Obrada informacija

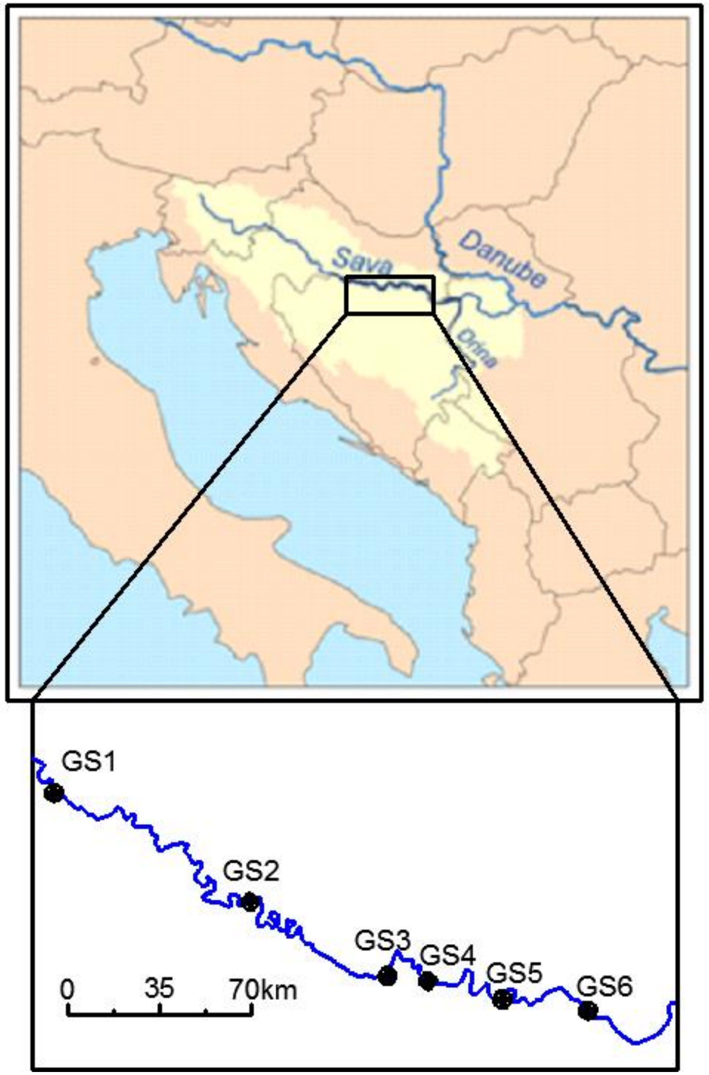
Doc. dr. sc. Ana Sović Kržić

Sadržaj

- Rijeka Sava
- Statistički podaci mjernih postaja
- Fourierova transformacija
- Fourierova transformacija na vremenskom prozoru
- Valićna transformacija

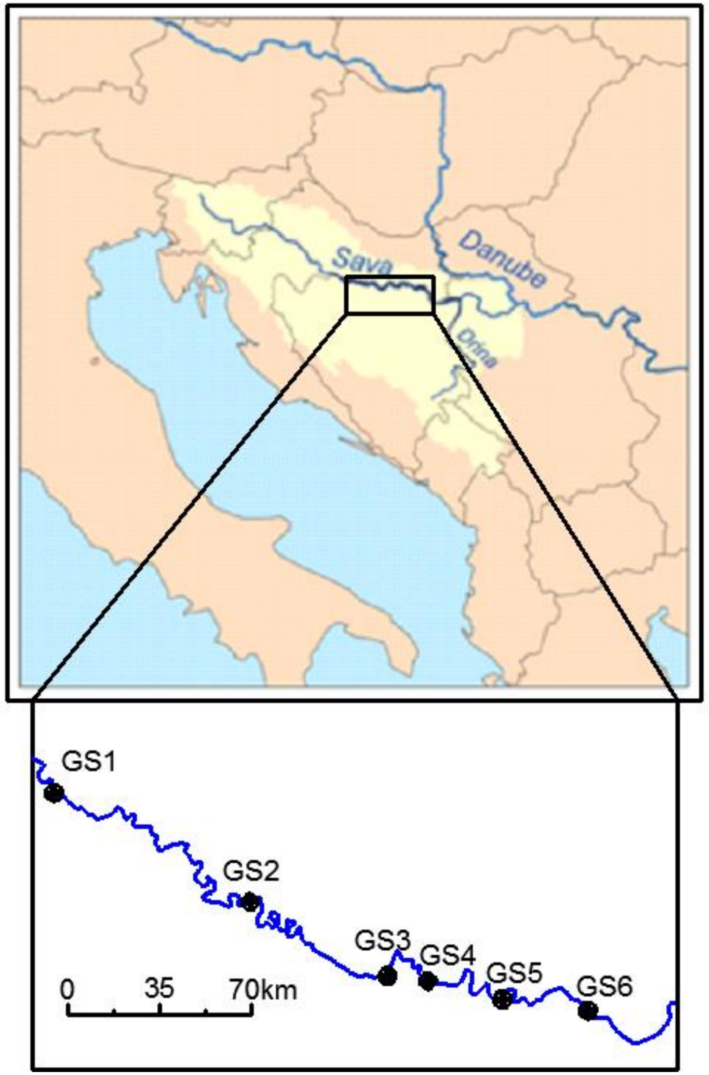
Rijeka Sava

- Rijeka Sava:
  - Duljina: 946 km (510 km u Hrvatskoj)
  - Područje sliva: 97.713 km^2, od toga 25.373 km^2 (26%) pripada Hrvatskoj
  - Širina korita: 100 m (Zagreb), 700 m (Šabac), 290 m (ušće)
  - Gospodarski značaj sliva:
    - Termoelektrane i nuklearne elektrane
    - Javna vodoopskrba
    - Poljoprivreda (navodnjavanje, ribogojilišta)
    - Industrija
    - Plovna od Siska



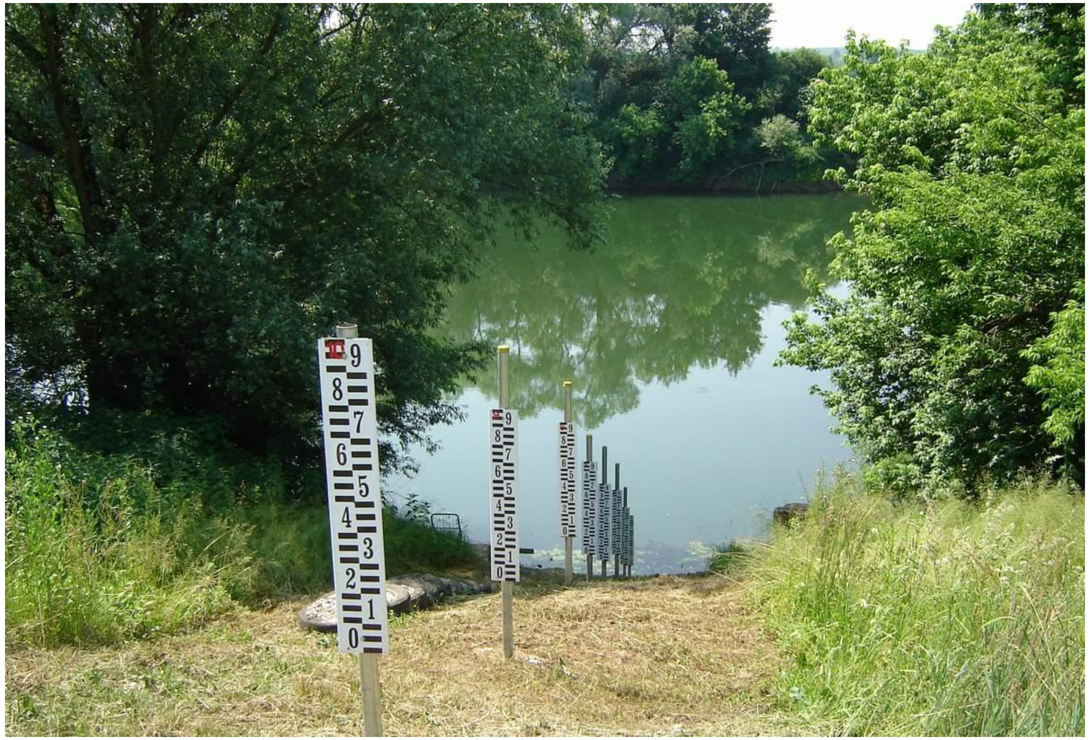
Problem

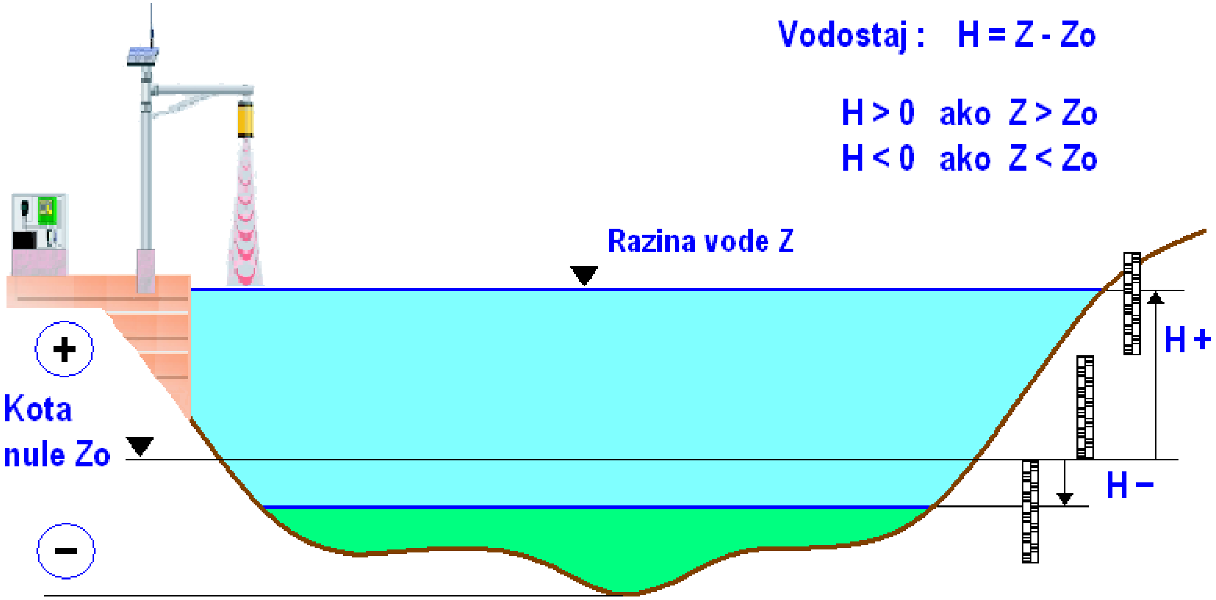
- Onečišćenje
- Erozije tla
- Poplave:
  - 1469. prva zabilježena poplava
  - 1651. se razlio potok Medveščak, 52 žitelja se utopilo
  - 1880. voda u Petrinjskoj visoka preko metar + potres 9.10.1880.
  - 26.10. 1964. – vodostaj 514 cm, najviši ikada, pucanje nasipa
  - gradnja oteretnog kanala “Odra”, ustave Prevlaka i oteretnog kanala “Lonja-Strug”
  - 2014. Gunja



Vodostaj

- Razina vode u moru, rijeci ili jezeru u cm
- Mjeri se pomoću vodomjerne letve





Mjerenje vodostaja

- hidrološke stanice raspoređene duž vodotoka → na svakom vodokaznom profilu postavljaju se vodokazne mjerne letve
- mjerne letve u koritu i na obalama vodotoka → raspon visina pokriva raspon promjena vodostaja
- referentna točka mjerenja vodostaja na kojoj se uzima da je očitanje vodostaja jednako nuli (kota nule)
- pozicija kote nule → na visini koja odgovara najnižem vodostaju na razmatranoj stanici (kako očitanja ne bi bila negativna)
- apsolutna nadmorska visina kote nule → određuje se geodetskim mjerenjem (moguća usporedba mjerenja vodostaja sa stanjem vodostaja na susjednim hidrološkim stanicama)
- u praksi se javljaju negativni vodostaji: korito vodotoka se proticanjem vode stalno mijenja, može doći do produbljavanja

