Multi-Node, Multi-GPU Radial GRAPPA Reconstruction for Online, Real-Time, Low-Latency MRI

Haris Saybasili[1,2], Daniel A. Herzka[2], Kestutis Barkauskas[3], Nicole Seiberlich[3], and Mark A. Griswold[1,3]

[1]*Radiology, Case Western Reserve University, Cleveland, OH, United States,* [2]*Biomedical Engineering, Johns Hopkins University School of Medicine, Baltimore, MD, United States,* [3]*Biomedical Engineering, Case Western Reserve University, Cleveland, OH, United States*

TARGET AUDIENCE: Cardiovascular MRI, Cardiologists, Radiologists, Computer Scientists, Those interested in image reconstruction.

PURPOSE: Non-Cartesian k-space trajectories combined with pMRI provide excellent spatial and temporal resolutions during real-time MRI studies. The computational demands of reconstruction methods of these datasets, such as through-time radial generalized autocalibrating partially parallel acquisitions (GRAPPA) [1] or conjugate gradient sensitivity encoding (CG-SENSE)[2] are very high. Both CG-SENSE and radial GRAPPA have been shown to work with low latency on graphical processing units (GPUs)[2,3]. However, performing faster-than-acquisition reconstructions is a challenge for modern MRI scanners since the number of elements in the receiver array becomes large. In this work, we present a hybrid (CPU- and GPU-based), distributed (multi-node, multi-GPU) implementation for through-time radial GRAPPA that is capable of reconstructing 32 coil, subsampled radial data sets much faster than data acquisition. Results on the performance for both radial GRAPPA weights calculation and image reconstructions are presented for varying number of nodes and GPUs.

METHODS: ***Software Implementation:*** Radial GRAPPA was implemented as suggested in [3], with two major differences. First, coil-selective calibration and reconstruction capabilities were added. That way, each node (workstation) could perform calibration/reconstruction for a subset of the acquisition coils, where each GPU had its own execution thread. Second, convolution gridding was implemented on the GPU, with coil-selective reconstruction capability. One node was reserved as *master* node to distribute the raw data to each reconstruction node and to gather partial results to obtain the final reconstructed image. Up to 4 reconstruction nodes were used. The Atlas library (http://math-atlas.sourceforge.net/) was used for matrix operations during calibration, and the OpenMP library (http://openmp.org) was used for parallelization of the calibration process. Partial reconstructions on each node were performed on the GPUs, using CUDA 4.0 (http:// http://developer.nvidia.com/cuda-toolkit-40), which includes the CuFFT library used for FFT operations. POSIX threads library was used for thread synchronization on each node during multi-GPU executions. A real-time product sequence was modified to enable radial GRAPPA acquisition. An additional module was implemented on the scanner to communicate with the master node. Unix sockets and TCP/IP protocol were used for network programming. Job distribution was completely transparent to the user: the list of reconstruction nodes was provided as a command line parameter. The number of GPUs on each node was automatically detected, and the calibration/reconstruction tasks were automatically distributed to each GPU. Partial results from each node were combined by the master node, and sent back to the scanner for display. ***Hardware:*** Each reconstruction node had an Intel Xeon X5660 CPUs (6 cores, 12 threads), 48 GB of RAM and two NVIDIA Fermi M2090 GPUs. Inter-node communications were accomplished via 10 Gbit/s ethernet connections. Communication with the scanner was performed using a 1 Gbit/s ethernet connection. ***Processing:*** Raw data were transferred from the scanner to the master node for distribution to each reconstruction node. Since GRAPPA process requires all-coil data, each node received full copy of the raw data. Subsequently, each reconstruction node automatically distributed the reconstruction task to its local GPUs for processing on a dedicated thread. Calibration was performed on the CPU. Partial images from each GPU were then combined, and forwarded to master node. Once the master node acquired all partial images, the final combined image was transferred to the scanner for display. Figure 1 depicts the distributed image reconstruction process. ***MRI:*** MRI was performed on a 1.5T Espree scanner (Siemens Medical Systems, Erlangen, Germany). Acquisition parameters were: radial acquisition matrix=144x256 (calibration), 16x256 (accelerated), acceleration rate (R)=8, image matrix=128x128, TR=2.64ms, FOV=300x300 mm^2, BW=1115Hz/px, number of coils: 32. Non-gated, free-breathing imaging was performed with prior written informed consent and local IRB approval in two healthy volunteers. 26 repetitions through-time with a segment size 8 read x 2 projections were used during the calibration.

