

# **Spatial moving sound source with headphone**

**Wenzheng Zhou**

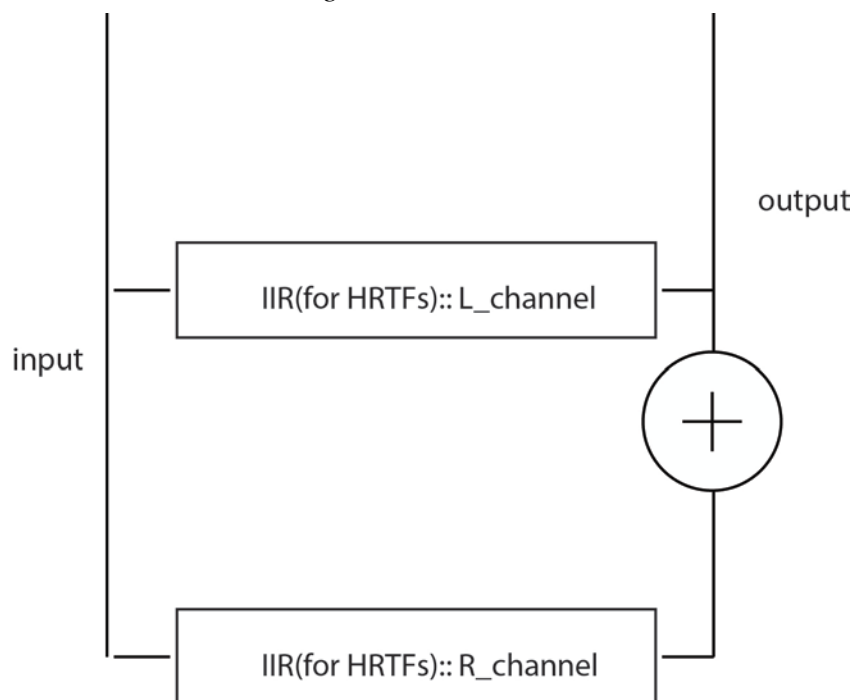
**420093312**

**Introduction:**

The 3D spatial sound with headphone, it offers psychological information relate the location to the audient psychological information. And also, this sort of spatial sound has been widely used in the game and film industries. however, once we watch movie or play game with headphone, it is easy to find the difference, the film is more fantasy and realistic than the game, especially the sound track of the movie makes the audience feel himself in the film, the game is hardly to catch movie. The reason for that is the film use more advanced technology to collect and develop sound effect or track.

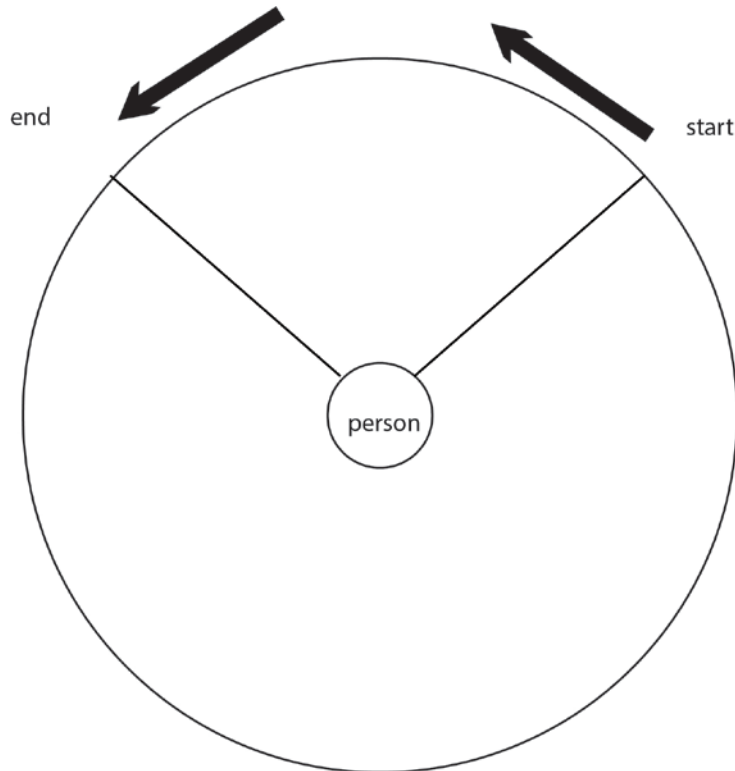
Fortunately, because of the gamers, we can have more advanced headphone today. If we notice the speak market share, we will find the headphones' brands are more concerning about their headphone for the game. In order to match the gamers' desires, the headphone is designed for precise location in tiny of period. Due to this kind of headphone, it is good opportunity to use that to create the music or sound like background music in the film. Therefore, I try to make a filter to simulate the moving sound source in the environment.

