

# Fourierova analiza

Miha Čančula

18. januar 2012



Slika 1: Znani francoski konvolucionar Jean Baptiste Joseph Fourier

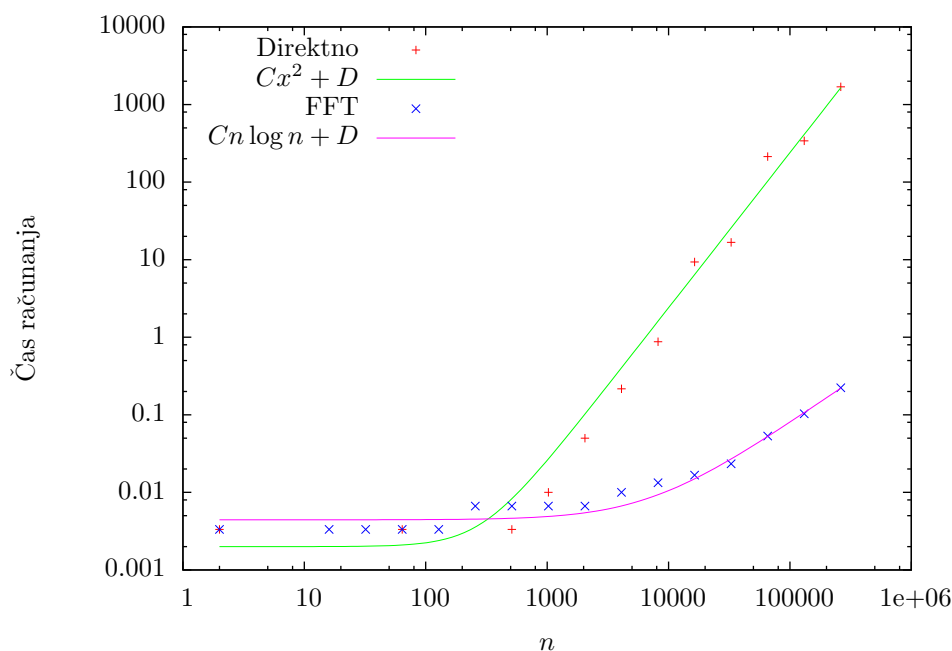
## 1 Konvolucija

Linearno padajočo funkcijo  $f(x) = 1 - x$  sem izvednostil v  $N$  diskretnih točkah, nato pa numerično računal konvolucijo te funkcije samo s sabo. Ta račun sem ponovil pri različnih vrednostih  $N$  in ga vsakih napravil na dva načina: enkrat po definiciji konvolucije, drugič pa z uporabo Fourierove transformacije. Za vse račune sem uporabil program GNU Octave.

### 1.1 Hitrost

Z grafa na sliki 2, se vidi, da je Fourierova transformacija mnogo hitrejša od direktnega računa konvolucije, ta razlika pa se vidi predvsem pri velikem številu točk  $N$ .

Na grafu sta narisani še ujemanju s krivuljo, ki narašča kot  $\mathcal{O}(\backslash^\epsilon)$  in krivuljo  $\mathcal{O}(\backslash \log \backslash)$ . V obeh primerih sem funkcijama dodal konstanto, saj že sam program Octave potrebuje nekaj časa za zagon. Ujemanje je dobro za obe napovedi, predvsem pri večjih  $N$ , ko lahko zanemarimo prispevke nižjih redov.



