Fourier transzformáció

Korszerű matematikai módszerek a geodéziában

2019.10.14.

Áttekintés

- Fourier-sor
- CFT (folytonos transzformáció)
- DFT (diszkrét transzformáció)
- FFT (gyors transzformáció)
- Példa: kerék gyorsulásmérő adatok FFT-je
- Alkalmazások

1D Fourier-sor

- *f*(*t*)-nek trigonometrikus alaprendszer szerinti felbontása (*t* idő vagy távolság)

$$f(t) = \frac{1}{2} a_0 + \sum_{n=1}^{\infty} \left(a_n \cos n\omega_0 t + b_n \sin n\omega_0 t \right)$$

- alap körfrekvencia (rad/s, rad/km)

$$\omega_0 = \frac{2\pi}{T}$$

Fourier-sor komplex alakban

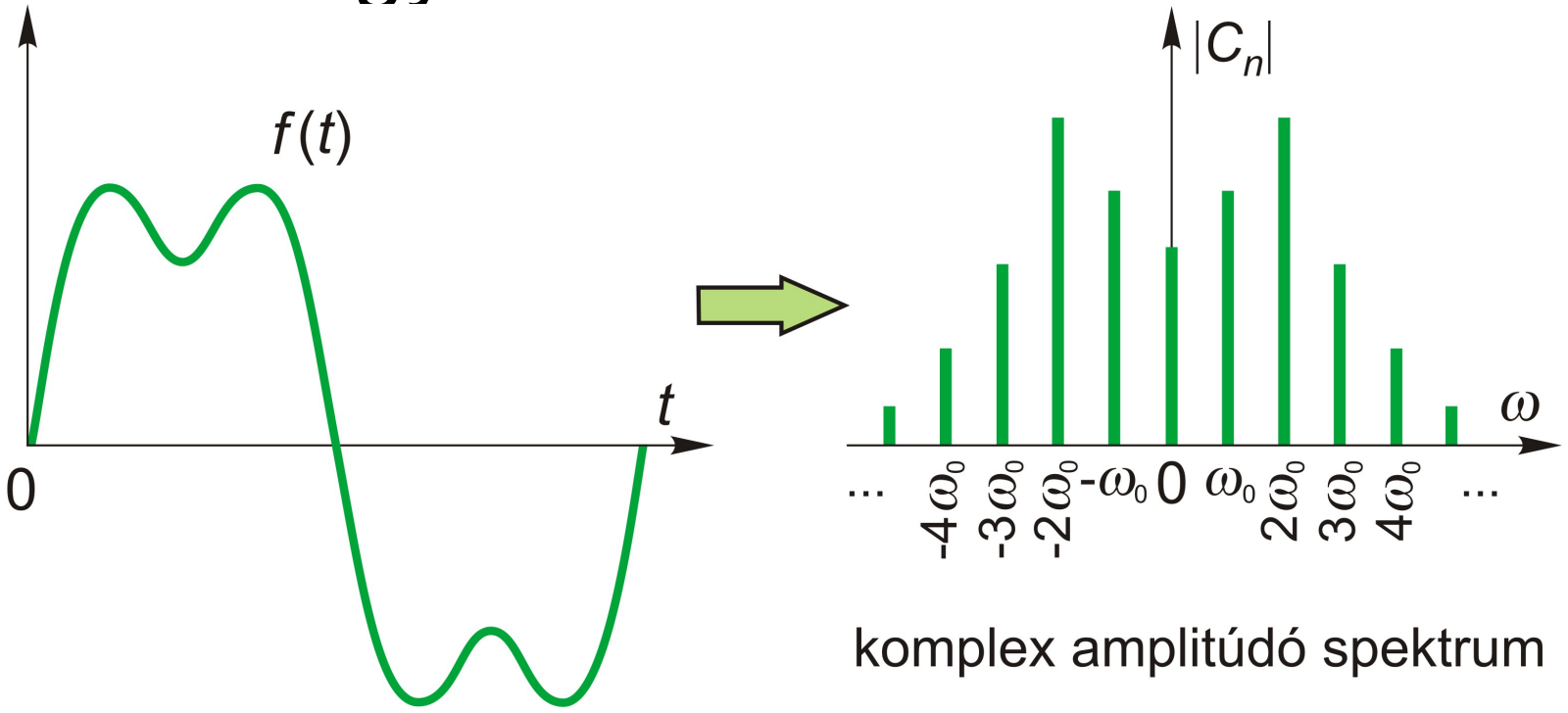
- Euler összefüggés: $e^{in\omega_0 t} = \cos n\omega_0 t + i \sin n\omega_0 t$
- komplex alak:

$$f(t) = \sum_{-\infty}^{\infty} c_n e^{in\omega_0 t}$$

- komplex együtthatók:

$$c_{\pm n} = \frac{1}{2}\left(a_n \mp i b_n\right)$$

Periódusos függvény leképezése diszkrét együtthatókra



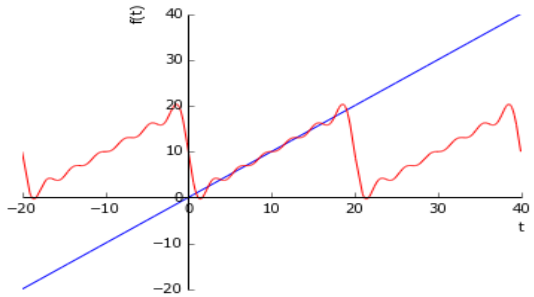
komplex amplitúdó spektrum

- az együtthatók kiszámítása (*T* periódus):

$$c_n = \frac{1}{T} \int_d^{d+T} f(t)\, e^{-in\omega_0 t}\, dt$$

1D Fourier-sor (Jupyter munkafüzet)

https://nbviewer.jupyter.org/github/gyulat/Fourier/blob/master/1DFourier.ipynb



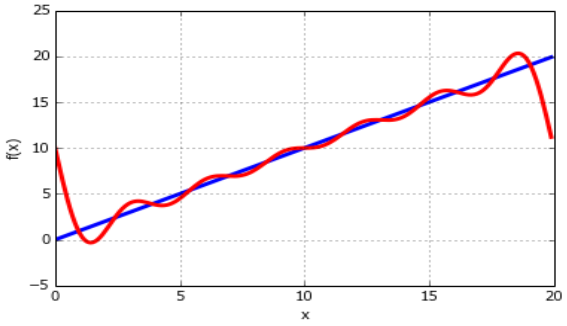
A szimbolikus megoldás helyett a numerikusan egyenértékű közelítő megoldást is előállíthatjuk a SymPy `mpmath` moduljával.

