is faster. The effect of p on reconstruction accuracy is described in the Results. The extreme case of the sparsest \mathbf{R} with full rank is that it has only one element in each of the k rows being non-zero. In that case, random projection with the sparsest \mathbf{R} is equivalent to randomly selecting k rows of the calibration matrix, which minimizes the computational cost of the random projection process. Such a strategy of randomly selecting equations has been used in solving large scale over determined systems and shown to improve convergence of the projection onto convex set (37). We use such a sparsest random projection for this study. An example of the sparsest random projection matrix is provided online in the Supporting Information.

Choice of k

The reduced dimension k directly affects the computational complexity of calibration. With random projection, the computational cost of the calibration phase is greatly reduced. Instead of solving an $m \times n$ equation, we solve a $k \times n$ equation, where $k \ll m$. We define a factor $\lambda = k/n > 1$, which represents how much the new equation is overdetermined after dimensionality reduction. Here we heuristically find the optimal λ to be in the

range of 2~4. That is, for the same number of unknowns n , the number of equations is reduced to be about 2~4 times that of unknowns. This optimal range is found to be independent of the parameters in GRAPPA. Further studies on the effect of λ are described in the Results. The exact computational savings depend on the specific problem using GRAPPA. For example, if we assume $k=2n$ and the commonly used parameters for 2D GRAPPA (e.g., 30 ACS, eight channels with ORF of 3), then the size of the equation is reduced from 6144×416 in Equation 3 to 832×416 in Equation 6. The calibration phase has a saving of approximately 5.3 times. With such a saving, the ratio between the computation time of the calibration and synthesis phases is reduced from 9.3 to 1.7. Compared with the conventional GRAPPA, the random projection brings in a total saving of 3.8 times in computational cost. If more data are involved in GRAPPA, as in the cases of 3D reconstruction and massive array coils, much more savings are expected.

Integrating Channel Reduction with Random Projection

Although PCA and random projection are both linear dimensionality reduction methods, they have different objectives and properties after the reduction. In PCA, the variance is to be maximized, which is ideal for channel reduction. While in random projection, the data distance needs to be maintained, which is ideal for reducing linear equations without changing the least squares solutions. In addition, the two methods are exploited at different stages of GRAPPA. PCA is performed before the reconstruction, and random projection is done during the calibration phase of reconstruction. The relationship is illustrated in Figure 1. Therefore, both methods can be integrated for a significant reduction in computational cost of GRAPPA.

METHODS

The computational benefit of the proposed random projection method for GRAPPA is evaluated on three scanned datasets. The first axial brain dataset was acquired on a GE 3T scanner (GE Healthcare, Waukesha, Wisconsin, USA) with an eight-channel head coil using a 2D spin echo sequence (echo time [TE]/pulse repetition time [TR]=11/700 ms; matrix size=256 × 256; field of view [FOV]=220 × 220 mm²). The second dataset was acquired on a Philips Ingenia 3T scanner (Philips Healthcare, Best, Netherlands) with 12-channel head coil using a 3D Fast Field Echo sequence (TE/TR=4.6/25 ms; matrix size=239 × 239 × 83; FOV=240 × 240 × 130 mm³). In the third dataset, a set of axial brain data was acquired on a Siemens 3T scanner (Siemens Healthcare, Erlangen, Germany) with a 32-channel head coil using a 2D gradient echo sequence (TE/TR=2.29/100 ms; flip angle=25°; matrix size=256 × 256; slice thickness=3 mm; FOV=240 × 240 mm²).

These datasets were acquired in full and the square-root of sum-of-squares of the images from fully sampled data of all coils was used as a reference. The full k-space data were then manually undersampled retrospectively to simulate the accelerated acquisition for GRAPPA. We compared the GRAPPA reconstructions with and without random projection to demonstrate the computational