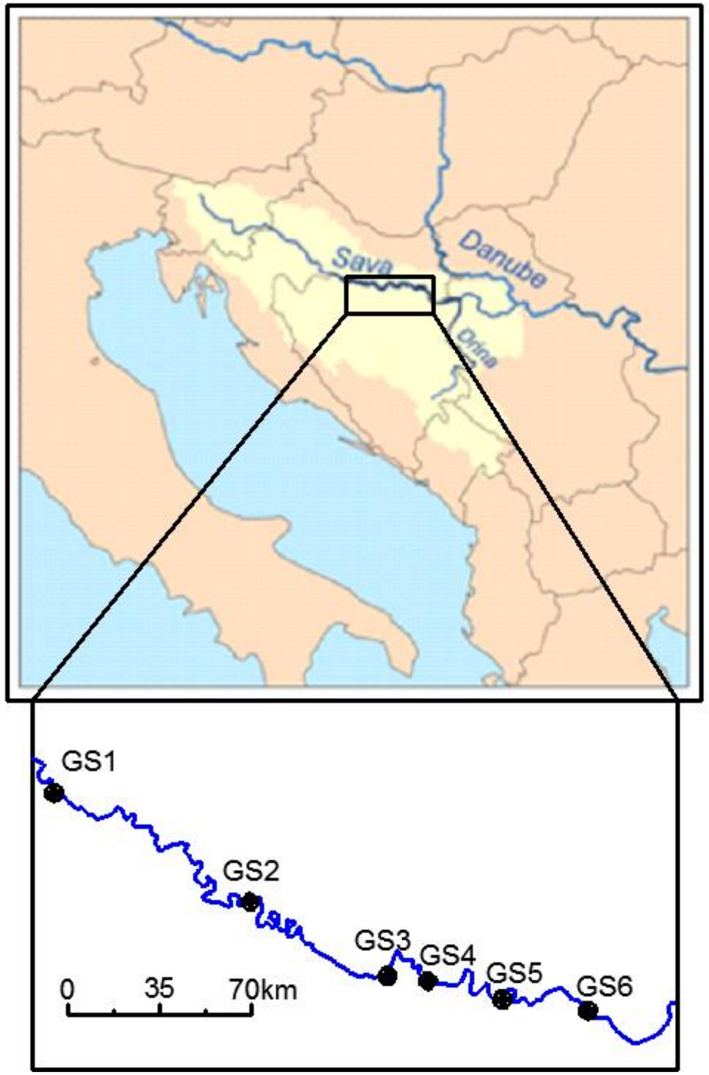
Mjerenje

- Podaci Državnog hidrometeoro-loškog zavoda RH
- Mjerenje u 07:00h
- * Promjena stanja u odnosu na vodostaje prethodnog dana u 07:00

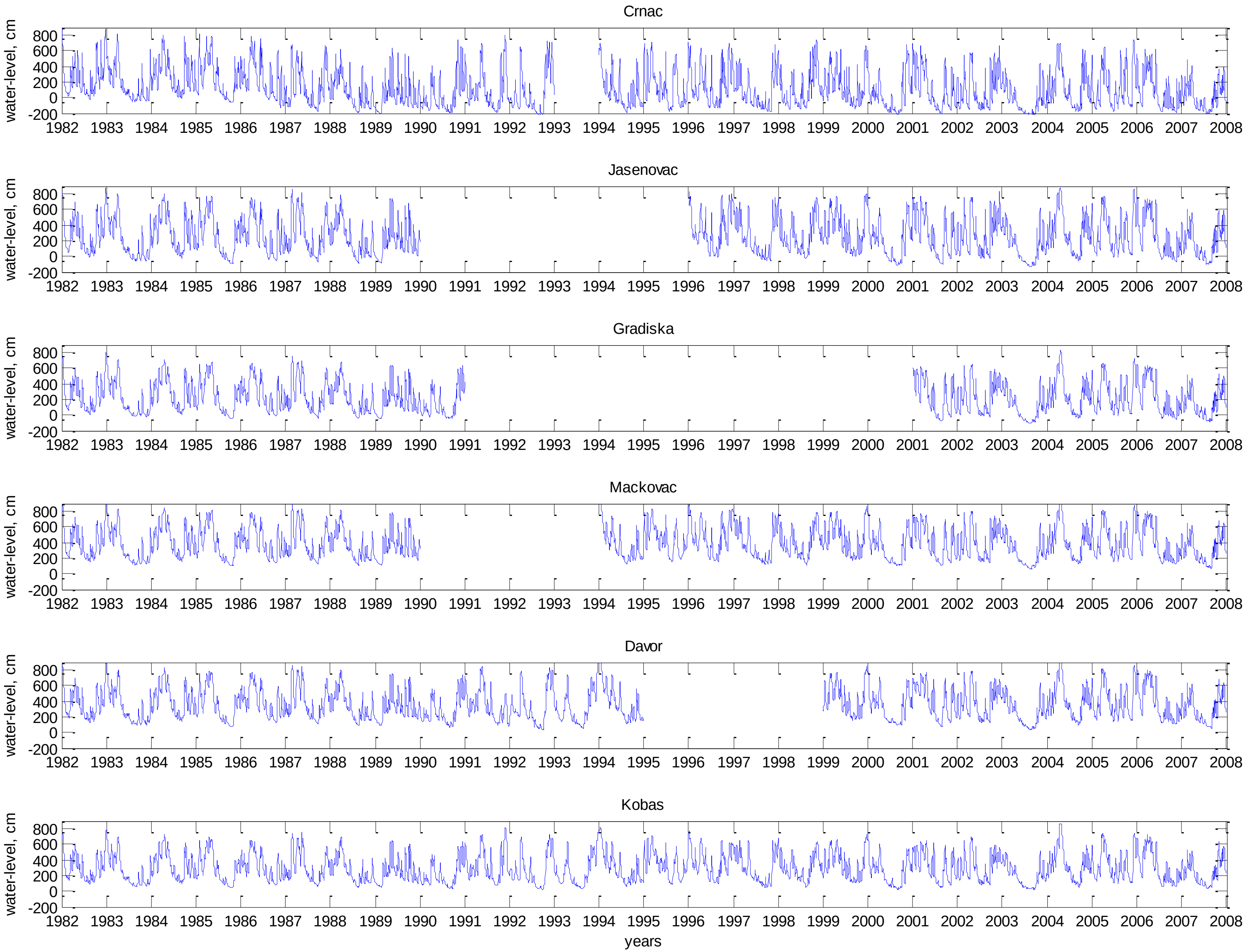
Vodotok	Postaja	Vodostaj (cm)	Tendencija* (cm) (promjena stanja)	Protok (m^3/s)	Zadnji vodostaj (cm)	Zadnji protok (m^3/s)	Sat zadnjeg podatka
DUNAV	BATINA	214	-6	2499	211	2480	13
	ALJMAŠ	276	-8	3140	273	3120	13
	DALJ	430	-6	3205	428	3190	13
	VUKOVAR	242	-4	3200	240	3184	13
	ILOK MOST	261	+0	3187	263	3203	13
DRAVA	BOTOVO	193	+26	725.4	198	740.5	13
	TEREZINO POLJE	---	---	---	---	---	---
	DONJI MIHOLJAC C.S.	116	-43	672.1	115	669.8	13
	BELIŠĆE	241	-30	717.6	232	694.8	13
	OSIJEK	99	-14	---	88	---	13
MURA	MURSKO SREDIŠĆE	244	+28	272.9	252	290.3	13
	GORIČAN	185	+5	227.9	210	271.9	13
SAVA	DRENJE BRDOVEČKO	8	+144	551.9	26	608.1	13
	ZAGREB	-11	+208	667.6	-19	646.4	13
	CRNAC	41	-43	---	65	---	13
	JASENOVAC	197	-52	605.9	188	589.6	13
	MAČKOVAC USTAVA	326	-41	651.2	316	626.9	13
	DAVOR C.S.	320	-37	736.7	308	705.9	13
	SLAVONSKI BROD	185	-26	777.8	177	753.5	13
	SLAVONSKI ŠAMAC	-32	-21	---	-39	---	13
	ŽUPANJA STEPENICA	193	-13	868.0	186	849.0	13

Promatrane mjerne postaje

- GS1 Crnac
- GS2 Jasenovac
- GS3 Stara Gradiška
- GS4 Mačkovac
- GS5 Davor
- GS6 Kobaš

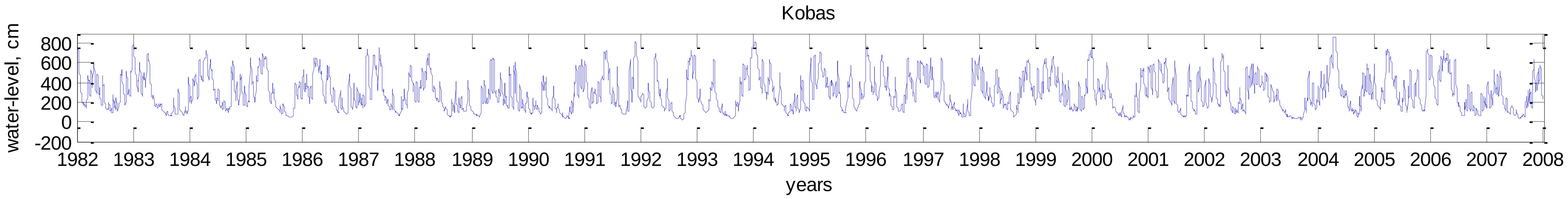


	Godine za koje nedostaju podaci	Udaljenost od ušća [km]	Udaljenost prehodne stanice [km]	Vertikalna referentna točna mjerne stanice [m]
GS 1 - Crnac	1993	575	74.5	91.338
GS 2 - Jasenovac	1990-1995	500.5	47.1	82.820
GS 3 - Gradiška	1991-2000	453.4	14.4	85.467
GS 4 - Mačkovac	1990-1993	439	21	83.645
GS 5 - Davor	1995-1998	418	27.5	82.784
GS 6 - Kobaš	-	390.5	-	82.750


Crnac
Jasenovac
Gradiska
Mackovac
Davor
Kobas
water-level, cm
800
600
400
200
0
-200
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
years

Kobaš

- Minimalan vodostaj: ljeti
- Maksimalni vodostaji: proljeće, jesen
- Amplitude niže – sušna godina (1988-1991)
- Amplitude veće – vlažna godina (1994)



Diskretni signal

- Prvih 10 mjerenja na stanici Kobaš:

$$x(n) = \{\underline{661}, 674, 685, 693, 702, 712, 726, 735, 740, 736\}$$

- Diskretni signal, $x\colon \mathbb{Z} \to \mathbb{R}$, duljine *N*

$$x(n) = \{\underline{x(0)}, x(1), x(2), x(3), \ \ldots, x(N-1)\}$$

Srednja vrijednost

$$\overline{x} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)$$

- Primjer za Kobaš: $\mathrm{x}(n) = \{\underline{661}, 674, 685, 693, 702, 712, 726, 735, 740, 736\}$

$$\overline{x} = \frac{661+674+685+693+702+712+726+735+740+736}{10} = 706.4$$

Standardna devijacija

$$\sigma = \sqrt{\frac{1}{N}\sum_{n=0}^{N-1}\left(\overline{x} - x(n)\right)^2}$$

- Primjer za Kobaš: $x(n) = \{\underline{661}, 674, 685, 693, 702, 712, 726, 735, 740, 736\}$

$$\sigma = \sqrt{\frac{(706.4 - 661)^2 + (706.4 - 674)^2 + \cdots + (706.4 - 736)^2}{10}} = 26.47$$

- $\overline{x} - 26.47 = 679.93$, $\overline{x} = 706.4$, $\overline{x} + 26.47 = 732.87$ → 6 od 10 uzoraka je unutar intervala $\pm\sigma$

Medijan

- Vrijednost središnjeg podatka → podatke poredane po veličini dijeli u dva jednako brojna dijela
  - Broj podataka NEPARAN = medijan je središnji podatak
  - Broj podataka PARAN = medijan je aritmetička sredina dva središnja podatka
- U sortiranom nizu 50% uzoraka ima vrijednost manju ili jednaku medijanu i 50% uzoraka ima vrijednost veću ili jednaku medijanu
- Mali utjecaj graničnih podataka (outliers)

- Primjer za Kobaš: $x(n) = \{\underline{661}$, 674, 685, 693, 702, 712, 726, 735, 740, 736}
- Sortirano: 661, 674, 685, 693, 702, 712, 726, 735, 736, 740
- Parna duljina → medijan = $\frac{702+712}{2} = 707$

Minimalna i maksimalna vrijednost

- Primjer za Kobaš: $x(n) = \{\underline{661}, 674, 685, 693, 702, 712, 726, 735, 740, 736\}$

- $\min\big(f(n)\big) = 661$ → najniži vodostaj
- $\max\big(f(n)\big) = 740$ → najviši vodostaj

	Srednja vrijednost [cm]	Standardna devijacija [cm]	Min [cm]	Max [cm]
GS 1 - Crnac	115	235	-241	895
GS 2 - Jasenovac	256	239	-137	884
GS 3 - Gradiška	223	209	-106	830
GS 4 - Mačkovac	389	203	64	976
GS 5 - Davor	358	204	35	992
GS 6 - Kobaš	294	183	17	878

Obrada signala u Pythonu

- Python osmislio Nizozemac Guido van Rossum
- Naziv po grupi i TV emisiji Monty Python
- https://www.python.org/downloads/
- razvojno okruženje za Python – IDLE (Integrated DeveLopment Environment, prezime komičara iz grupe Monty Python koji se zove Eric Idle)
- IDLE se instalira zajedno s Python

Moduli (1)

