

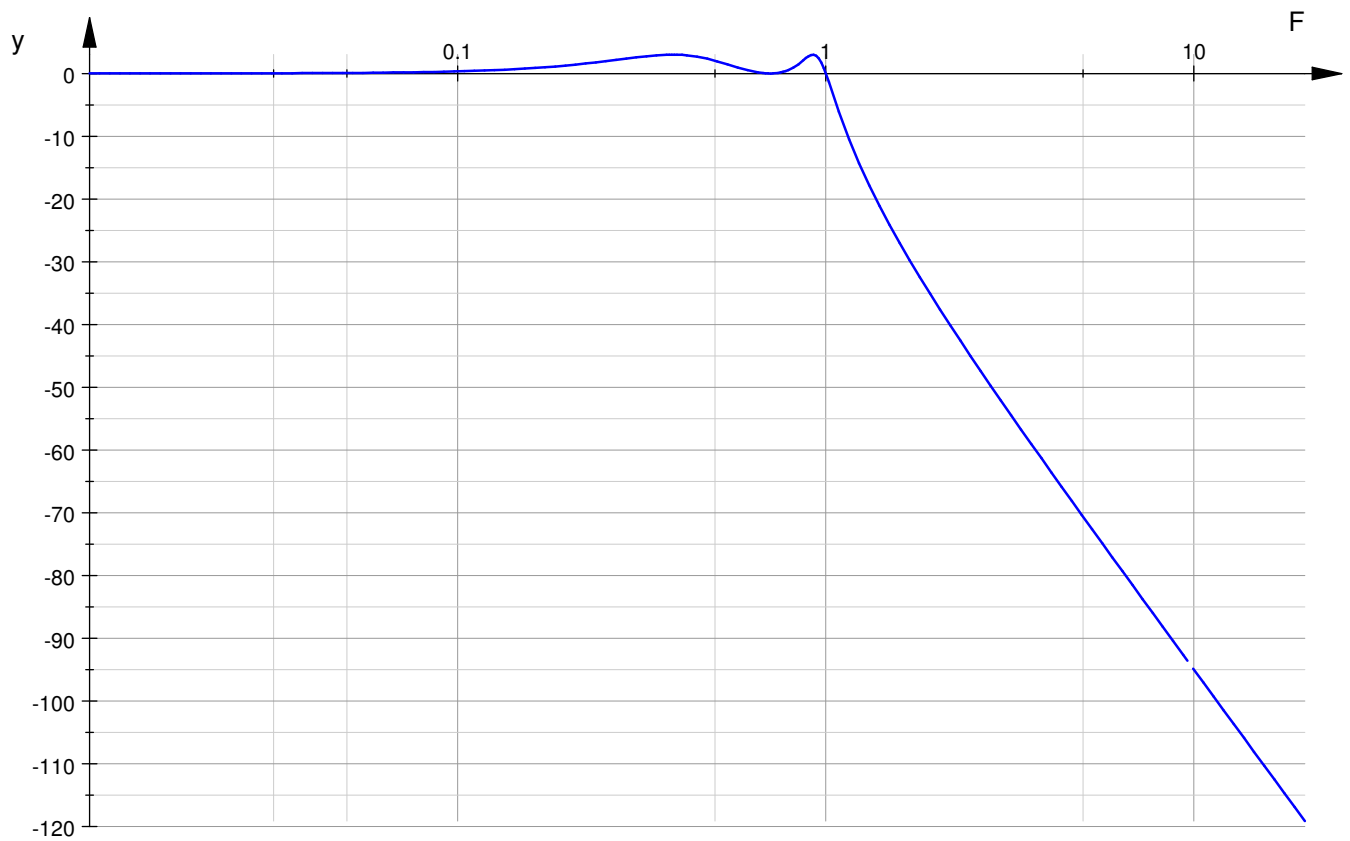
## Amplitudenverlauf eines TSCHEBYSCHEFF-Filters und eines JACOBI-Filters im Vergleich und