Slika 2: Čas, potreben za izračun konvolucije treh linearno padajočih funkcij po obeh algoritmih

## 1.2 Natančnost

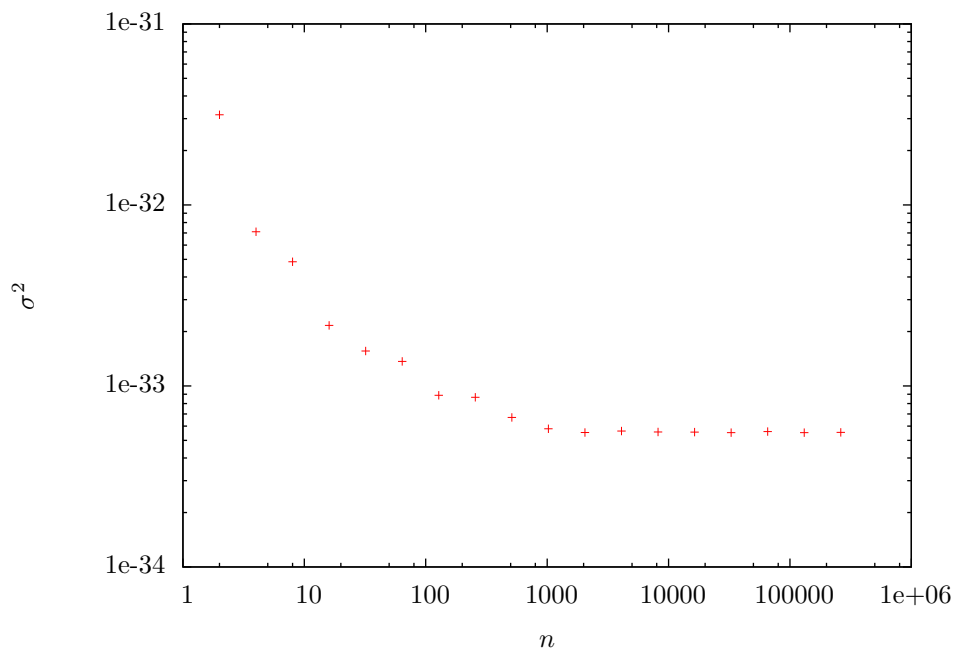
Preveril sem tudi, kako natančna je metoda s Fourierovo transformacijo. Tako direktna kot FFT metoda ne delata nobenih približkov, edini vir napake je končna natančnost računalniškega zapisa, na katerega pa sta lahko metodi različno občutljivi. Kot mero za napako metode sem izbral

$$\sigma^2 = \sum_{i=0}^{kN-1} (f_i^F - f_i^D)^2 \quad (1)$$

kjer so  $f_i^F$  in  $f_i^D$  koeficienti, izračunani s Fourierovo oz. direktno metodo. Če sem začel z diskretizacijo na  $N$  točk in izračunal konvolucijo  $k$ -tih enakih funkcij, ima končni izraz  $kN$  koeficientov.

## 2 Dekonvolucija signala

Tokrat je bila naloga obratna, poiskati izviren signal ob poznavanju izhodnega signala in prehodne funkcije. Prehodna funkcija  $G(t)$  pa je imela en neznan parameter  $\beta$ , ki sem ga določil tako, da je bil izvorni signal čim lepši. Lepoto signala pa sem definiral na različne načine.



Slika 3: Napaka izračuna s Fourierovo transformacijo

## 2.1 Čim manjše spremembe

Ena možna izbira je, da iščemo signal, ki se čim počasneje spreminja. Naravna izbira je takšna, da je izraz