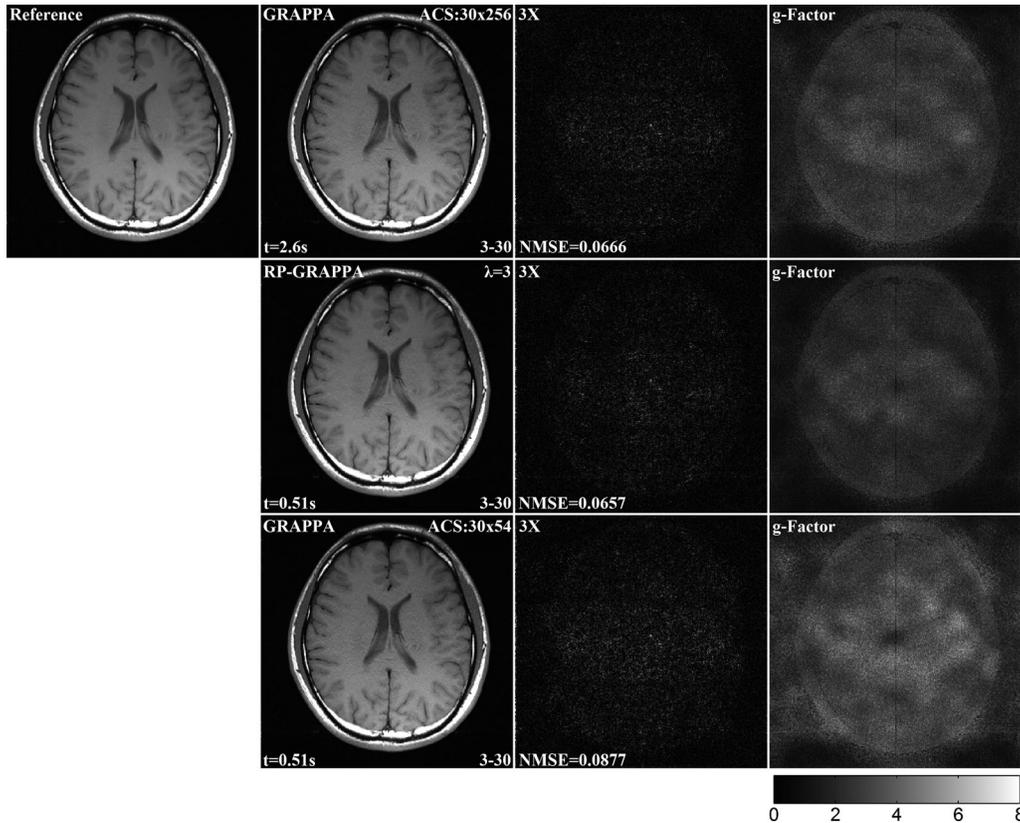


FIG. 2. Axial brain images reconstructed from a set of eight-channel data with an ORF of 3 and 30 ACS lines using GRAPPA and RP-GRAPPA. The corresponding difference images are shown in the middle column ($3\times$ amplification) and g-factor maps are shown in the right column. The proposed RP-GRAPPA can achieve a similar reconstruction quality to GRAPPA with much less CPU time or take about the same CPU time as GRAPPA but with better SNR.

savings of the proposed random projection method over the conventional GRAPPA. In particular, we compared the two approaches when the same amount of ACS data was used as well as when both had the same computational complexity but with reduced ACS data for conventional GRAPPA. The reconstruction quality was evaluated both visually and quantitatively using the normalized mean squared error (NMSE) with the reference image as the gold standard. The computational savings are measured in terms of both the computational complexity calculated analytically as the total number of multiplications and the computational time measured in CPU time. Due to the random nature of the proposed method, the NMSE and CPU time were both obtained from the average results of 50 executions. For the proposed random projection method, we varied the parameter $\lambda = k/n$. The smaller the λ was, the less the equation was overdetermined and the less the computation was needed. For the 32-channel dataset, we also combined random projection with channel reduction methods. Specifically, we used the PCA channel reduction method to compress the 32 channels to fewer source channels and target channels (8–11), then applied the proposed random projection method to GRAPPA reconstruction.

All methods were implemented in MATLAB (Mathworks, Natick, Massachusetts, USA) and run on a PC with an Intel i7-3700 3.4GHz 8-core CPU with a 16-GB memory, except that the 3D data set was processed on a

workstation with an Intel Xeon X5492 3.4GHz dual 4-core CPU with a 32-GB memory. To reduce the CPU time in 3D GRAPPA, parallel computing was implemented in the synthesis phase using the MATLAB Parallel Computing Toolbox.

RESULTS

Random Projection for 2D GRAPPA

We evaluated the computational savings of random projection on GRAPPA (RP-GRAPPA) for 2D reconstruction using the first dataset. The data were undersampled along the phase encoding direction with an ORF of 3 with 30 ACS lines. Both GRAPPA and RP-GRAPPA were used to reconstruct the final image, with the number of blocks d_y being 4 and the number of columns d_x 13. The CPU times of GRAPPA and RP-GRAPPA were 2.60 s and 0.51 s, respectively. The CPU time ratio between GRAPPA and RP-GRAPPA was about 5. Such a saving in computation time roughly agrees with our theoretical analysis. In addition, we also reduced the number of columns of ACS data in GRAPPA reconstruction to make the computational time to be about the same as that of RP-GRAPPA. The reconstructions, their corresponding difference images with the reference, and the g-factor maps are shown in Figure 2. The NMSE were 0.0666 for GRAPPA, 0.0657 for RP-GRAPPA ($\lambda = 3$), and 0.0877 for ACS-reduced GRAPPA. The proposed random projection

