Magnetic Resonance in Medicine 61:1425–1433 (2009)

HTGRAPPA: Real-Time B_1-Weighted Image Domain TGRAPPA Reconstruction

Haris Saybasili,[1,2*] Peter Kellman,[1] Mark A. Griswold,[3] J. Andrew Derbyshire,[1] and Michael A. Guttman[1]

The temporal generalized autocalibrating partially parallel acquisitions (TGRAPPA) algorithm for parallel MRI was modified for real-time low latency imaging in interventional procedures using image domain, B_1-weighted reconstruction. GRAPPA coefficients were calculated in *k*-space, but applied in the image domain after appropriate transformation. Convolution-like operations in *k*-space were thus avoided, resulting in improved reconstruction speed. Image domain GRAPPA weights were combined into composite unmixing coefficients using adaptive B_1-map estimates and optimal noise weighting. Images were reconstructed by pixel-by-pixel multiplication in the image domain, rather than time-consuming convolution operations in *k*-space. Reconstruction and weight-set calculation computations were parallelized and implemented on a general-purpose multicore architecture. The weight calculation was performed asynchronously to the real-time image reconstruction using a dedicated parallel processing thread. The weight-set coefficients were computed in an adaptive manner with updates linked to changes in the imaging scan plane. In this implementation, reconstruction speed is not dependent on acceleration rate or GRAPPA kernel size. Magn Reson Med 61: 1425–1433, 2009. © 2009 Wiley-Liss, Inc.

Key words: GRAPPA; TGRAPPA; real-time MRI; parallel MRI

Parallel MRI (pMRI) methods aim to increase imaging speed using multiple receiver coils and *k*-space undersampling. Each receiver coil is placed at a different location on the patient, thus acquiring the same data modulated by its unique spatial sensitivity pattern. For accelerated imaging, *k*-space is undersampled in the phase-encode direction, which causes aliasing in the image domain. The aliasing is then corrected using the coil sensitivity information. Several pMRI methods have been proposed, including the well-known image domain sensitivity encoding (SENSE) (1) and *k*-space domain generalized autocalibrating partially parallel acquisitions (GRAPPA) (2) algorithms. The proposed approach uses the GRAPPA algorithm for *k*-space weight calculation, which is beneficial when a smaller field of view (FOV) is desired, as this method is more tolerant of image wrap (3). Autocalibration was performed using a time interleaved acquisition scheme similar to that employed in TSENSE (4) by combining several R (acceleration factor) consecutive frames to form a low temporal resolution full *k*-space dataset for B_1 mapping, as done in temporal GRAPPA (TGRAPPA) (5).

Real-time TSENSE implementation has been demonstrated to work well with interventional applications (6). However, to the best of our knowledge TGRAPPA has not yet been deployed in such applications with real-time, low-latency image reconstruction. This may be due to the fact that the TGRAPPA algorithm uses convolution-like operations to estimate the missing *k*-space data, requiring significantly greater computation than TSENSE. In the originally described GRAPPA, both the calculations of weights and the reconstruction take place in *k*-space. However, as the convolution operation in the *k*-space domain is equivalent to multiplication in the image-domain, computational demands of the GRAPPA algorithm can be greatly reduced if a hybrid *k*-space and image-domain reconstruction is used (7–9). Additionally, with this hybrid approach it is possible to remove dependency of reconstruction speed on kernel size and acceleration rate, which improves the applicability of the algorithm to real-time interventional applications.

In TGRAPPA reconstruction, the final image is obtained by combining per coil images using root sum of squares (RSS) in order to avoid dephasing. In this work, we replaced the RSS combining with adaptive B_1-weighted combining (10), which further increases the computational speed over previous approaches (7) and improves signal-to-noise ratio (SNR) characteristics. In this approach, *k*-space domain GRAPPA weights for each coil are calculated and transformed into the image domain. After the transformation, these per-coil image domain weights are linearly combined into a composite set of unmixing coefficients using an estimated sensitivity map. These precalculated weights are then applied to the aliased images to reconstruct a single unaliased image (as in SENSE) with significant computational reduction over previous GRAPPA implementations. A noise correlation matrix is calculated and used for noise-weighting the GRAPPA coefficients in order to improve SNR in the reconstructed images.

In this work, we demonstrate a parallelized hybrid TGRAPPA implementation, HTGRAPPA, with improved reconstruction speed and continuously updated autocalibration for real-time, multislice, interactive MRI.

MATERIALS AND METHODS

The TGRAPPA algorithm uses data from all coils in the estimation of the missing data of a particular coil. The amount of missing data depends on the acceleration rate R, as it determines the number of phase-encoding steps to

The human subjects research protocol (ClinicalTrials.gov Indentifier: NCT00720460) was approved by the NHLBI Institutional Review Board. All subjects consented to participate in writing.

[1]National Institutes of Health, National Heart, Lung, and Blood Institute, DHHS, Bethesda, Maryland.

[2]Bogazici University, Biomedical Engineering Institute, Istanbul, Turkey.

[3]Case Western Reserve University, Cleveland, Ohio.

*Correspondence to: Haris Saybasili, National Institutes of Health, Bldg. 10, Rm. B1D416, Bethesda, MD 20892-1061. E-mail: saybasilih@mail.nih.gov

Received 11 July 2008; revised 16 October 2008; accepted 14 November 2008.

DOI 10.1002/mrm.21922

Published online 7 April 2009 in Wiley InterScience (www.interscience.wiley.com).