### Berechnung der Polstellen für beide Filtertypen mit MuPAD

----- 18. Mai 2007, Ingenieurbüro Baumann -----

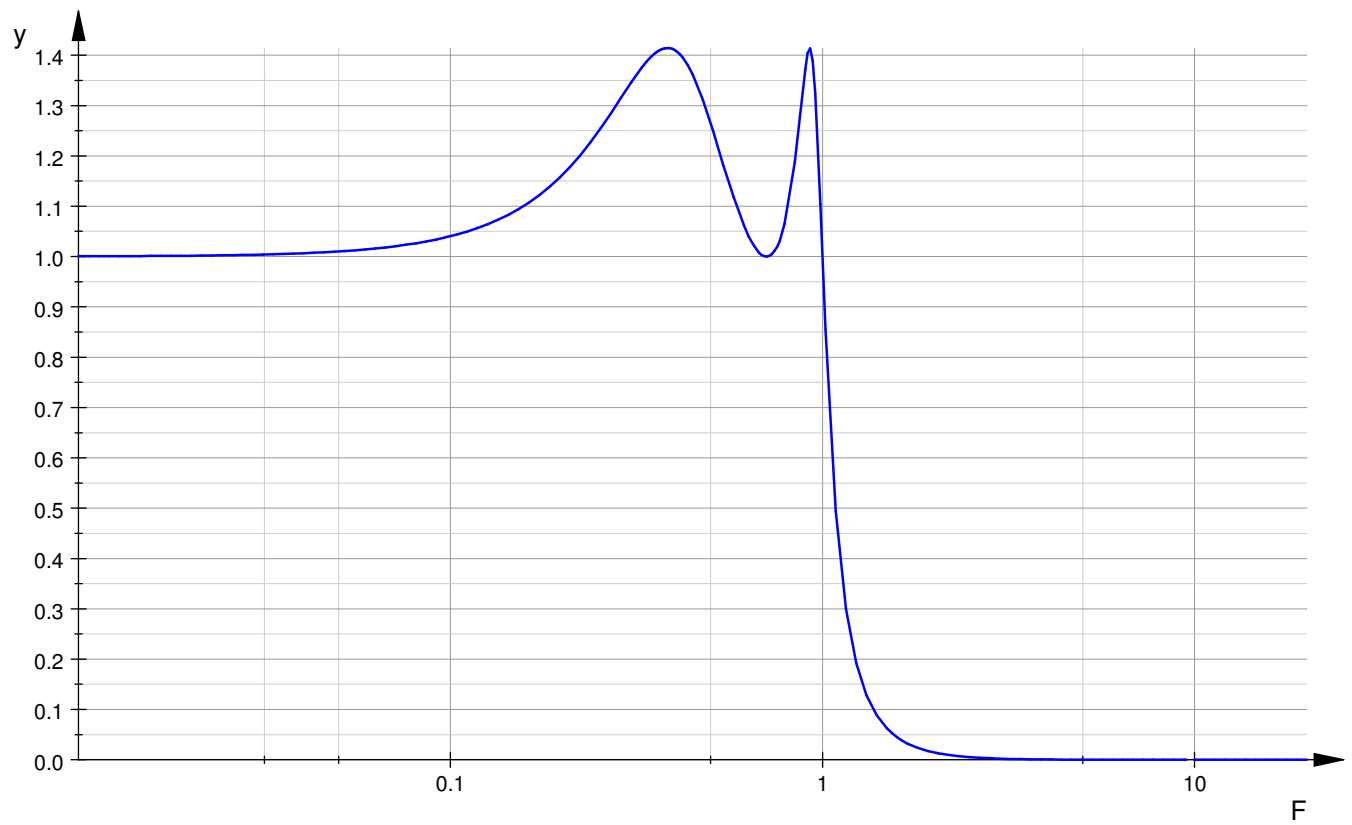
- `reset():DIGITS:=32:`
- `n:=4:RippledB:=3.01:epsilonT:=sqrt(10^(0.1*RippledB)-1):`
- `KT:=(F)->cos(n*arccos(F)):gcr:=limit(KT(F),F=0,Right):`
- `print("der Grenzwert der char. Funktion von TSCHEBYSCHEFF gcr =  
",gcr," ( n gerade -> 1, n ungerade -> 0 )"):`  
  
