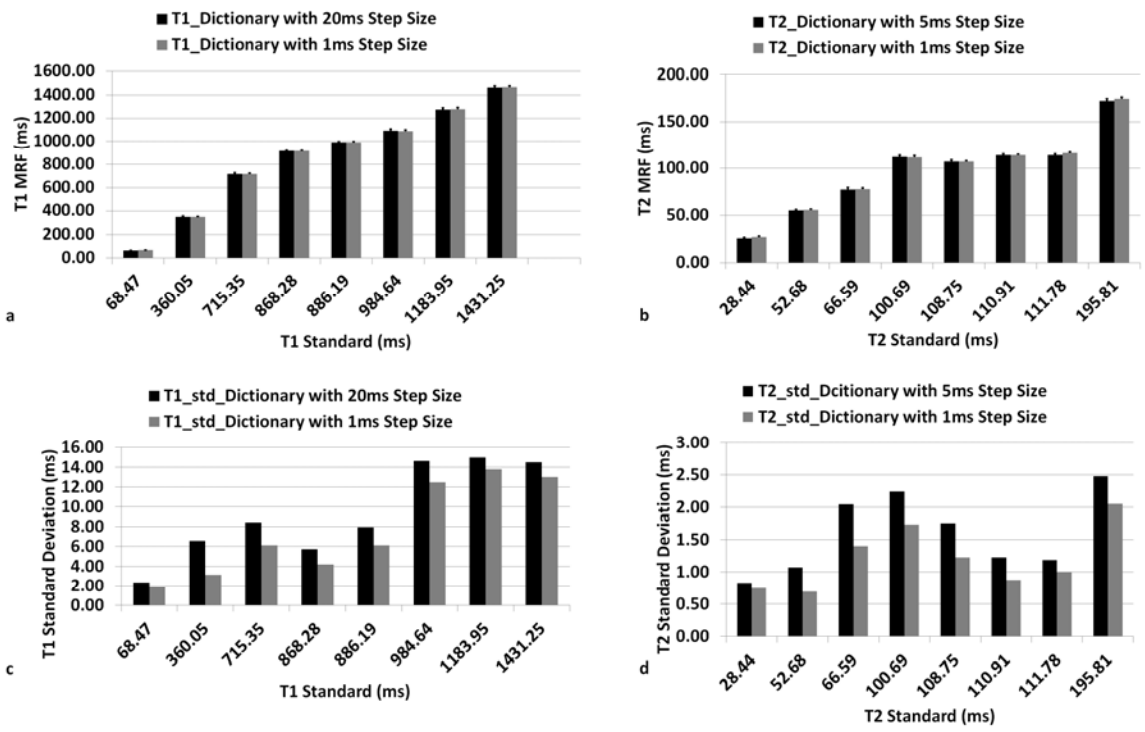
Supplemental Material

1. The Effect of Dictionary Resolution on MRF Performance

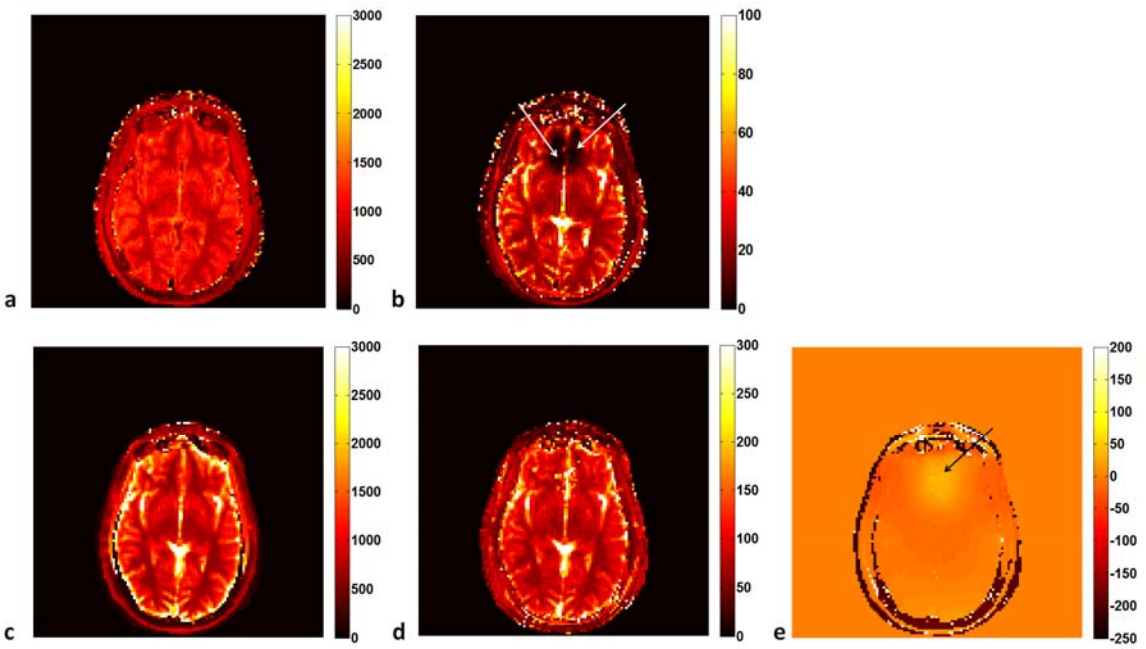
Since a larger or finer-resolved dictionary in general requires a longer computation time, the effects of the dictionary resolution on the accuracy of T1 and T2 were estimated in the phantom study as shown in Supplementary Figure 1. The MRF results shown in Figure 5 were estimated using the dictionary with 20 ms step size for T1 values from 20 to 2000 ms, and 5 ms step size for T2 values from 20 to 300 ms. Based on the values of T1 and T2 shown in Figure 5, a finer dictionary with only 1 ms step size was generated targeted on each of the phantoms. The finer resolution dictionary included the range of values within a T1 range of ±50 ms and a T2 range of ±20 ms from the original results. (Performing such a calculation over the full range of values would be computationally impractical with our current setup.) Means and standard deviations of each phantom using the dictionaries with different resolutions were compared. As can be seen below, the mean values remained unchanged. However, the standard deviation across the pixels of the phantom decreased as the dictionary step size became smaller. This can be intuitively understood by examining an example situation where a pixel contains a material that sits directly between two dictionary entries. In this case, any added noise will cause the selected dictionary entry to switch between the dictionary entries on either side of the true value. The further apart these dictionary entries are, the worse the apparent error is. Thus a finer resolved dictionary can potentially provide higher quality results at the expense of increased computation time.