- Mnoge korisne funkcije nisu sastavni dio jezgre Pythona, ali se instaliraju zajedno s Pythonom ili se mogu skinuti s interneta – to su MODULI
- Unutar jednog modula se nalaze funkcije koje su međusobno srodne
- Funkcije iz pojedinog modula moramo najaviti prije korištenja:

```
import ime_modula
```

- Funkcije se pozivaju:

```
ime_modula.ime_funkcije(ulaz)
```

```
>>> x = 25
>>> y = sqrt(x)

Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    y = sqrt(x)
NameError: name 'sqrt' is not defined
>>> import math
>>> y = sqrt(x)

Traceback (most recent call last):
  File "<pyshell#4>", line 1, in <module>
    y = sqrt(x)
NameError: name 'sqrt' is not defined
>>> y = math.sqrt(x)
>>> y
5.0
```

Moduli (3)

- Ako želimo koristiti samo jednu funkciju iz nekog modula to najavljujemo:

```
from ime_modula import ime_funkcije1,
     ime_funkcije2
```

- Funkciju pozivamo:

```
ime_funkcije1(ulaz)
```

```
>>> from math import sqrt
>>> x = 25
>>> y = sqrt(x)
>>> y
5.0
```

Moduli (4)

- Ako želimo uvesti sve funkcije, bez pisanja naziva modula:

```
from ime_modula import *
```

- Funkciju pozivamo:

```
ime_funkcije1(ulaz)
```

```
>>> from math import *
>>> x = 25
>>> y = sqrt(x)
>>> y
5.0
>>> z = y * pi
>>> z
15.707963267948966
```

Izmjena imena modula i njegovih funkcija

- U programu se već koristi isto ime za nešto drugo
- Drugi modul koji se koristi ima iste nazive funkcija
- Skratiti dugačko ime modula

```
import ime_modula as novo_ime_modula
```

- Funkciju pozivamo:

```
novo_ime_modula.ime_funkcije(ulaz)
```

```
>>> import math as m
>>> x = 25
>>> y = sqrt(x)
Traceback (most recent call last):
  File "<pyshell#2>", line 1, in <module>
    y = sqrt(x)
NameError: name 'sqrt' is not defined
>>> y = m.sqrt(x)
>>> y
5.0
```

Pomoć

- Ako nas zanima popis svih funkcija koje se nalaze u nekom modulu:

    `help('ime_modula')`

```
>>> help('math')
Help on built-in module math:

NAME
    math

FILE
    (built-in)

DESCRIPTION
    This module is always available.  It provides access to the
    mathematical functions defined by the C standard.

FUNCTIONS
    acos(...)
        acos(x)

        Return the arc cosine (measured in radians) of x.

    acosh(...)
        acosh(x)

        Return the hyperbolic arc cosine (measured in radians) of x.

    asin(...)
        asin(x)

        Return the arc sine (measured in radians) of x.
```

Preuzimanje modula s interneta

- U terminalu računala pozvati preuzimanje i instalaciju

```
pip install numpy
pip install matplotlib
pip install scipy
pip install PyWavelets
```

NumPy

- NumPy (**Numerical Python**)
- standard za rad s numeričkim podacima u Pythonu
- NumPy biblioteka radi s multidimenzionalnim nizovima i matricama te matematičke operacije na nizovima
- intenzivno se koristi u kombinaciji s Pandas, SciPy, Matplotlib, scikit-learn, scikit-image i drugim paketima iz područja znanosti o podacima

```
import numpy as np
```

CREATING ARRAYS

`np.array([1,2,3])` - One dimensional array

`np.array([(1,2,3),(4,5,6)])` - Two dimensional array

`np.zeros(3)` - 1D array of length `3` all values `0`

`np.ones((3,4))` - `3x4` array with all values `1`

`np.eye(5)` - `5x5` array of `0` with `1` on diagonal (Identity matrix)

`np.linspace(0,100,6)` - Array of `6` evenly divided values from `0` to `100`

`np.arange(0,10,3)` - Array of values from `0` to less than `10` with step `3` (eg `[0,3,6,9]`)

`np.full((2,3),8)` - `2x3` array with all values `8`

INSPECTING PROPERTIES

`arr.size` - Returns number of elements in `arr`

`arr.shape` - Returns dimensions of `arr` (rows, columns)

`arr.dtype` - Returns type of elements in `arr`

`arr.astype(dtype)` - Convert `arr` elements to type `dtype`

`arr.tolist()` - Convert `arr` to a Python list

`np.info(np.eye)` - View documentation for `np.eye`

STATISTICS

`np.mean(arr,axis=0)` - Returns mean along specific axis

`arr.sum()` - Returns sum of `arr`

`arr.min()` - Returns minimum value of `arr`

`arr.max(axis=0)` - Returns maximum value of specific axis

`np.var(arr)` - Returns the variance of array

`np.std(arr,axis=1)` - Returns the standard deviation of specific axis

`arr.corrcoef()` - Returns correlation coefficient of array

SCiPy

- kolekcija numeričkih algoritama, koji uključuju obradu signala, optimizacije, statistiku, linearnu algebru, numeričku integraciju i drugo.

```
from scipy.fft import fft, ifft
from scipy import io, signal
```

Filter design

bilinear(b, a[, fs])	Return a digital IIR filter from an analog one using a bilinear transform.
bilinear_zpk(z, p, k, fs)	Return a digital IIR filter from an analog one using a bilinear transform.
findfreqs(num, den, N[, kind])	Find array of frequencies for computing the response of an analog filter.
firls(numtaps, bands, desired[, weight, nyq, fs])	FIR filter design using least-squares error minimization.
firwin(numtaps, cutoff[, width, window, ...])	FIR filter design using the window method.
firwin2(numtaps, freq, gain[, nfreqs, ...])	FIR filter design using the window method.
freqs(b, a[, worN, plot])	Compute frequency response of analog filter.
freqs_zpk(z, p, k[, worN])	Compute frequency response of analog filter.
freqz(b[, a, worN, whole, plot, fs, ...])	Compute the frequency response of a digital filter.
freqz_zpk(z, p, k[, worN, whole, fs])	Compute the frequency response of a digital filter in ZPK form.

Spectral analysis

periodogram(x[, fs, window, nfft, detrend, ...])	Estimate power spectral density using a periodogram.
welch(x[, fs, window, nperseg, noverlap, ...])	Estimate power spectral density using Welch's method.
csd(x, y[, fs, window, nperseg, noverlap, ...])	Estimate the cross power spectral density, Pxy, using Welch's method.
coherence(x, y[, fs, window, nperseg, ...])	Estimate the magnitude squared coherence estimate, Cxy, of discrete-time signals X and Y using Welch's method.
spectrogram(x[, fs, window, nperseg, ...])	Compute a spectrogram with consecutive Fourier transforms.
lombscargle(x, y, freqs)	Computes the Lomb-Scargle periodogram.
vectorstrength(events, period)	Determine the vector strength of the events corresponding to the given period.
stft(x[, fs, window, nperseg, noverlap, ...])	Compute the Short Time Fourier Transform (STFT).
istft(Zxx[, fs, window, nperseg, noverlap, ...])	Perform the inverse Short Time Fourier transform (iSTFT).
check_COLA(window, nperseg, noverlap[, tol])	Check whether the Constant OverLap Add (COLA) constraint is met
check_NOLA(window, nperseg, noverlap[, tol])	Check whether the Nonzero Overlap Add (NOLA) constraint is met

MatPlotLib

- Biblioteka za kreiranje statičkih, animiranih i interaktivnih vizualizacija (grafovi, slike, polja, statistike, anotacije)

```
import matplotlib.pyplot as plt
```

`figure`	Create a new figure, or activate an existing figure.
`fill`	Plot filled polygons.
`fill_between`	Fill the area between two horizontal curves.
`fill_betweenx`	Fill the area between two vertical curves.
`findobj`	Find artist objects.
`gca`	Get the current axes, creating one if necessary.

`stem`	Create a stem plot.
`step`	Make a step plot.
`streamplot`	Draw streamlines of a vector flow.
`subplot`	Add a subplot to the current figure.
`subplot2grid`	Create a subplot at a specific location inside a regular grid.
`subplot_mosaic`	Build a layout of Axes based on ASCII art or nested lists.
`subplot_tool`	Launch a subplot tool window for a figure.
`subplots`	Create a figure and a set of subplots.
`subplots_adjust`	Adjust the subplot layout parameters.

`xlabel`	Set the label for the x-axis.
`xlim`	Get or set the x limits of the current axes.
`xscale`	Set the x-axis scale.
`xticks`	Get or set the current tick locations and labels of the x-axis.
`ylabel`	Set the label for the y-axis.
`ylim`	Get or set the y-limits of the current axes.
`yscale`	Set the y-axis scale.
`yticks`	Get or set the current tick locations and labels of the y-axis.

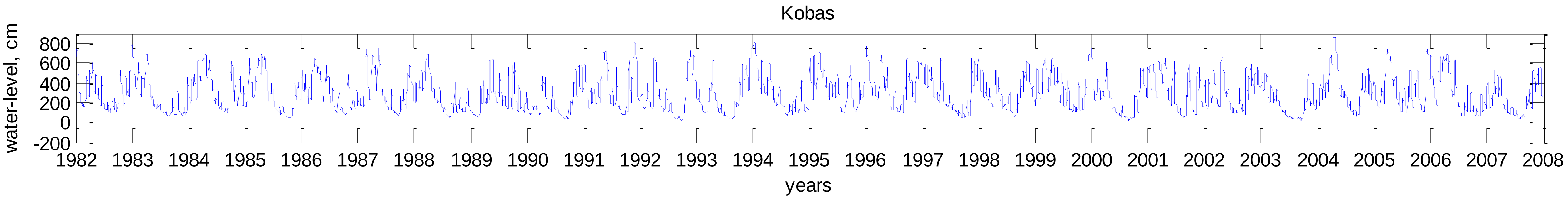
Primjer: operacije na diskretnom signalu

```python
import numpy as np

kobas = [661 , 674, 685, 693, 702, 712, 726, 735, 740, 736]

print(np.mean(kobas))   # srednja vrijednost
print(np.std(kobas))    # standardna devijacija
print(np.min(kobas))    # minimalna vrijednost
print(np.max(kobas))    # maksimalna vrijednost
print(np.sort(kobas))   # sortiranje
print(np.median(kobas)) # medijan signala
```

Primjer: učitavanje podataka iz .mat

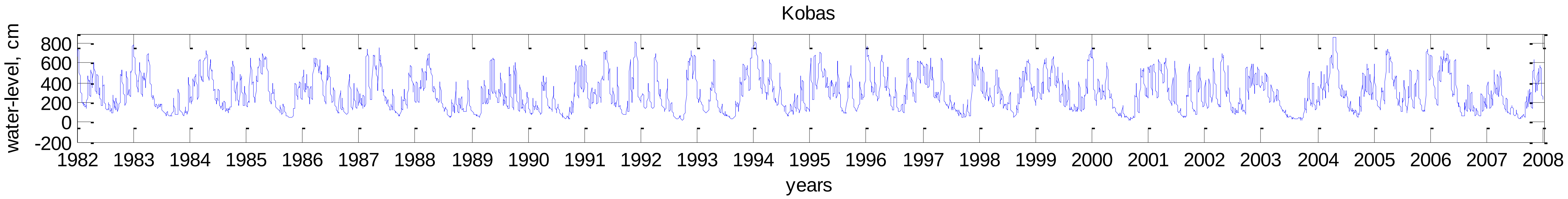
```
import numpy as np
from scipy import io

# ucitavanje podataka iz .mat - tip podataka je rjecnik
kobas_iz_matlaba = io.loadmat('OIkobas.mat')

# izdvajanje jednog kljuca i pripadne vrijednosti iz rjecnika
kobas = kobas_iz_matlaba['kobas']

# smanjivanje dimenzije liste 'kobas'
kobas = np.squeeze(kobas)

print(kobas.size)  # broj podataka u listi 'kobas'
print(kobas.shape) # dimenzije matrice 'kobas'

# stvaranje liste vremena n, prvi element = 0, zadnji = kobas.size-1, korak = 1
n = np.arange(0,kobas.size,1)
print(n.size)  # broj podataka u listi vremena 'n'
print(n.shape) # dimenzije matrice 'kobas'

print(np.mean(kobas))   # srednja vrijednost
print(np.std(kobas))    # standardna devijacija
print(np.min(kobas))    # minimalna vrijednost
print(np.max(kobas))    # maksimalna vrijednost
print(np.sort(kobas))   # sortiranje
print(np.median(kobas)) # medijan signala
```