`"der Grenzwert der char. Funktion von TSCHEBYSCHEFF gcr = ", 1,  
" ( n gerade -> 1, n ungerade -> 0 )"`
- `FT:=(F)->1/sqrt(1+(epsilonT*KT(F))^2):a0T:=1/limit(FT(F),F=0,Right):`
- `print("das a0 des Betrages der TSCHEBYSCHEFF-Amplitudenfunktion a0T  
= ",a0T,"und ergibt sich aus RippledB"):`  
  
`"das a0 des Betrages der TSCHEBYSCHEFF-Amplitudenfunktion a0T = ",  
1.4141647250701539511753990774471, "und ergibt sich aus RippledB"`
- `FTN:=(F)->a0T*FT(F):`
- `FTNdB:=(F)->20*log(10,FTN(F)):print("die Filtersteilheit [dB/Dekade]  
= ",float(FTNdB(10)-FTNdB(1))):`  
  
`"die Filtersteilheit [dB/Dekade] = ", -  
94.964013368115336211739808993993`
- `plotfunc2d(FTNdB(F), F=1/100..20, LegendVisible=FALSE,  
CoordinateType=LogLin,  
GridVisible=TRUE, SubgridVisible=TRUE,  
Height=120*unit::mm, Width=180*unit::mm,  
Header="Amplitudenfunktion des TSCHEBYSCHEFF-Filters, doppelt  
logarithmisch"):`

Amplitudenfunktion des TSCHEBYSCHEFF-Filters, doppelt logarithmisch



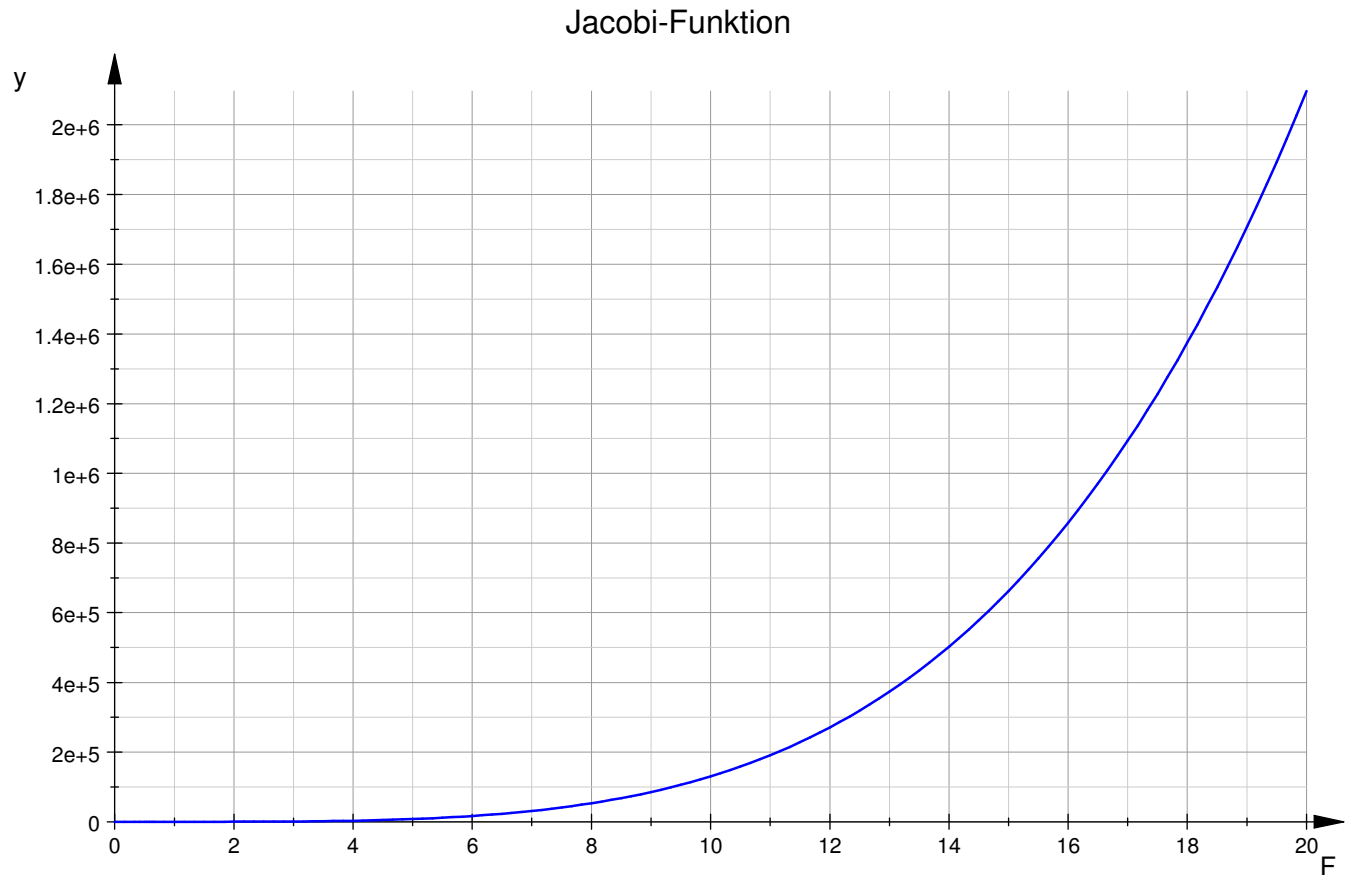
- `plotfunc2d(FTN(F), F=1/100..20, LegendVisible=FALSE, CoordinateType=LogLin, GridVisible=TRUE, SubgridVisible=TRUE, Height=120*unit::mm, Width=180*unit::mm, Header="Amplitudenfunktion des TSCHEBYSCHEFF-Filters, einfach logarithmisch über F"):`