In [5]:
```
import mpmath

cs = sympy.mpmath.fourier(f, [t0, t0+T], N)

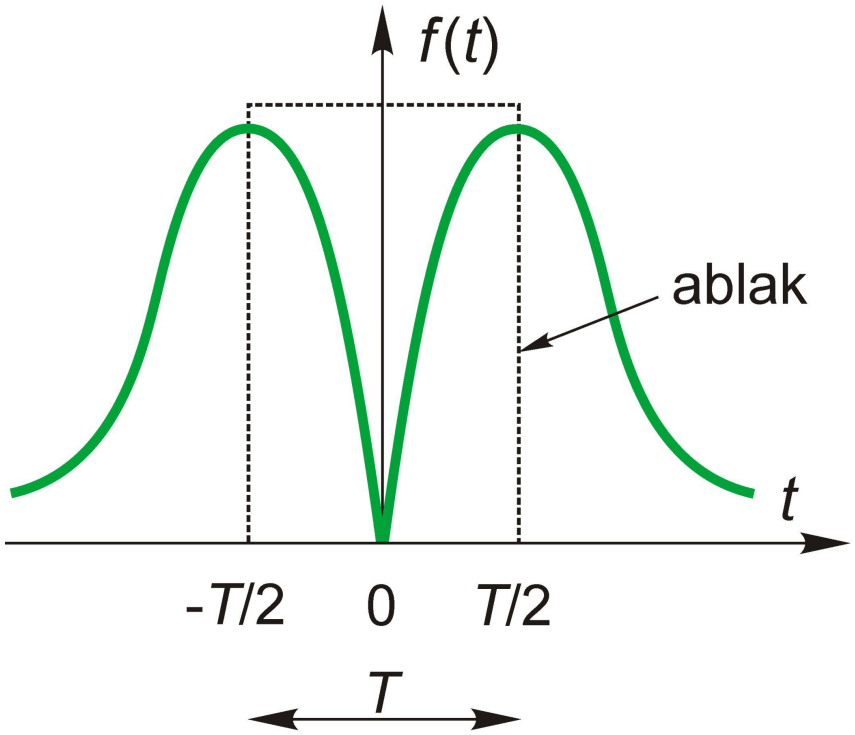
def numeric_approx(t):
    return mpmath.fourierval(cs, [t0, t0+T], t)

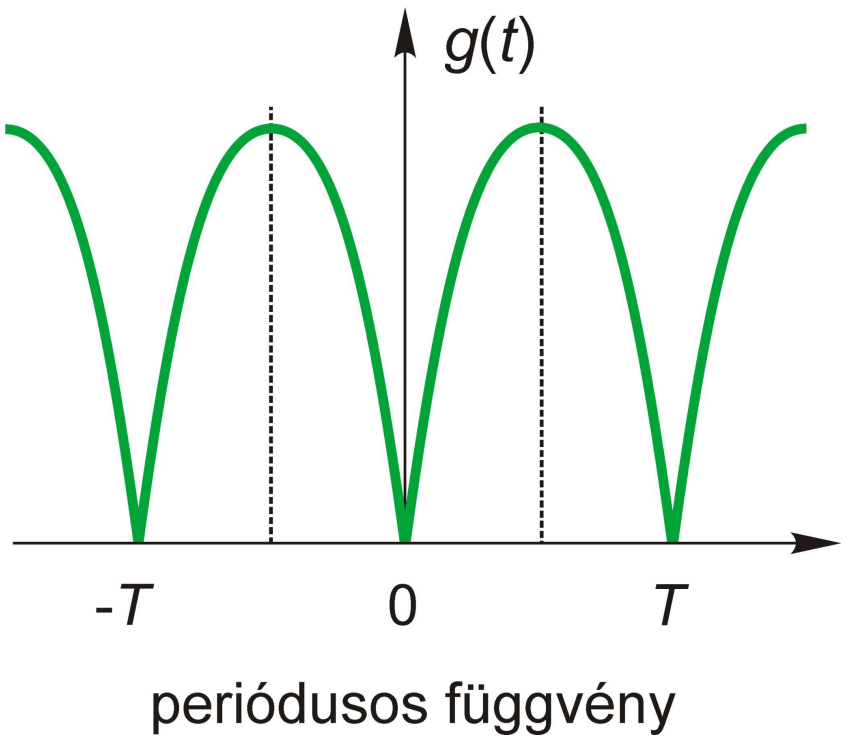
mpmath.plot([f, numeric_approx], [t0, t0+T])
```



Folytonos Fourier-transzformáció (CFT)

- egy $g(t)$ periódusos függvény ($T \to \infty$) határátmenetre nemperiódusossá válik:





Fourier-transzformáció (CFT)

- direkt és inverz transzformált pár:

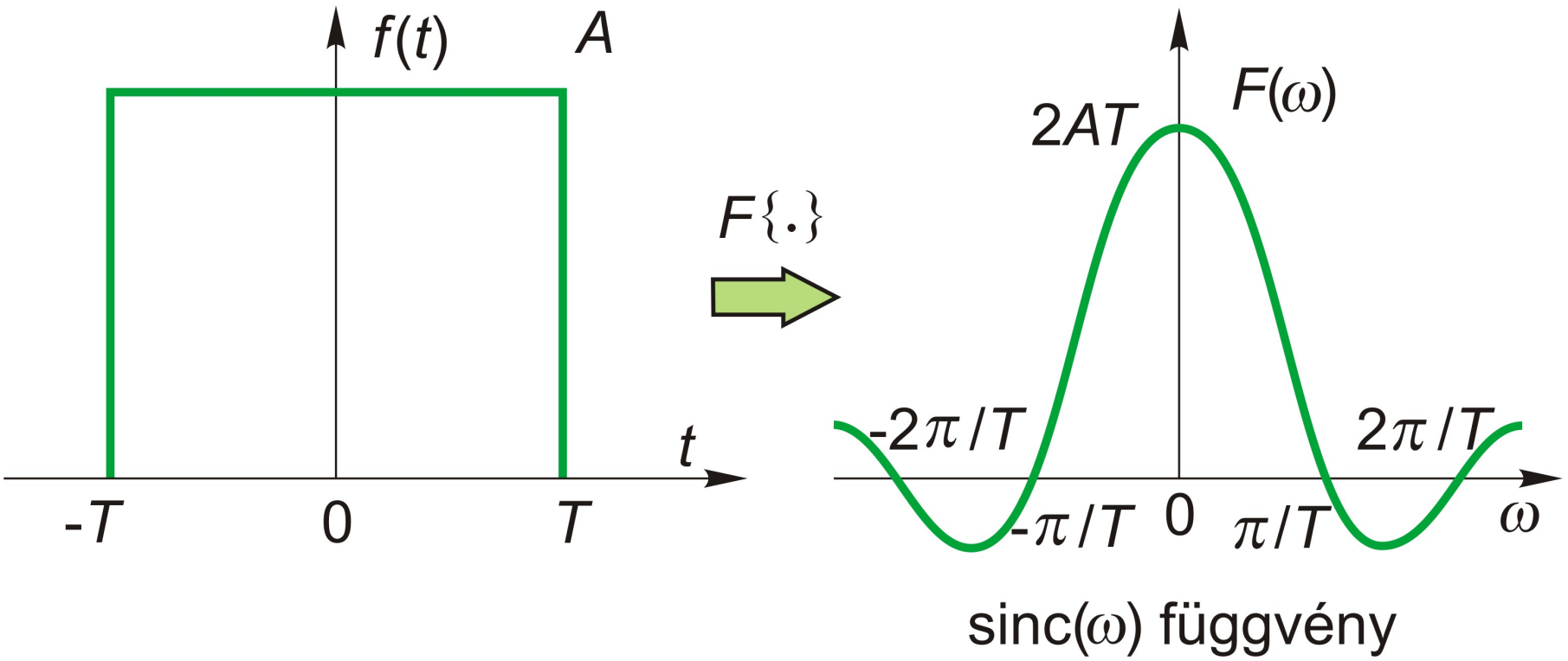
$$F(\omega) = \int_{-\infty}^{\infty} f(t)\, e^{-i\omega t}\, dt = \mathbf{F}\{f(t)\}$$

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)\, e^{i\omega t}\, d\omega = \mathbf{F}^{-1}\{F(\omega)\}$$

$e^{i\omega t} = \cos \omega t + i \sin \omega t$ **periódusos** függvény!

A CFT tulajdonságai

- **páros** *f(t)* függvény *F(ω)* transzformáltja **valós**



sinc(ω) függvény

- **linearitás**

$$\mathbf{F}\{\alpha f(t)+\beta g(t)\}=\alpha\mathbf{F}\{f(t)\}+\beta\mathbf{F}\{g(t)\}=\alpha F(\omega)+\beta G(\omega)$$

- **eltolás (fázis változás)**

$$\mathbf{F}\{f(t-t_0)\}=\mathbf{F}\{f(t)\}e^{-i\omega t_0}=F(\omega)\,e^{-i\omega t_0}$$

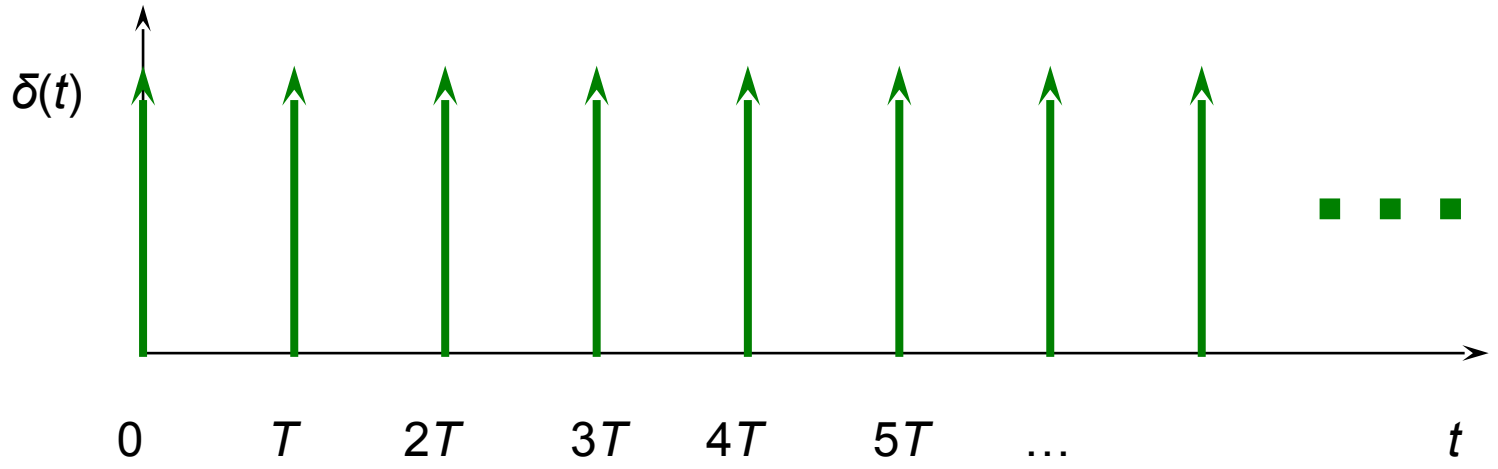
- **konvolúciótétel**

$$\mathbf{F}\{f(t)*h(t)\}=F(\omega)\,H(\omega)$$

A konvolúció Fourier-transzformáltja
a tényező függvények Fourier-transzformáltjainak
szorzata.