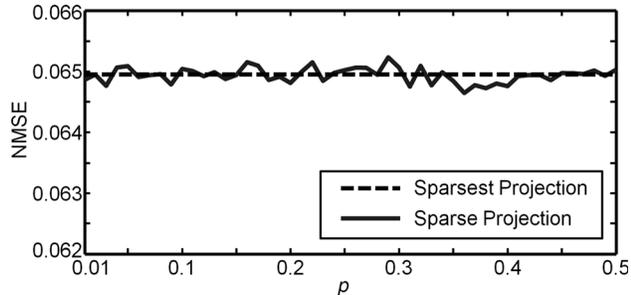


FIG. 3. NMSE of RP-GRAPPA versus the sparsity-control parameter ρ of the random projection matrix (eight-channel dataset; ACS=30, ORF=3, $\lambda=3$) suggests that the sparsity of random projection barely affects the reconstruction accuracy.

method was able to reduce the GRAPPA reconstruction time by a factor of 5 without compromising the image quality. Equivalently, the proposed method was able to improve the GRAPPA reconstruction quality with the same computational time.

We also studied how the parameters of p and λ in random projection affected the reconstruction accuracy. Specifically, we first changed the value for the parameter p from 0.01 to 0.5, which controlled the sparsity of the random projection matrix \mathbf{R} . We also used the sparsest \mathbf{R} , which randomly selects rows of the calibration equation. The corresponding reconstruction errors are plotted in Figure 3 when λ was fixed to 3, which controlled the level of overdeterminedness after projection. These results show that the sparsity of the random projection matrix barely affected the reconstruction accuracy. Similar reconstruction errors were achieved with both non-sparse ($p=0.5$) and the sparsest random projections. For these two extreme cases, we also compared the reconstruction errors as well as the CPU time in Figure 4 when λ went from 1.1 to 4. The nonsparse random projection was more accurate for small λ , but the sparsest projection became more accurate when $\lambda > 2.2$. On the other hand, the sparsest random projection had a significant benefit in computational efficiency, especially at a large λ . These results suggest that the projection with randomly selected rows is a good choice in practice.

With random row selection, we also calculated the overall (both calibration and synthesis) computation complexity as well as the NMSE of RP-GRAPPA as λ increased from 1.1 to 4, with a step size of 0.05. The CPU time was also recorded. The results of NMSE, complexity, and CPU time are shown on the same plot in Figure 5. The NMSE decreased rapidly (approximately exponentially) as λ increased and became sufficiently low and remained steady for $\lambda > 2.2$. On the other hand, the computation complexity increased linearly as λ increased. The almost linear trend of CPU time with respect to λ was consistent with that of the computation complexity. Figure 5 suggests that λ has to be sufficiently large to maintain the reconstruction quality. When λ increased, reconstruction error was suppressed exponentially, while the CPU time increased only linearly. For $\lambda \geq 3$, the NMSE varied slowly. The optimal value for λ to balance the tradeoff between NMSE and CPU time is seen to be 3.

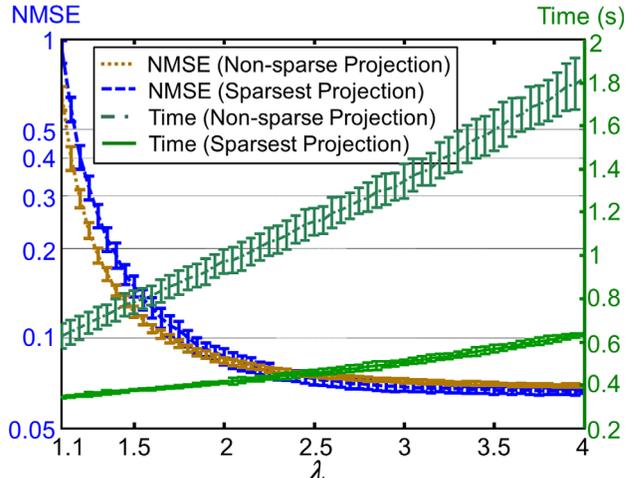


FIG. 4. NMSE and CPU time of RP-GRAPPA versus λ with sparsest and nonsparse random projection matrices (eight-channel dataset; ACS=30, ORF=3). All curves were acquired by averaging the results from 50 \times experiments. The error bars represent the standard deviation.

Random Projection for 3D GRAPPA

In 3D reconstruction, the calibration phase becomes even more time consuming due to the larger amount of data involved in the equation. We use the second dataset to evaluate the computational savings in 3D GRAPPA reconstruction. For this 12-channel dataset with a matrix size of $239 \times 239 \times 83$, an ORF of 2×2 with 35×33 ACS lines was used. The number of blocks d_y and d_z were both 4, and the number of columns d_x was 13. The central 101 columns (frequency encodings) of these ACS lines were initially used for calibration because the high computation complexity involved such a large set of 3D calibration data. The CPU times for 3D RP-GRAPPA ($\lambda=2$) and 3D GRAPPA were 1540 s and 13743 s, respectively, and the computational complexities were 4.59 and 32.45