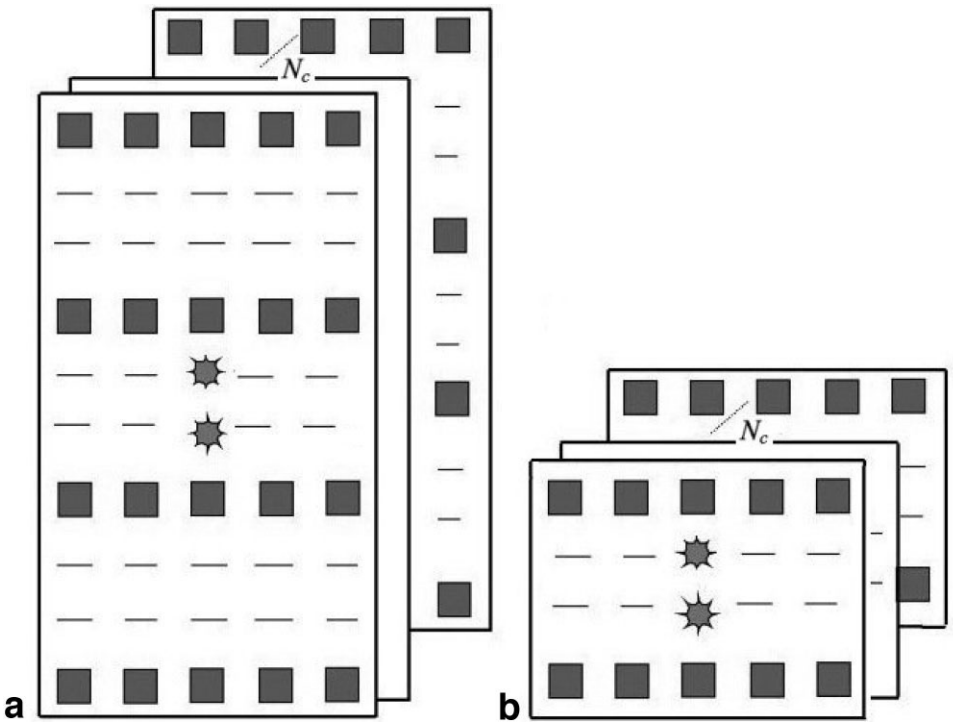

FIG. 1. Squares represent source (acquired) samples, and stars represent the target points (missing samples to be estimated). **a:** $R = 3$, block size [4 × 5] (4 lines, 5 source points per line, and 2 target points). **b:** $R = 3$, block size [2 × 5] (2 lines, 5 source points per line, and 2 target points).

skip during the acquisition. Let N_c denote the number of receiver coils. After acquisition of k-space data from each coil (c, k_x, k_y), $R - 1$ samples per measured sample must be synthesized in order to fill out the acquisition matrix. The estimates of these missing samples for a given coil are calculated using blocks of $[Y \times X]$ neighboring acquired samples from all the receiver coils. The size of the block is an important parameter that has impact on the image quality and reconstruction speed. Figure 1 represents examples of [4 × 5] and [2 × 5] blocks at one k-space location with the missing k-space samples for $R = 3$.

$R - 1$ unknown points of a particular block B of a particular coil c' are estimated by least squares solution as a linear combination of the source samples of all the receiver coils in the block B. Thus, each target point in block B requires $(X \times Y \times N_c)$ GRAPPA weighting coefficients. Once these weights are computed, they are used for estimating the target points in each block.

$S(c,i,j)$ denotes the $(i, j)^{\text{th}}$ element of the acquired data in the corresponding block of data from coil c (i.e., the square samples in Fig. 1). Let $W_{c'p}(c,i,j)$ be the set of GRAPPA weight coefficients that is used to weight the samples $S(c,i,j)$ in order to synthesize $T_{c'p}$, the p^{th} missing sample of the target block c' (i.e., the stars in Fig. 1). This relation can be expressed as:

$$\sum_{c=1}^{N_c}\sum_{i=1}^{X}\sum_{j=1}^{Y} S(c,i,j)\,W_{c'p}(c,i,j) = T_{c'p}. \qquad [1]$$

Equation 1 can also be represented in matrix form, $\mathbf{SW} = \mathbf{T}$, where $\mathbf{S}$ is the matrix containing the data acquired by all coils, $\mathbf{W}$ is the weight-set coefficients matrix for all coils, and $\mathbf{T}$ is the matrix of the missing sample estimates. If $\mathbf{S}$ and $\mathbf{T}$ are known, $\mathbf{W}$ can be calculated by $\mathbf{W} = (\mathbf{S}^H\mathbf{S})^{-1}\mathbf{S}^H\mathbf{T}$, with $\mathbf{S}^H$ being the complex conjugate transpose of $\mathbf{S}$. Therefore, weighting coefficients $W_{c'p}(c,i,j)$ are calculated after filling the matrices $\mathbf{S}$ and $\mathbf{T}$ with autocalibration signal (ACS) data $S^{ACS}(c,i,j)$ using least squares estimation of the overdetermined system formed by considering all the blocks of the fully sampled autocalibration data.

As the number of sample points (block size) from the autocalibration data increases, the accuracy of the weight-sets also increases. If the region of support for which the weights are estimated is not enough, artifact suppression may be insufficient. The fully sampled ACS data used in Eq. [1] is obtained by sliding window of R frames as described in Ref. (5).

Even though the weight-set calculation is computationally demanding, it is not required to perform it for each acquired frame. Once it is calculated, it can be used to fill the missing points in k-space for multiple frames. However, it is desired to update the weights as often as possible so that they adapt to the changes in the coil sensitivity profiles. If the image encoding process changes (e.g., due to a change of scan plane or movement of the receiver coil array), a new weight-set is required.

Application of the weights to estimate the missing k-space samples (using Eq. [1]) is computationally less demanding than calculating the weight-sets. In particular, for real-time reconstruction this calculation must be very rapid. The calculation of GRAPPA coefficients in the conventional manner (2) is computationally more demanding than corresponding image-based parallel imaging methods (e.g., SENSE), due to the fact that the GRAPPA algorithm is based on a convolution type process. Also, the requirement to reconstruct each individual coil further increases both computational and data management demands. In (T)GRAPPA, the size of the acquisition matrix; the number of receiver coils N_c; the acceleration rate, R; and the block

size [$Y \times X$] are the parameters that define the reconstruction speed.

TGRAPPA Reconstruction in the Image Domain

The convolution-type operation expressed in Eq. [1] is performed in the *k*-space domain. By transforming the weights into the image domain, the convolution-type operation can be replaced by a direct multiplication, reducing the dependency of the reconstruction speed to just the acquisition matrix size and the number of receiver coils N_c. Since the TGRAPPA weights are calculated in the *k*-space, but the reconstruction is realized in the image-domain, this method is called HTGRAPPA (hybrid TGRAPPA) reconstruction.

The transformation starts by combining the $p = [1 : R\text{-}1]$ weight-sets $W_{c'p}(c, \cdot, \cdot)$, relating the acquired data in coil c to the missing data from coil c'. This combination yields N_c *k*-space convolution kernels $W_{c'c}(k_x,k_y)$ for each coil, c'. In the next step the kernel is zero-filled to the same matrix size as the actual images and inverse Fourier transform is applied to the *k*-space kernels for transforming them into the image-domain.

$$w_{c'c}^{img} = F^{-1}(W_{c'c}) \quad [2]$$

Once N_c image-domain kernels are obtained for each coil, these kernels must be combined into one composite unmixing coefficient kernel per coil. The composite unmixing coefficients u_c^{comp} of coil c is obtained by pixel-by-pixel multiplication of each $w_{c'c}^{img}$ by complex conjugate of B_1-map estimates $b1_{c'}$, calculated from the temporal average of data from each coil and summing over coils. This relation is expressed as:

$$u_c^{comp} = \sum_{c'=1}^{N_c} w_{c'c}^{img} \cdot b1_{c'}^{*} \quad [3]$$

where $\cdot$ denotes pixel-by-pixel multiplication operator.