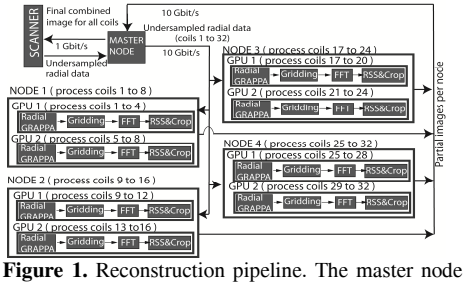

Figure 1. Reconstruction pipeline. The master node forwards raw data as it is to each node for partial processing (GRAPPA requires all coil data). Each node distributes the task to its local GPUs, and sends its partial image product to the master node after recon. All partial images are combined and sent to the scanner for display.

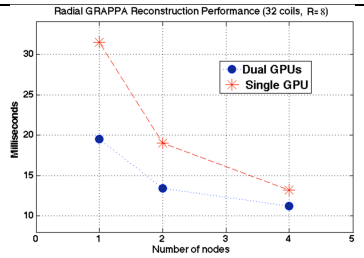

Figure 2. Reconstruction performances (in ms, including network transfers) from 32 coil, 16x256 radial data for various number of nodes with single/dual GPU configurations.

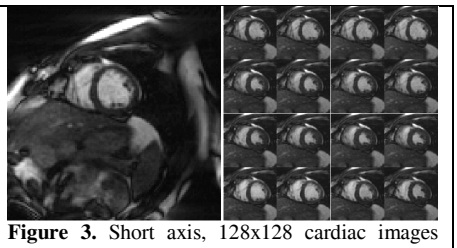

Figure 3. Short axis, 128x128 cardiac images from a healthy volunteer reconstructed from a 32-coil 16x256 radial data set using radial GRAPPA. Acquisition time: 42 ms, reconstruction time with 4 nodes (2 GPUs on each node): 11.2 ms.

RESULTS: The calibration process took 32 seconds. Figure 2 shows the execution times for image reconstructions on 1-4 nodes, using 1-2 GPUs. The timer started when data reached the master node (before distribution to reconstruction nodes), and stopped when final image was combined on the master node. Please note that data transfer times between master node and worker nodes were also included in the performance measurements (0.2-0.5 ms). Short-axis full FOV, systolic and diastolic cardiac images (R=8, 16 projections, 42 ms temporal resolution) are presented in Figure 3.

DISCUSSION: We present an automatically distributed (multi-node, multi-GPU), low-latency through-time radial GRAPPA reconstruction pipeline using multi-threaded CPU and GPU programming on multiple nodes. Our implementation provided faster-than-acquisition reconstruction performances on 32 coil highly accelerated (42 ms acquisition time, 11.2 ms reconstruction time) undersampled radial datasets. Images were reconstructed online, in real-time, and were displayed on the scanner with very low-latency. Since the reconstruction task distribution was completely transparent for the user, our implementation would easily adapt to more challenging reconstruction scenarios (e.g. larger number of acquisition coils, or higher acceleration rates), by utilizing more reconstruction nodes and/or GPUs.

REFERENCES: **[1]** Seiberlich et al. Magn Reson Med. 2010; 65:492-505. **[2]** Sorensen et al. IEEE Trans Med Imag. 2009; 28: 1974-1985. **[3]** Saybasili et al. Proc ISMRM 2012. Pg. 2554.

FUNDING: This project was funded by NIH/NIBIB R00EB011527, and NIH 1RO1HL094557.