Supplemental Figure 1: The effects of dictionary resolution on the accuracy of T1 and T2. **a)** T1 values retrieved from the dictionary with 20ms step size were compared with the values from the dictionary with 1ms step size. **b)** T2 values retrieved from the dictionary with 5ms step size were compared with the values from the dictionary with 1ms step size. c) The standard deviation of T1 values retrieved from the dictionary with 20 ms step size were compared with the values from the dictionary with 1 ms step size. d) The standard deviation of T2 values retrieved from the dictionary with 5 ms step size were compared with the values from the dictionary with 1 ms step size.

2. Effects of Severe Off-Resonance on MRF and DESPOT2 Acquisitions

The bSSFP based sequence is known to be sensitive to field inhomogeneities, where the resonance frequency is different from the frequency of exciting RF field. This off-resonance effects will cause signal voids at the phase evolution of π or multiples of π per TR[1]. In order to show the impact of this effect on a conventional acquisition, an *in vivo* accelerated MRF scan (using the sequence pattern as shown in Figure 1d), and the combined DESPOT1 and DESPOT2 (acquisition parameters are provided from Deoni et al[2]) was performed in a lower part of the brain, near the air-tissue interfaces in the frontal lobe region, that well known to display large off-resonances. The TR of the bSSFP sequence for DESPOT2 was changed to be the same as the TR of MRF, which is 10.8 ms). As shown in Supplementary Figure 2b, the T2 map from DESPOT2 has signal voids in the frontal lobe region, while this off-resonance effect is not shown in the MRF T2 map in Supplementary Figure 2d. The off-resonance map from MRF in Supplementary Figure 2e shows the high off-resonance values in the corresponding area as indicated by arrows.



Supplementary Figure 2: Correction of Severe off-resonance Effects from bSSFP based Sequence. (a) and (b) show the T1 and T2 maps acquired from DESPOT1 and DESPOT2 methods, respectively. Arrows indicate areas of signal voids due to the off-resonance effect. (c-e) show the T1, T2 and off-resonance maps acquired from MRF method. T2 map in (d) does not have signal voids because the off-resonance effect has been considered in the off-resonance map shown in (e) by arrows.