To perform image reconstruction in the image domain, the undersampled *k*-space data, $D_c(k_x,k_y)$ from each coil, $c = [1{:}N_c]$, is transformed into the image-domain, yielding N_c aliased images $I_c^{aliases}$:

$$I_c^{aliased} = F^{-1}(D_c) \quad [4]$$

These aliased images are combined into the final estimated image by pixel-by-pixel multiplication with composite unmixing coefficients:

$$I^{final} = \sum_{c=1}^{N_c} I_c^{aliased} \cdot u_c^{comp} \quad [5]$$

which is a phased array combiner as in SENSE unmixing. Equation [5] implies that the reconstruction speed for our implementation does not depend on either R or the block size. Furthermore, computation is greatly reduced as the convolution operation is replaced by a pixel-by-pixel multiplication, and multicoil root sum-of-squares magnitude operation is no longer required for final image formation.

Noise-Weighted Unmixing Coefficients

Optimal noise-weighted combining may be used to maximize the SNR of the phased array combined image (11). Let $\mathbf{R}_n$ be the noise correlation matrix defined as follows (12):

$$R_{ij} = \frac{1}{N}\sum_{k=1}^{N} n_i(k) n_j^{*}(k) \quad [6]$$

where R_{ij} represents a component of the matrix $\mathbf{R}_n$, n_i(k) represents the k-th sample of noise data for coil *i*. The optimum noise-weighted unmixing coefficients are computed using $\mathbf{b1}\mathbf{R}_n{}^{-1}$. Noise weighting improves the SNR of the reconstructed images with no additional computation required in image reconstruction, and only minor additional computation in weight-set calculation.

Implementation

Weight-set calculation (both *k*-space and image domain) and image reconstruction algorithms are implemented in C/C++ for high performance. Weight-set calculation and image reconstruction were computed (asynchronously) in parallel using general-purpose 8 dual-core AMD Opteron 8220 processor (2.8 GHz) on Linux-2.6.16.46-0.12-smp. Pthreads (http://pasc.org) library is used for the parallelization of the reconstruction algorithm, and OpenMP (http://openmp.org) library is used for the parallelization of the weight-set calculation. The code compiled on GCC (http://gcc.gnu.org) v. 4.2.2 for OpenMP support. For displaying the reconstructed images, the OpenGL library (http://opengl.org) is used. Matrix inversions, required for the least-squares calculation of the weight-set coefficients from the autocalibration data, were calculated using the ATLAS library (http://math-atlas.sourceforge.net). The FFTW library (http://fftw.org) was used for fast Fourier transformations.

Weight-set calculation is the most time-consuming step of the whole process. Thus, this calculation must be separated from the reconstruction: it is performed asynchronously to the image reconstruction in order not to slow reconstruction speed. During the reconstruction the most current weights are used. The reconstruction process uses N_c threads created by Pthreads library. Reconstruction threads are created once at the beginning and the same threads are used during the whole execution. That way, continuous context switching, which means decreased performance, is avoided. Another thread recalculates the weights using the most current autocalibration data permitting the reconstruction to track the changes in the coil sensitivities. At initialization, and when scan-plane is changed, new weight-sets must be calculated as quickly as possible in order to produce useful images as soon as possible. At that point, multithreaded weight-set calculation process uses all system CPUs for faster calculation. The view-sharing method (13) is used until the first weight-set calculation is over. The number of CPUs used for weight-set calculation may be decreased dynamically following weight-set initialization to a minimum number of CPUs, thereby devoting more CPU resources to image reconstruction. Dynamic allocation of the thread number

