horizontal Dipol über perfektem Grund

Antennencharakteristik im Raum

also:In c[the,phi] und ParametricPlot3D bei phi Trigonometie wechseln, Winkel ab x,y-Ebene zu z-Achse pos. zählen

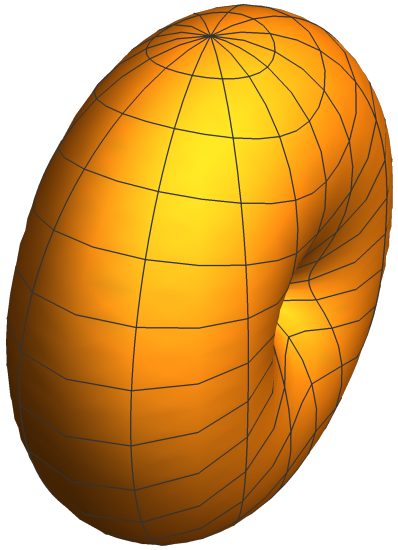
In[1]:=
```
time = TimeUsed[]; h = 1 / 2; b2 = 1 / 2; l = 1; wv = 0 * Degree; wh = 90 * Degree;
```
verbrauchte Zeit · Grad · Grad

In[2]:=
```
c[the_, phi_] := Abs[(Cos[Pi * h / l * Cos[the] * Cos[phi]] - Cos[Pi * h / l]) / (Sqrt[1 - Cos[the]^2 * Cos[phi]^2])];
```
Abs··· · Ko·· · Kreiszahl π · Kosinus · Kosinus · Ko·· · Kreiszahl π · Quadra··· · Kosinus · Kosinus

In[3]:=

In[4]:=
```
ParametricPlot3D[{Cos[the] * Cos[phi] * c[the, phi], Sin[the] * Cos[phi] * c[the, phi], Sin[phi] * c[the, phi]},
 {phi, -Pi / 2, Pi / 2}, {the, 0, 2 * Pi}, Axes → None, Boxed → False, PlotPoints → 15, Mesh → Full, PlotRange → All]
```

Out[4]=



Horizontal-Diagramm, horiz. rel. Strahlungsleistungsdichte u. Maximum bei Winkel auch dBi

In[5]:=
```
PolarPlot[c[the, wv], {the, 0, 2 * Pi}, PlotRange → All]
```
Polardarstellung Krei··· Koordinatenb··· alle

```
Plot[c[the * Degree, wv]^2, {the, 0, 360}]
```
stelle Funktion·· Grad

```
Maximum = N[FindMaximum[c[the, wv]^2, {the, 45 * Degree}]];
```
·· ermittle Maximum Grad

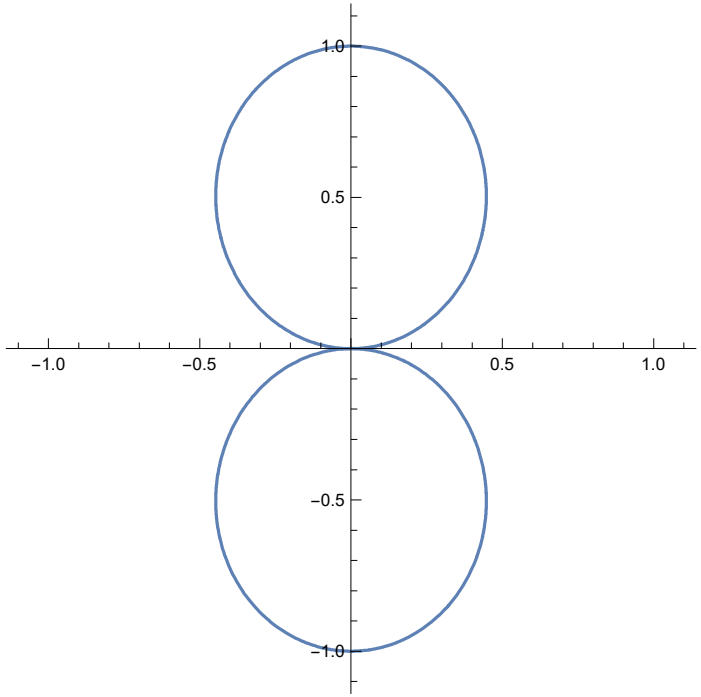
```
N[Maximum[[1]]]
```
numerischer Wert

```
N[10 * Log[10, Maximum[[1]]] + 2.15]
```
nume·· Logarithmus