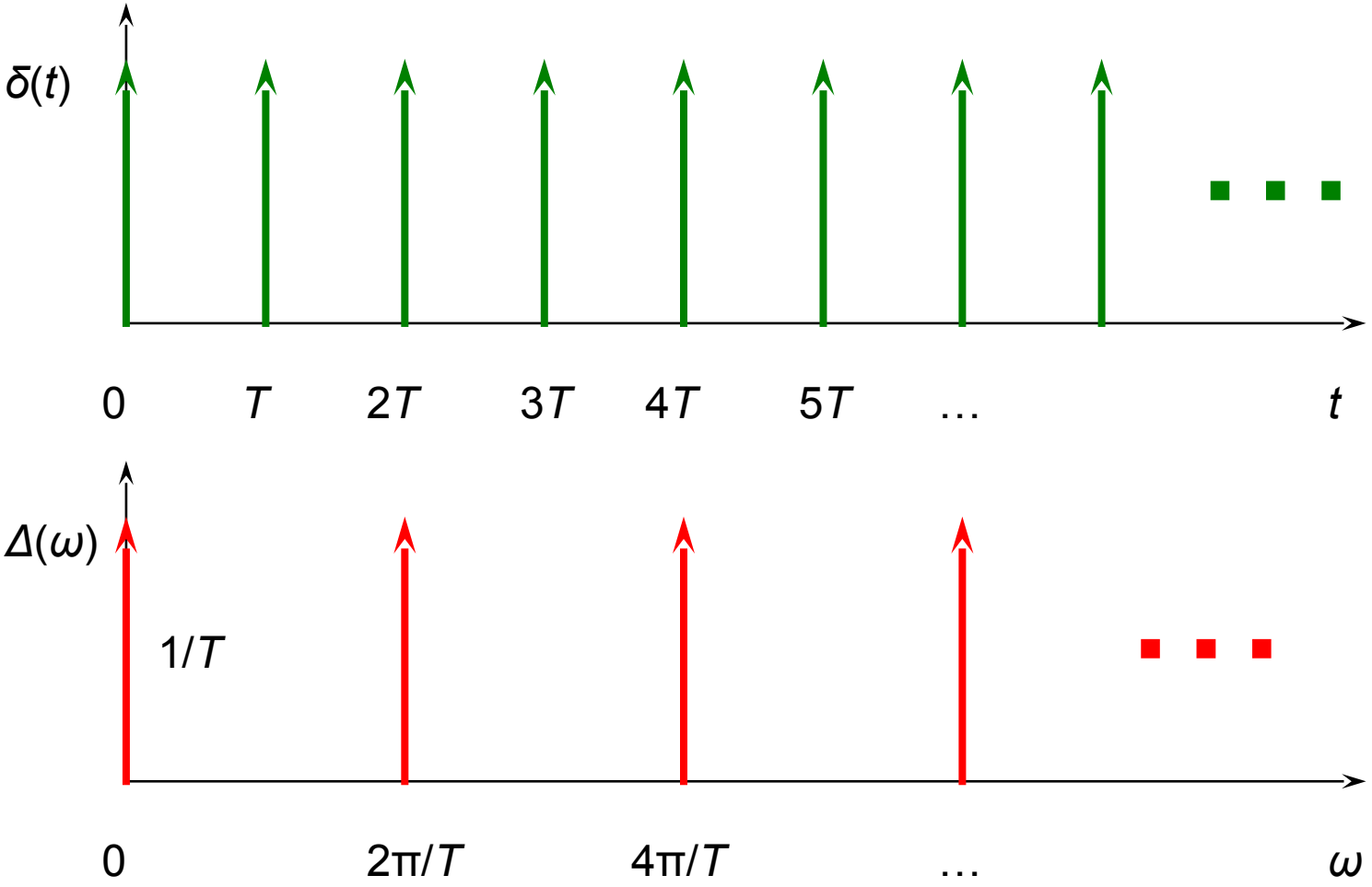
Dirac-féle egységimpulzus

- Dirac-féle egységimpulzus „függvény” $\delta(t)$: mindenhol zérus, kivéve a t helyet, ahol végtelen nagy, de integrálja véges
- Hasznos absztrakció mint pl. a tömegpont
- Impulzus sorozat:



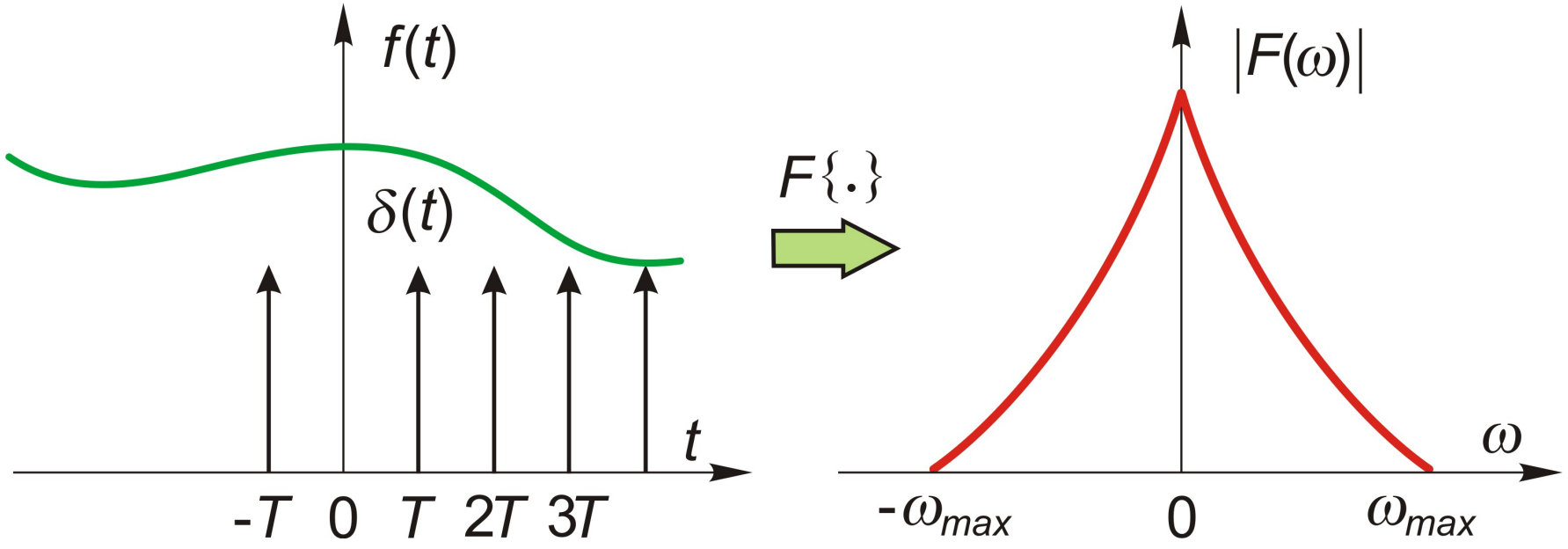
Impulzus sorozat Fourier-transzformáltja

- Impulzus sorozat CFT-je egy másik impulzus sorozat:



Mintavételezés

- a mintavételezés $f(t)$ impulzussorozattal való szorzását jelenti:

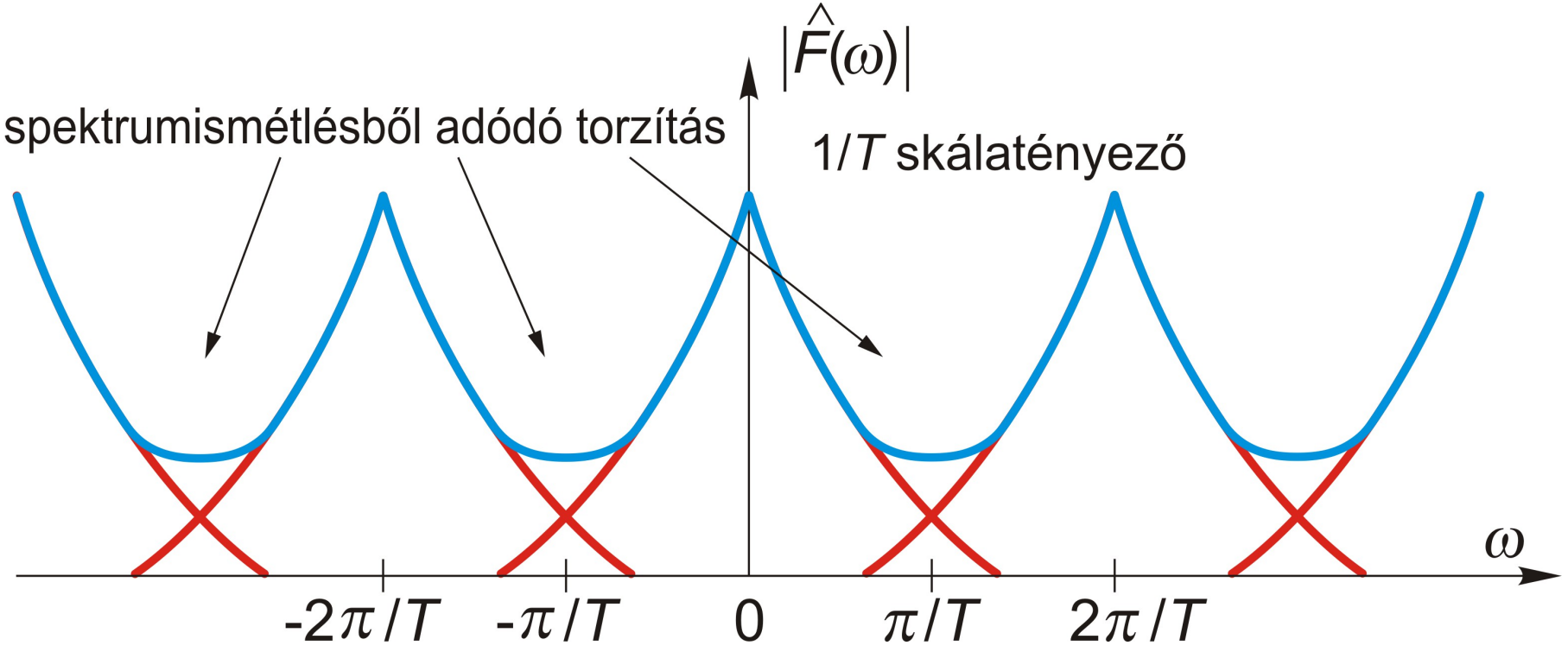


Mintavételezett jel Fourier-transzformáltja (CFT-je)

- a mintavételezett függvény CFT-je (az ‘inverz’ konvolúció tétel szerint) $f(t)$-nek az impulzussorozat Fourier-transzformáltjával vett konvolúciója lesz
- $f(t)$ függvény $F(\omega)$ transzformáltjainak ($1/T$-vel átskálázott) másolatait a mintavételezési $2\pi/T$ (kör)frekvenciának megfelelő közönként egymás mellé helyezzük és összeadjuk
- ez a diszkrét Fourier-transzformáció (**DFT**)