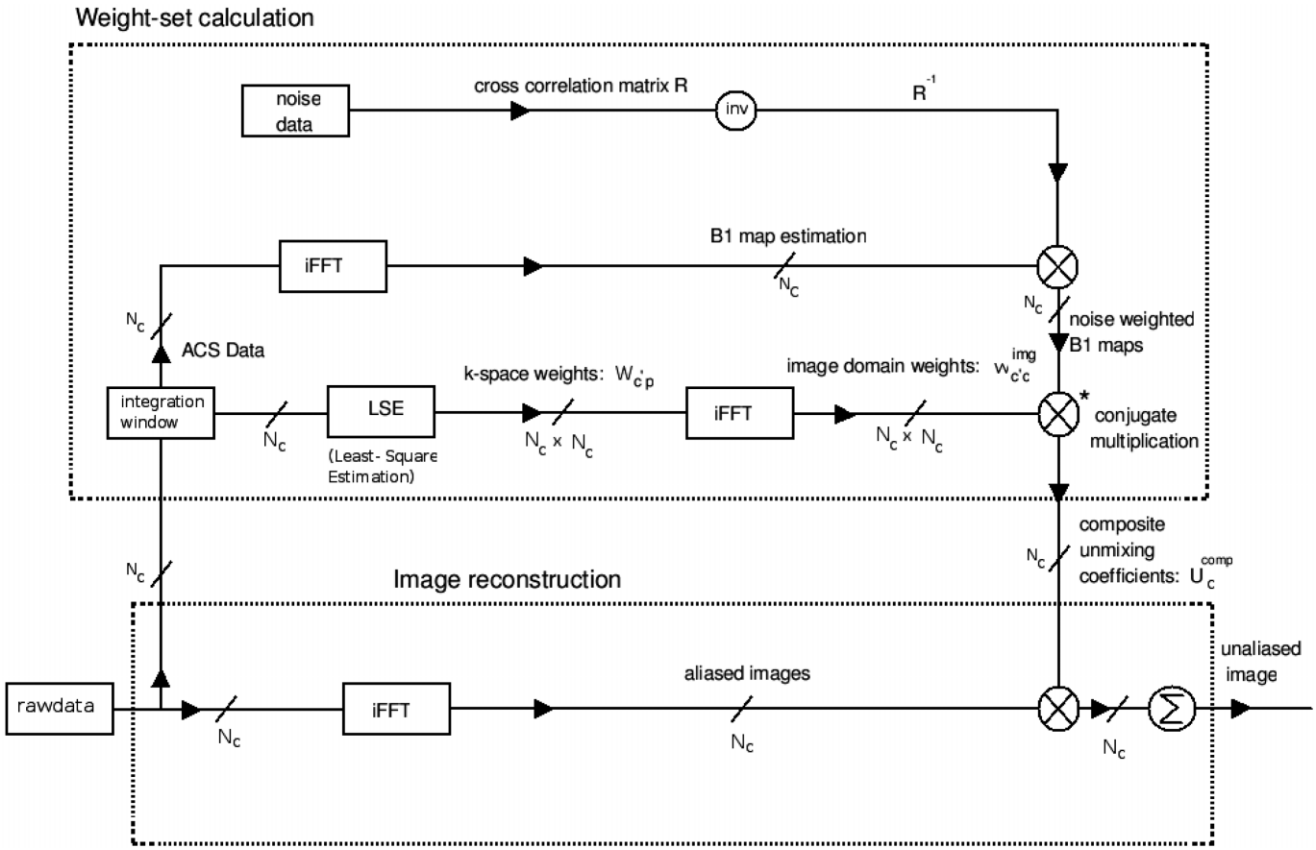

FIG. 2. Block diagram of the reconstruction algorithm is given. The upper dotted rectangle represents the weight-set calculation process, and the lower dotted rectangle represents the reconstruction process. These processes are executed asynchronously.

is realized by OpenMP library. The use of OpenMP improves the scalability of our reconstruction program and makes it possible to deploy the same program on different multicore architectures.

Our implementation works with real-time multislice acquisitions and can be controlled interactively: acceleration rate R, the number of CPUs to use during the weight-set calculation, block size, and reference data size to use can be changed dynamically to suit different imaging requirements. When the slice plane has changed, view-sharing is used until the weight-sets are ready to deploy. The block diagram of the reconstruction scheme is presented in Fig. 2.

RESULTS

Real-time cardiac images from healthy individuals were acquired using a Siemens Magnetom Avanto 1.5T and short, wide-bore Siemens Magnetom Espree 1.5T (Siemens Medical Solutions, Erlangen, Germany). The following parameters were used during the experiments: TR = 3.06 ms, flip angle = 45°, bandwidth = 800 Hz/pixel, acquisition matrix = 192 × 108. First, MR signals were collected using 18 receiver coils; 6 elements on the chest and 12 under the spine. A balanced steady state free precession (SSFP) (14) sequence was used with the S^{5}FP (15) method for real-time suppression of fat and flow artifacts during this experiment. Second, data were acquired using a 32-element array (Invivo, Birmingham, MI), 16 elements on the chest, 16 elements under the spine. Final data with an acquisition matrix of 192 × 108 was acquired using 30 receiver coils. The HTGRAPPA imaging technique was employed to improve temporal resolution with acceleration factors up to rate $R = 4$.

Performance

Our hybrid TGRAPPA implementation's performance results for the weight-set calculation on the first dataset (acquisition matrix of 192 × 108, N_c=18) are given in Table 1 for acceleration rates of $R = 2, 3, 4$, using various block sizes [2 × 3], [2 × 5], [2 × 7], [4 × 3], and [4 × 5]. In all, 48 autocalibration lines were used for weight-set calculations. HTGRAPPA reconstruction performance results are independent of acceleration factor R and block sizes, as

Table 1
Weight-set Calculation Times (in sec) for HTGRAPPA

Rate	[2 × 3]	[2 × 5]	[2 × 7]	[4 × 3]	[4 × 5]
2	0.495	0.702	0.969	0.726	1.320
3	0.510	0.712	0.973	0.779	1.339
4	0.530	0.751	0.989	0.795	1.428

Different *acceleration (R)* values and block sizes of [2 × 3], [2 × 5], [2 ×7], [4 × 3], and [4 × 5] when 16 CPUs are used (18 acquisition coils, acquisition matrix of 192 × 108, 48 ACS lines).

Table 2
Reconstruction Times (in sec) for Conventional TGRAPPA and for HTGRAPPA

Rate	[2 × 3] TGRAPPA	[2 × 3] HTGRAPPA	[2 × 5] TGRAPPA	[2 × 5] HTGRAPPA	[2 × 7] TGRAPPA	[2 × 7] HTGRAPPA	[4 × 3] TGRAPPA	[4 × 3] HTGRAPPA	[4 × 5] TGRAPPA	[4 × 5] HTGRAPPA
2	0.360	0.008	0.520	0.008	0.730	0.008	0.630	0.008	1.140	0.008
3	0.430	0.008	0.660	0.008	0.880	0.008	0.790	0.008	1.310	0.008
4	0.480	0.008	0.750	0.008	1.060	0.008	0.880	0.008	1.480	0.008

Different *acceleration (R)* values and block sizes of [2 × 3], [2 × 5], [2 ×7], [4 × 3], and [4 × 5] (18 acquisition coils, acquisition matrix of 192 × 108).

our algorithm removes the acceleration rate and block size dependency of the image reconstruction step. The HTGRAPPA weight-set calculation results given in Table 1 include the calculation of the *k*-space GRAPPA weights, the transformation of the *k*-space weight-sets into image-domain, and the calculation of the noise-weighted composite unmixing coefficients. If noise weighting is disabled, the calculations are 0.11 sec faster than the values indicated in Table 1.

The images from the first dataset were also reconstructed using a conventional TGRAPPA algorithm with asynchronous weight-set calculation, and compared to our hybrid algorithm. Table 2 represents side-by-side reconstruction performance comparison (in seconds) of TGRAPPA with HTGRAPPA.

Images

Figure 3 shows real-time reconstructed images from our first dataset (N_c = 18, acquisition matrix of 192 × 108) for 2 interleaved slices (long and short axis views) using the HTGRAPPA method at acceleration rates 2, 3, and 4. Parallel imaging artifacts are not apparent in each case. Furthermore, long axis images, which were intentionally prescribed with reduced FOV to cause prefolding, are reconstructed robustly demonstrating a benefit of GRAPPA over SENSE. When smaller blocks are used (e.g., [2 × 3]), artifact suppression is slightly degraded. However, the slight difference between small and large block sizes that is apparent during dynamic imaging is less apparent in still images, and therefore are not shown.

