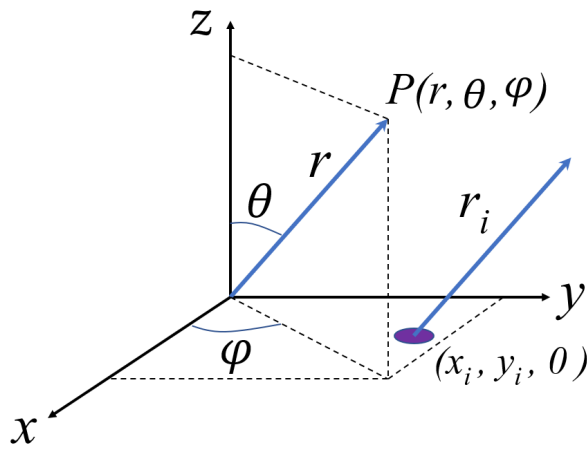
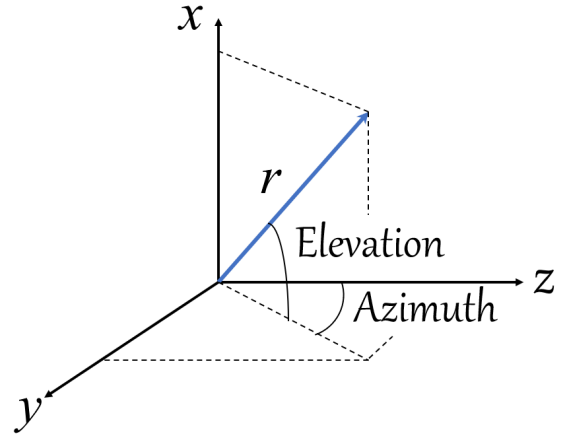


【アレイ信号処理・ビーム走査 Beam Scanning】

1次元パターン 座標系



2次元パターン 座標系



P を表すベクトル \mathbf{r} は, $\mathbf{r} = (r \sin \theta \cos \varphi, r \sin \theta \sin \varphi, r \cos \theta)$.
 r, θ, φ 方向の単位ベクトル $\mathbf{a}_r, \mathbf{a}_\theta, \mathbf{a}_\varphi$ は,

$$\mathbf{a}_r = \frac{1}{\left\| \frac{\partial \mathbf{r}}{\partial r} \right\|} \frac{\partial \mathbf{r}}{\partial r} = (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta)$$

$$\mathbf{a}_\theta = \frac{1}{\left\| \frac{\partial \mathbf{r}}{\partial \theta} \right\|} \frac{\partial \mathbf{r}}{\partial \theta} = (\cos \theta \cos \varphi, \cos \theta \sin \varphi, -\sin \theta)$$

$$\mathbf{a}_\varphi = \frac{1}{\left\| \frac{\partial \mathbf{r}}{\partial \varphi} \right\|} \frac{\partial \mathbf{r}}{\partial \varphi} = (-\sin \varphi, \cos \varphi, 0)$$

指向性

$$R = \sum_i R_0 \frac{e^{-jkr_i}}{r_i} = \sum_i R_0 \frac{e^{-jk(r - \mathbf{a}_r \cdot \mathbf{r}_i)}}{r_1} = R_0 \frac{e^{-jkr}}{r_1} \sum_i e^{jk(x_i \sin \theta \cos \varphi + y_i \sin \theta \sin \varphi)}$$