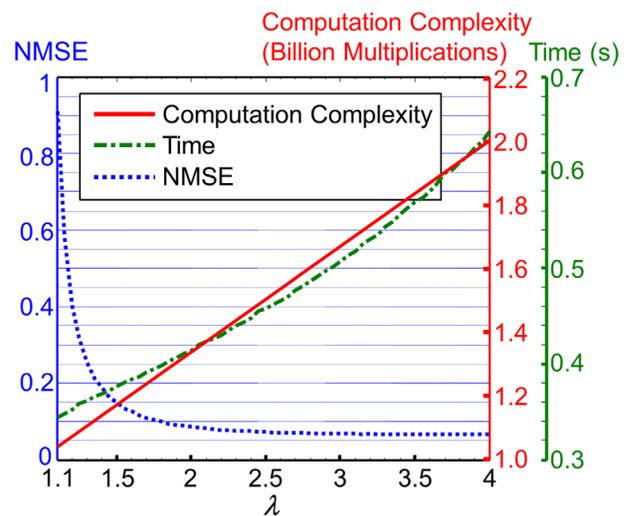


FIG. 5. NMSE and CPU time of RP-GRAPPA versus λ (eight-channel dataset; ACS=30, ORF=3). The CPU time curve is approximately linear, which agrees with the theoretical curve of computation complexity.

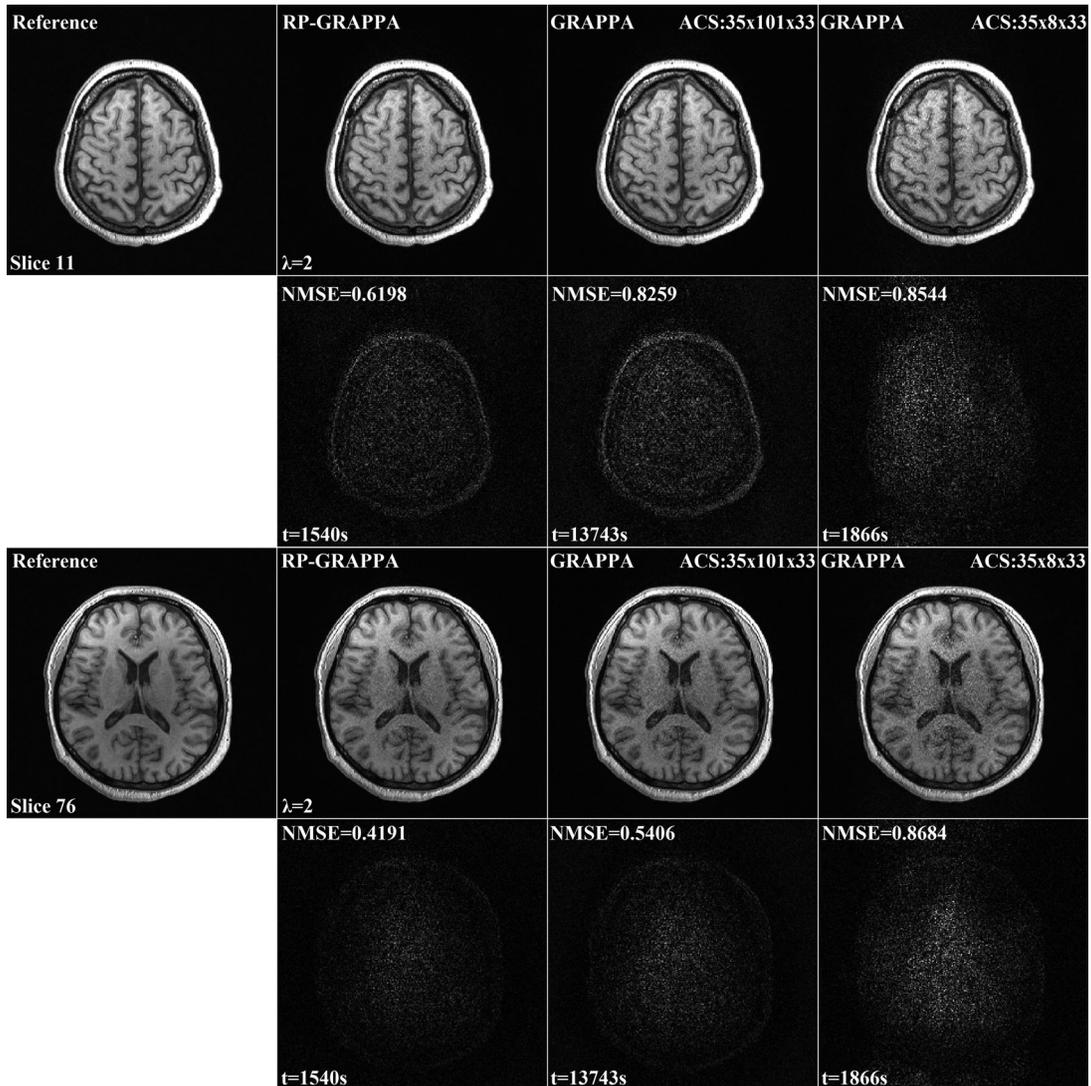


FIG. 6. 3D reconstruction using RP-GRAPPA (101 ACS columns) and GRAPPA (101 and 8 ACS columns) with 2D undersampling (12-channel dataset; ACS = 35 × 33, ORF = 2 × 2). The proposed 3D RP-GRAPPA with 101 ACS columns takes about the same CPU time as 3D GRAPPA with eight ACS columns, but achieves better SNR.

