Obtaining B_1 Distributions by Encoding in B_1 Instead of Image Space

Kalina V Jordanova[1], Dwight G Nishimura[1], and Adam B Kerr[1]
[1]*Electrical Engineering, Stanford University, Stanford, California, United States*

PURPOSE: The BEAR method[1] is a recently proposed phase-based B_1 mapping method. An important feature of BEAR is that the phase sensitivity to B_1 can be manipulated with sequence parameters. We describe a new method that calculates the B_1 distribution in a volume from multiple 1D BEAR projections, where each projection has a different B_1 sensitivity. In essence, we are replacing spatial phase encoding with B_1 phase encoding. An estimate of the B_1 distribution in each projected pixel is then calculated using a convex optimization formulation. With this method, an estimate of the B_1 distribution in a volume can be attained faster than by acquiring a 2D B_1 map. Additionally, this method may resolve rapid B_1 variations in space better than with a 2D image. We validate this method through simulations and in vivo at 3T.

METHODS: BEAR acquires doubly refocused spin-echo images with phase that is sensitive to B_1 variations using two HSn[2] adiabatic full-passage pulses. BEAR's phase sensitivity to B_1 can be tuned via the HSn refocusing pulse parameters by varying n, which determines the shape of the pulses, and δ, the ratio of the magnitude of the pulses (Fig. 1).

Multiple successive projection measurements are acquired while varying δ and n_2 to change the B_1 sensitivity. Using the projection measurements and known B_1 sensitivity, an estimate of the B_1 distribution in the volume can be calculated. A given B_1 distribution in a volume can be binned into K bins, each representing a particular B_1 range. If the fraction of spins in the volume with B_1 values in bin k is represented by c_k, then a projection measurement can be written as $p = \sum_k c_k e^{i\Phi k}$, where Φ_k represents the phase corresponding to $B_{1,k}$, given the known sensitivity, and $k = 1,2,\ldots K$. If J projections with different sensitivities are acquired, then each projection is $p_j = \sum_k c_k e^{i\Phi j,k}$ where $j = 1,2,\ldots J$. In matrix form: $\mathbf{p} = \mathbf{\Phi c}$, with $\mathbf{p}$ a Jx1 vector containing the measured projections, $\mathbf{\Phi}$ a JxK phase matrix, and $\mathbf{c}$ a Kx1 vector representing the unknown B_1 distribution. To solve for $\mathbf{c}$, a convex optimization formulation[3] with additional constraints can be set up as

$$\min_c: \quad \sum \text{hub}(\mathbf{p}-\mathbf{\Phi c}) + \lambda_1\|\mathbf{D_1 c}\|_2 + \lambda_2\|\mathbf{D_2 c}\|_2, \quad \text{subject to:} \quad \mathbf{c} \geq \mathbf{0}$$

restricting the solution to be nonnegative real, as desired. This formulation uses the Huber penalty function[3] for robustness against outliers. Smoothing constraints have been included as finite difference matrices in the B_1 direction ($\mathbf{D_1}$), and for adjacent pixels in the 1D projection ($\mathbf{D_2}$). L-curve corners were used to find $\lambda_{1,2}$ for each pixel, and the mean $\lambda_{1,2}$ were used for optimization. For smoothing, the solution for each pixel used two immediately adjacent pixels.

For both simulated and in vivo data, 11 B_1 bins, and 11 sensitivities, uniformly distributed in the range of δ = [1,0.8], were used. The number of samples and range in δ are analogous to the number of Y phase encodes and ky excursion for 2D imaging. The B_1 map used for the simulated data was of previously acquired in vivo 7T data. To compare actual and measured distributions, the Earth Mover's Distance (EMD)[4] metric is calculated for each projected pixel, which estimates the work needed to transform one distribution into another. A smaller EMD value indicates greater similarity between distributions.

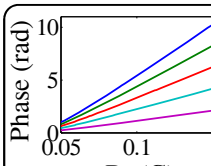

Figure 1: Simulated phase as a function of B_1 for δ= [0.8, 0.84, 0.88, 0.92, 0.96] n_1=4 and n_2=[4.255, 4.205, 4.155, 4.102, 4.051] from top to bottom.

RESULTS: Fig. 1 shows a range of B_1 sensitivities used in this study (not all sensitivities are shown). The convex optimization B_1 distribution solution closely matches the actual B_1 distribution for simulated data (Fig. 2b-d), with an average EMD of 0.0024 G·(relative counts). In vivo (Fig. 2f-h), there is more variation between the measured and actual distributions, with an average EMD of 0.0104 G·(relative counts). Simulated and in vivo scanned 2D B_1 maps are shown in Fig. 2a,e for reference. For both datasets, errors in the histograms were concentrated near the actual B_1 distributions of the 2D maps.

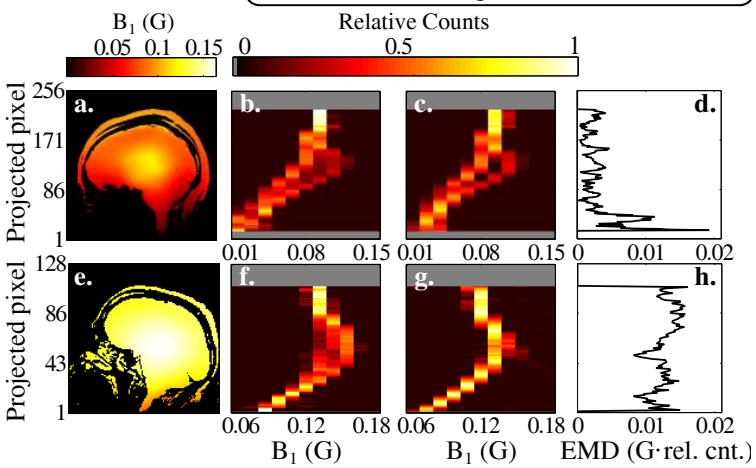

Figure 2: **a-d**: Simulated and **e-h**: in vivo data and results. **a,e**: Masked 2D B_1 maps. **b,f**: Actual and **c,g**: convex optimization results for the B_1 distribution match visually, with more variation in vivo. **d,h**: Earth Mover's Distance between (b,f) and (c,g), with mean distances of 0.0024 G·(relative counts) (simulated) and 0.0104 G·(relative counts) (in vivo).

DISCUSSION/CONCLUSION: We have demonstrated a new method that estimates the B_1 distribution in a volume without acquiring 2D image data. By encoding the signal data in B_1, we are able to use multiple measurements to calculate the B_1 distribution. This method was validated in simulation and in vivo, with most of the histogram error concentrated along the B_1 distribution. This suggests that errors are most likely due to misplaced counts in consecutive bins, with small errors in B_1. The convex optimization formulation is a general approach that would allow for the estimation of more B_1 bins than number of encodings, as well as the use of nonuniform B_1 binning. Compared to 2D B_1 mapping methods, this new method may be useful to acquire a faster estimate of the B_1 distribution, or to estimate a distribution where B_1 is varying rapidly in space, such as near a conducting guidewire.

Acknowledgment: The authors thank Bob Dougherty for access to Stanford CNI's 3T scanner.

References: [1] Jordanova *et al.*, Proceedings of ISMRM, Salt Lake City, p. 370, 2013. [2] Tannus *et al*, NMR in Biomed, 10:423-434, 1997. [3] Michael Grant *et al.*, CVX: MATLAB software. http://cvxr.com/cvx, September 2013. [4] Rubner *et al.*, Int. Journal of Comp. Vision, 40:99-121, 2000.