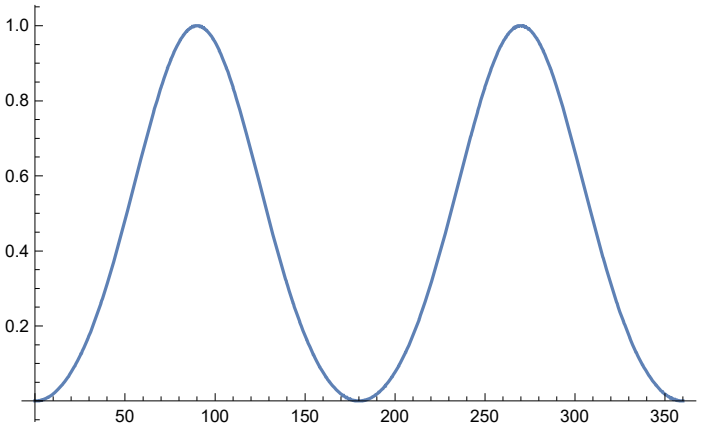
```
N[Maximum[[2]][[1, 2]] / Degree]
```
Grad

··· Power: Infinite expression $\frac{1}{0.}$ encountered.

Out[5]=



Out[6]=



Out[8]= 1.

Out[9]= 2.15

Out[10]= 90.

Vertikal-Diagramm, vertikal. rel. Strahlungsleistungsdichte u. Maximum bei Winkel auch dBi

In[11]:=
```
PolarPlot[c[wh, phi], {phi, 0, 2 * Pi}, PlotRange → All]
```
Polardarstellung | Krei… | Koordinatenb… | alle

```
Plot[c[wh, phi * Degree]^2, {phi, 0, 180}]
```
stelle Funktion grap… | Grad

```
Maximum = N[FindMaximum[c[wh, phi]^2, {phi, Pi / 2}]];
```
… | ermittle Maximum | Kreiszahl π

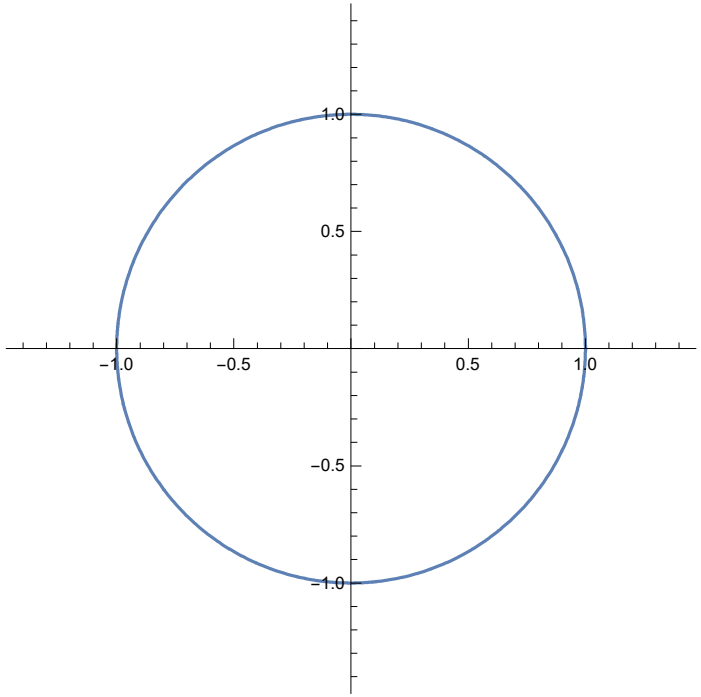
```
N[Maximum[[1]]]
```
numerischer Wert

```
N[10 * Log[10, Maximum[[1]]] + 2.15]
```
Logarithmus

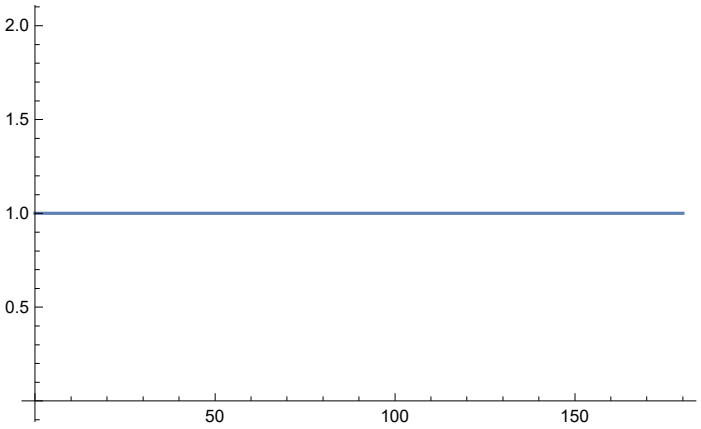
```
N[Maximum[[2]][[1, 2]] / Degree]
```
Grad

```
Print[TimeUsed[] - time, " CPU-Time in seconds"]
```
verbrauchte Zeit

Out[11]=



Out[12]=



FindMaximum: Encountered a gradient that is effectively zero. The result returned may not be a maximum; it may be a minimum or a saddle point.

Out[14]= 1.

Out[15]= 2.15

Out[16]= 90.

0.391 CPU-Time in seconds