trillion multiplications, respectively. These findings indicate that the proposed random projection brings a saving of about 8.9 times in 3D GRAPPA reconstruction time. Due to such inhibiting time for 3D GRAPPA without random projection, much fewer ACS data are usually used in practice. We repeated GRAPPA with only eight columns of ACS data such that the resulting calibration time (1866 s CPU time and 4.73 trillion multiplications) was approximately the same as that of RP-GRAPPA with 101 columns. We compared the 3D GRAPPA and 3D RP-GRAPPA reconstructions with their corresponding references in Figure 6. The results suggest that directly reducing ACS columns for fast 3D GRAPPA reconstruction leads to poorer SNR, while 3D RP-GRAPPA achieves fast reconstruction without compromising SNR.

Random Projection with Channel Reduction

We used the third dataset with 32 channels to demonstrate the benefit of integrating channel reduction (CR)

with random projection for GRAPPA reconstruction (CR-RP-GRAPPA), where PCA was used for channel reduction (10). The data were manually undersampled with ACS = 48 and ORF = 5 (net acceleration of 2.84). The number of columns d_x was 15. For PCA-based channel reduction, the source channel number and target channel number were chosen to be $N_{c-s} = 24$ and $N_{c-t} = 8$, respectively. In Figure 7, the results of CR-RP-GRAPPA are compared with those of RP-GRAPPA, CR-GRAPPA, and conventional GRAPPA in terms of visual quality, NMSE, and reconstruction time. These findings indicate that random projection alone is more effective than channel reduction alone in reducing CPU time without compromising image quality. When the random projection and channel reduction are combined, the reconstruction time can be significantly reduced. In this particular case with $\lambda = 2.5$, a factor of up to 11 in savings can be achieved for the GRAPPA reconstruction time.