Figure 4 compares, from left to right, images reconstructed from our second dataset (N_c = 32, acquisition matrix of 192 × 108, R = 4) using TSENSE, TGRAPPA (non-real-time) and HTGRAPPA (real-time) algorithms. Both large and small FOV images are given for comparing the prefolding artifact robustness of HTGRAPPA with TGRAPPA and TSENSE algorithms. Upper images represent larger FOV (380 mm), and lower images represent images with smaller FOV (320 mm).

Our reconstruction system adapts dynamically to the slice orientation changes during the scan by switching to view-sharing for a few imaging frames. Figure 5 shows a series of images reconstructed in real time from our third dataset (N_c = 30, acquisition matrix of 192 × 108, R = 4) with HTGRAPPA, where slice orientation is changed during scan-time. The behavior of our reconstruction scheme after the scan plan change is as follows: our reconstruction system switches to view-sharing while the new weight-set calculation is in progress, and switches back to HTGRAPPA when the new weight-sets are ready to use.

Figure 6 shows a series of rate 4 images of full heart cycle (16 images), with matrix size of 192 × 108 and $\frac{3}{4}$ partial phase Fourier acquisition with TR = 3.06 ms. Each reconstruction is performed after the acquisition of only 20 lines, resulting in a frame rate of ≈16/sec. The images show that HTGRAPPA can be used to image full cardiac cycle without aliasing artifacts or temporal blurring.

DISCUSSION

With image domain hybrid TGRAPPA reconstruction, image-domain reconstruction speed is unaffected by increased acceleration rates and increased block sizes. Thus, larger GRAPPA kernels may be employed for superior artifact suppression and still achieve real-time reconstruction. Reconstruction speed of 0.008 sec per frame (125 frames/sec) is demonstrated on current computer hardware with 18 channel data and acquisition matrix of 192 × 108. Thus, the reconstruction system is easily capable of handling the rate 4 data acquisition speed of 16 frames/sec. The benefit of hybrid TGRAPPA image domain reconstruction comes at the cost of an increase in the weight calculation: 1) transform *k*-space domain GRAPPA weights to the image domain, 2) calculate noise-weighted B_1 estimates, 3) combine image domain weights using B_1 estimates. This added computation depends on the number of receiver coils and image resolution. With our imaging parameters, the increase in the time for weight-set calculation was measured to be 0.42 sec on data acquired with 18 receiver coils, and 0.76 sec on data acquired with 30 receiver coils. We considered this to be reasonable for our application given the benefit: although transforming *k*-space weights to the image domain, and forming the composite unmixing coefficients add computation to the algorithm, a dramatic speed-up in overall reconstruction time is achieved. More than 120 frames/sec is achieved when reconstructing rate 4 dataset acquired with 18 receiver coils, and more than 75 frames/sec is achieved during the reconstruction of rate 4 dataset acquired with 30 receiver coils.

Table 1 represents the performance values for HTGRAPPA reconstruction scheme for the weight-set calculation processes on our first dataset, where all threads for weight-set calculation and reconstruction are sharing the system CPUs. When reconstruction threads are disabled the performance of weight-set calculation increased by 10%. Disabling weight-set calculation threads increased

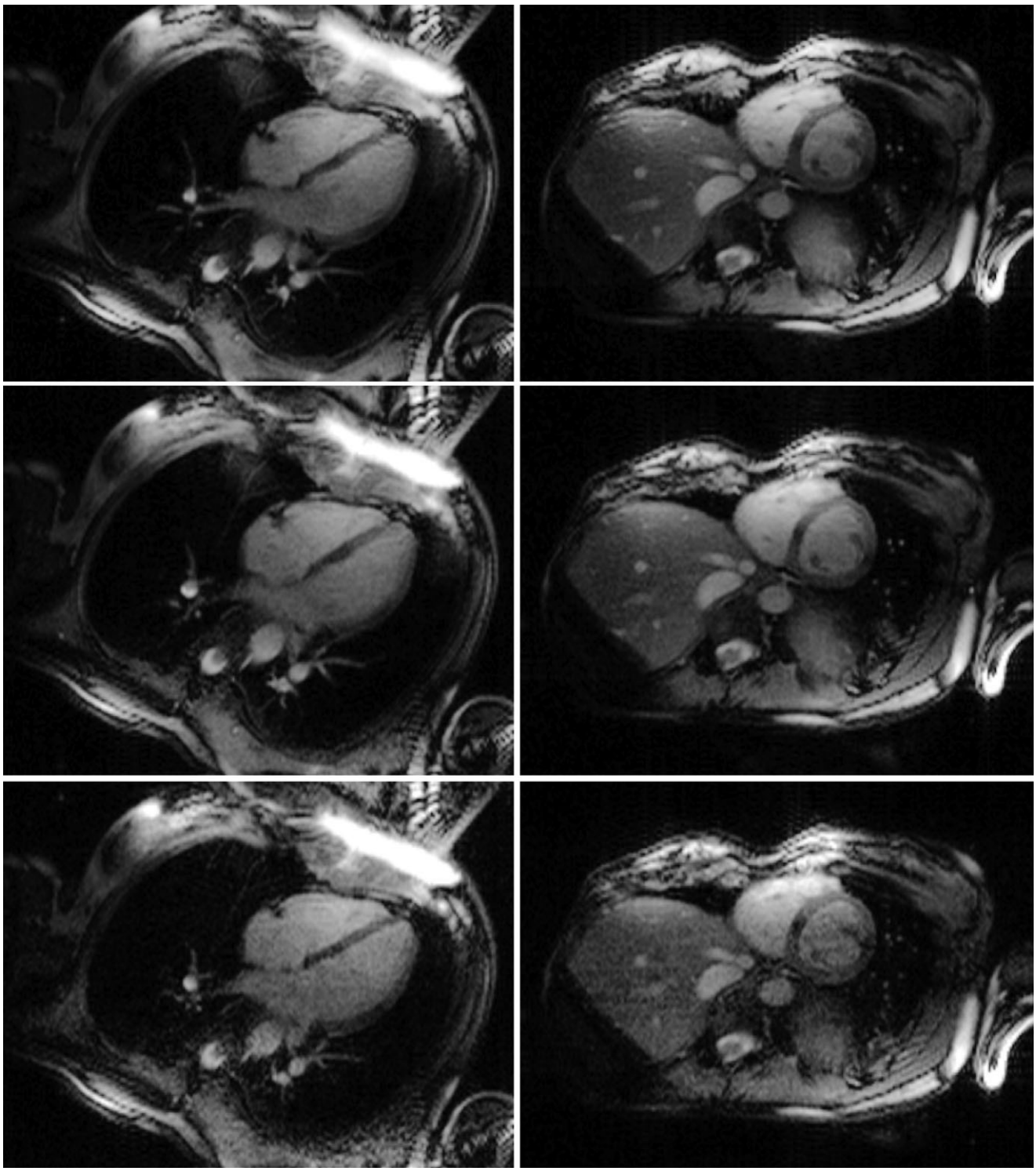

FIG. 3. Real-time images reconstructed with HTGRAPPA in 0.016 sec with acquisition of two interleaved slices (18 acquisition coils, acquisition matrix of 192 × 108). From left to right: long axis and short axis heart images. From top to bottom, $R = 2$, $R = 3$ and $R = 4$. Darkening in the myocardium from saturation can be seen where the images intersect, since they are acquired serially.

the reconstruction performance by 40%. However, the reconstruction is much faster than the acquisition rate and the reconstruction threads are idle most of the time, leaving much of the system resources to the threads for weight-set calculation.