$$\sum_{i=0}^{N-1} |s_i - s_{i-1}|^2 \quad (2)$$

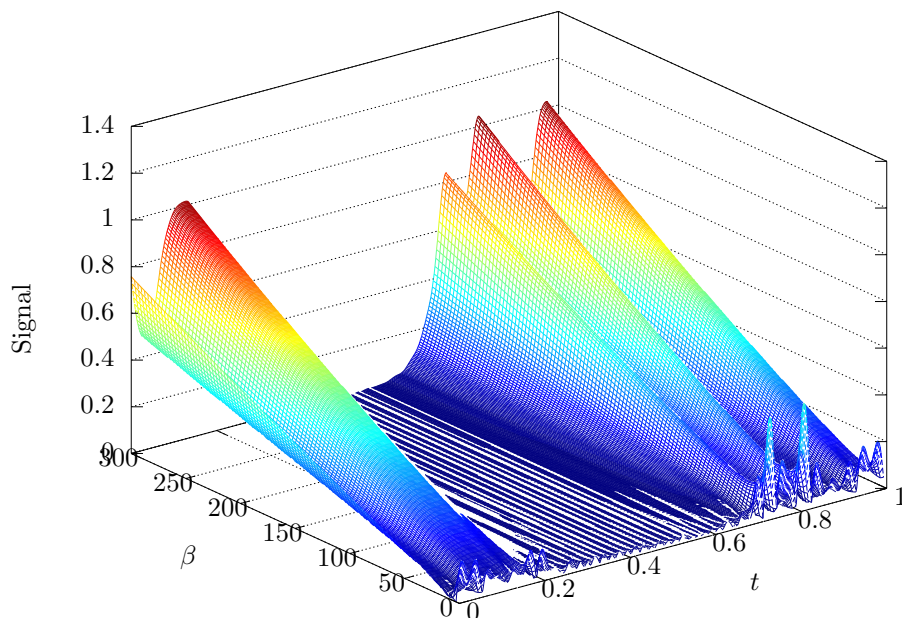
čim manjši. Izkaže se, da je to pri vrednosti  $\beta = 34.852$ , vhodni signal pa je tedaj tak kot na sliki 5.

## 2.2 Čim nižje frekvence

Lepoto signala pa lahko ocenimo že po frekvenčnem spektru, tako da si želimo čim manj visokih frekvenc. V ta namen sem minimiziral izraz

$$\sum_{i=-N/2}^{N/2-1} i \cdot |S_i| \quad (3)$$

kjer so  $S_i$  komponentne Fourierove transformacije signala  $s_i$ . Ta kriterij je v bistvu zelo podoben prejšnjemu, zato je tudi optimalna vrednost  $\beta$  blizu, in sicer  $\beta = 36.475$ .



Slika 4: Izračunani izvirni signal z upoštevanjem različnih eksponentov  $\beta$

### 2.3 Čim manjša amplituda

Za primerjavo sem iskal tudi signal z najmanjšo amplitudo. Če privzamemo, da je signal približno sinusen, lahko njegovo amplitudo ocenimo v vsaki točki.

$$s(t) = A \sin \omega t \quad (4)$$

$$\dot{s}(t) = A\omega \cos \omega t \quad (5)$$

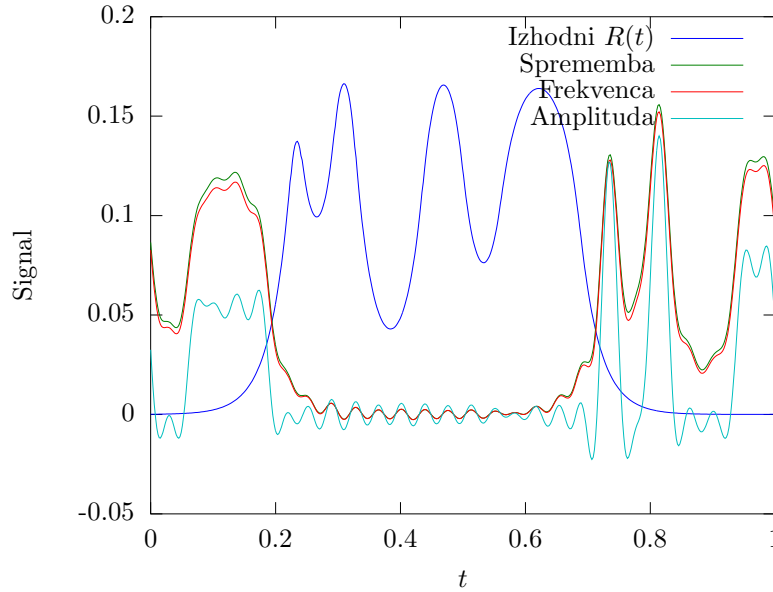
$$A^2 = \omega^2 s^2 + (\dot{s})^2 \quad (6)$$

Odvod  $\dot{s}$  lahko približamo s končno diferenco, frekvenčo  $\omega$  pa kot po absolutni vrednosti največjo komponentno Fourierove transformiranke. Seveda naš signal ni povsem sinusen, zato se amplituda spreminja s časom,  $A = A(t)$ . Podobno kot v prvem primeru sem na koncu minimiziral povprečen kvadrat amplitude.

