q6

March 12, 2018

0.1 Decript signal using fourier transform

- Each message is shuffled in the fourier domain.
- We have to take the fft, permute and listen to each n

0.1.1 Function to compute permutation for each signal and check if its correct

```
function [right_perm] = Listen_perm(audiofile)

x = audioread(audiofile);

% Plot the orignal signal and its fft
figure;
subplot(2,1,1);
plot(x);
title('Original Sound');

% Only work in fourier domain
f_y = fft(x);
subplot(2,1,2);
plot(abs(f_y));
title('Ft of original');


% We do not need the conjugate
f_y = f_y(1:end/2);

% Divide the whole into 4 parts
pause
d = size(f_y,1)/4;
f_y = reshape(f_y,d,4);

or = perms([1,2,3,4]);
    for i = [1:24]
        or(i,:)
        % Permute and reshape to original shape
        new_x = perm_ord(f_y,or(i,:),d);
```

```
        subplot(2,1,2);
        plot(new_x);
        title('New Sound');

        sound(new_x,41400);
        pause
    end

    right_perm = [2,3,1,4]
end
```

0.1.2 Funtion to print the corect order

```
function [new_x] = perm_ord(f_y,ord,d)


        new_y = [f_y(:,ord(1)), f_y(:,ord(2)), f_y(:,ord(3)), f_y(:,ord(4))];
        new_y = reshape(new_y,d*4,1);

        % add the flip version
        y = zeros(2*size(new_y,1),1);
        y(1:end/2) = new_y;
        y(end/2 + 1:end) = conj(flipud(new_y(:)));

        figure;
        subplot(2,1,1);
        plot(abs(y));
        title('Frequency domain');

        new_x = real(ifft(y));
end
```

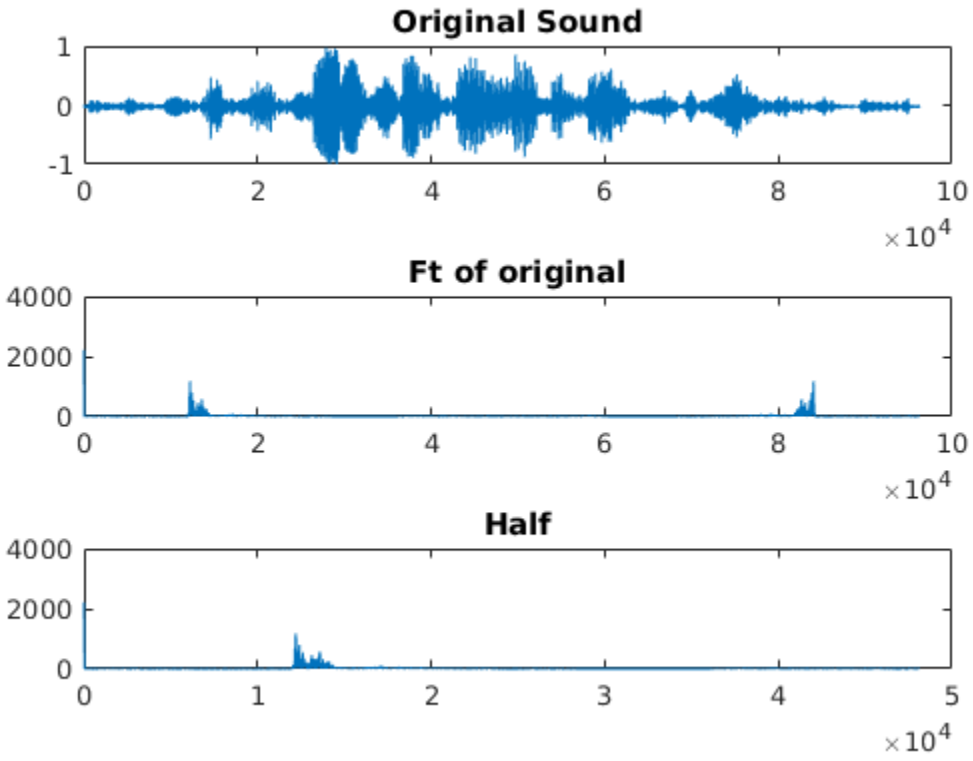
0.1.3 Message 1

=> If you are good at something never do it for free

```
In [39]: x = audioread('./message1.wav');

         figure;
         subplot(3,1,1);
         plot(x);
         title('Original Sound');
         % Only work in fourier domain
         f_y = fft(x);
         %
         subplot(3,1,2);
         plot(abs(f_y));
         title('Ft of original');
```

```
% We do not need the conjugate
f_y = f_y(1:end/2);
subplot(3,1,3);
plot(abs(f_y));
title('Half');
% Divide the whole into 4 parts
d = size(f_y,1)/4;
f_y = reshape(f_y,d,4);
```



In [40]:
```
ord = [2,3,3,4];
new_x = perm_ord(f_y,ord,d);
sound(new_x,43400);
```