Table 2 compares conventional TGRAPPA to HTGRAPPA in terms of reconstruction performance on our first dataset. Even with acceleration rate 2 and with a relatively small block size of [2 × 3], TGRAPPA is 45 times slower than HTGRAPPA. As the rate increases, and the

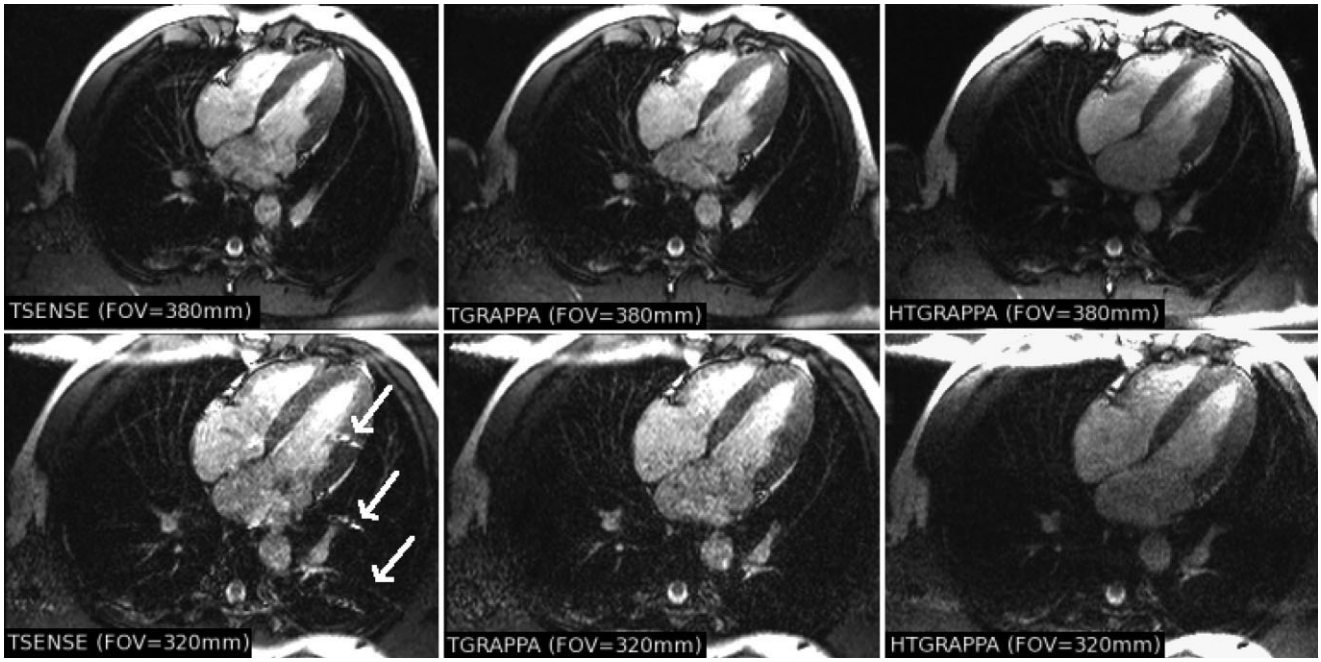

FIG. 4. HTGRAPPA, TSENSE, and TGRAPPA compared. From left to right: TSENSE, TGRAPPA, HTGRAPPA reconstructed images from data acquired with 32 acquisition coils with acquisition matrix of 192 × 108. Upper images represent large FOV (380), and lower images represent small FOV (320). Note that artifacts from prefolding are seen with TSENSE, but not with TGRAPPA and HTGRAPPA.

bigger blocks are deployed, the standard implementation of TGRAPPA becomes too slow for use in real-time imaging. At rate 4, HTGRAPPA is nearly 180× faster than conventional TGRAPPA reconstruction, since its performance does not depend on acceleration rate or GRAPPA block size.

Figure 4 represents large (upper images) FOV (380 mm) and small (lower images) FOV (320 mm) images reconstructed with TSENSE, TGRAPPA (non-real-time) and HTGRAPPA (in real-time) algorithms (from left to right). Image quality of HTGRAPPA is on a par with TGRAPPA and TSENSE. Additionally, when using a small FOV, HTGRAPPA images do not exhibit artifacts from prefolding as is the case for TSENSE. HTGRAPPA allows us to perform real-time imaging using 30 coils with an acceleration rate of 4 without sacrificing image quality.

In interventional applications, scan plane changes occur often to keep track of invasive device positions. However, parallel imaging algorithms require some time for adapting to these changes, and during this transition time aliasing cannot be removed from the images. In order to overcome this issue, our system automatically switches to view-sharing when a change in slice orientation is detected. Weight-sets for the new scan plan are calculated in the background while the images are reconstructed using view-sharing. This way, images with reasonable quality are obtained when waiting for the new weight-set. When the weight-sets are calculated, our system switches back to the HTGRAPPA algorithm, and new weight-sets are deployed to continue parallel imaging. The behavior of our system to a dynamic slice plane change is shown in Fig. 5 as a series of images reconstructed from data acquired with acceleration rate of 4, 30 receiver coils, and an acquisition matrix of 192 × 108. At frame 3, scan plane is dynamically rotated by 90° during scan-time. Our system reacts to this change by switching to view-shared reconstruction, while the new weights are being calculated in the background. At frame 21, reconstruction switches back to HTGRAPPA when the new weight-sets became available. This approach makes it possible to continue parallel imaging without interruption when scan plane change occurs.