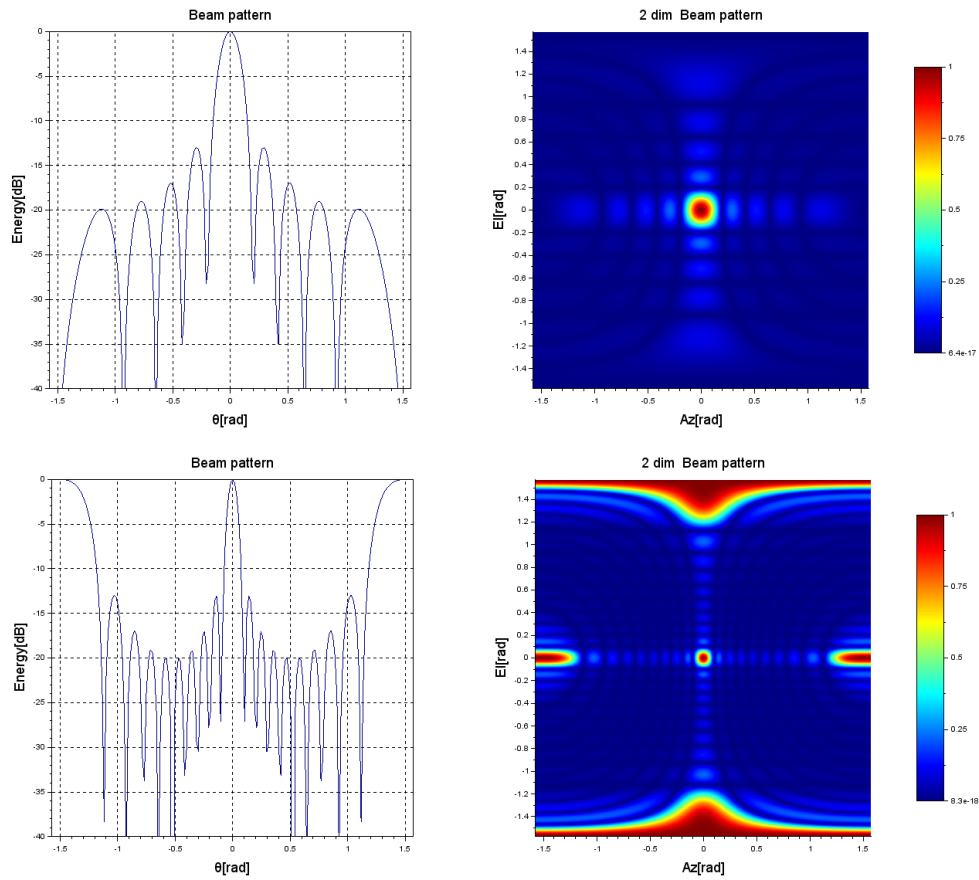


Figure 1: Scilab 実行結果
正面方向 素子間隔 半波長, 1 波長

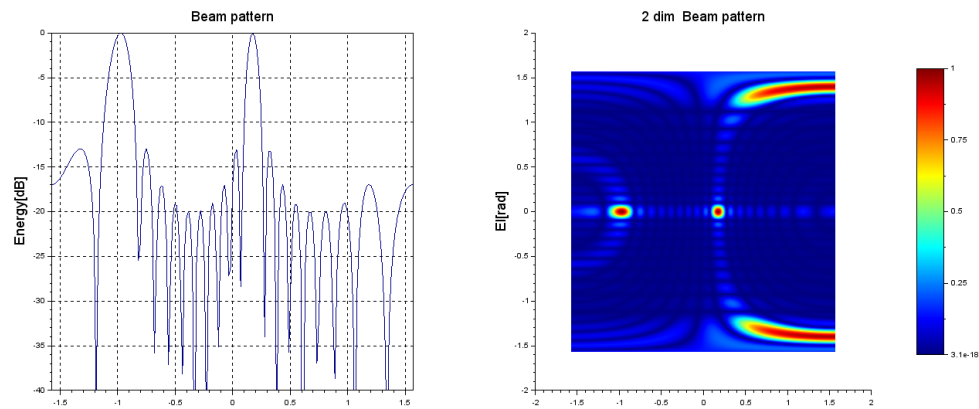


Figure 2: Scilab 実行結果
ビーム走査 (θ 方向, Az 方向に 10[deg])

Source Code 1: Scilab

```

////////////////////////////////////
//   ビームパターン走査
//   Beam Scanning
//
//                               M.Tsutsui
////////////////////////////////////

