

---

# MATLAB code

## Table of Contents

Setup .....	1
Average in the frequency domain .....	1
Find the marble's frequency and filter about it .....	1
Use filter to denoise data and locate the marble .....	2

Brian de Silva 1422824

## Setup

```
clear all; close all; clc;
load Testdata

L=15;           % spatial domain
n=64;           % Fourier modes
x2=linspace(-L,L,n+1); x=x2(1:n); y=x; z=x; % Spatial grid axes
k=(2*pi/(2*L))*[0:(n/2-1) -n/2:-1]; % Scaled frequencies
[X,Y,Z]=meshgrid(x,y,z); % 3D spatial grid
[Kx,Ky,Kz]=meshgrid(k,k,k); % 3D frequency grid

ave = zeros(n,n,n);
```

## Average in the frequency domain

Take 3D DFT of the data, then average to remove noise (noise assumed to be white)

```
for j=1:20
    Uj=reshape(Undata(j,:),n,n,n);
    ave = ave + fftn(Uj);

    % % Plot noisy data (takes a long time to run)
    %     close all, isosurface(X,Y,Z,abs(Uj),0.4)
    %     axis([-20 20 -20 20 -20 20]), grid on, drawnow
    %     pause(1)
end

% % Plot the noisy data at the final time (doesn't mesh well with publish
% % feature)
% close all, isosurface(X,Y,Z,abs(Uj),0.4)
% axis([-20 20 -20 20 -20 20]), grid on, drawnow
% title('Noisy data at final time'), xlabel('x'), ylabel('y'), zlabel('z')
```

## Find the marble's frequency and filter about it

Get indices of frequency of largest magnitude (center frequency)

```
ave = ave / 20;
[M, linearInd] = max(abs(ave(:)));
[I,J,K] = ind2sub([n n n], linearInd);

% Get frequency components for the center frequency
ki = Kx(I,J,K);
kj = Ky(I,J,K);
kk = Kz(I,J,K);

% Compute filter centered at the discovered center frequency
filter = exp(-1 * ((Kx - ki).^2 + (Ky - kj).^2 + (Kz - kk).^2));
```

## Use filter to denoise data and locate the marble

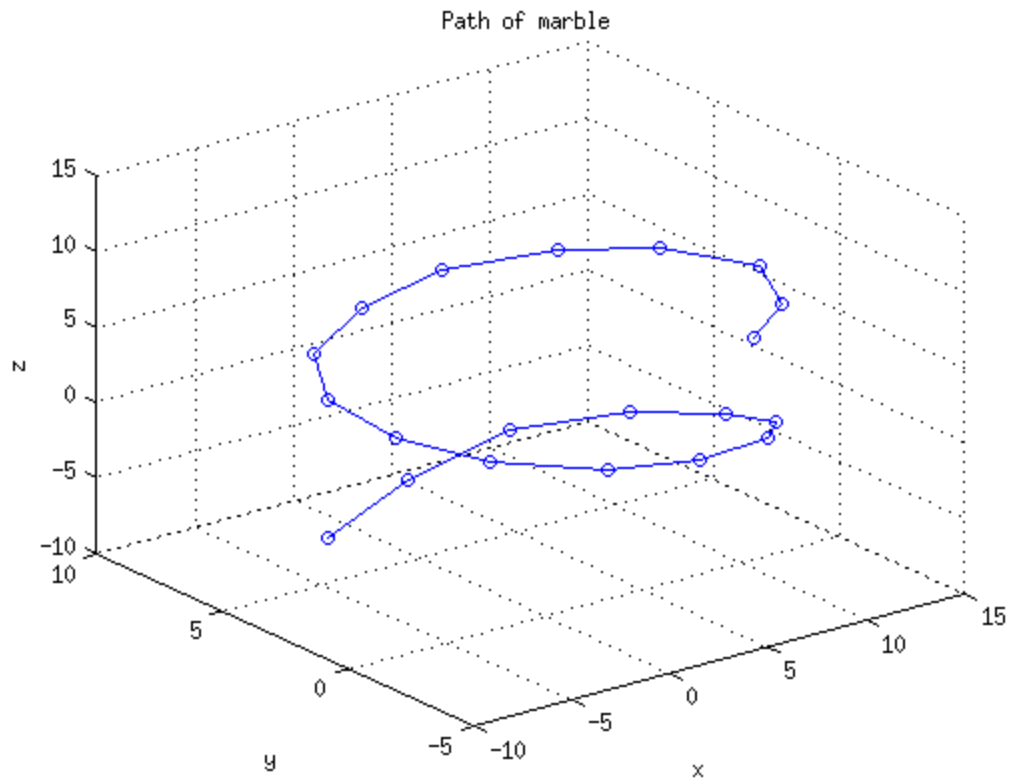
Apply filter to denoise each frequency matrix then take ifft

```
marble = zeros(3,20);
for j=1:20
    Uj = reshape(Undata(j,:),n,n,n);
    Ujtf = fftn(Uj) .* filter;
    Ujf = ifftn(Ujtf);

    % Store coordinates of marble at each (time) step
    [~, ind] = max(abs(Ujf(:)));
    [marblei, marblej, marblek] = ind2sub([n n n], ind);
    marble(1,j) = X(marblei, marblej, marblek);
    marble(2,j) = Y(marblei, marblej, marblek);
    marble(3,j) = Z(marblei, marblej, marblek);
end

% % Plot the denoised data at the final time (doesn't mesh well with publish
% % feature)
% figure(), isosurface(X,Y,Z,abs(Ujf) / max(abs(Ujf(:))),0.9)
% axis([-20 20 -20 20 -20 20]), grid on, drawnow
% title('Cleaned data at final time'), xlabel('x'), ylabel('y'), zlabel('z')

% Plot the marble's path
figure()
plot3(marble(1,:), marble(2,:), marble(3,:), 'b-o'), grid on;
title('Path of marble'), xlabel('x'), ylabel('y'), zlabel('z')
```



*Published with MATLAB® 7.14*