This is my filter design draft, we can see the signal has been divide into two channels and beginning of the processing, and then , the IIR filter use the signal process relate to the specific angle to obtain the result, at the end of the process, the function combine both channel together.



Generally Speaking, the filter focuses on taking advantage of the 3D spatial sound, but I develop a function in order to specify the track of sound source follow.

Picture shows how the filter works, the sensitive in function called twochannel is segments between the end and start point, usually the big number is easy to get smooth sound (such as 20);



**Function:**

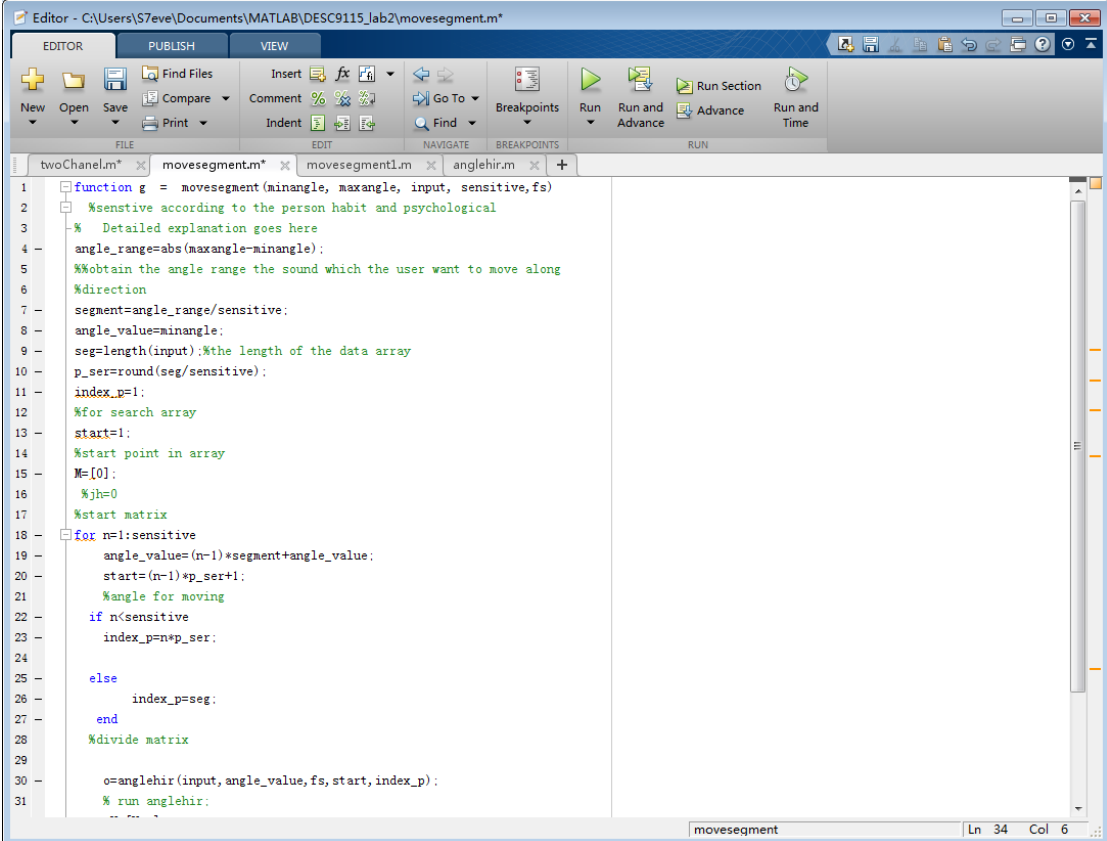
The filter is consisted by four functions, anglehir, movesegment, movesegment1 and twochannel, anglehir function is used to obtain the matrix on the specific angle;

```

Editor - C:\Users\S7eve\Documents\MATLAB\DESC9115_lab2\anglehir.m
EDITOR PUBLISH VIEW
Find Files Insert
New Open Save Compare Comment Go To Breakpoints Continue Step Step In Function Call Stack
Print Indent Find Run to Cursor Step Out Step Out twoChannel
FILE EDIT NAVIGATE BREAKPOINTS DEBUG
twoChannel.m* movesegment.m* movesegment1.m* anglehir.m*
1 function a = anglehir(input, angle1, fs, start, index_p)
2 %angle base on the view angle in 2D dimension according to concept
3 angle=angle1+90;
4 angle_ref=150;
5 a_min=0.05;
6 c=344;
7 as=0.05;
8 w=c/a;
9 delay=0;
10 azimuth=1+a_min+(1-a_min)*cos(angle/angle_ref*pi);
11 b=[(w/fs+azimuth)/(1+w/fs), (w/fs-azimuth)/(1+w/fs)];
12 %numerator for impulse
13 a=[1, (w/fs-1)/(w/fs+1)];
14 %denominator for impulse
15 if abs(angle)<90
16 delay=-fs/w*(cos(angle*pi/180)-1);
17 else
18 delay=fs/w*(cos(abs(angle)-90)*pi/180+1);
19 end
20
21 d=(1-delay)/(1+delay);
22 magt=filter(b, a, input);
23 E=filter(d, 1, [1, d], magt);
24 %veimp(b, a);
25 %fvtool(b, a);
26 %dceconv(V, 512);
27
28 oE(start:index_p, 1);
29 end
30
31
Ln 1 Col 1