```

```

clear;

stacksize('max');

funcprot(0);
function [C]=crdt(itv_f,x_end,y_end) //itv_f:音源間隔 ,x_end:x座標音源数 ,y_end:y座標音源数
//__座標作成関数____//

    C=[];
    for j=0:itv_f:(y_end-1)*itv_f;
        for i=0:itv_f:(x_end-1)*itv_f;
            C=[C;i,j];
        end
    end

endfunction

funcprot(0);
function [BF_m]=BF_F(BF_ang,d_size) //BF_ang:  $\theta$  方向走査角度 d_size:データサイズ mainプログラム size2

//※[size1,size2]=size(u)
//__1次元パターン_ビーム走査用関数__//

    fai=0;//  $\phi$ 
    BF_p=[sin(BF_ang)*cos(fai);sin(BF_ang)*sin(fai)];

    BF_m=[];

    for cnt=1:1:d_size;
        BF_m=[BF_m,BF_p];
    end

endfunction

pi=%pi;//  $\pi$ 
j=%i;//虚数

T=25;//摂氏温度
c=331.5+0.61*T;//音速
f=1000;//周波数
wl=c/f;//波長
k=(2*pi)/wl;//波数

//_____座標_____//
itv=wl; //音源間隔

C=crdt(itv,10,10);

sita_min=-pi/2; //  $\theta$  下端
sita_max=pi/2; //  $\theta$  上端
st=sita_min:pi/180:sita_max;//  $\theta$ 
fai=0; //  $\phi$ 

u=[sin(st)*cos(fai);sin(st)*sin(fai)];//単位ベクトルより [sin  $\theta$  cos  $\phi$  ; sin  $\theta$  sin  $\phi$  ]

[size1,size2]=size(u);

```

```

//___ビーム走査___//
BF=pi/18; //θ方向 走査角度[rad]

cal=C*(u-BF.F(BF,size2));//行列計算

H=exp(j*k*cal);

[size3,size4]=size(cal);

W=abs(ones(1,size3)*H);//加算,絶対値

I=20*log10(W/max(W));

subplot(1,2,1)
plot(st,I);
set(gca(),'data_bounds',[sita_min,-40;sita_max,0]);
set(gca(),'tight_limits','on');
xlabel("θ [rad]", "fontsize", 4);
ylabel("Energy [dB]", "fontsize", 4);
title('Beam_pattern','fontsize',4);
xgrid();

//_____2次元パターン_____//
AzAz=st;//Azimuth
ElEl=st;//Elevation
[El,Az]=meshgrid(AzAz,ElEl);

K=0;
for I=1:1:size3;
    loop=C(I,1)*sin(El)+C(I,2)*cos(El).*sin(Az)-(C(I,1)*sin(0)+C(I,2)*sin(BF)).*cos(0);
    K=K+exp(j*k*loop);
end

abv=abs(K');
subplot(1,2,2)
Sgrayplot(AzAz,ElEl,abv/max(max(abv)));
set(gcf(),'color_map',jetcolormap(256));
colorbar(min(abv/max(max(abv))),max(abv/max(max(abv))));
xlabel('Az[rad]',_,'fontsize',_4);
ylabel('El[rad]',_,'fontsize',_4);
title("2_dim_Beam_pattern",'fontsize',4);

```

Source Code 2: Python

```

#-----
# Module Name:Beam pattern
# Author:m.tsutsui
#-----

#_Library_Import_____
from numpy import*
import math, numpy as np
import matplotlib.pyplot as plt
#_Library_Import_end_____

def crdt(itv_f,x_end,y_end):
    C=[]
    for j in np.arange(0,y_end*itv_f,itv_f):
        for i in np.arange(0,x_end*itv_f,itv_f):
            buf=i,j
            C.append(buf)