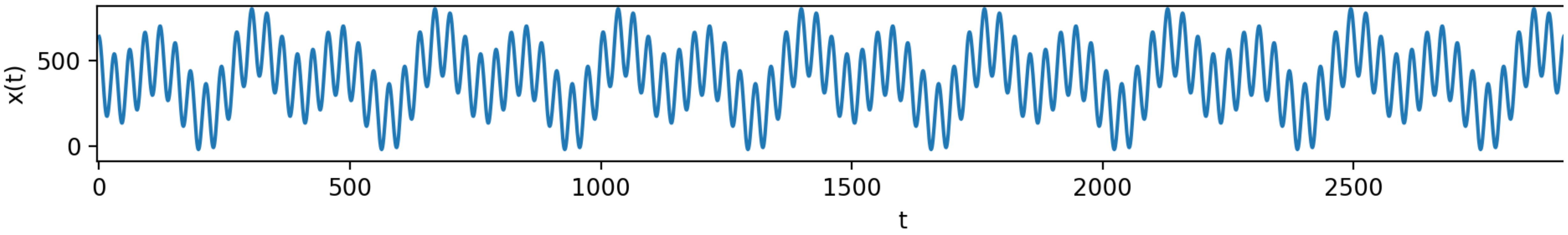
Kobaš

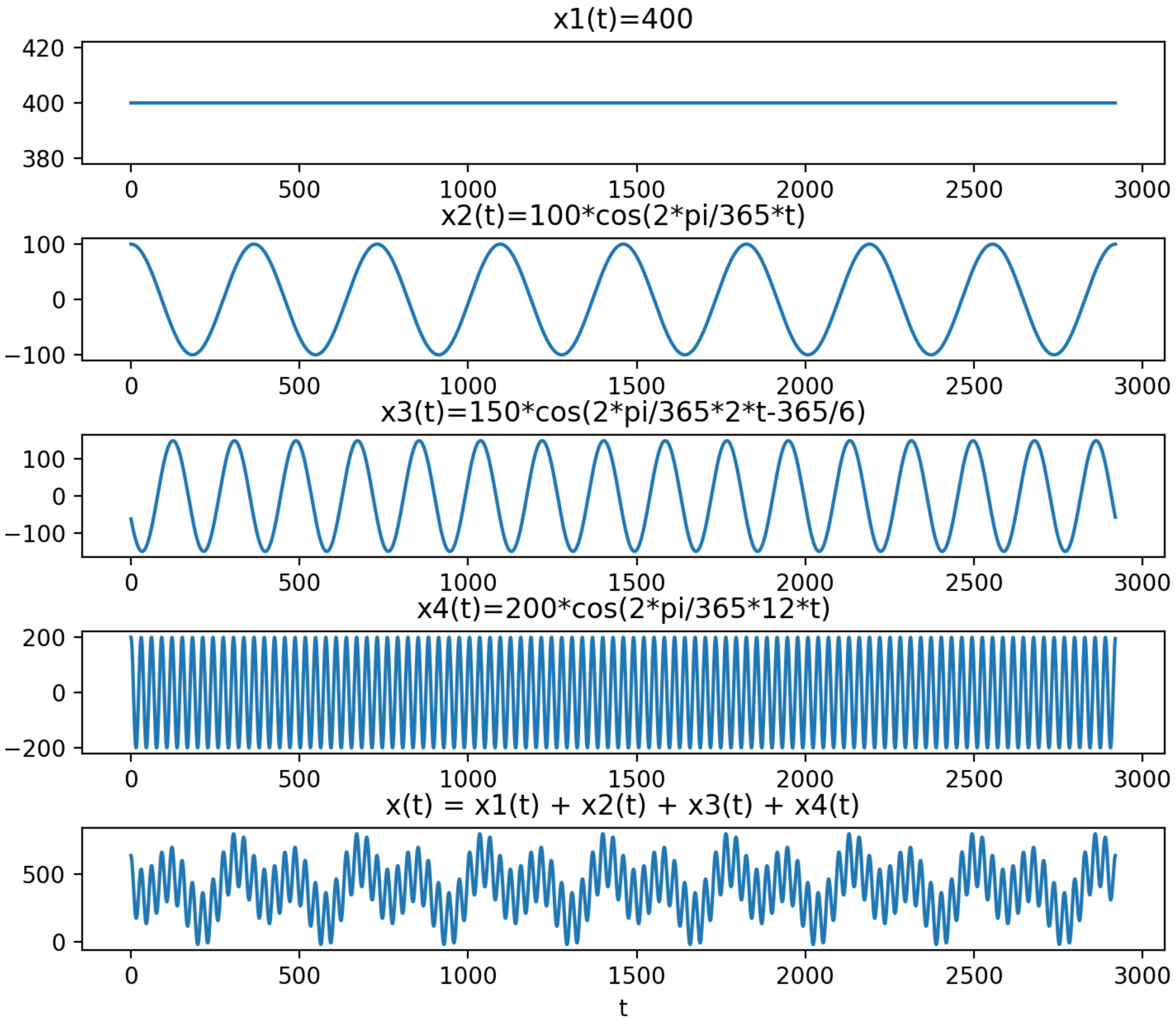


Kobas
water-level, cm
800
600
400
200
0
-200
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
years

Vodostaj ≈ zbroj sinusa






x1(t)=400
x2(t)=100*cos(2*pi/365*t)
x3(t)=150*cos(2*pi/365*2*t-365/6)
x4(t)=200*cos(2*pi/365*12*t)
x(t) = x1(t) + x2(t) + x3(t) + x4(t)
420
400
380
100
0
−100
200
−200
500
0
500
1000
1500
2000
2500
3000
t

Periodičan signal

$$x(t) = 400 + 100\cos\left(\frac{2\pi}{365}t\right) + 150\cos\left(\frac{2\pi}{365}2t - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}12t\right)$$

- Period drugog dijela = 365, trećeg = 365/2, četvrtog = 365/12
- Temeljni period: T = 365
- Frekvencija: $\omega = \frac{2\pi}{365} = 0.0172$

Podsjetnik

- Veza kompleksne eksponencijale i sinusa:

$$e^{j\omega_0 t} = \cos(\omega_0 t) + j\sin(\omega_0 t)$$
$$e^{-j\omega_0 t} = \cos(\omega_0 t) - j\sin(\omega_0 t)$$

- Zbrajanjem, odnosno oduzimanjem se dobiva:

$$\cos(\omega_0 t) = \frac{1}{2}\left(e^{j\omega_0 t} + e^{-j\omega_0 t}\right)$$
$$\sin(\omega_0 t) = \frac{1}{2j}\left(e^{j\omega_0 t} - e^{-j\omega_0 t}\right)$$

$$x(t) = 400 + 100\cos\left(\frac{2\pi}{365}t\right) + 150\cos\left(\frac{2\pi}{365}2t - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}12t\right)$$

- Zamijenimo cos s eksponencijalnom funkcijom:

$$x(t) = 400e^{j0} + 100 \cdot \frac{1}{2}\left(e^{j\frac{2\pi}{365}t} + e^{-j\frac{2\pi}{365}t}\right) +$$
$$+150 \cdot \frac{1}{2}\left(e^{j\left(\frac{2\pi}{365}2t - \frac{365}{6}\right)} + e^{-j\left(\frac{2\pi}{365}2t - \frac{365}{6}\right)}\right) + 200 \cdot \frac{1}{2}\left(e^{j\frac{2\pi}{365}12t} + e^{-j\frac{2\pi}{365}12t}\right)$$

- Upotrijebimo oznaku T=365:

$$x(t) = 400e^{j\frac{2\pi}{T}0t} + 100 \cdot \frac{1}{2}\left(e^{j\frac{2\pi}{T}t} + e^{-j\frac{2\pi}{T}t}\right) +$$
$$+150 \cdot \frac{1}{2}\left(e^{j\left(\frac{2\pi}{T}2t - \frac{365}{6}\right)} + e^{-j\left(\frac{2\pi}{T}2t - \frac{365}{6}\right)}\right) + 200 \cdot \frac{1}{2}\left(e^{j\frac{2\pi}{T}12t} + e^{-j\frac{2\pi}{T}12t}\right)$$

- Bez zagrada:

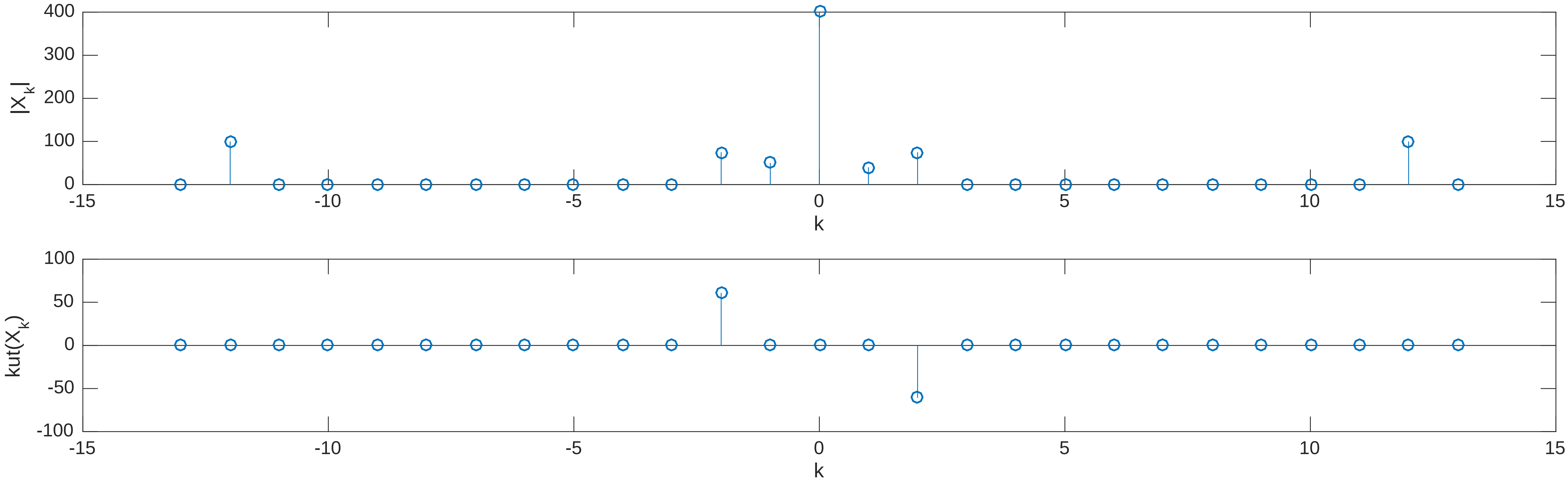
$$x(t) = 400e^{j\frac{2\pi}{T}0t} + 50e^{j\frac{2\pi}{T}t\cdot 1} + 50e^{j\frac{2\pi}{T}t\cdot(-1)} +$$
$$+75e^{-\frac{365}{6}j}e^{j\frac{2\pi}{T}t\cdot 2} + 75e^{\frac{365}{6}j}e^{j\frac{2\pi}{T}t\cdot(-2)}$$
$$+100e^{j\frac{2\pi}{T}t\cdot 12} + 100e^{j\frac{2\pi}{T}t\cdot(-12)}$$

- Sažeti zapis:

$$x(t) = \sum_{k=-\infty}^{\infty} X_k e^{j\frac{2\pi}{T}\cdot t\cdot k}$$

- Koeficijenti su:

$$X_0 = 400, X_1 = 50, X_{-1} = 50,$$
$$X_2 = 75e^{-\frac{365}{6}j}, X_{-2} = 75e^{\frac{365}{6}j},$$
$$X_{12} = 100, X_{-12} = 100$$



- U zadanom signalu imamo frekvencije, koje možemo očitati i sa slike:
  - Za k=1 → $\omega_1 = \frac{2\pi}{T} = 0.0172$
  - Za k=2 → $\omega_2 = 2 \cdot \frac{2\pi}{T} = 0.0344$
  - Za k=12 → $\omega_{12} = 12 \cdot \frac{2\pi}{T} = 0.2066$

Vremenski kontinuiran Fourierov red

- Za periodičan kontinuirani signal x(t)
- Fourierov red vremenski kontinuiranih signala (CTFS)

$$X_k = \frac{1}{T} \int_T x(t) e^{-j\frac{2\pi}{T}\cdot t\cdot k} dt$$

- Inverzni Fourierov red vremenski kontinuiranih signala (ICTFS)

$$x(t) = \sum_{k=-\infty}^{\infty} X_k e^{j\frac{2\pi}{T}\cdot t\cdot k}$$

- Interpretacija: “Bilo koja periodička funkcija se može predstaviti sumom sinusa i kosinusa, različitih frekvencija, amplituda i faza.”

- **Fourierova transformacija** – proširuje skup signala koji se mogu rastaviti na sinuse i kosinuse na bilo koji aperiodičan signal
- Interpretacija: „Fourierova transformacija je kao prizma: rastavlja funkciju u sve moguće frekvencije koje su skrivene u toj funkciji, kao što prizma rastavlja svjetlo u boje.”