Figure 6 shows a cardiac cine image series across a single heart cycle acquired nontriggered with free-breathing using rate 4 parallel imaging, with partial phase Fourier acquisition (20 *k*-space lines are actually acquired per frame), with TR = 3.06 ms. This TR value implies 3.06 × 20 = 61.20 ms acquisition time per cine image (temporal resolution); hence, 16.34 frames/sec is achieved in real time using these parameters. The average heart-beat of the volunteer was ≈60 bps for this study. Our results show that a fixed set of unmixing coefficients may be used across the cardiac cycle without aliasing artifacts or temporal blurring, as shown previously (4,5).

On our current computer hardware the weight-set calculation performance dropped disproportionally when the number of coils exceeded 31. Several tests were performed using rate 4 datasets with acquisition matrix of 192 × 108, block sizes of [4 × 5] and [2 × 5] and 48 ACS lines for the weight-set calculation. The cause of the problem was revealed to be greatly increased "*L2 cache misses*" during the execution. Given the performance shown in Fig. 7, it is evident that our current computer hardware is not capable of handling more than 31 coil data in real time with reasonable weight-set updating. However, this limitation did not appreciably reduce image quality at acceleration rate 4, and thus the system performance is sufficient for our applications. Advances in CPU technology with larger cache

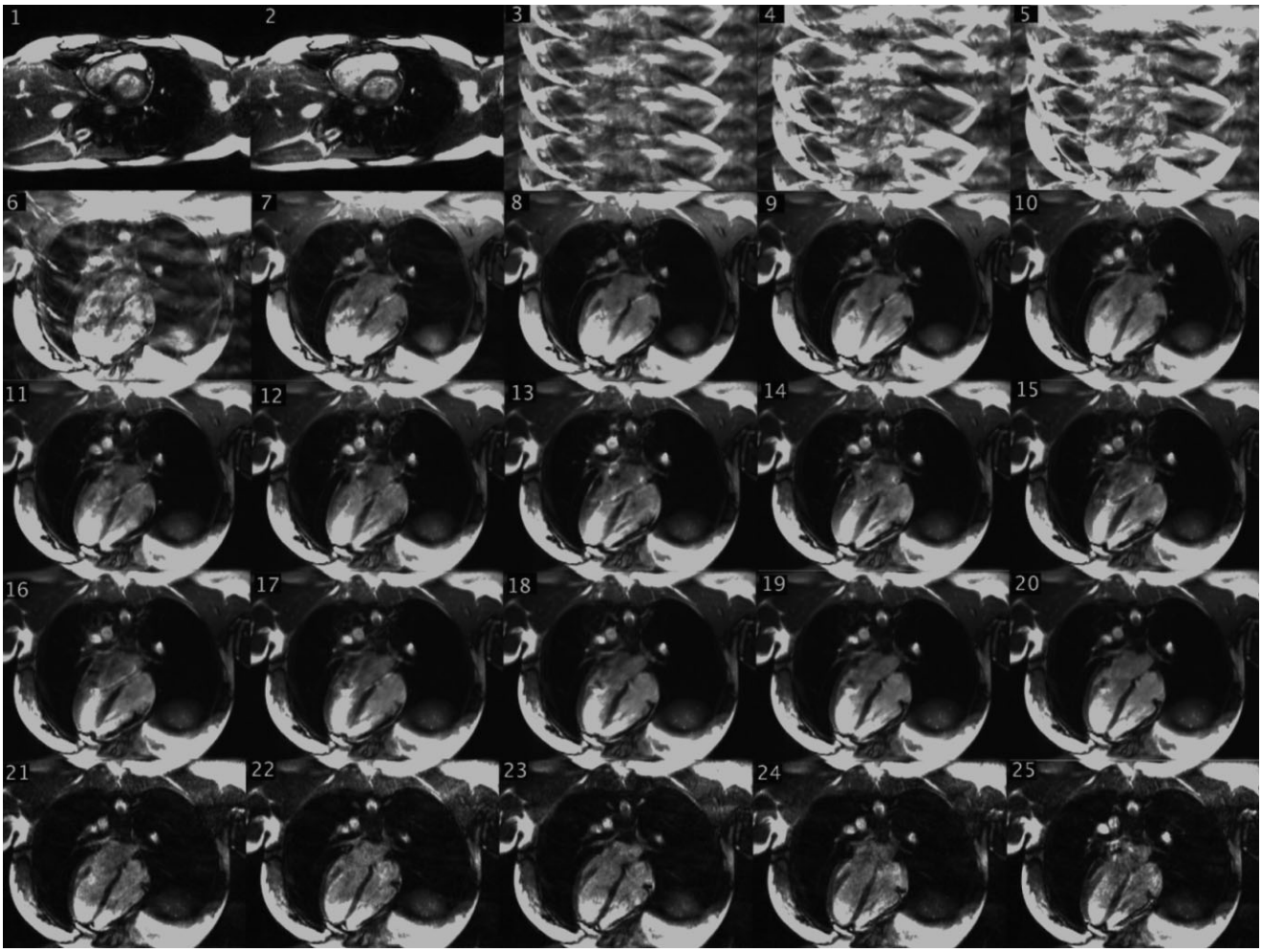

FIG. 5. Series of images (reconstructed from data acquired with $R = 4$, $N_c = 30$, acquisition matrix of 192×108) showing the response of our reconstruction system to dynamic scan plane change. At frame 3 the scan plan was changed perpendicularly. Our system switched to view-sharing reconstruction while the new weight-sets were being calculated. At frame 21 reconstruction switched back to HTGRAPPA when the weight-sets were ready to use.

memories will improve the performance of HTGRAPPA with more receiver coils.

A significant benefit of TGRAPPA is the tolerance to prefolding within the prescribed FOV of the autocalibration data. Furthermore, acceleration rate and block size independent reconstruction scheme provide better image quality with faster reconstruction rates. The flexible general purpose software implementation using parallel threads permits the scaling of the number of CPUs to readily tradeoff cost and speed. Our approach can also be implemented using low cost graphics cards (GPUs), as was done in Ref. (16) for the SENSE algorithm.