Amplitudenfunktion des TSCHEBYSCHJEFF-Filters, einfach logarithmisch über F



- `k:=1:epsilonC:=sqrt(10^(0.1*RippledB)-1):`
- `Jc:=(F)->orthpoly::jacobi(n,1,1,F):`

- `plotfunc2d(Jc(F), F=0..20, LegendVisible=FALSE, CoordinateType=LinLin, GridVisible=TRUE, SubgridVisible=TRUE, Height=120*unit::mm, Width=180*unit::mm, Header="Jacobi-Funktion"):`



- `FC:=(F)->1/sqrt(1+(epsilonC*Jc(F))^2):a0C:=1/FC(0):`
- `FCN:=(F)->a0C*FC(F):a0C:=1/FC(0):`
- `FCNdB:=(F)->20*log(10,FCN(F)):alpha:=op(numeric::solve(FCNdB(F)=0,F=7/10..2,RestrictedSearch),1):`

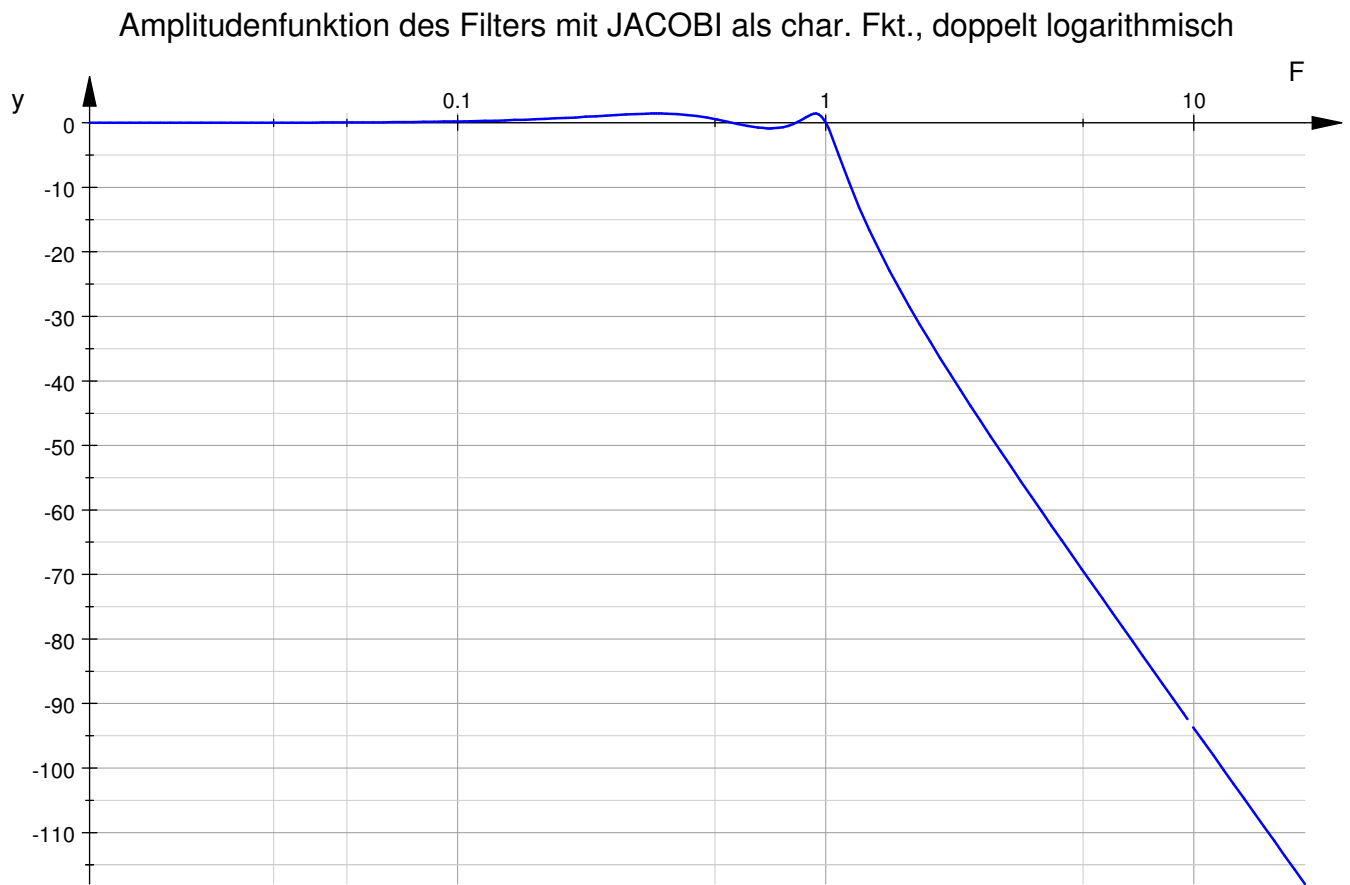
0.81649658092772603273242802490196

- `FCNdBk:=(F)->20*log(10,FCN(alpha*F)):print("die Filtersteilheit [dB/Dekade] = ",FCNdBk(10)-FCNdBk(1)):`