Vremenski kontinuirana Fourierova transformacija

- Za aperiodičan kontinuirani signal x(t)
- Fourierova transformacija vremenski kontinuiranih signala (CTFT)

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$$

- Inverzna Fourierova transformacija vremenski kontinuiranih signala (ICTFT)

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega$$

Vremenski diskretan Fourierov red

- Za periodičan diskretan signal x(n)
- Fourierov red vremenski diskretnih signala (DTFS)

$$X_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}\cdot n\cdot k}$$

- Inverzni Fourierov red vremenski diskretnih signala (IDTFS)

$$x(n) = \sum_{k=0}^{N-1} X_k e^{j\frac{2\pi}{N}\cdot n\cdot k}$$

Vremenski diskretna Fourierova transformacija

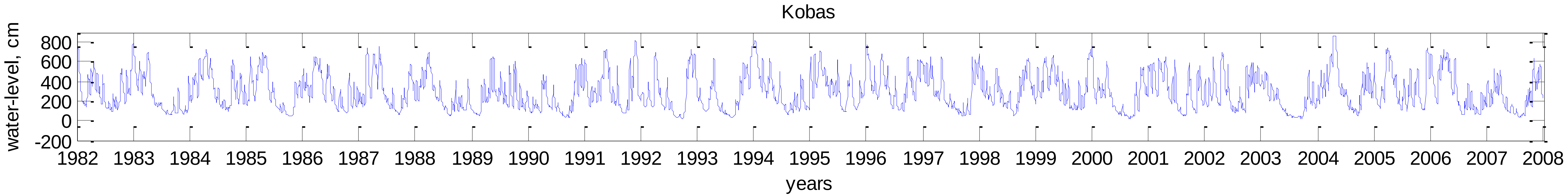
- Za aperiodičan diskretan signal x(n)
- Fourierova transformacija vremenski diskretnih signala (DTFT)

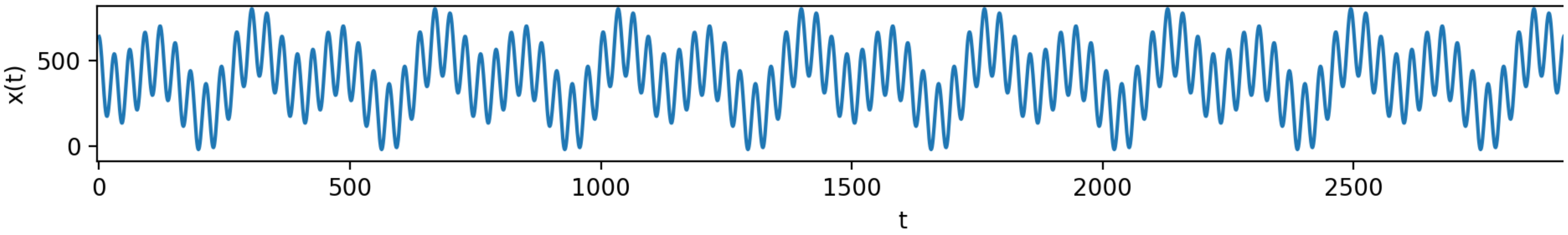
$$X(\Omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\Omega n}$$

- Inverzna Fourierova transformacija vremenski diskretnih signala (IDTFT)

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\Omega) e^{j\Omega n} d\Omega$$

Vodostaj ≈ zbroj sinusa





- Diskretan signal, konačnog trajanja → želimo diskretan spektar konačnog trajanja - DFT

Diskretna Fourierova transformacija - DFT

- Za aperiodičan diskretni signal x(n) duljine N,

$$x(n) = 0, \text{za}\ n < 0 \text{ i } n \geq N$$

- Diskretna Fourierova transformacija

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}kn}, k = 0,1,\ldots,N-1$$

- Inverzna diskretna Fourierova transformacija

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}kn}, n = 0,1,\ldots,N-1$$

DFT primjer

- Odredite DFT za diskretni signal $x(n) = \{\underline{2}, 4,1,4\}$ → N=4

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn} = \sum_{n=0}^{3} x(n)e^{-j\frac{2\pi}{4}kn}$$

$$= \sum_{n=0}^{3} x(n)\left(\cos\left(\frac{\pi}{2}\right) - \mathrm{j}\sin\left(\frac{\pi}{2}\right)\right)^{kn}$$

$$= \sum_{n=0}^{3} x(n)(-j)^{kn}$$

$$= x(0)(-j)^{0\cdot k} + x(1)(-j)^{1\cdot k} + x(2)(-j)^{2\cdot k} + x(3)(-j)^{3\cdot k}$$

DFT primjer

- Odredite DFT za diskretni signal $x(n) = \{\underline{2}, 4,1,4\}$ → N=4

$$X(k) = x(0)(-j)^{0\cdot k} + x(1)(-j)^{1\cdot k} + x(2)(-j)^{2\cdot k} + x(3)(-j)^{3\cdot k}$$

$$X(0) = x(0) + x(1) + x(2) + x(3) = 2 + 4 + 1 + 4 = 11$$

$$X(1) = x(0) + x(1)(-j)^{1\cdot 1} + x(2)(-j)^{2\cdot 1} + x(3)(-j)^{3\cdot 1}$$

$$= 2 - 4j - 1 + 4j = 1$$

$$X(2) = x(0) + x(1)(-j)^{1\cdot 2} + x(2)(-j)^{2\cdot 2} + x(3)(-j)^{3\cdot 2}$$

$$= 2 - 4 + 1 - 4 = -5$$

$$X(3) = x(0) + x(1)(-j)^{1\cdot 3} + x(2)(-j)^{2\cdot 3} + x(3)(-j)^{3\cdot 3}$$

$$= 2 + 4j - 1 - 4j = 1$$

$$X(k) = \{\underline{11}, 1, -5, 1\}$$

IDFT primjer

- Odredite IDFT za spektar X$(k) = \{\underline{11}, 1, -5,1\}$ → N=4

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi}{N}kn} = \frac{1}{4}\sum_{k=0}^{3} X(k)e^{j\frac{2\pi}{4}kn}$$

$$= \frac{1}{4}\sum_{k=0}^{3} X(k)\left(\cos\left(\frac{\pi}{2}\right) + \mathrm{j}\sin\left(\frac{\pi}{2}\right)\right)^{kn}$$

$$= \frac{1}{4}\sum_{n=0}^{3} X(k)(j)^{kn}$$

$$= \frac{1}{4}\left(X(0)(j)^{0\cdot k} + X(1)(j)^{1\cdot k} + X(2)(j)^{2\cdot k} + X(3)(j)^{3\cdot k}\right)$$

IDFT primjer

- Odredite IDFT za spektar $\mathrm{X}(k) = \{\underline{11}, 1, -5, 1\}$ → N=4

$$\mathrm{x}(n) = \frac{1}{4}\left(X(0)(j)^{0\cdot k} + X(1)(j)^{1\cdot k} + X(2)(j)^{2\cdot k} + X(3)(j)^{3\cdot k}\right)$$

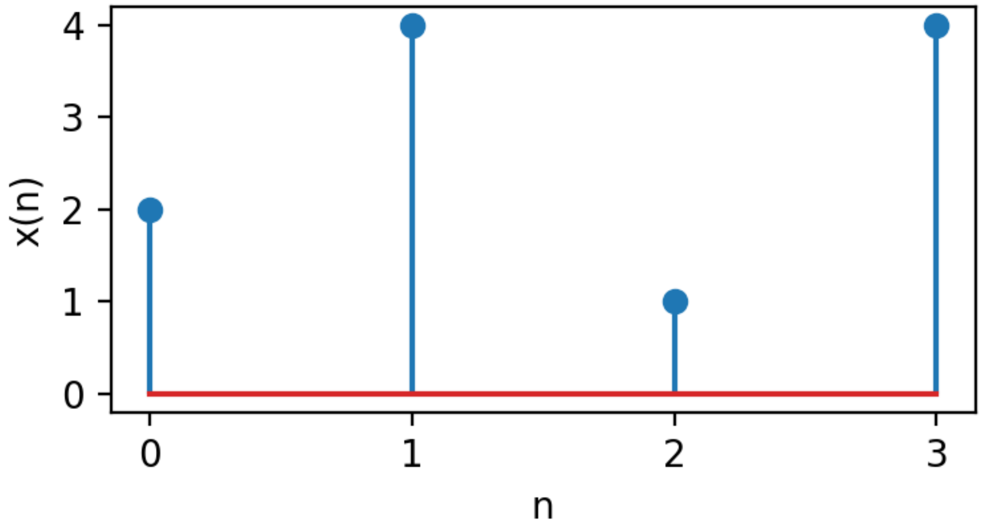
$$\mathrm{x}(0) = (X(0) + X(1) + X(2) + X(3))/4 = (11 + 1 - 5 + 1)/4 = 2$$

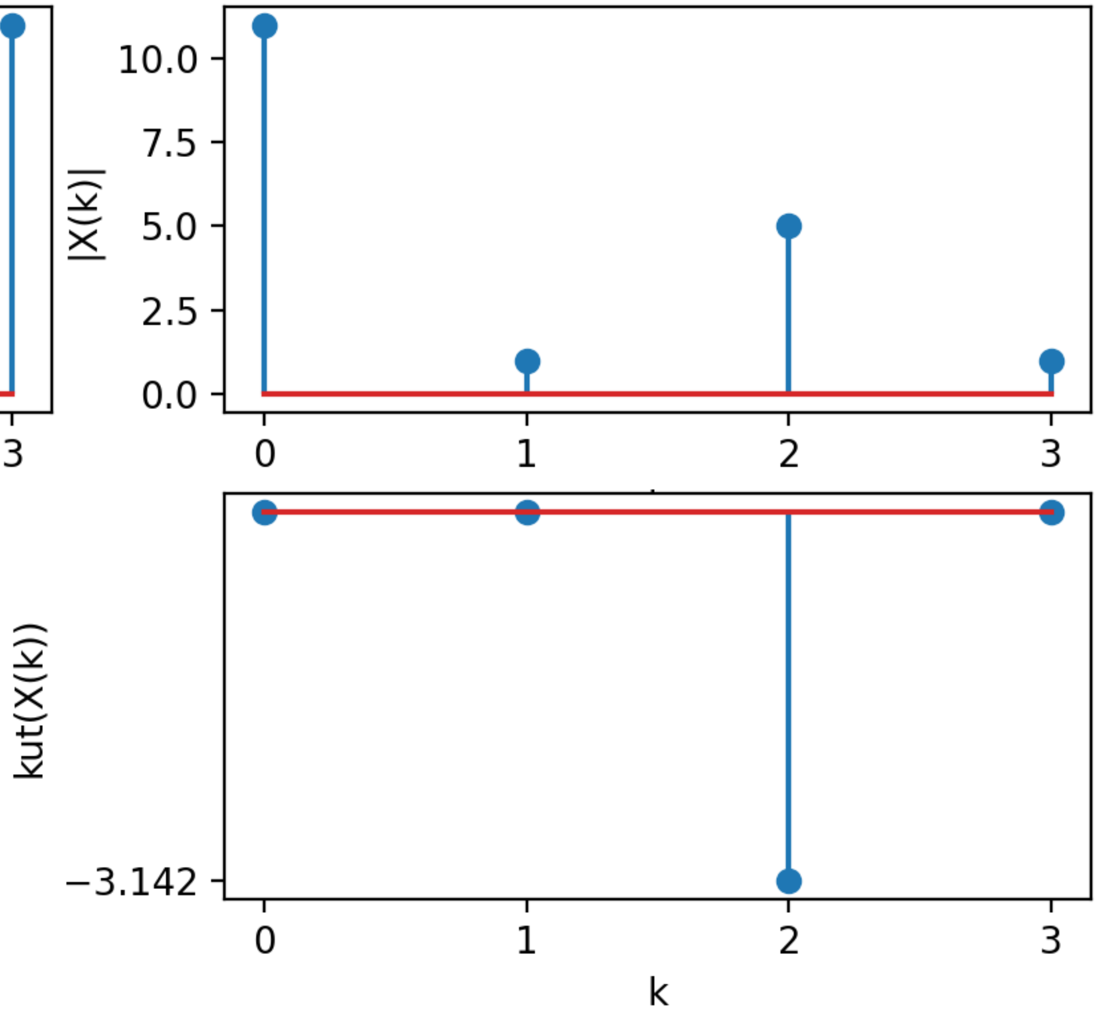
$$\mathrm{x}(1) = (X(0) + X(1)(j)^{1\cdot 1} + X(2)(j)^{2\cdot 1} + X(3)(j)^{3\cdot 1})/4$$
$$= (11 + 1j + 5 - 1j)/4 = 4$$

$$\mathrm{x}(2) = (X(0) + X(1)(j)^{1\cdot 2} + X(2)(j)^{2\cdot 2} + X(3)(j)^{3\cdot 2})/4$$
$$= (11 - 1 - 5 - 1)/4 = 1$$

$$\mathrm{x}(3) = (X(0) + X(1)(j)^{1\cdot 3} + X(2)(j)^{2\cdot 3} + X(3)(j)^{3\cdot 3})/4$$
$$= (11 - 1j + 5 + 1j)/4 = 4$$

$$x(n) = \{\underline{2}, 4, 1, 4\}$$





$$x(n) = \{\underline{2}, 4, 1, 4\}$$
$$X(k) = \{\underline{11}, 1, -5, 1\}$$

Primjer: DFT u Python-u

```
import numpy as np
from scipy.fft import fft, ifft
import matplotlib.pyplot as plt

x = np.array([2, 4, 1, 4])  # stvaranje niza = uzorci diskretnog signala
print(x)                    # ispis
X=fft(x, 4)                 # DFT signala x u 4 tocke
print(X)
xr = ifft(X, 4)             # IDFT spektra X u 4 tocke
print(xr)
```