CONCLUSION

A parallelized hybrid TGRAPPA implementation for low latency, real-time interventional applications was developed and demonstrated. For improved reconstruction

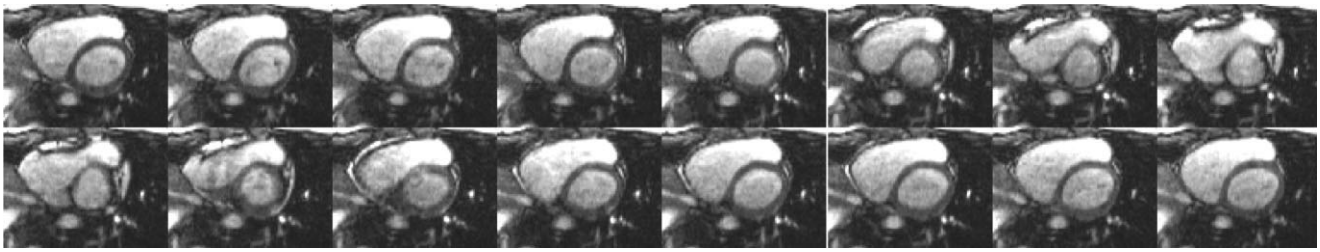

FIG. 6. Cardiac cine image series reconstructed from rate 4 dataset in real-time using HTGRAPPA.

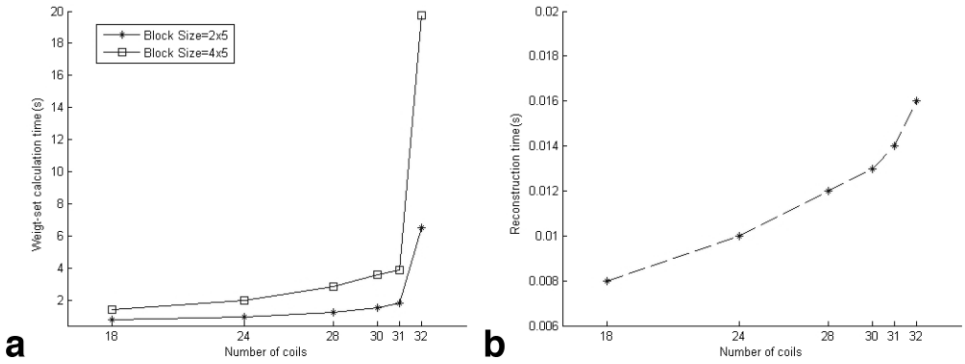

FIG. 7. Performance plots of HTGRAPPA for weight-set calculation and reconstruction with different number of acquisition coils (acquisition matrix of 192 × 108, acceleration rate of 4). **a:** Weight-set calculation performances in seconds with block sizes of [2 × 5] and [4 × 5]. **b:** Reconstruction performances in seconds.

speeds, *k*-space domain convolution type operations are converted to pixel-wise multiplication in the image domain after proper transformation. This improvement essentially provides a constant reconstruction speed for a given acceleration rate and block size. Our implementation can keep pace with acquisition rates up to $R = 8$, which was not previously possible to achieve using conventional TGRAPPA reconstruction in our tests. This approach requires additional steps to the calculation of the weight-sets; however, the weight-sets are calculated in the background, asynchronous to image reconstruction. Also, we can respond to interactive changes in slice plane by first producing view-shared images, then resuming HTGRAPPA once new weight-sets have been calculated. This method allows use of the TGRAPPA algorithm for interactive real-time MRI with very low latency. On current computer hardware, our system permits reconstructions at up to 120 frames/sec, even with high acceleration rates, and up to 31 receiver coils. With more CPUs, weight-set calculation will become even faster with increased number of threads dedicated to this calculation. As a result, better adaptation to the coil sensitivity changes can be achieved.

ACKNOWLEDGMENT

Grant sponsor: Intramural Research Program of the NIH, National Heart, Lung, and Blood Institute, and a Cooperative Research and Development Agreement between the National Heart, Lung, and Blood Institute and Siemens Medical Solutions.

REFERENCES

1. Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P. SENSE: sensitivity encoding for fast MRI. Magn Reson Med 1999;42:952–962.
2. Griswold MA, Jakob PM, Heidemann RM, Nittka M, Jellus V, Wang K, Kiefer B, Haase A. Generalized autocalibrating partially parallel acquisitions (GRAPPA). Magn Reson Med 2002;47:1202–1210.
3. Griswold MA, Kannengiesser S, Heidemann RM, Wang J. Field-of-view limitations in parallel imaging. Magn Reson Med 2004;52:1118–1126.
4. Kellman P, Epstein FH, McVeigh ER. Adaptive sensitivity encoding incorporating temporal filtering (TSENSE). Magn Reson Med 2001;45:846–852.
5. Breuer FA, Kellman P, Griswold MA, Jakob PM. Dynamic autocalibrated parallel imaging using temporal GRAPPA (TGRAPPA). Magn Reson Med 2005;53:981–985.
6. Guttman MA, Kellman P, Dick AJ, Lederman RJ, McVeigh ER. Real-time accelerated interactive MRI with adaptive TSENSE and UNFOLD. Magn Reson Med 2003;50:315–321.
7. Griswold MA. Advanced k-space techniques. In: Proc Second International Workshop on Parallel MRI. Zurich, 2004.
8. Wang JM, Zhang B. Siemens Aktiengesellschaft, Munich Germany. Fast generalized autocalibrating partially parallel acquisition image reconstruction algorithm for magnetic resonance imaging. US Patent 7279895; 2005.
9. Brau AC, Beatty PJ, Skare S, Bammer R. Comparison of reconstruction accuracy and efficiency among autocalibrating data-driven parallel imaging methods. Magn Reson Med 2008;59:382–395.
10. Walsh DO, Gmitro AF, Marcellin MW. Adaptive reconstruction of phased array MR imagery. Magn Reson Med 2000;43:682–690.
11. Roemer PB, Edelstein WA, Hayes CE, Souza SP, Mueller OM. The NMR phased array. Magn Reson Med 1990;16:192–225.
12. Kellman P, McVeigh ER. Image reconstruction in SNR units: a general method for SNR measurement. Magn Reson Med 2005;54:1439–1447 [erratum, Magn Reson Med 2007;58:211–212].
13. Fredrickson JO, Pelc NJ. Temporal resolution improvement in dynamic imaging. Magn Reson Med 1996;35:621–625.
14. Oppelt A, Graumann R, Barfuss H, Fischer H, Hartl W, Shajor A. FISP — a new fast MRI sequence. Electromedica 1986:54:15–18.
15. Derbyshire JA, Herzka DA, McVeigh ER. S5FP: spectrally selective suppression with steady state free precession. Magn Reson Med 2005;54:918–928.
16. Hansen MS, Atkinson D, Sorensen TS. Cartesian SENSE and k-t SENSE reconstruction using commodity graphics hardware. Magn Reson Med 2008;59:463–468.