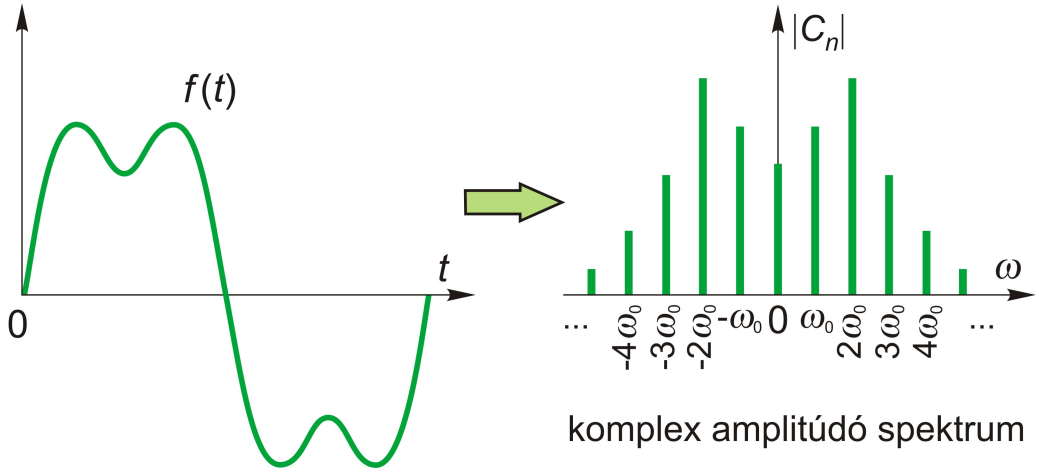
Diszkrét Fourier-transzformáció (DFT)

- a DFT $2\pi/T$ körfrekvenciával periódusos
- az egymás mellé tett másolatok átlapolódhatnak:



Diszkrét ↔ periódusos

- Elv: ami az egyik tartományban diszkrét, az a másikban periódusos ($t \leftrightarrow \omega$)

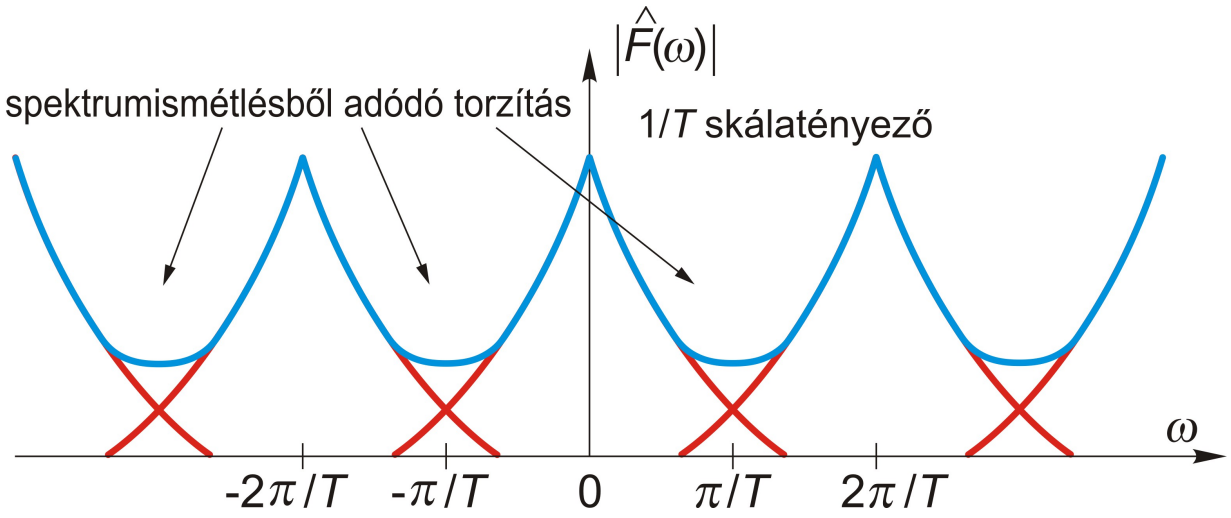


- Példa: a gömb két irányban is periódusos, ezért a gömbfüggvénysor kétszeresen diszkrét: $\sum \sum$

Mintavételezési tétel

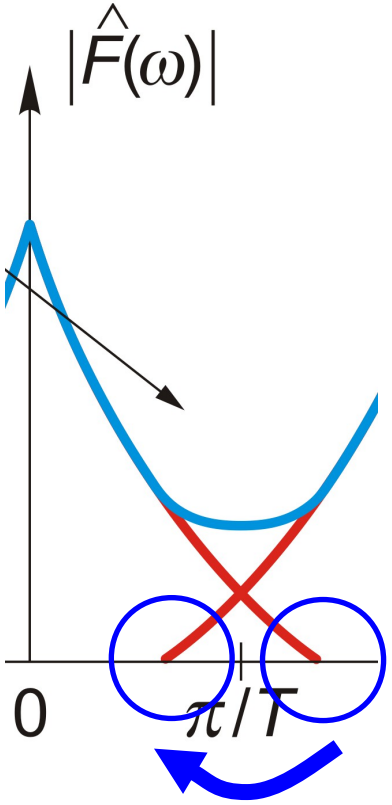
- Akkor nincsen spektrumismétlésből adódó torzítás (átlapolódás, aliasing), ha

$$\omega_{\max} \leq \frac{\pi}{T}$$



Átlapolódás

- Az átlapolódott magas frekvenciájú összetevők alacsony frekvenciájú torzításként jelennek meg.





eredeti kép



hullám (Moiré) mintázat

Nyquist-Shannon tétel

- szükséges minimális mintavételezési (Nyquist) periódus:

$$T_{\min} = \frac{\pi}{\omega_{\max}}$$

- szükséges minimális mintavételezési (Nyquist) körfrekvencia:

$$\omega_{\text{Nyquist}} = \frac{2\pi}{T_{\min}} = 2\omega_{\max}$$

a jelben előforduló legmagasabb frekvencia legalább **kétszeresével** kell a jelet mintavételezni

Következmény (DFT)

- Ha a Nyquist-Shannon feltétel teljesül, vagyis a mintavételezési frekvencia legalább kétszerese a jelben előforduló legnagyobb frekvenciának, akkor $F(\omega)$ másolatai elkülönülnek, nincs spektrumismétlésből adódó torzítás (aliasing), és elég a másolatokból csak eggyel foglalkozni, ha azt T-vel átskálázzuk (megszorozzuk).