Primjer: crtanje u Python-u

```
plt.figure(1) # otvaranje novog prozora za slike
plt.subplot(2,2,1) # u jednom prozoru ce biti 2 retka x 2 slike, crtamo 1. sliku
plt.stem(x) # peteljkasti prikaz diskretnog signala
plt.xlabel('n') # opis x osi
plt.ylabel('x(n)') # opis y osi
plt.xticks(np.arange(0,4,step=1)) # koje vrijednosti obiljeziti na x osi

plt.subplot(2,2,2) # u jednom prozoru ce biti 2 retka x 2 slike, crtamo 2. sliku
plt.stem(abs(X)) #moguce zadati i ovako: plt.stem([0,1,2,3],abs(X))
plt.xlabel('k')
plt.ylabel('|X(k)|')
plt.xticks(np.arange(0,4,step=1))

plt.subplot(2,2,4) # u jednom prozoru ce biti 2 retka x 2 slike, crtamo 4. sliku
plt.stem(np.angle(X)) #ili plt.stem([0,1,2,3],np.angle(X))
plt.xlabel('k')
plt.ylabel('kut(X(k))')
plt.xticks(np.arange(0,4,step=1))
plt.yticks(np.arange(-np.pi,0,step=np.pi)) # koje vrijednosti obiljeziti na y osi

plt.show() # prikazi slike
```

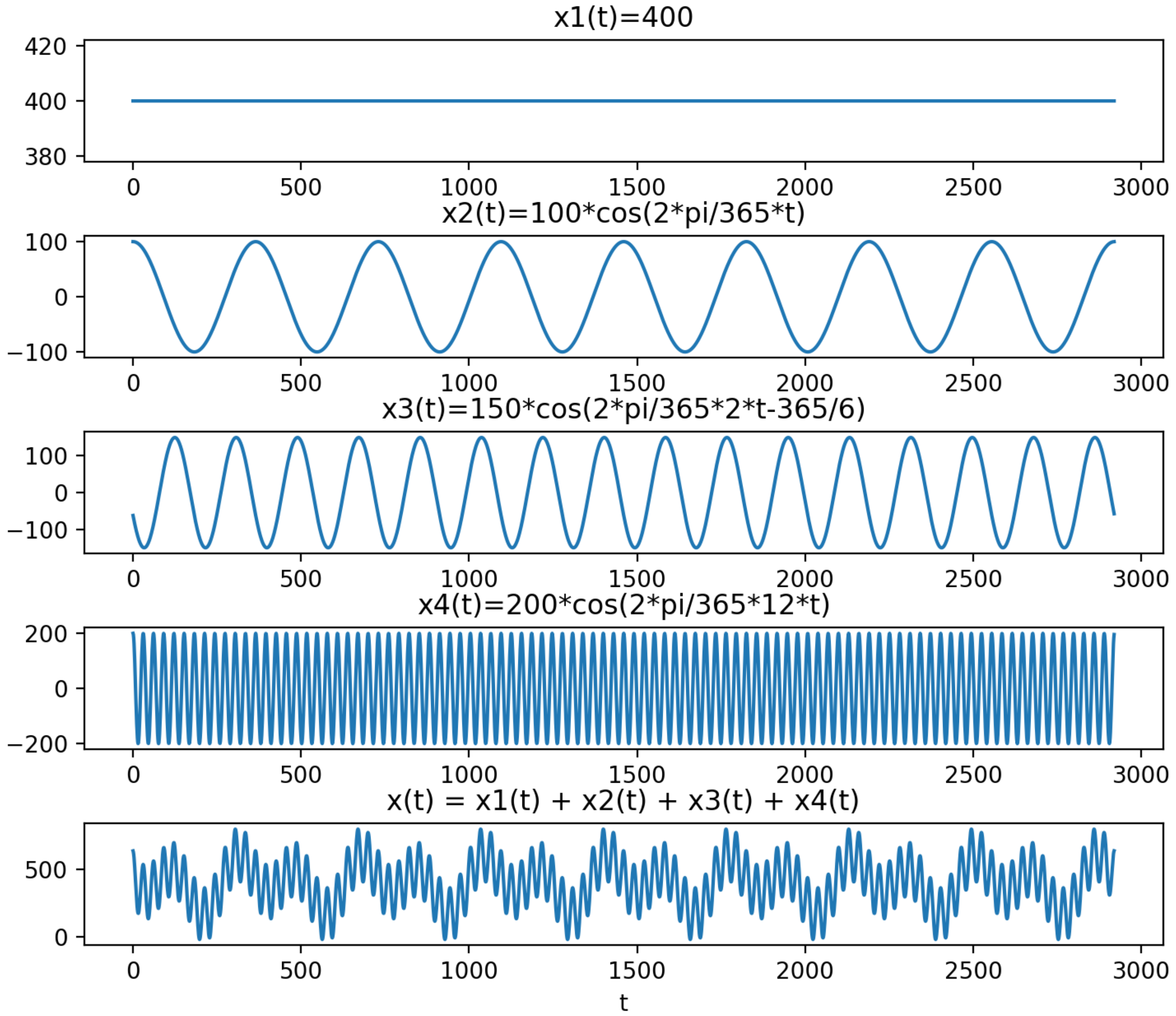
Zadatak za vježbu

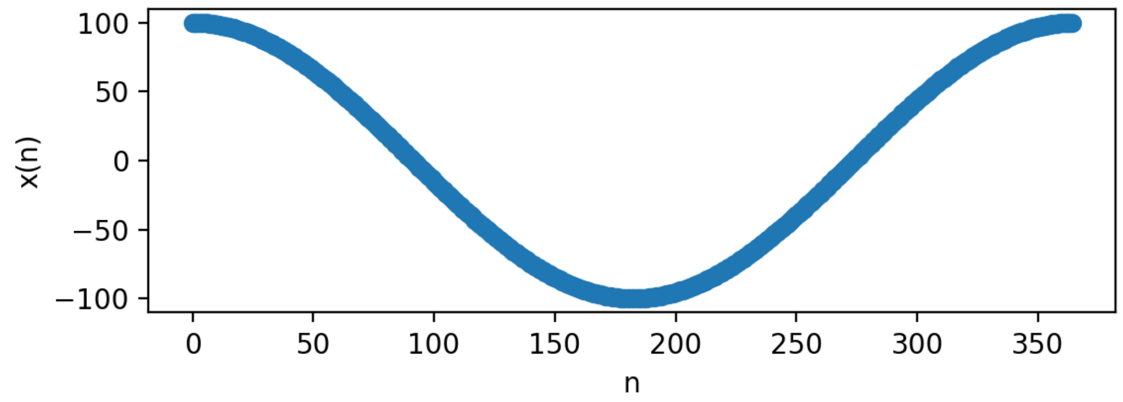
IDFT

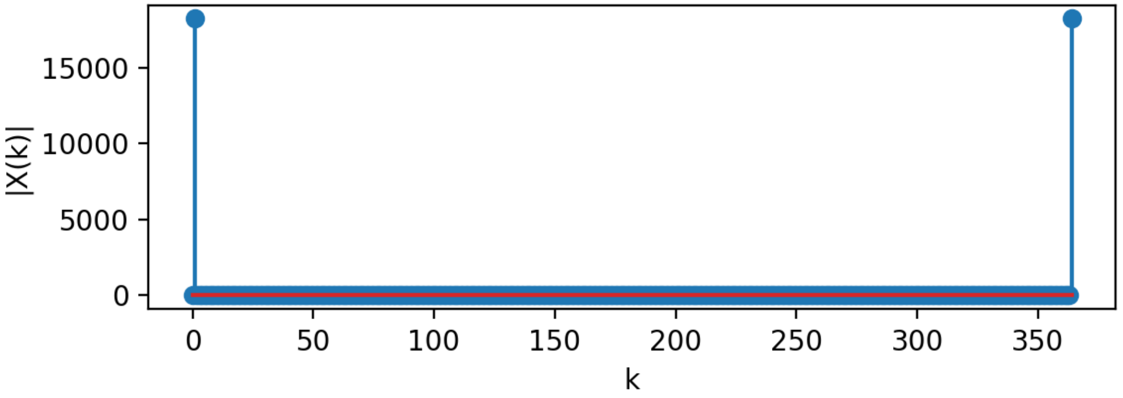
$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi}{N}kn}, n = 0,1,\dots, N-1$$

- Odredite DFT signala $x(n) = \cos\left(\frac{2\pi}{5}n\right)$, uz N=5, na dva načina:
  - Koristeći formulu za DFT i uvrštavajući vrijednosti *n*
  - Rastavljanjem kosinusa na kompleksne eksponencijale
- Poklapaju li se rezultati?
- Rj. $X(0) = 0, X(1) = 2.5, X(2) = 0, X(3) = 0, X(4) = 2.5$
- Primijetite: $k = 1$ je vezano uz frekvenciju $\frac{2\pi}{5}$ → amplituda pripadajućeg koeficijenta je $X(1) = \frac{5}{2} = \frac{N}{2}$
- $k = 4$ → dolazi iz $e^{j\frac{2\pi}{5}n(-1)} = e^{j\frac{2\pi}{5}n(4-5)} = e^{j\frac{2\pi}{5}n\cdot 4}\ e^{j\frac{2\pi}{5}n(-5)} = e^{j\frac{2\pi}{5}n\cdot 4}$

- Pogledajmo komponente umjetnog vodostaja
- Očitavanje signala: $t = n \cdot T_o$, gdje je T_o period očitavanja
- Ovdje: $T_o = 1$ dan







IDFT

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j\frac{2\pi}{N}kn}, n = 0,1,\dots,N-1$$

- Kontinuirani signal:

$$x_1(t) = 100\cos\left(\frac{2\pi}{365}t\right)$$

- Očitani signal, uz $T_o = 1$ dan:

$$x_1(n) = 100\cos\left(\frac{2\pi}{365}n\right)$$

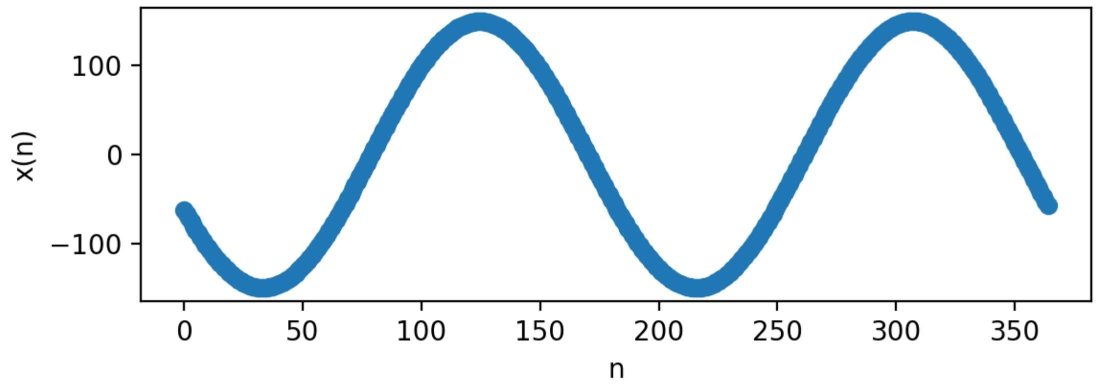
- Promatramo 1 godinu: $N = 365$

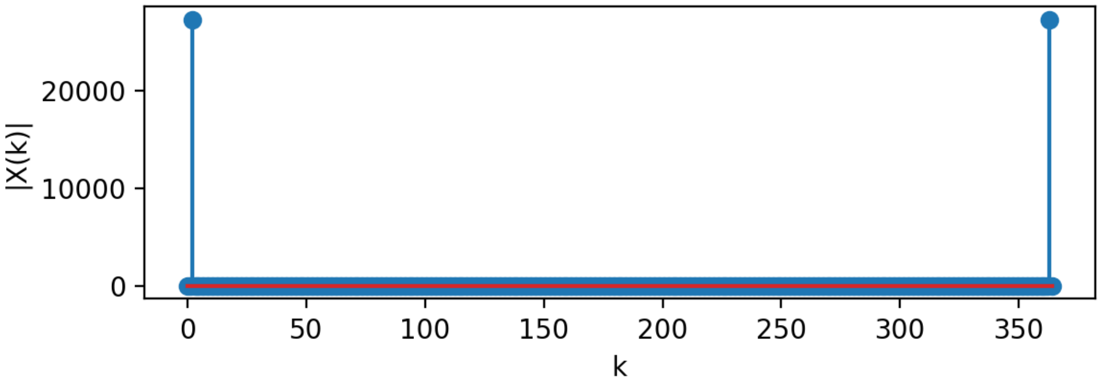
- $x_1(n) = 100\cos\left(\frac{2\pi}{365}n\right) =$

$$= \frac{1}{365}\left(50\cdot 365 e^{j\frac{2\pi}{365}n} + 50\cdot 365 e^{j\frac{2\pi}{365}n\cdot 364}\right)$$

$$= \frac{1}{365}\left(18250 e^{j\frac{2\pi}{365}n} + 18250\, e^{j\frac{2\pi}{365}n\cdot 364}\right)$$