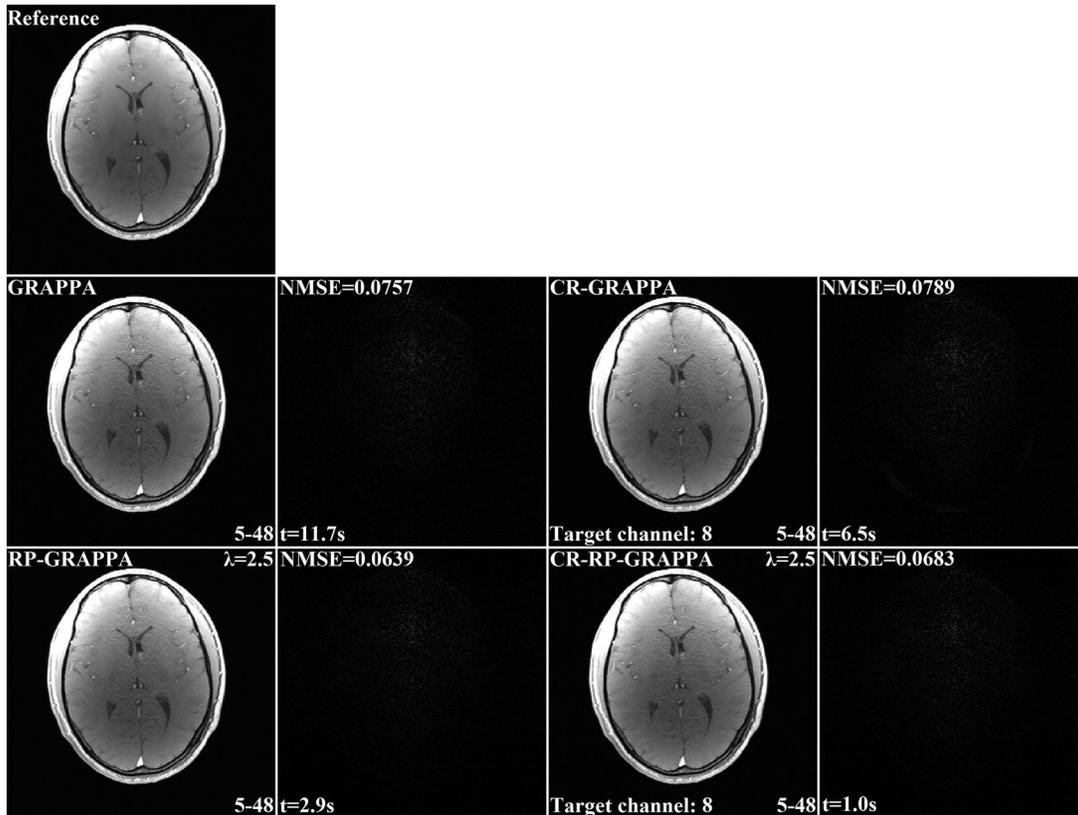


FIG. 7. Brain images reconstructed from a set of 32-channel data with an ORF of 5 and 48 ACS lines using GRAPPA, RP-GRAPPA, CR-GRAPPA, and CR-RP-GRAPPA. PCA was used in channel reduction to reduce to eight target channels. The random projection method when integrated with existing channel reduction method shows even lower CPU time.

DISCUSSION

There are three ways to reduce the number of calibration equations in order to reduce the computational complexity. The most straightforward way is to reduce the amount of ACS data. In this study, we show that a large reduction usually leads to degradation in reconstruction quality. The other two approaches are channel reduction and random projection. The current study shows that the channel reduction method is very effective when there are a large number of channels. However, very little savings can be achieved when only a few physical channels are used. This is because the channel reduction method only reduces the data redundancy cross channels. To this end, the proposed random projection method complements the channel reduction method perfectly for a large savings in computation because it reduces the redundancy among the calibration equations. It is worth noting that both PCA, which is used for channel reduction, and random projection are widely used dimension reduction methods. The reason that PCA is appropriate only for channel reduction is that its own computational complexity is large when the size of the data becomes large. On the other hand, random projection is effective only when the dimension of the data is very high, and is thereby not effective in channel reduction.

Interestingly, Figures 2, 6, and 7 show that the NMSE of the GRAPPA reconstructions with random projection are usually lower than those without. This is possibly

because random projection has denoising capability at a certain level. When the error primarily comes from the noise in \mathbf{b} of Equation 3, random projection is able to reduce the error. In order to understand the observation more comprehensively, we compare in Figure 8 the NMSE of GRAPPA and RP-GRAPPA at different levels of SNR, where white Gaussian noise is added onto the acquired k-space data (assumed to be noise-free) of the first dataset to simulate different SNR. To avoid aliasing artifacts at low SNR, we choose 48 ACS lines. Other parameters remain the same as those used in Figure 2. It is seen that when the SNR is low (e.g., below 30 dB), the proposed random projection method results in poorer NMSE. As SNR increases, random projection is able to improve the NMSE. This is because at high SNR, the noise reduction from \mathbf{Rb} overweights the system error caused by \mathbf{RA} in Equation 6. When the SNR is low, the calibration matrix \mathbf{A} is affected by larger noise, and consequently, the system error due to \mathbf{RA} becomes dominant. Nevertheless, the NMSE difference is so small that GRAPPA and RP-GRAPPA can be regarded to have the same reconstruction quality.

We have also studied the effect of different random selections (i.e., different \mathbf{R} matrices) on the reconstruction quality in terms of NMSE. Table 1 shows the means and standard deviations of 50 executions for two different random selections. One is pure random used in our study, and the other is variable density random which

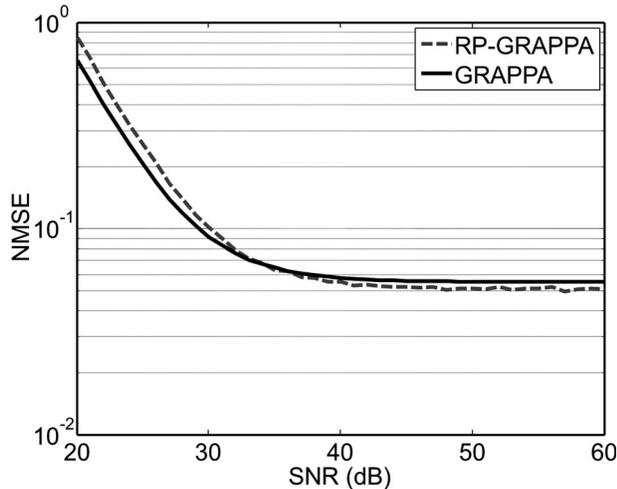


FIG. 8. NMSE of RP-GRAPPA and GRAPPA versus SNR (eight-channel dataset; ACS = 48, ORF = 3, $\lambda = 2$).

selects more equations from the center ACS. The same 2D dataset and parameters were used as in Figure 2. The worst case NMSE of pure random selection was 0.0729, which is even lower than the mean NMSE of the other case. It is evident that the pure random selection is superior to the variable density random selection. The results are also consistent with those in Figure 2, where GRAPPA calibration using the center ACS is inferior only to that using randomly selected ACS.