3. The Potential of MRF to Resolve Multiple Components from Single Voxels

Most conventional quantitative imaging methods assume that each voxel contains a pure, single material. One issue that is common to many quantitative imaging methods is how to deal with voxels with mixed content. This general class of problems is referred to as "partial volume". If there is a mixture of materials in a single voxel, one typically observes a single component with the volume-weighted average T1, T2, etc. Several investigators have also used multiexponential fitting to attempt to derive separate components of a voxel[3–6]. However, these methods are highly unstable and sensitive to noise. One of the fundamental bases for this problem is that the different signal evolutions for the different components of the voxel fundamentally "look" the same (e.g. exponential decays) and thus they are difficult to separate.

MRF provides new opportunities to analyze partial volume results because of the incoherence between different signal evolutions. In general, the goal of MRF is to design a pulse sequence that generates signal evolutions that are as independent as possible. In an ideal case, the signal evolutions from the different components of a voxel would be orthogonal, which would make them trivial to separate. However, in general there will be some correlation between the different materials, and thus some other method must be used. To this end, one could perform OMP and derive multiple components from each pixel. Alternatively, this could be formulated as a L1-minimization commonly found in the compressed sensing literature[7]. Bayesian estimation methods are also well suited[8]. Or finally, one could use a neural network based pattern matching algorithm which could output the probability of each material type[9] . However, in all of these cases, the maps for the 2nd component (and higher) are noisy and difficult to interpret, since there is only signal in those voxels that are strongly inhomogeneous.

Here we present an alternative formulation that is both easy to understand and easy to visualize. If we are working in a system where the major substances of interest are known *a priori*, one can ask "how much of each substance is in each voxel?", since the target signal

evolutions are completely known. In this case, we can model the signal from a single voxel (S_{voxel}) as a linear superposition of the known components:

$$S_{voxel} = \sum_i w_i D_i \qquad \text{[S.3.1]}$$

where D_i is the signal evolution for material i as calculated directly from Bloch equations or extracted from a dictionary, and w_i is the unknown weight for the individual components. This can be written in matrix form

$$S_{voxel} = Dw \qquad \text{[S.3.2]}$$

This can then be solved though an appropriate inverse solution to solve for the individual component weights, such as:

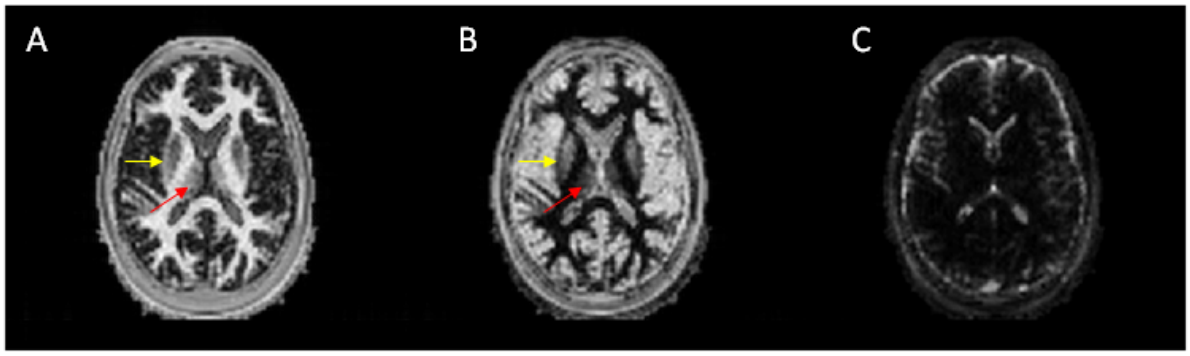
$$(D^H D)^{-1} D^H S_{voxel} = w \qquad \text{[S.3.3]}$$

where a superscript H represents the Hermetian adjoint or complex transpose operator. The weights are then normalized such that the sum of the weights has unit magnitude.