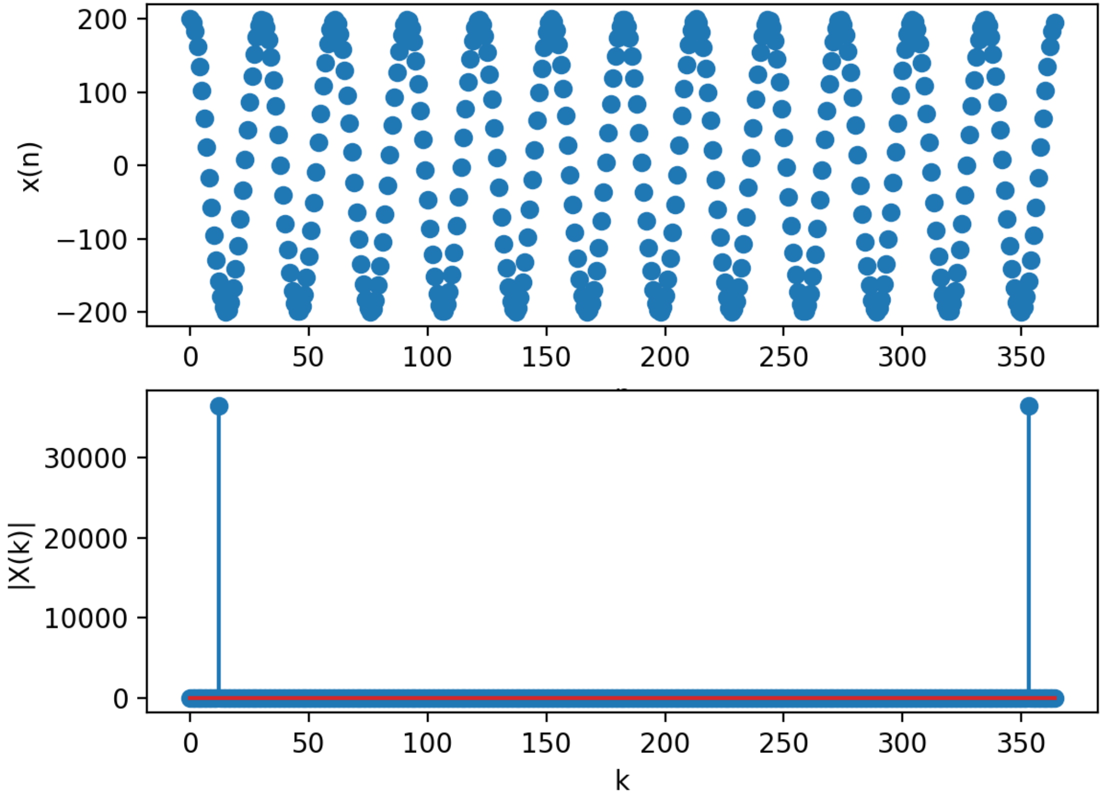
- Ako znamo spektar: $|X(1)| = 18250$, znamo i amplitudu signala $\frac{18250}{365}\cdot 2 = 100$ i njegovu frekvenciju $\frac{2\pi}{N}k = \frac{2\pi}{365} = 0.0172$





- Kontinuirani signal:
$$x_2(t) = 150\cos\left(\frac{2\pi}{365}2t - \frac{365}{6}\right)$$
- Očitani signal, uz $T_o = 1$ dan:
$$x_2(n) = 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right)$$
- Promatramo 1 godinu: $N = 365$

- $x_2(n) = 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) =$
$$= \frac{1}{365}\left(75 \cdot 365 e^{j\left(-\frac{365}{6}\right)} e^{j\frac{2\pi}{365}2n} + 75 \cdot 365 e^{j\frac{365}{6}} e^{j\frac{2\pi}{365}n\cdot 363}\right)$$
$$= \frac{1}{365}\left(27375 e^{j\left(-\frac{365}{6}\right)} e^{j\frac{2\pi}{365}2n} + 27375 e^{j\frac{365}{6}} e^{j\frac{2\pi}{365}n\cdot 363}\right)$$
- Ako znamo spektar: $|X(2)| = 27375$, znamo i amplitudu signala $\frac{27375}{365} \cdot 2 = 150$ i njegovu frekvenciju $\frac{2\pi}{N} \cdot k = \frac{2\pi}{N} \cdot 2 = \frac{4\pi}{365} = 0.0344$



- Kontinuirani signal:

$$x_3(t) = 200\cos\left(\frac{2\pi}{365}12t\right)$$

- Očitani signal, uz $T_o = 1$ dan:

$$x_3(n) = 200\cos\left(\frac{2\pi}{365}12n\right)$$

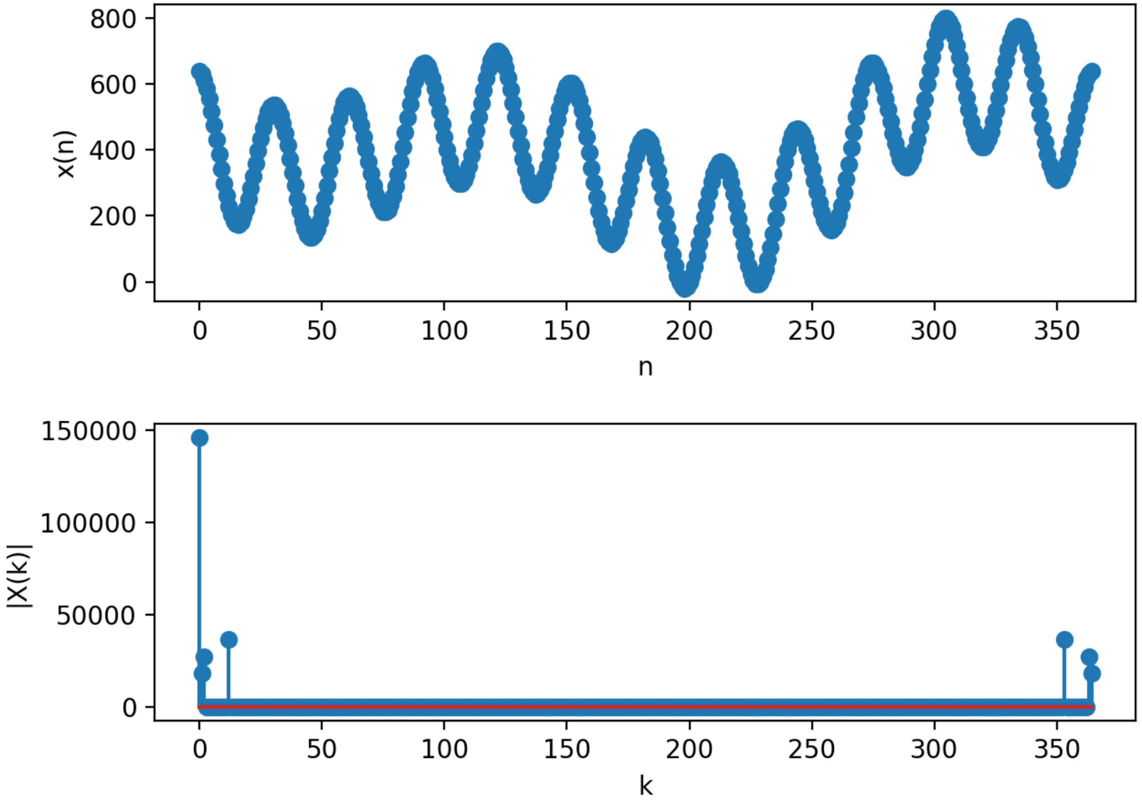
- Promatramo 1 godinu: $N = 365$

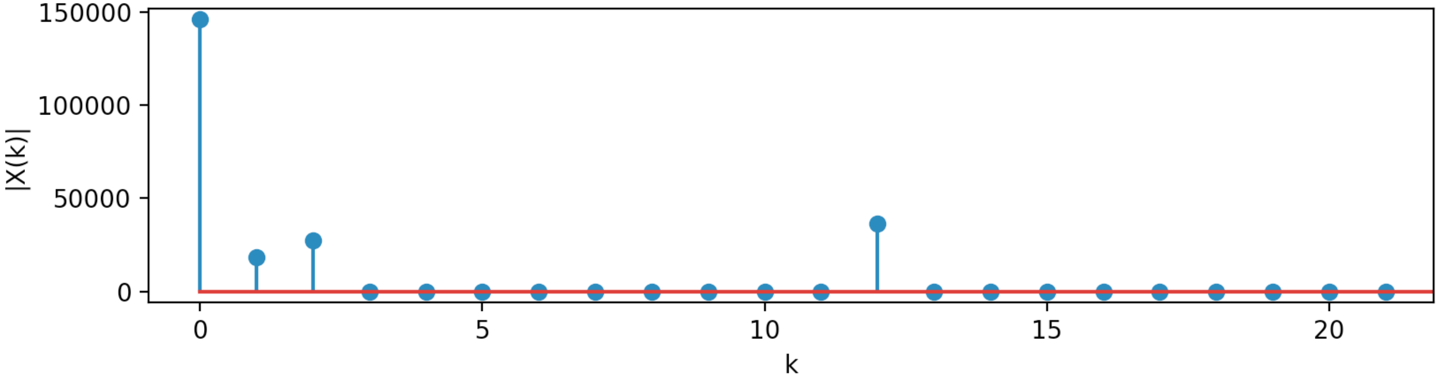
- $x_3(n) = 200\cos\left(\frac{2\pi}{365}12n\right) =$

$$= \frac{1}{365}\left(100 \cdot 365 e^{j\frac{2\pi}{365}12n} + 100 \cdot 365 e^{j\frac{2\pi}{365}n \cdot 353}\right)$$

$$= \frac{1}{365}\left(36500 e^{j\frac{2\pi}{365}12n} + 36500\ e^{j\frac{2\pi}{365}n \cdot 353}\right)$$

- Ako znamo spektar: $|X(12)| = 36500$, znamo i amplitudu signala $\frac{36500}{365} \cdot 2 = 200$ i njegovu frekvenciju $\frac{2\pi}{N} \cdot k = \frac{2\pi}{N} \cdot 12 = \frac{24\pi}{365} = 0.2066$





- Cijeli signal:

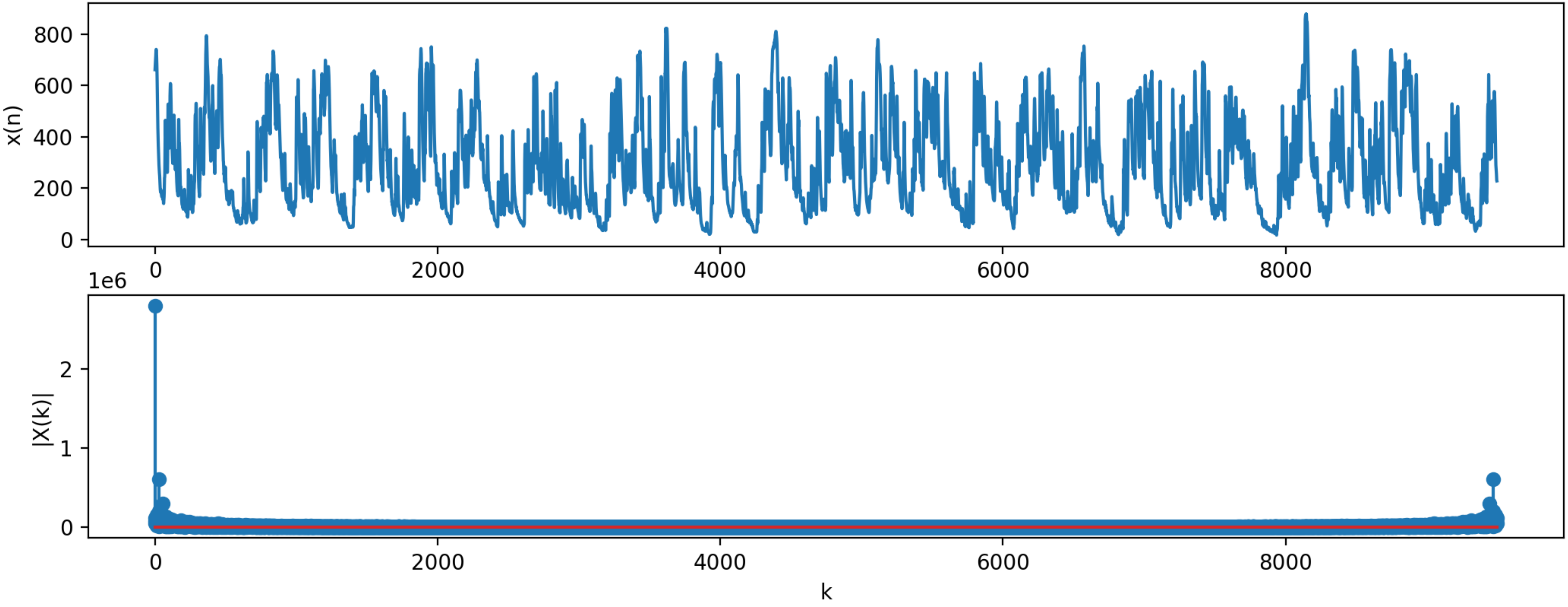
$$x(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}12n\right)$$