```

```

return array(C)

if __name__ == '__main__':

    T=25 #temp
    c=331.5+0.61*T #speed of sound
    f=1000 #freq
    wl=c/f #wave length
    k=(2*pi)/wl #wave number

    #-----Coordinates-----#
    itv=wl/2 #Array_Interval

    C=crdt(itv,10,10)
    #-----#

    sita_min=-pi/2
    sita_max=pi/2
    st=array(np.arange(sita_min,sita_max,pi/180))
    fai=0

    u1=array(sin(st)*cos(fai))
    u2=array(sin(st)*sin(fai))

    u=np.c_[u1, u2].T

    BF=0#pi/18 #scan angle[rad]
    BF_m=array([[sin(BF)*cos(fai)],[sin(BF)*sin(fai)])]

    cal=matrix(C)*(matrix(u)-matrix(BF_m))

    size1,size2=cal.shape

    H=exp(1j*k*cal)

    W_p=abs(ones((1,size1))*matrix(H))

    W=W_p/np.max(W_p)

    I=array(20*log10(W))

#-----2dim_pattern-----//
    AzAz=st #Azimuth
    ElEl=st #Elevation
    Az,El=meshgrid(AzAz,ElEl)

    K=0
    for L in np.arange(0,size1,1):
        loop=C[L,0]*sin(El)+C[L,1]*cos(El)*sin(Az)-(C[L,0]*sin(0)+C[L,1]*sin(BF)*cos(0))
        K+=exp(1j*k*loop)

    abv=abs(K)

#_Plot_-----
    plt.figure(facecolor='w')
    plt.subplot(121)
    plt.plot(st,I.T)
    plt.axis([sita_min, sita_max, -40, 0])
    plt.xlabel("$\Theta$ [rad]")
    plt.ylabel("Energy [dB]")
    plt.title('Beam_pattern')

```

```

plt.grid()

plt.subplot(122)
plt.contourf(Az,El,abv,100)
plt.xlabel("Az [rad]")
plt.ylabel("El [rad]")
plt.title('2_dim_Beam_pattern')
plt.grid()

plt.show()

```

Source Code 3: Python

```

#-----
# Module Name:Beam Scanning
# Author:m_tsutsui
#-----

#_Library_Import-----
from numpy import*
import math, numpy as np
import matplotlib.pyplot as plt
#_Library_Import_end-----

def crdt(itv_f,x_end,y_end):
    C=[]
    for j in np.arange(0,y_end*itv_f,itv_f):
        for i in np.arange(0,x_end*itv_f,itv_f):
            buf=i,j
            C.append(buf)

    return array(C)

if __name__ == '__main__':

    T=25 #temp
    c=331.5+0.61*T #speed of sound
    f=1000 #freq
    wl=c/f #wave length
    k=(2*pi)/wl #wave number

    #-----Coordinates-----#
    itv=wl #Array_Interval

    C=crdt(itv,10,10)
    #-----#

    sita_min=-pi/2
    sita_max=pi/2
    st=array(np.arange(sita_min,sita_max,pi/180))
    fai=0

    u1=array(sin(st)*cos(fai))
    u2=array(sin(st)*sin(fai))

    u=np.c_[u1, u2].T

    BF=pi/18 #scan angle[rad]
    BF_m=array([[sin(BF)*cos(fai)],[sin(BF)*sin(fai)])]

    cal=matrix(C)*(matrix(u)-matrix(BF_m))

```

```

size1,size2=cal.shape

H=exp(1j*k*cal)

W_p=abs(ones((1,size1))*matrix(H))

W=W_p/np.max(W_p)

I=array(20*log10(W))

#-----2dim_pattern-----//
AzAz=st #Azimuth
ElEl=st #Elevation
Az,El=meshgrid(AzAz,ElEl)

K=0
for L in np.arange(0,size1,1):
    loop=C[L,0]*sin(El)+C[L,1]*cos(El)*sin(Az)-(C[L,0]*sin(0)+C[L,1]*sin(BF)*cos(0))
    K+=exp(1j*k*loop)

abv=abs(K)

#_Plot-----
plt.figure(facecolor='w')
plt.subplot(121)
plt.plot(st,I.T)
plt.axis([sita_min, sita_max, -40, 0])
plt.xlabel("$\Theta$ [rad] ")
plt.ylabel("Energy [dB] ")
plt.title('Beam_pattern')
plt.grid()

plt.subplot(122)
plt.contourf(Az,El,abv,100)
plt.xlabel("Az [rad] ")
plt.ylabel("El [rad] ")
plt.title('2_dim_Beam_pattern')
plt.grid()

plt.show()

```