The proposed method can be integrated with any linear equation solvers (e.g., LU decomposition) in addition to the pseudo-inverse used here to improve the computational efficiency of GRAPPA calibration, because the number of rows of \mathbf{A} affects the complexity of all solvers. In addition to GRAPPA, the proposed random projection method is also applicable to any algorithm that involves solving a large, overdetermined linear equation. For example, it can be applied to most of the GRAPPA extensions (e.g., refs. 38–41). However, if the calibration phase does not dominate the total reconstruction time as in SPIRiT (39), the computational savings are limited. On the other hand, when calibration is time consuming, the benefit is prominent. For example, robust GRAPPA (41) is known to be robust to outliers but has lengthy calibration times. When the proposed random projection method is applied to robust GRAPPA, there is a huge time savings. Table 2 compares both NMSE and CPU times of robust GRAPPA with and without random projections using the first dataset. The NMSE of robust GRAPPA with random projection is nearly the same as that without random projection, but the former saves a significant amount of computation time (~ 34 times).

Table 1
Comparison of Pure Random with Variable-Density Random Selections (ACS = 30, ORF = 3, $\lambda = 3$)

RP-GRAPPA NMSE	Pure random	Variable-density random
Mean	0.0685	0.0819
Standard deviation	0.0018	0.0012

Table 2

Comparison of Robust GRAPPA with and without RP (ACS = 30, ORF = 3, $\lambda = 2$)

Robust GRAPPA	NMSE	CPU time (s)
With RP	0.0618	53
Without RP	0.0645	1835

CONCLUSIONS

In this study, we propose a random projection method to reduce the dimension of the overdetermined equations in GRAPPA calibration and thus save computation time. The proposed method is easily integrated with the existing channel reduction methods. Experimental results demonstrate that random projection can reduce CPU time by a factor of up to 5 for 2D GRAPPA and 8.9 for 3D GRAPPA without compromising the reconstruction quality, and when combined with PCA channel reduction, a factor of up to 11. Equivalently, when the computation time is the same, random projection is able to improve the GRAPPA reconstruction quality by taking advantage of more ACS data.

REFERENCES

- Sodickson DK, Manning WJ. Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays. *Magn Reson Med* 1997;38:591–603.
- Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P. SENSE: sensitivity encoding for fast MRI. *Magn Reson Med* 1999;42:952–962.
- Griswold MA, Jakob PM, Heidemann RM, Nittka M, Jellus V, Wang J, Kiefer B, Haase A. Generalized autocalibrating partially parallel acquisitions (GRAPPA). *Magn Reson Med* 2002;47:1202–1210.
- Ying L, Liang Z-P. Parallel MRI using phased array coils. *IEEE Signal Processing Magazine* 2010;27:90–98.
- Schmitt M, Potthast A, Sosnovik DE, Polimeni JR, Wiggins GC, Triantafyllou C, Wald LL. A 128-channel receive-only cardiac coil for highly accelerated cardiac MRI at 3 Tesla. *Magn Reson Med* 2008;59:1431–1439.
- Brau AC, Beatty PJ, Skare S, Bammer R. Comparison of reconstruction accuracy and efficiency among autocalibrating data-driven parallel imaging. *Magn Reson Med* 2008;59:382–395.
- King SB, Varosi SM, Duensing GR. Optimum SNR data compression in hardware using an eigencoil array. *Magn Reson Med* 2010;63:1346–1356.
- Buehrer M, Pruessmann KP, Boesiger P, Kozierke S. Array compression for MRI with large coil arrays. *Magn Reson Med* 2007;57:1131–1139.
- Doneva M, Börner P. Automatic coil selection for channel reduction in SENSE-based parallel imaging. *Magn Reson Mater Phys* 2008;21:187–196.
- Huang F, Vijayakumar S, Li Y, Hertel S, Duensing GR. A software channel compression technique for faster reconstruction with many channels. *Magn Reson Imaging* 2008;26:133–141.
- Huang F, Lin W, Duensing GR, Reykowski A. A hybrid method for more efficient channel-by-channel reconstruction with many channels. *Magn Reson Med* 2012;67:835–843.
- Feng S, Ji J. Channel reduction in massive array parallel MRI. In *Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medical and Biological Society, Minneapolis, Minnesota, USA, 2009*. p. 4045–4048.
- Zhang T, Pauly JM, Vasanawala SS, Lustig M. Coil compression for accelerated imaging with Cartesian sampling. *Magn Reson Med* 2013;69:571–582.
- Bahri D, Uecker M, Lustig M. ESPIRiT-Based Coil Compression for Cartesian Sampling. In *Proceedings of the 21st Annual Meeting of ISMRM, Salt Lake City, Utah, USA, 2013*. Abstract 2657.
- Uecker M, Lai P, Murphy MJ, Virtue P, Elad M, Pauly JM, Vasanawala SS, Lustig M. ESPIRiT—an eigenvalue approach to