"die Filtersteilheit [dB/Dekade] = ", -  
93.798602027019429919443527093475

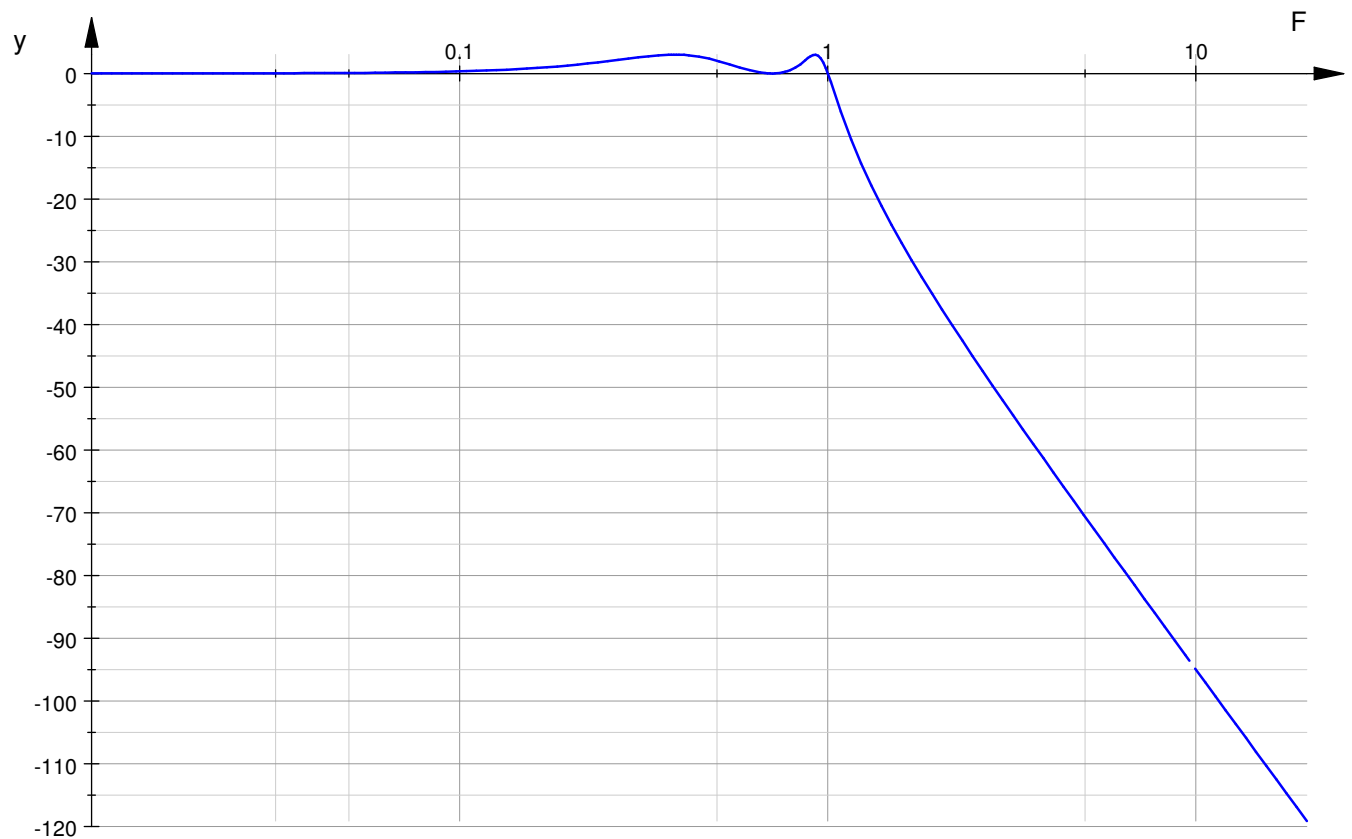
- `plotfunc2d(FCNdB(alpha*F), F=1/100..20, LegendVisible=FALSE, CoordinateType=LogLin, GridVisible=TRUE, SubgridVisible=TRUE,`

```
Height=120*unit::mm, Width=180*unit::mm,  
Header="Amplitudenfunktion des Filters mit JACOBI als  
