```matlab
% iteratively calculate scaling function and wavelet

 % load coefficients
 str=load('db3.mat'); varnames=fieldnames(str); db=getfield(str,varnames{1});
 h = db(:)/sum(db)*sqrt(2);       % normalize (optional)

 % create matrix, and look for proper eigenvector
 H = convmtx(h,length(h)); H = H(1:2:end,:);
 [V,L] = eig(H);
 [dum,indx] = min(abs(diag(L)-1/sqrt(2)));
 v = V(:,indx);                   % the proper evec
 v = v/sum(v);                    % resolve evec gain ambiguity

 % iterate scaling eqn
 T = length(h)-1;                 % scaling fxn time support
 t = [0:T];                       % sample times (currently integers)
 phi = v;                         % scaling fxn sampled at times t
 figure(1); clf;
 subplot(3,2,1); stem(t,phi,'.');
 title('iteration 1');
 for i=0:8,
   u = 2.^i;                                          % upsampling factor
   h_up = zeros(u*length(h)-(u-1),1); h_up(1:u:end) = h;% upsampled filter
   phi = sqrt(2)*conv(phi,h_up);                       % sampled scaling fxn
   t = [0:2*u*length(h)-2*u]/u/2;                      % sample times
   if i<4,
     subplot(3,2,i+2); stem(t,phi,'.');
     title(['iteration ',num2str(i+2)]);
   end;
 end;
 [dum,indx] = max(abs(phi));
 dt = t(2)-t(1);  %dt = 1/2/u;
 %phi = phi/sqrt(norm(phi)^2*dt);       % normalize (optional)
 subplot(3,2,6); stairs(t,phi);
 title(['iteration ',num2str(i+2)]);
 orient tall;

 % calculate wavelet (via wavelet scaling eqn)
 g = flipud(h.*((-1).^[1:length(h)]).');
 psi = zeros(length(t),1);
 for n=0:length(g)-1,
   psi([find(t==n/2):find(t==n/2+T/2)]) =...
       psi([find(t==n/2):find(t==n/2+T/2)]) + sqrt(2)*phi(1:2:end)*g(n+1);
 end;

 % plot psi, phi, and CTFT magnitudes
 figure(2);
 Fmax = 6;
 N_fft = 32768;
 subplot(221);
   plot(t,phi); xlabel('sec'); title('scaling fxn: \phi(t)');
 subplot(222);
   Phi = abs(fft(phi,N_fft)*dt);
   plot([0:ceil(Fmax*dt*N_fft)-1]/N_fft/dt,...
        Phi(1:ceil(Fmax*dt*N_fft)));
   xlabel('Hz'); title('scaling fxn CTFT: |\Phi(\Omega)|');
 subplot(223);
   plot(t,psi); xlabel('sec'); title('wavelet: \psi(t)')
 subplot(224);
   Psi = abs(fft(psi,N_fft)*dt);
   plot([0:ceil(Fmax*dt*N_fft)-1]/N_fft/dt,...
        Psi(1:ceil(Fmax*dt*N_fft)));
   xlabel('Hz'); title('wavelet CTFT: |\Psi(\Omega)|')

return;
```

```
% verify orthonormality of mother scaling fxn at various shifts
for n=0:T-1,
   sum(phi([find(t==0):find(t==T-n)]).*phi([find(t==n):find(t==T)]))*dt
end;

% verify orthonormality of scaling fxn at level one
t1 = 2*t;
dt1 = t1(2)-t1(1);
phi1 = phi/sqrt(2);
for n=0:T-1,
   sum(phi1([find(t1==0):find(t1==2*T-2*n)]).*...
      phi1([find(t1==2*n):find(t1==2*T)]))*dt1
end;

% verify orthogonality of scaling fxn and wavelet
for n=0:T-1,
   sum(psi([find(t==0):find(t==T-n)]).*phi([find(t==n):find(t==T)]))*dt
end;
```