

Ingenieurbüro Baumann --- www.leobaumann.de --- Markt 6, 46282 Dorsten

manuelle Berechnung eines vert. Quads vor einem Reflektor über Grund

h = Länge, b2 = Höhe über Grund (Unterkante), d = Distanz Parallele, d1 = Distanz Reflektor, l = Wellenlänge

- `reset():digits:=16:w:=90*PI/180:vw:=59*PI/180:wh:=90*PI/180:h:=1/2:d:=1/2:d1:=1/2:b2:=1/2:l:=1:`

Richtdiagramm im Kugelraum als Funktion der Winkel

- `c:=(the,phil) -> abs((cos(PI*h/l*cos(phil))-cos(PI*h/l))/sin(phil))
*2*abs(cos(PI*d/l*cos(the)*sin(phil)))
*2*abs(cos(PI*d1/l*cos(the)*sin(phil)))
*2*abs(cos(PI*2*(b2+h/2)/l*cos(phil)))
+abs((cos(PI*d/l*cos(the)*sin(phil))-
cos(PI*d/l))/sqrt(1-cos(the)^2*sin(phil)^2))
*2*abs(sin(PI*h/l*cos(phil)))
*2*abs(sin(PI*d1/l*sin(phil)*sin(the)))
*2*abs(sin(PI*2*(b2+h/2)/l*cos(phil))):`

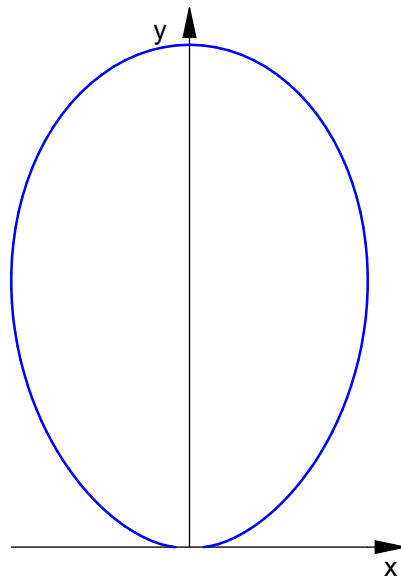
Antennenimpedanz nach 4nec2 einseitig mittengespeist

- `Z:=115+I*17.4;`

`115.0 + 17.4 · i`

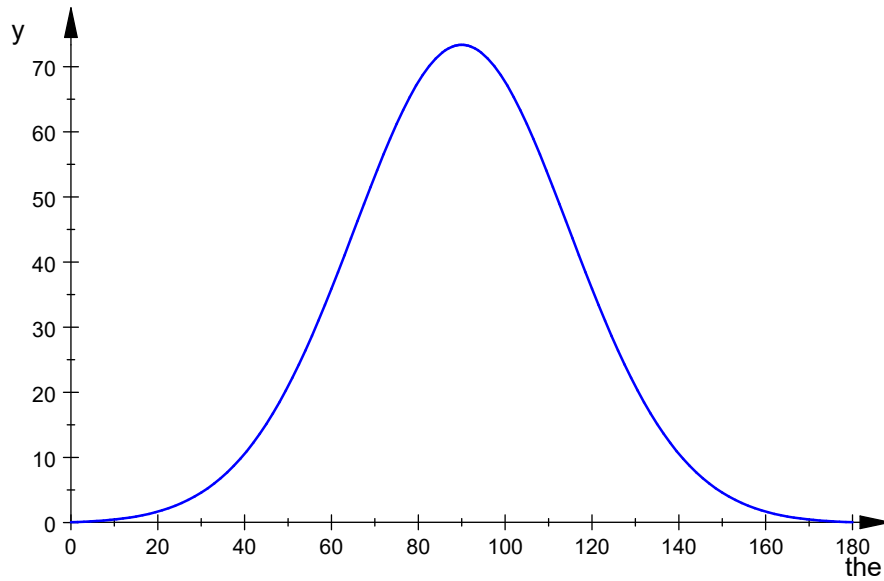
Horizontaldiagramm

- `plot(plot::Polar([c(the,wv),the], the = 0..PI, TicksNumber=None,
Scaling=Constrained, AdaptiveMesh=4));`



horizontale relative Strahlungsleistungsdichte

- `plotfunc2d(c(the*PI/180,wv)^2, the = 0..180):`



Maximalwert der relativen Strahlungsleistungsdichte , auch in dBi

- ```

ghmax:=0:ghwmax:=0:for m from 0 to 2880 step 1 do
gh:=float(c(m*PI/5760,wv)^2);
if gh>ghmax then
 ghmax:=gh;
 ghwmax:=float(m/32);
end_if;
end_for:ghmax;float(10*log(10,ghmax)+2.15);ghwmax;