char. Fkt., doppelt logarithmisch"):
```



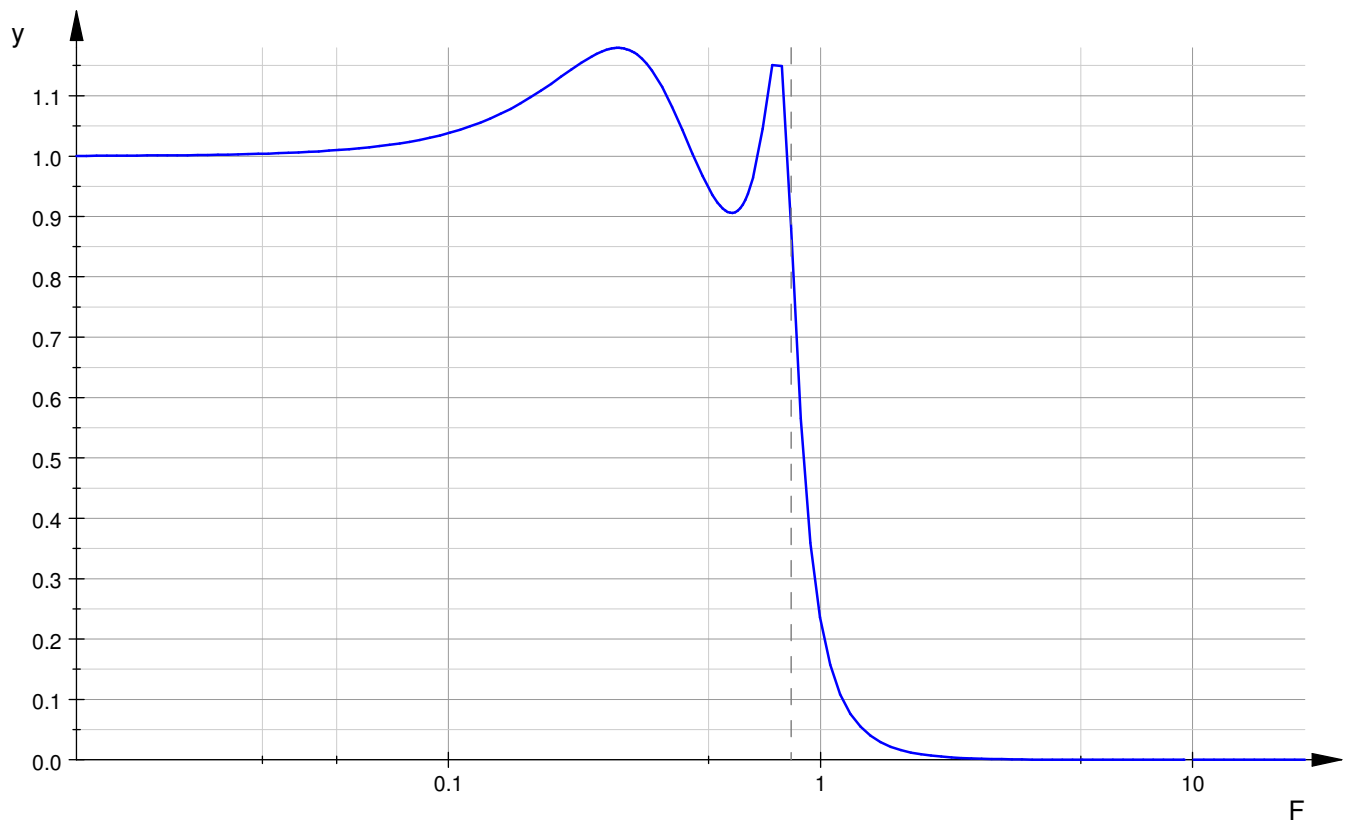
- ```
plotfunc2d(FTNdB(F), F=1/100..20, LegendVisible=FALSE,  
CoordinateType=LogLin,  
GridVisible=TRUE, SubgridVisible=TRUE,  
Height=120*unit::mm, Width=180*unit::mm,  
Header="Amplitudenfunktion des TSCHEBYSCHJEFF-Filters, doppelt  
logarithmisch, zum Vergleich"):
```