## 3 Filtriranje

### 3.1 Celoten interval

Iz frekvenčnega spektra originalnih signalov na sliki 6 lahko vidimo, da sta oba vrhova v spektru signala iz datoteke `val2.dat` ostra, medtem ko so vrhovi v



Slika 5: Vhodni in izhodni signal pri  $\beta = 34.852$

spektru `val3.dat` razširjeni. Iz tega lahko sklepamo, da smo pri merjenju prvega signala ujeli celo število period, pri merjenju drugega pa ne.

Oba spektra lahko približamo z uporabo okenske funkcije. Najprej sem poskusil z enostavno sinusno funkcijo

$$w(n) = \sin\left(\frac{n\pi}{N-1}\right) \quad (7)$$

Ta funkcija se na robovih dotakne 0, ima pa neničelen odvod. Frekvenčni spekter obeh signalov, pomnoženih s takšnim oknom, je na sliki 7.

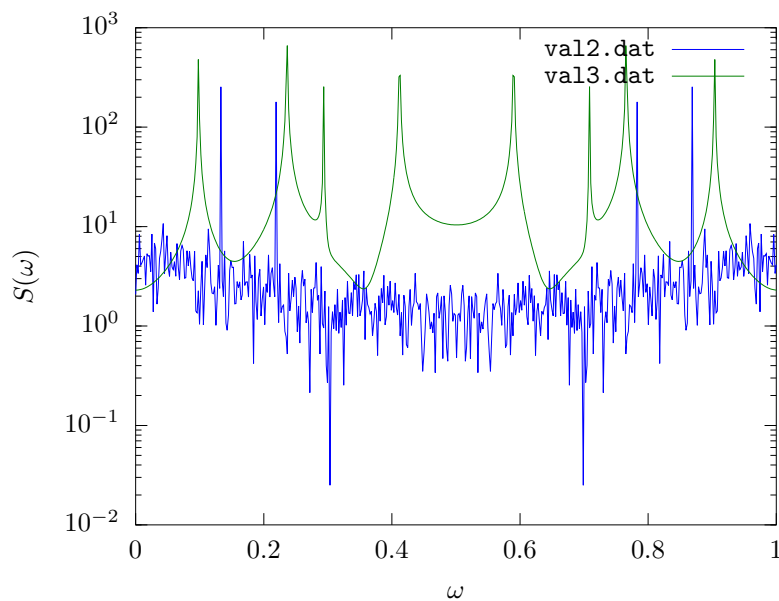
Hitro vidimo, da so se vrhovi prej ostrega signala razširili, vrhovi signala s širokimi vrhovi pa zožili. Med signaloma je še vedno jasna razlika, vrhovi so po širini sedaj bolj podobni kot prej, ne pa še enaki.

Nato sem preizkusil Hammingovo okensko funkcijo. Za razliko od prejšnje ta niti na robovih ne pade čisto na 0, ima pa tam odvod enak 0, zato bo tako obdelan signal lahko zvezen in periodičen. Koeficiente okna dobimo s formulo

$$w(n) = 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) \quad (8)$$

Če okensko funkcijo uporabimo v Fourierovem prostoru, je predznak drugega člena odvisen od izbire intervala za  $n$ . V tem primeru pa smo jo uporabili v časovnem prostoru, tako da  $n \in [0, N-1]$  in je drugi člen s pozitivnim predznakom. Želimo namreč, da je vrednost okna na robovih intervala enaka 0, na sredini pa 1. Rezultat uporabe zgornje funkcije je na sliki 8.