```

73.34691118

20.80381829

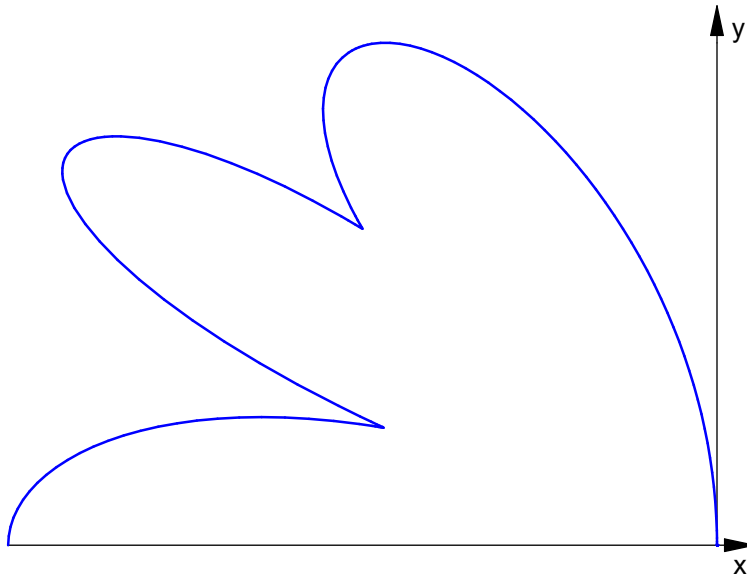
90.0

Vertikaldiagramm

- ```

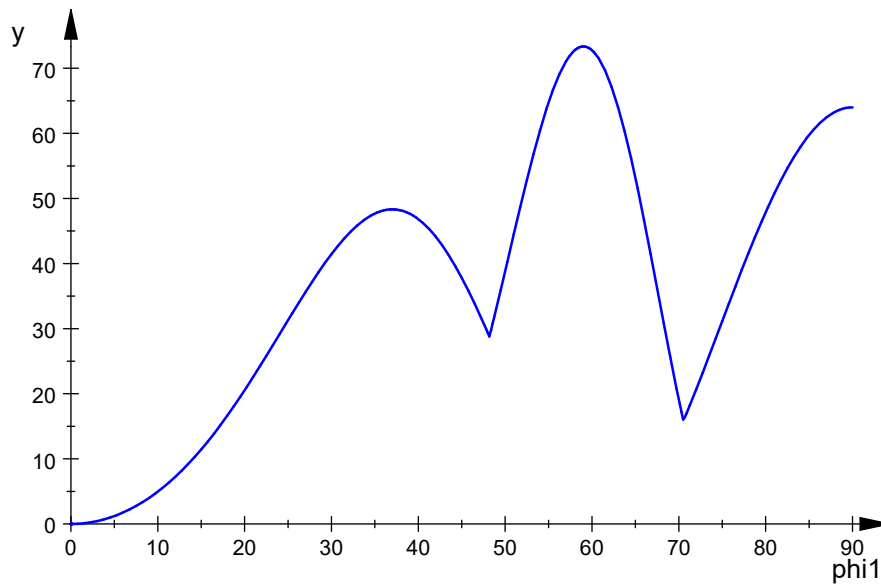
plot(plot::Polar([c(wh,phil),phil+PI/2], phil = 0..PI/2,
    TicksNumber=None, Scaling=Constrained, AdaptiveMesh=4));

```



vertikale relative Strahlungsleistungsdichte

- `plotfunc2d(c(wh,phi1*PI/180)^2, phi1 = 0..90):`



Maximalwert der relativen Strahlungsleistungsdichte , auch in dBi

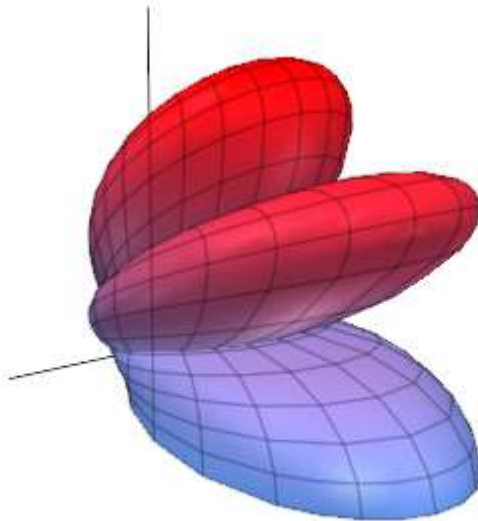
- `gvmax:=0:gvwmax:=0:for m from 1 to 2880 step 1 do
gv:=float(c(wh,m*PI/5760)^2);
if gv>gvmax then
gvmax:=gv;
gvwmax:=float(m/32);
end_if;
end_for:gvmax;float(10*log(10,gvmax)+2.15);gvwmax;`

73.34691118

20.80381829

59.0

- `graph:=plot::Surface([cos(the)*sin(phi)*c(the,phi),sin(the)*sin(phi)*c(the,phi),cos(phi)*c(the,phi)],the=0..PI,phi=-PI/2..0,Axes=Origin,TicksNumber=None,Scaling=Constrained,AdaptiveMesh=4):`
- `plot(graph);`



•