Here we have generated fractional maps in the brain assuming that the primary tissues are white matter, gray matter and CSF as derived directly from the highly undersampled MRF data set shown in Figure 3. The images were interpolated to 256x256 prior to this analysis using straightforward Fourier interpolation. At each voxel, the off-resonance value was determined using the standard MRF method. Based on this value, the individual signal evolutions were derived assuming the following values for T1 and T2: white matter (T1=660 ms, T2=70), gray matter (T1=1200 ms, T2=90 ms) and CSF (T1=5000 ms, T2=700 ms). The resulting maps derived from the direct application of equation S.3.3 are shown in Supplemental Figure 3. The maps show the expected distribution of the various tissue types throughout the brain. This is

particularly evident in the visualization of the deep nuclei in the brain. One can see that the putamen (yellow arrows) displays a reduced, but non-zero, white matter fraction and a large gray matter fraction. The thalamus (red arrows) also shows a significant fraction of gray matter. Areas of partial volume in and around the sulci also appear as expected. CSF is incorrectly visualized in both of the other maps to a certain extent. Our hypothesis is that this is due to CSF pulsation, which is not taken into account in our current model.

While probabilistic image segmentation such as this can be performed using other methods based on multiple image acquisitions, we believe that this is the first method that is able to translate raw signals directly to quantitative material fractions. In medical applications, we believe that this visualization might be useful whenever one can identify MRF signatures that are specific for certain disease states. One could generate images that directly predict the probability of finding a certain disease state within any given voxel, even when the volume of the affected tissue is less than a single voxel. This could lead to a dramatic paradigm shift in the way clinical MRI is performed.



Supplemental Figure 3: Fractional tissue maps for A) white matter B) gray matter and C) CSF. The intensity of each pixel in these maps represents the fraction of that tissue in that voxel. The putamen (yellow arrow) shows the anticipated reduced white matter fraction, with a large gray matter fraction. The thalamus (red arrow) shows a similar trend, as expected.

4. Simplified Matlab Code for Calculation of MRF Dictionaries and Template Matching

```
function dict=makeMRFdictionary(RFpulses ,TR ,T1, T2, df);
%
% This function calculates the MRF signal evolution for an IR-bSSFP-based readout
% for a single material (T1, T2, df) for the input RF pulse train and TR.
%
% This code is based on the single isochromat method popularized by Klaus
% Scheffler. This works for bSSFP as long as one can assume that both the
% flip angle and off-resonance frequency are essentially homogeneous inside
% a voxel.
%
%         INPUTS:
%                       RFpulses        Complex RF pulse train
%                                           (1 x #timepoints)
%
%                       TR              TR  (1 x #timepoints)
%
%                       T1, T2, df      T1, T2, and off-resonance frequency
%
%                                       Make sure these are all in the same
%                                       time units!
%
%         OUTPUTS:
%
%                       dict            Calculated dictionary
%
%
%   Nov. 15, 2012   Dan Ma, Mark Griswold
%
%

if nargin<5
    df=0;
end

if nargin<4
    T2=50;
end

if nargin<3
    T1=1000;
end

if nargin<2
    TR=rand(1,1000)*4+10;
end

if nargin<1
    RFpulses=10*pi./180*randn(1,1000);
end

N=length(TR);

rf=abs(RFpulses);
rph=angle(RFpulses);

dict=zeros(1,N);

 m1=[0 0 -1].';    % This assumes a perfect inversion pulse with no delay
```

```
for i=1:N

    rx=    [   1.0 0.0 0.0;        %rotation matrix for pulse
               0.0 cos(rf(i)) sin(rf(i));
               0.0 -sin(rf(i)) cos(rf(i))];

    rdzp=  [   cos(rph(i)) sin(rph(i)) 0.0; %RF phase
               -sin(rph(i)) cos(rph(i)) 0.0;
               0.0 0.0 1.0];

    rdzm=  [   cos(-rph(i)) sin(-rph(i)) 0.0;
               -sin(-rph(i)) cos(-rph(i)) 0.0;
               0.0 0.0 1.0];

    e1=exp(-TR(i)./T1);        % relaxation terms
    e2=exp(-TR(i)./T2);


    beta=df.*TR(i)*2*pi;

    rbeta= [   cos(beta./2) sin(beta./2) 0.0; % 1/2 off-resonance rotation
               -sin(beta./2) cos(beta./2) 0.0;
               0.0 0.0 1.0];



    e12=   [   e2 0.0 0.0;
               0.0 e2 0.0;
               0.0 0.0 e1];



    m1=rdzp*rx*rdzm*m1;    % do RF pulse

    m1=(e12*m1+(1-e1)*[0 0 1].');  %relax

    m1=rbeta*m1;   % Off-resonance for first 1/2 of TR

    dict(i)=m1(1)+j.*m1(2);  %Sample assuming TE=TR/2


    m1=rbeta*m1;  % Off-resonance for 2nd 1/2 of TR
end
```

```
function matchout=templatematch(dict,sig);
%
% This function calculates a template match for use in MR Fingerprinting
% applications. Here this is calculated using a simple dot-product. The
% output is the index of the most likely dictionary entry for each pixel.
%
%       INPUTS:         dict        (# entries, timepoints)
%                       sig         (timepoints, #pixels)
%
%       OUTPUTS:        matchout    (1, #pixels)
%
% Please note that this code requires the generation of an intermediate
% matrix of size (#entries, #pixels), which can be quite large. If this is
% beyond the memory capability of your computer, simply parse the data into
% smaller sets of pixels and calculate them serially.
%
%   Nov. 15, 2012   Dan Ma, Mark Griswold
%


if nargin<2
    disp('Error - 2 inputs are required!');
    break
end

% First normalize the dictionary and signal

for i=1:size(dict(:,:),2)
    dict(:,i)=dict(:,i)./norm(dict(:,i));
end

for i=1:size(sig(:,:),2)
    sig(:,i)=sig(:,i)./norm(sig(:,i));
end

% Calculate inner product

innerproduct=dict(:,:)*sig(:,:);

% Take the maximum value and return the index
[maxval,matchout]=max(abs(innerproduct));
```