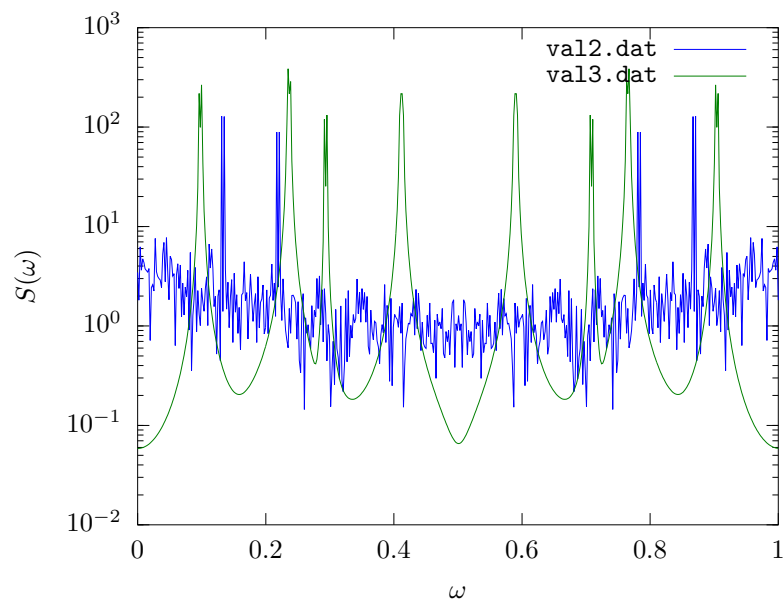
Širini vrhov pa sta skoraj enaki. Na sliki je sicer vidna razlike med njimi, ampak le v področju med vrhovi, kjer je spekter sedaj mnogo nižji kot pri



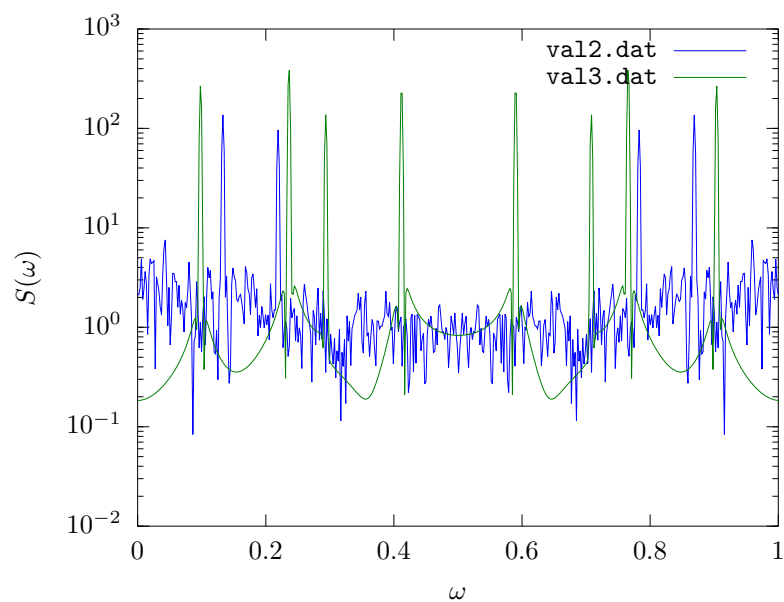
Slika 6: Frekvenčni spekter obeh signalov

izvirnem signalu. Opazimo lahko tudi, da v signalu `val3.dat` tik ob najvišjih vrhovih sedaj dobimo skoke (angl. side-lobes), čeprav je Hammingova funkcija optimizirana tako, da čimbolj zmanjša višino prvega skoka.