Amplitudenfunktion des TSCHEBYSCHJEFF-Filters, doppelt logarithmisch, zum Vergleich



- `plotfunc2d(FCN(F), F=1/100..20, LegendVisible=FALSE, CoordinateType=LogLin, GridVisible=TRUE, SubgridVisible=TRUE, Height=120*unit::mm, Width=180*unit::mm, Header="Amplitudenfunktion des Filters mit JACOBI als char. Fkt., einfach logarithmisch über F"):`

Amplitudenfunktion des Filters mit JACOBI als char. Fkt., einfach logarithmisch über F



- `print("die TSCHEBYSCHJEFF-Filter-Polstellen stimmen mit anderen Angaben überein"):`

`"die TSCHEBYSCHJEFF-Filter-Polstellen stimmen mit anderen Angaben überein"`

- `ListeT:=[-Im(op(float(solve(1+(epsilonT*KT(F))^2=0,F)),i))-I*Re(op(float(solve(1+(epsilonT*KT(F))^2=0,F)),i)) $ i=1..n];`

`[0.085010532860532360295502149566263 +  
0.94640066225562318945384215532251 I`

`,`

`0.20523358137646088767152950098637 -  
0.39201198974515803710368905539601 I`

`,`

`- 0.20523358137646088767152950098637 +  
0.39201198974515803710368905539601 \`  
`I`

`,`

```

- 0.085010532860532360295502149566263 -
0.94640066225562318945384215532251\
I

]

• print("die JACOBI-Filter-Polstellen"):

      "die JACOBI-Filter-Polstellen"

• ListeJ:=[-Im(op(float(solve(1+(epsilonC*Jc(alpha*F))^2=0,F)),i))-
I*Re(op(float(solve(1+(epsilonC*Jc(alpha*F))^2=0,F)),i)) $ i=1..n];

[0.10341967955034225498031136707334 -
0.96875699108021788580750958738221 I

,

- 0.26531506887863222498701113454525 -
0.37762098475266371966586030148751 \
I

,

0.26531506887863222498701113454525 +
0.37762098475266371966586030148751 I

,

- 0.10341967955034225498031136707334 +
0.96875699108021788580750958738221 \
I

]

•

```