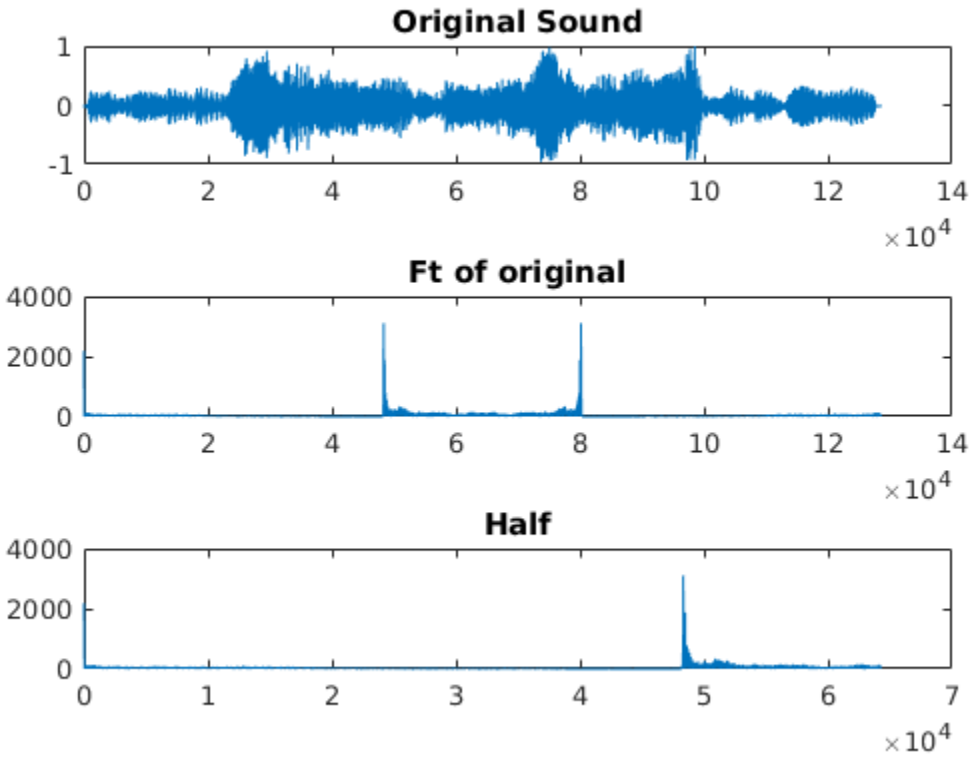
0.1.4 Message 2

=> Why so serious

In [42]:
```
x = audioread('./message2.wav');

figure;
subplot(3,1,1);
```

```
plot(x);
title('Original Sound');
% Only work in fourier domain
f_y = fft(x);
%
subplot(3,1,2);
plot(abs(f_y));
title('Ft of original');
% We do not need the conjugate
f_y = f_y(1:end/2);
subplot(3,1,3);
plot(abs(f_y));
title('Half');
% Divide the whole into 4 parts
d = size(f_y,1)/4;
f_y = reshape(f_y,d,4);
```



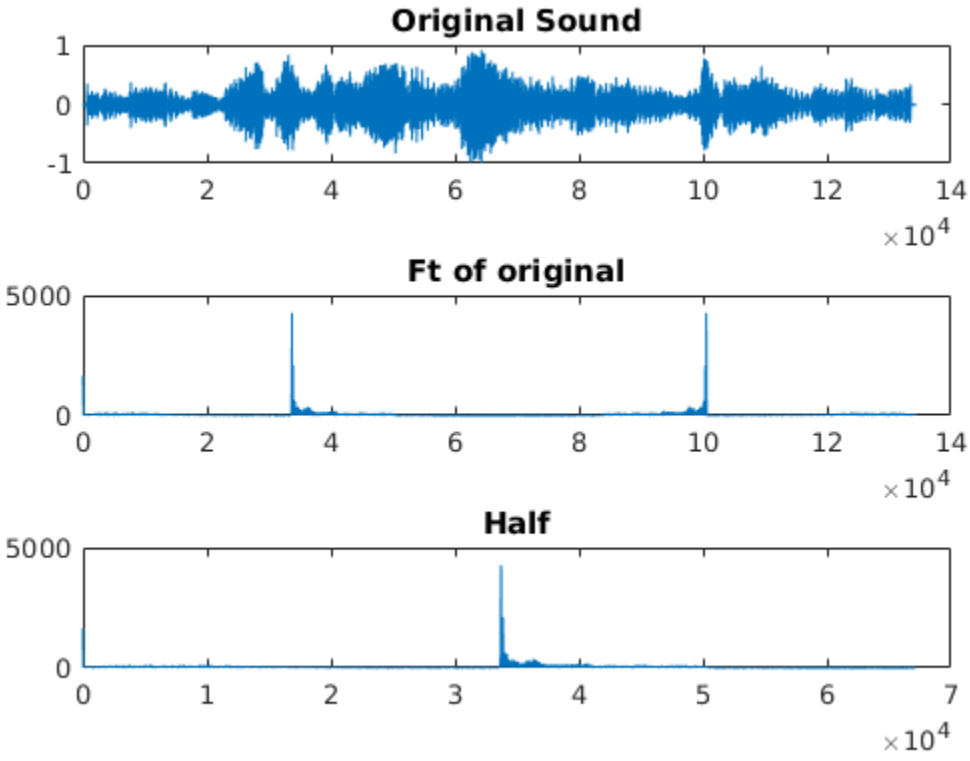
In [41]:
```
ord = [4,3,2,3];
new_x = perm_ord(f_y,ord,d);
sound(new_x,43400);
```

0.1.5 Message 3

=> Lets put a smile on that face

In [56]:
```
x = audioread('./message3.wav');

figure;
subplot(3,1,1);
plot(x);
title('Original Sound');
% Only work in fourier domain
f_y = fft(x);
%
subplot(3,1,2);
plot(abs(f_y));
title('Ft of original');
% We do not need the conjugate
f_y = f_y(1:end/2);
subplot(3,1,3);
plot(abs(f_y));
title('Half');
% Divide the whole into 4 parts
d = size(f_y,1)/4;
f_y = reshape(f_y,d,4);
```



In [57]:
```
ord = [3,2,2,2];
new_x = perm_ord(f_y,ord,d);
sound(new_x,43400);
plot(new_x);
```