Mintapélda (Jupyter munkafüzet)

https://nbviewer.jupyter.org/github/gyulat/Fourier/blob/master/FFT.ipynb

Bázisfüggvények frekvenciája

Mivel a felbontandó jel frekvenciája (1Hz) nem felel meg egyetlen bázisfüggvény (szinusz, koszinusz) frekvenciájának sem (ezek 0.32222222 Hz értékközzel növekszenek), ezért a jel amplitúdója az 1 Hz-hez közeli bázisfüggvények amplitúdóiban jelenik meg.

Nézzük mi történik, ha a jel frekvenciája a mintavételezési frekvenciának pontos többszöröse:

In [27]:
```
num_s = 31 # minták száma
mps = (num_s-1)/tmax # minta/másodperc
print "%.5f minta/másodperc (Hz)" % mps

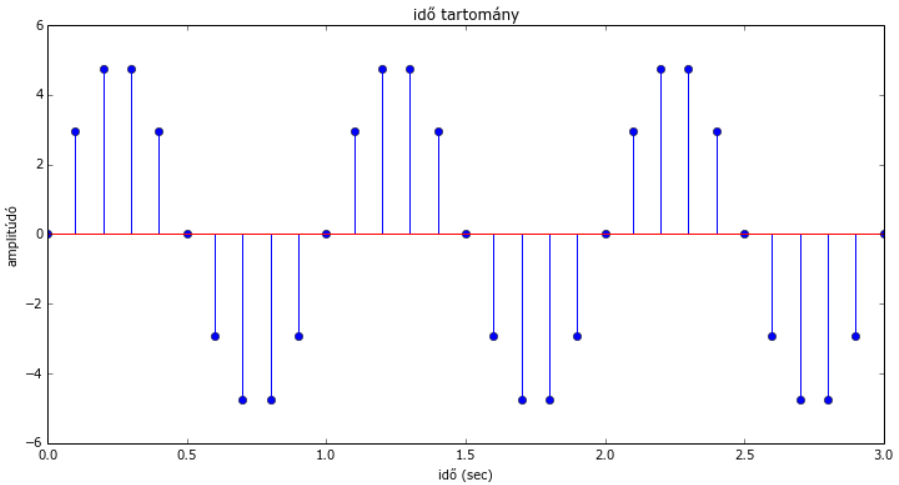
t = np.linspace(0, tmax, num_s)
f = lambda x: ampl * np.sin(f_hz * 2*np.pi*x)
s_f = [f(i) for i in t]  # új mintavételezett f
setup_graph(title='idő tartomány', x_label='idő (sec)', y_label='amplitúdó', fig_size=(12,6))
plt.stem(t, s_f)

# valós FFT számítása
rfft_f = np.fft.rfft(s_f)
rf = [(i*1.0/(num_s-1))*mps for i in range(num_s/2+1)]
rfft_mag = np.abs(rfft_f)/len(rfft_f)

setup_graph(title='FFT eredmény', x_label='frekvencia (Hz)', y_label='magnitúdó', fig_size=(12,6))
plt.stem(rf, rfft_mag)
```

```
10.00000 minta/másodperc (Hz)
```