5. Captions for Supplemental Movies

Supplementary Movie 1:

This movie shows the time resolved *in vivo* images acquired from fully sampled MRF scan. The movie covers the first 50 time frames. Oscillations in signal intensity appear across all of the time frames. (MPG animation: 980 KB)

Supplementary Movie 2:

This movie shows the time resolved *in vivo* images generated from an accelerated MRF scan that acquired only $1/48^{th}$ of the normally required data. The movie covers the first 50 time frames out of 1000. High intensity but incoherent undersampling errors are present in all time frames. (MPG animation: 1.5 MB)

Supplementary Movie 3:

This movie shows the motion corrupted scan. The subject started to move after 12 seconds of a 15 seconds acquisition. The movie clearly shows the motion as well as severe aliasing artifacts from the highly undersampled data. (MPG animation: 1.5 MB)

6. Supplemental Material References

1. Schmitt, P. *et al.* A Simple Geometrical Description of the TrueFISP Ideal Transient and Steady-State Signal. *Magnetic Resonance in Medicine* **55**, 177–86 (2006).

2. Deoni, S. C. L., Peters, T. M. & Rutt, B. K. High-Resolution T1 and T2 Mapping of the Brain in a Clinically Acceptable Time with DESPOT1 and DESPOT2. *Magnetic Resonance in Medicine* **53**, 237–41 (2005).

3. Deoni, S. C. L., Rutt, B. K., Arun, T., Pierpaoli, C. & Jones, D. K. Gleaning Multicomponent T1 and T2 Information From Steady-State Imaging Data. *Magnetic Resonance in Medicine* **60**, 1372–87 (2008).

4. Gareau, P. J., Rutt, B. K., Karlik, S. J. & Mitchell, J. R. Magnetization transfer and multicomponent T2 relaxation measurements with histopathologic correlation in an experimental model of MS. *Journal of magnetic resonance imaging : JMRI* **11**, 586–95 (2000).

5. Qian, Y., Williams, A. a, Chu, C. R. & Boada, F. E. Multicomponent T2* Mapping of Knee Cartilage: Technical Feasibility Ex Vivo. *Magnetic Resonance in Medicine* **64**, 1426–31 (2010).

6. Yu, H. *et al.* Combination of Complex-Based and Magnitude-Based Multiecho Water-Fat Separation for Accurate Quantification of Fat-Fraction. *Magnetic Resonance in Medicine* **66**, 199–206 (2011).

7. Goldstein, T. & Osher, S. The Split Bregman Method for L1-Regularized Problems. *SIAM Journal on Imaging Sciences* **2**, 323 (2009).

8. Bilgic, B., Goyal, V. K. & Adalsteinsson, E. Multi-Contrast Reconstruction with Bayesian Compressed Sensing. *Magnetic Resonance in Medicine* **66**, 1601–15 (2011).

9. Reddick, W. E., Glass, J. O., Cook, E. N., Elkin, T. D. & Deaton, R. J. Automated Segmentation and Classification of Multispectral Magnetic Resonance Images of Brain using Artificial Neural Networks. *IEEE transactions on medical imaging* **16**, 911–8 (1997).