```

*movesegment works for controlling the sound source movement for one channel;*



```
1 function g = movesegment(minangle, maxangle, input, sensitive, fs)
2     %sensitive according to the person habit and psychological
3     -% Detailed explanation goes here
4     angle_range=abs(maxangle-minangle);
5     %obtain the angle range the sound which the user want to move along
6     %direction
7     segment=angle_range/sensitive;
8     angle_value=minangle;
9     seg=length(input):%the length of the data array
10    p_ser=round(seg/sensitive);
11    index_p=1;
12    %for search array
13    start=1;
14    %start point in array
15    M=[0];
16    %jh=0
17    %start matrix
18    for n=1:sensitive
19        angle_value=(n-1)*segment+angle_value;
20        start=(n-1)*p_ser+1;
21        %angle for moving
22        if n<sensitive
23            index_p=n*p_ser;
24        else
25            index_p=seg;
26        end
27        %divide matrix
28
29
30        o=anglehir(input, angle_value, fs, start, index_p);
31        % run anglehir:
```

*movesegment1 is similar to movesegment, but work for other channel, twochannel function is used to discrete two channel matrix and output value*

*The function called twochannel is the API function, it offer the user several arguments in order to user customize the sound.*

```

1 function s = twoChanel( voice, minangle, maxangle, sensitive, fs )
2 %change the channel for two channel sounds
3 x1=voice(:,1);
4 x2=voice(:,2);
5 t=0;
6 xa1=movesegment(minangle, maxangle, x1, sensitive, fs);
7
8 xa2=movesegment1(minangle, maxangle, x1, sensitive, fs);
9 %s=[xa1,xa2];
10
11 x1=[x1;t];
12 plot(xa1,x1);
13 plot2=plot(xa2);
14 wavwrite(s, fs, 'output.wav');
15 end
16
17

```

*voice: it is work for input data;*  
*minangle: input the initial angle(related to head) which you want to put sound source on*  
*maxangle: angle of the end position of the sound source.*  
*sensitive: related to the density of sound on different angle.*

**Further analysis:**

*The filter has already worked in Matlab, also can simulate the movement of sound source. However, the sound lacks some reverberations and multiple movement (such as up to down) function. In the DAFX, it introduces the torso and shoulder as important factors that humans realize the position of sound, I may use*

$$\gamma_{sh} = 1.2 * (180^\circ - azimuth) / 180^\circ * (1 - 0.000004 * (\phi - 80) * 180^\circ / (180^\circ + azimuth))$$

*Obtains the delay in time domain, and then, using the Fourier transform to calculate the reverberation value in frequency domain,*

*And also, we have to notice the Left and Right ears help humans to locate the sound source, in other words, if we can deeply understand the delay difference and sensitive difference between two ears, we can simulate more realistic and reasonable sound*

**Bibliography**

Zölzer, Udo, and Amatriain, Xavier. DAFX digital audio effects. Chichester, England: Wiley, 2002.

Rumsey F. Spatial audio[M]. Taylor & Francis, 2001.  
Print.