- autocalibrating parallel MRI: where SENSE meets GRAPPA. *Magn Reson Med* 2014;71:990–1001.
16. Feng S, Zhu Y, Ji J. Efficient large-array k-domain parallel MRI using channel-by-channel array reduction. *Magn Reson Imaging* 2011;29:209–215.
 17. Beatty PJ, Sun W, Brau AC. Direct Virtual Coil (DVC) Reconstruction for Data-Driven Parallel Imaging. In Proceedings of the 16th Annual Meeting of ISMRM, Toronto, Canada, 2008. Abstract 8.
 18. Beatty PJ, Chang S, Holmes JH, Wang K, Brau AC, Reeder SB, Brittain JH. Design of k-space channel combination kernels and integration with parallel imaging. *Magn Reson Med* 2014;71:2139–2154.
 19. Chen W, Hu P, Meyer CH. Rapid Partially Parallel Reconstruction Using Single Synthetic Target Coil. In Proceedings of the 16th Annual Meeting of ISMRM, Toronto, Canada, 2008. Abstract 1296.
 20. Beatty PJ, Brau AC, Chang S, Joshi SM, Michelich CR, Bayram E, Nelson TE, Herfkens RJ, Brittain JH. A Method for Autocalibrating 2-D Accelerated Volumetric Parallel Imaging with Clinically Practical Reconstruction Times. In Proceedings of the 15th Annual Meeting of ISMRM, Berlin, Germany, 2007. Abstract 1749.
 21. Arriaga RI, Vempala S. An algorithmic theory of learning: robust concepts and random projection. *Machine Learning* 2006;63:161–182.
 22. Kaski S. Dimensionality reduction by random mapping. In the 1998 IEEE International Joint Conference on Neural Networks Proceedings, Anchorage, Alaska, USA, 1998. p. 413–418.
 23. Candès EJ, Romberg J, Tao T. Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. *IEEE Trans Inf Theory* 2006;52:489–509.
 24. Donoho DL. Compressed sensing. *IEEE Trans Inf Theory* 2006;52:1289–1306.
 25. Lyu J, Chang Y, Ying L. Efficient GRAPPA reconstruction using random projection. In Proceedings of the 10th IEEE International Symposium on Biomedical Imaging, San Francisco, California, USA, 2013. p. 696–699.
 26. Lyu J, Chang Y, Ying L. A Random Projection Approach to Highly Efficient GRAPPA Reconstruction. In Proceedings of the 21st Annual Meeting of ISMRM, Salt Lake City, Utah, USA, 2013. Abstract 2653.
 27. Geladi P, Kowalski BR. Partial least-squares regression: a tutorial. *Analytica Chimica Acta* 1986;185:1–17.
 28. Tipping ME, Bishop CM. Probabilistic principal component analysis. *J R Statist Soc B* 1999;61:611–622.
 29. Kwak N. Principal component analysis based on L1-norm maximization. *IEEE Trans Pattern Analysis and Machine Intelligence* 2008;30:1672–1680.
 30. Johnson WB, Lindenstrauss J. Extensions of Lipschitz mapping into Hilbert space. Conference in Modern Analysis and Probability, Contemporary Mathematics, American Mathematical Society 1984;26:189–206.
 31. Achlioptas D. Database-friendly random projections: Johnson-Lindenstrauss with binary coins. *J comput Syst Sci* 2003;66:671–687.
 32. Matoušek J. On variants of the Johnson-Lindenstrauss lemma. *Random Structures and Algorithms* 2008;33:142–156.
 33. Li P, Hastie TJ, Church KW. Very sparse random projections. In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, New York, USA, 2006. p. 287–296.
 34. Krahmer F, Ward R. New and improved Johnson-Lindenstrauss embeddings via the restricted isometry property. *SIAM J Math Anal* 2011;43:1269–1281.
 35. Frankl P, Maehara H. The Johnson-Lindenstrauss lemma and the sphericity of some graphs. *J Combin Theory Ser B* 1988;44:355–362.
 36. Baraniuk R, Davenport M, DeVore R, Wakin M. A simple proof of the restricted isometry property for random matrices. *Constructive Approximation* 2008;28:253–263.
 37. Strohmer T, Vershynin R. A randomized Kaczmarz algorithm with exponential convergence. *J Fourier Anal Appl* 2009;15:262–278.
 38. Weller DS, Polimeni JR, Grady L, Wald LL, Adalsteinsson E, Goyal VK. Denoising sparse images from GRAPPA using the nullspace method. *Magn Reson Med* 2012;68:1176–1189.
 39. Lustig M, Pauly J. SPIRiT: iterative self-consistent parallel imaging reconstruction from arbitrary k-space. *Magn Reson Med* 2010;64:457–471.
 40. Chang Y, Ying L. Nonlinear GRAPPA: a kernel approach to parallel MRI reconstruction. *Magn Reson Med* 2012;68:730–740.
 41. Huo D, David LW. Robust GRAPPA reconstruction and its evaluation with the perceptual difference model. *J Magn Reson Imaging* 2008;27:1412–1420.