Out[27]:
```
<Container object of 3 artists>
```

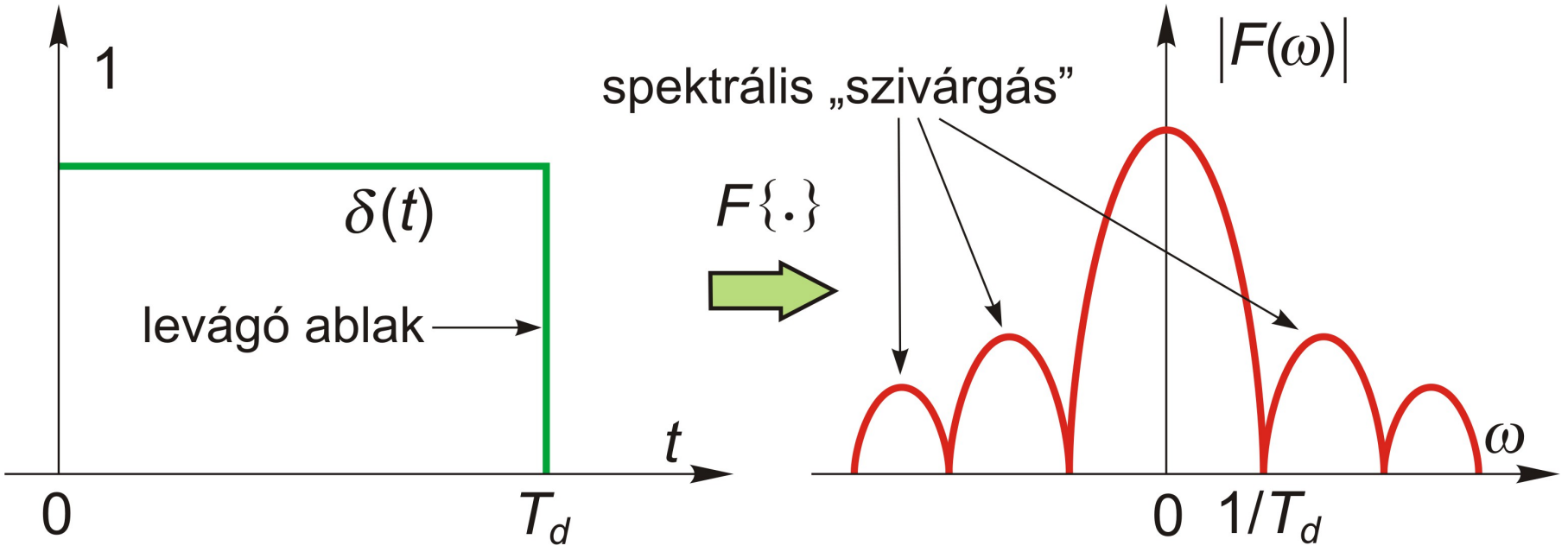


Csak véges sok minta van

- a mintavételezés eredményeként elvileg végtelen sok mintával kellene rendelkeznünk
- a gyakorlatban csak egy korlátozott $t \in (t_{min}, t_{max})$ tartományból vannak adataink
- mi lesz ennek a hatása egy folytonos $f(t)$ jel esetében?

Spektrális szivárgás (leakage)

- levágó ablakkal való szorzás (→konvolúció!)



- a jel energiája ‘átszivárog’ a környező frekvenciákra

DFT transzformált pár

$$F_k = \sum_{n=0}^{N-1} f_n e^{-ink\Delta\omega T} \qquad k = 0, \ldots, N-1$$

$$f_n = \frac{1}{N} \sum_{k=0}^{N-1} F_k e^{ink\Delta\omega T} \qquad n = 0, \ldots, N-1$$

- a DFT műveletigénye N^2 db komplex szorzás és $N(N-1)$ db. komplex összeadás

A gyors Fourier-transzformáció (FFT)

- a DFT műveletigényét csökkenteni lehet
  - bináris ($N=2^k$) számú adatra a műveletigény N^2-ről N $\log_2 N$-re csökkenthető
- Cooley és Tukey (1965)
  - szeizmikus adatok idősoranalízise
- Danielson és Lánczos (1942)
- Gauss (1805)
  - aszteroida pályák interpolációja

Az FFT hatékonysága

- nem csak bináris számú (2^k) adatra működik (ez a több törzsténye zős FFT, és akkor hatékony, ha az adatok száma kis törzsszámok, pl. 2, 3, 5 szorzatára bontható)
- A különbség a DFT és az FFT műveletigénye között óriási lehet!
  - $N=10^9$ adat, 1 GFLOPS processzor (1 ns)
    - FFT-vel 45 másodperc
    - DFT-vel 63 év

Gyorsulásmérő adatok FFT spektruma (Jupyter munkafüzet)

https://nbviewer.jupyter.org/github/gyulat/Fourier/blob/master/FFT_demo.ipynb

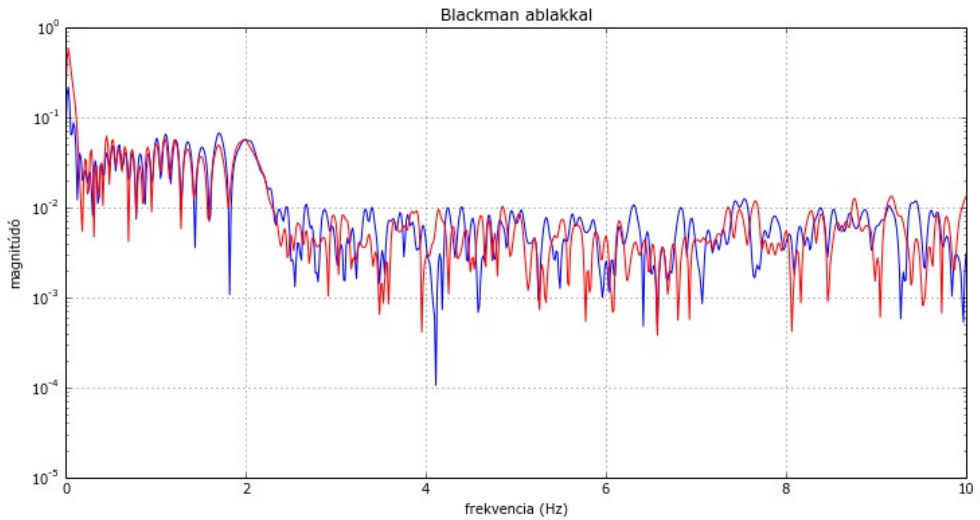
Nézzük meg a Blackman-féle ablakolás hatását:

In [5]:
```
from scipy.signal import blackman
N = len(t)
w = blackman(N)
Ax = np.fft.rfft(ax*w)
Ay = np.fft.rfft(ay*w)

Axm = np.abs(Ax)/len(Ax)
Aym = np.abs(Ay)/len(Ay)

plt.figure()
setup_graph(title='Blackman ablakkal', x_label='frekvencia (Hz)', y_label='magnitúdó', fig_size=(12,6))
plt.semilogy(frq,Axm,'b-')
plt.semilogy(frq,Aym,'r-')
plt.xlim(0,10)
plt.grid()
```

```
<matplotlib.figure.Figure at 0x7fc501a9bed0>
```



Alkalmazások

A Fourier transzformációt igen sok területen alkalmazzák, a teljesség igénye nélkül:

- digitális jelfeldolgozás, képfeldolgozás, szűrés, optikai rendszerek, telekommunikáció
- lineáris rendszerek analízise
- diszkrét konvolúció, dekonvolúció
- differenciálegyenletek
- antennák tervezése (interferometria, SAR, InSAR), képillesztés
- DNS szekvenciák illesztése (MAFFT)
- zárt görbék, felületek elemzése (orvosi képanalízis, karakter felismerés)

További interaktív Jupyter munkafüzetek

Digitális jelfeldolgozás:

http://nbviewer.jupyter.org/github/spatialaudio/digital-signal-processing-lecture/blob/master/index.ipynb

Jelfeldolgozás Pythonnal:

https://github.com/unpingco/Python-for-Signal-Processing/blob/master/README.md