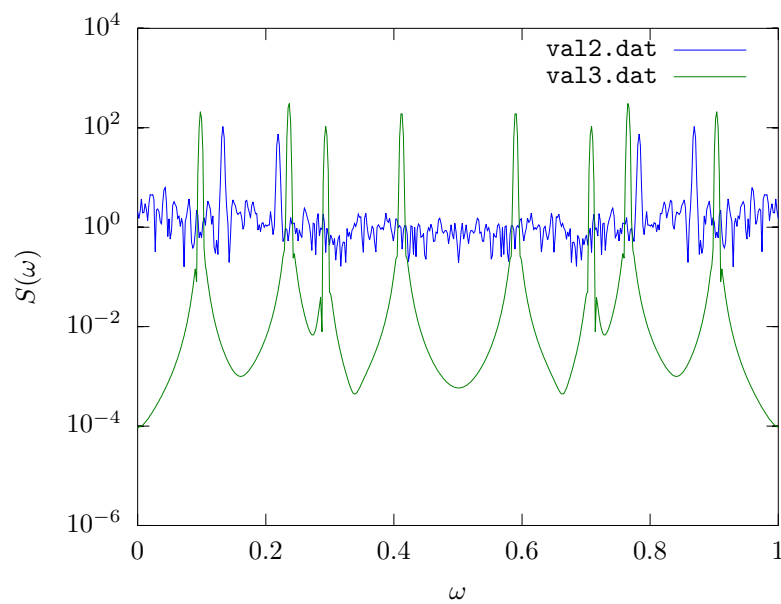
Nazadnje sem uporabil se Blackmanovo funkcijo, ki ima na robovih tako vrednost kot odvod enaka 0. Rezultati so na sliki 9.



Slika 7: Frekvenčni spekter obeh signalov, pomnoženih s sinusno okensko funkcijo



Slika 8: Frekvenčni spekter obeh signalov, pomnoženih s Hammingovo okensko funkcijo

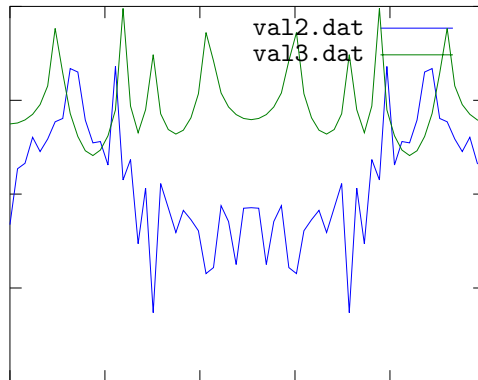


Slika 9: Frekvenčni spekter obeh signalov, pomnoženih z Blackmanovo okensko funkcijo

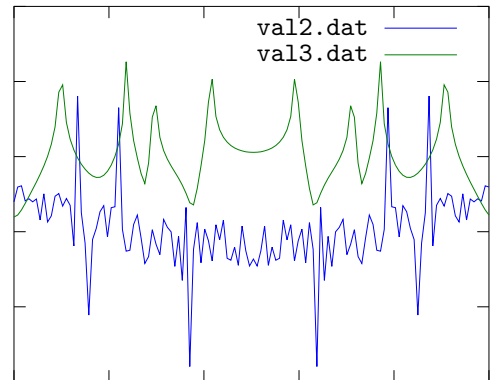


### 3.2 Krajši intervali

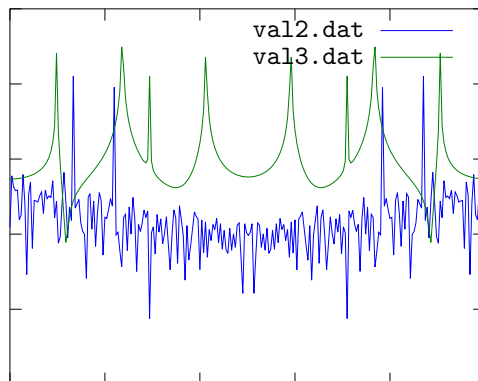
Ogledal sem si tudi, kaj se zgodi, če obravnavamo le manjše število točk. Iste izračune kot prej sem ponovil za celoten signal (512 točk) in za krajše podintervale (64, 128 in 256 točk). Tokrat sem prikazal le originalen signal in pa signal, dobljen s Hammingovo okensko funkcijo. Spektri so na slikah 10 in 11.



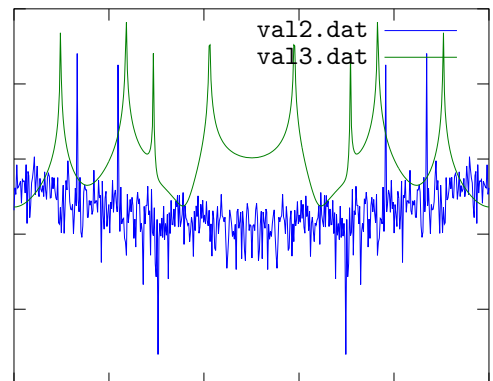
(a)  $M = 64$



(b)  $M = 128$



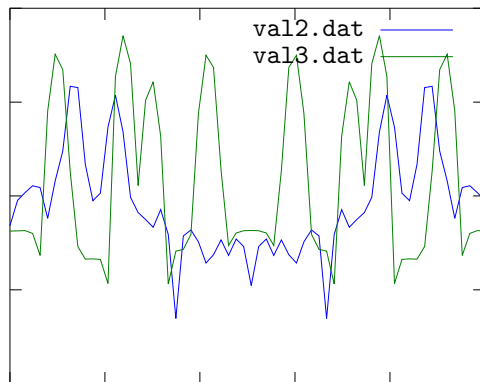
(c)  $M = 256$



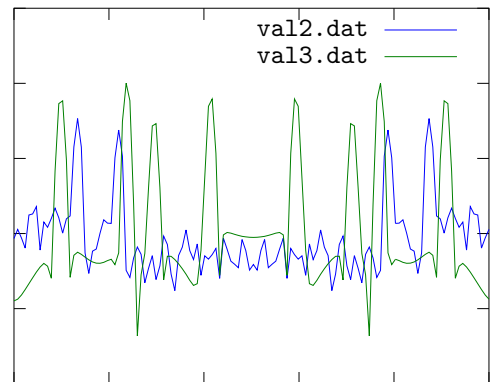
(d)  $M = 512$

Slika 10: Frekvenčni spektri originalnih signalov pri različnih dolžinah analiziranega intervala

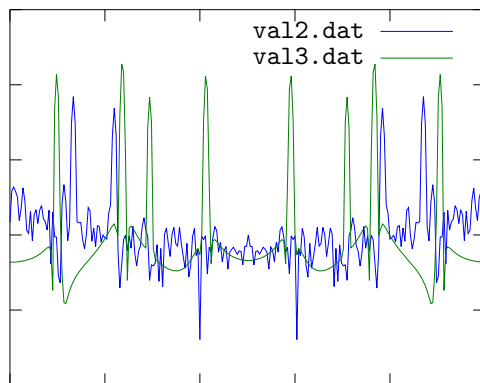
Okenska funkcija že pri manjšem številu točk poskrbi, da so vrhovi bolj razločni tudi na logaritemski sliki, poleg tega pa so tudi po obliki bolj podobni med seboj. Njen vpliv dobro viden pri  $M = 64$ , na sliki 10(a) sta vrhova modre funkcije vidno različno široka, medtem ko sta njuni širin na sliki 11(a) približno enaki.



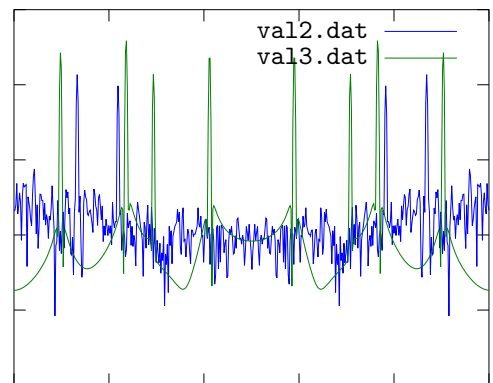
(a)  $M = 64$



(b)  $M = 128$



(c)  $M = 256$



(d)  $M = 512$

Slika 11: Frekvenčni spektri signalov, pomnoženih s Hammingovo okensko funkcijo, pri različnih dolžinah analiziranega intervala