- Spektar:

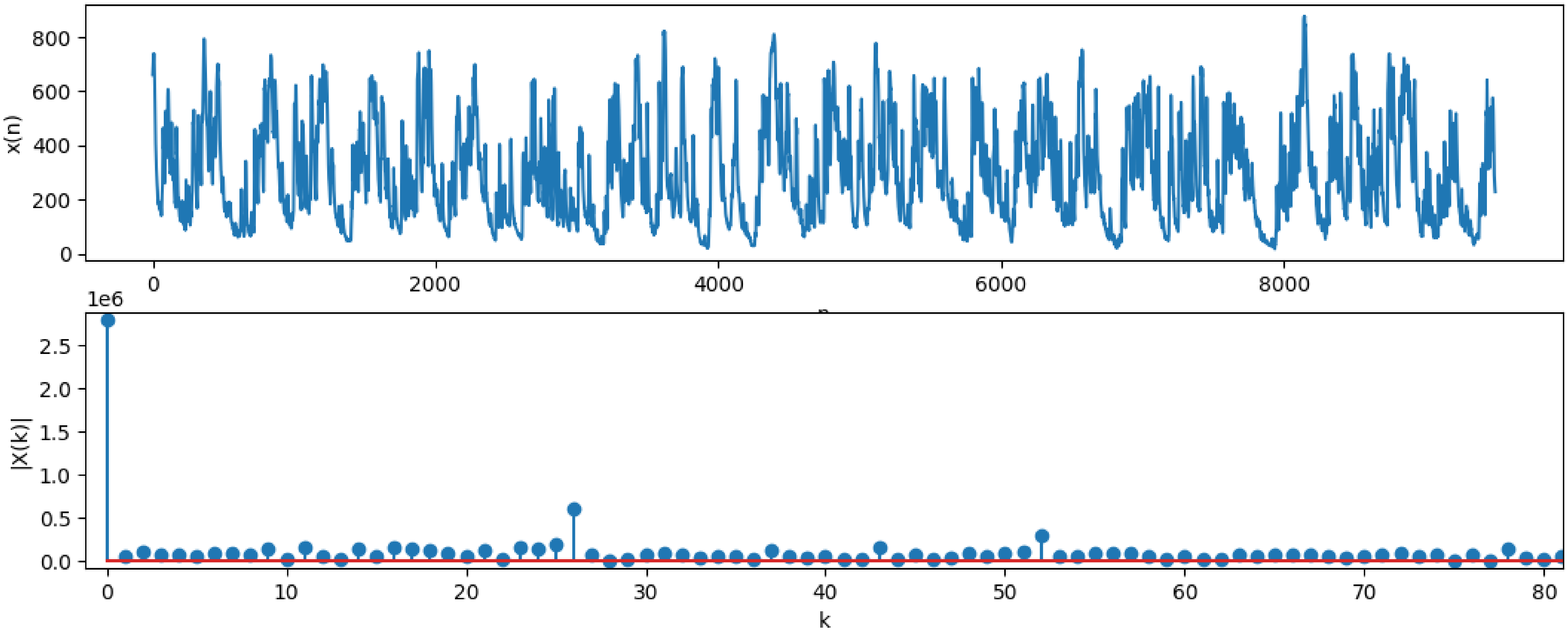
$$|X(0)| = 400 \cdot 365 = 146000$$
$$|X(1)| = 18250$$
$$|X(2)| = 27375$$
$$|X(12)| = 36500$$

- Frekvencije u signalu:
  - Za k=1 → $\frac{2\pi}{N} = 0.0172$
  - Za k=2 → $2 \cdot \frac{2\pi}{N} = 0.0344$
  - Za k=12 → $12 \cdot \frac{2\pi}{N} = 0.2066$

Analiza pravog vodostaja Kobaš

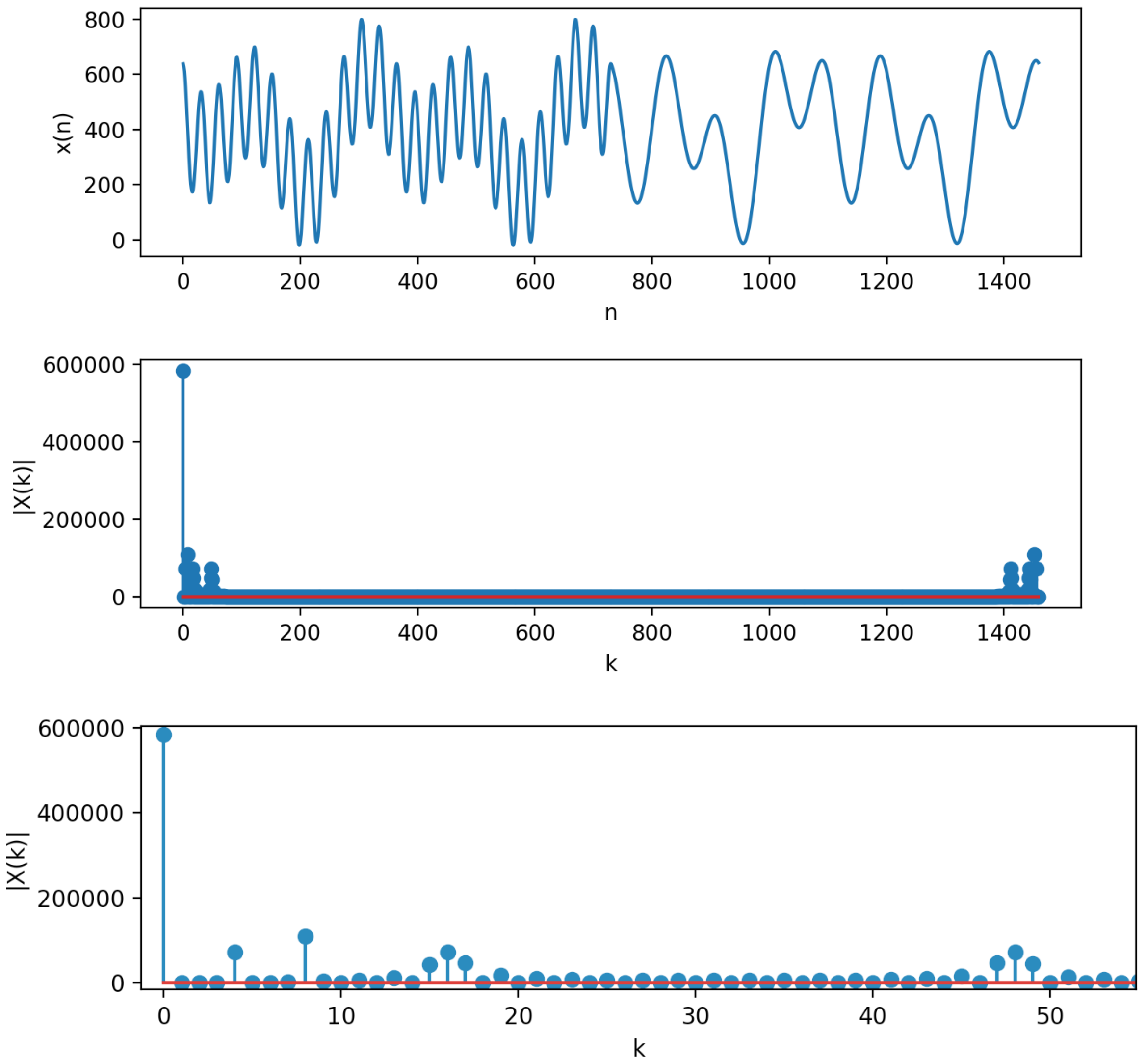


Analiza pravog vodostaja Kobaš



Signal koji se mijenja u vremenu

- $x_1(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}12n\right), \text{za } n \in \langle 0,2T]$
- $x_2(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}4n\right), \text{za } n \in \langle 2T,4T]$
- U DFT se ne mogu prepoznati dijelovi signala



Fourierova transformacija na vremenskom otvoru (STFT)

- zamijenimo $e^{j\omega t}$ → $g(t-\tau)e^{j\omega t}$
- $g(t)$ lokalni analizirajući otvor, τ pomak
- STFT

$$X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)g^*(t-\tau)e^{-j\omega t}dt$$

- * konjugacija → nije potrebna za realne otvore
- dvije dimenzije: vremenski pomak τ i frekvencija ω
- ovisi o izabranom otvoru $g(t)$

- CTFT

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$$

- ICTFT

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega$$

Inverzna Fourierova transformacija na vremenskom otvoru (ISTFT)

- STFT

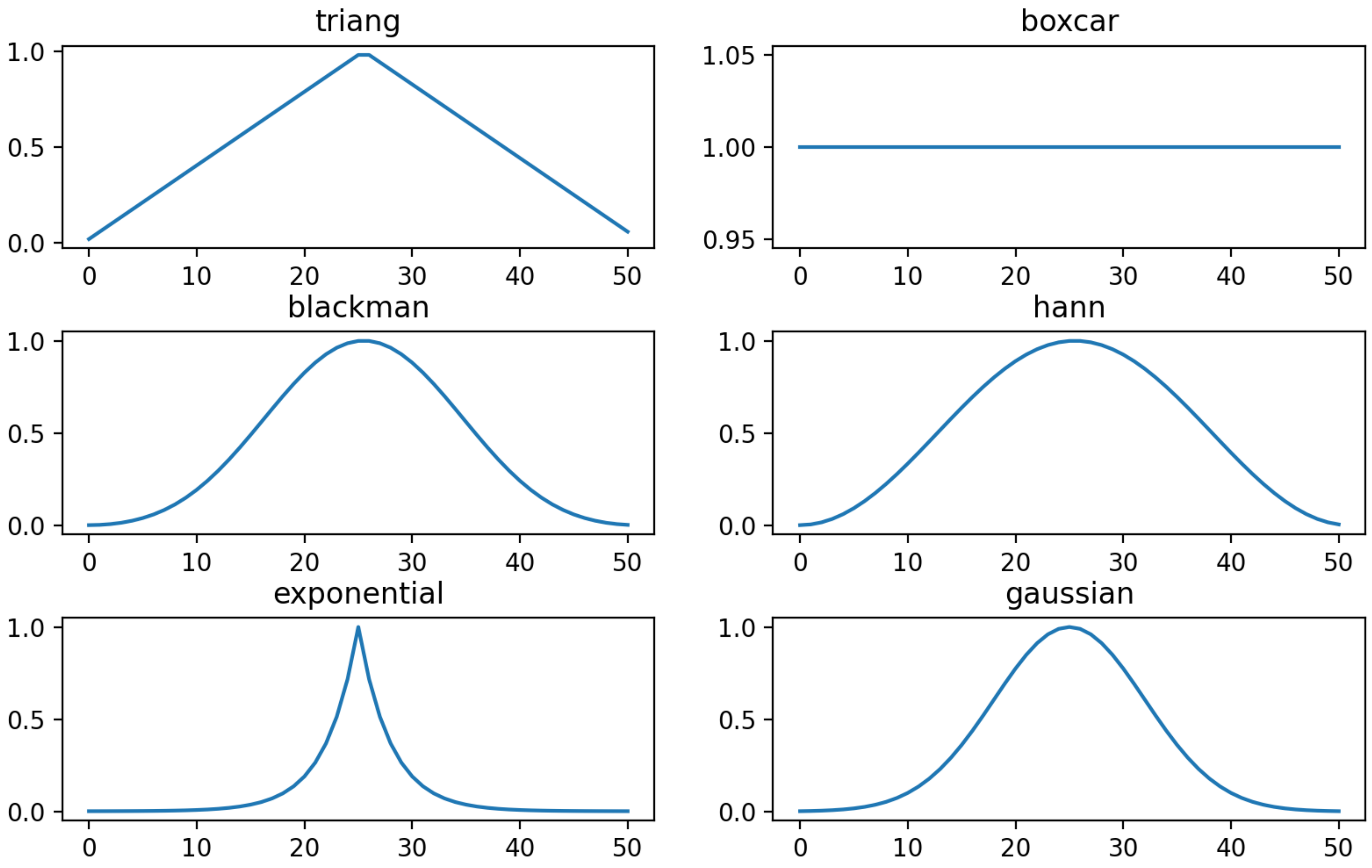
$$X(\tau, \omega) = \int_{-\infty}^{\infty} x(t) g^*(t-\tau) e^{-j\omega t} dt$$

- ISTFT

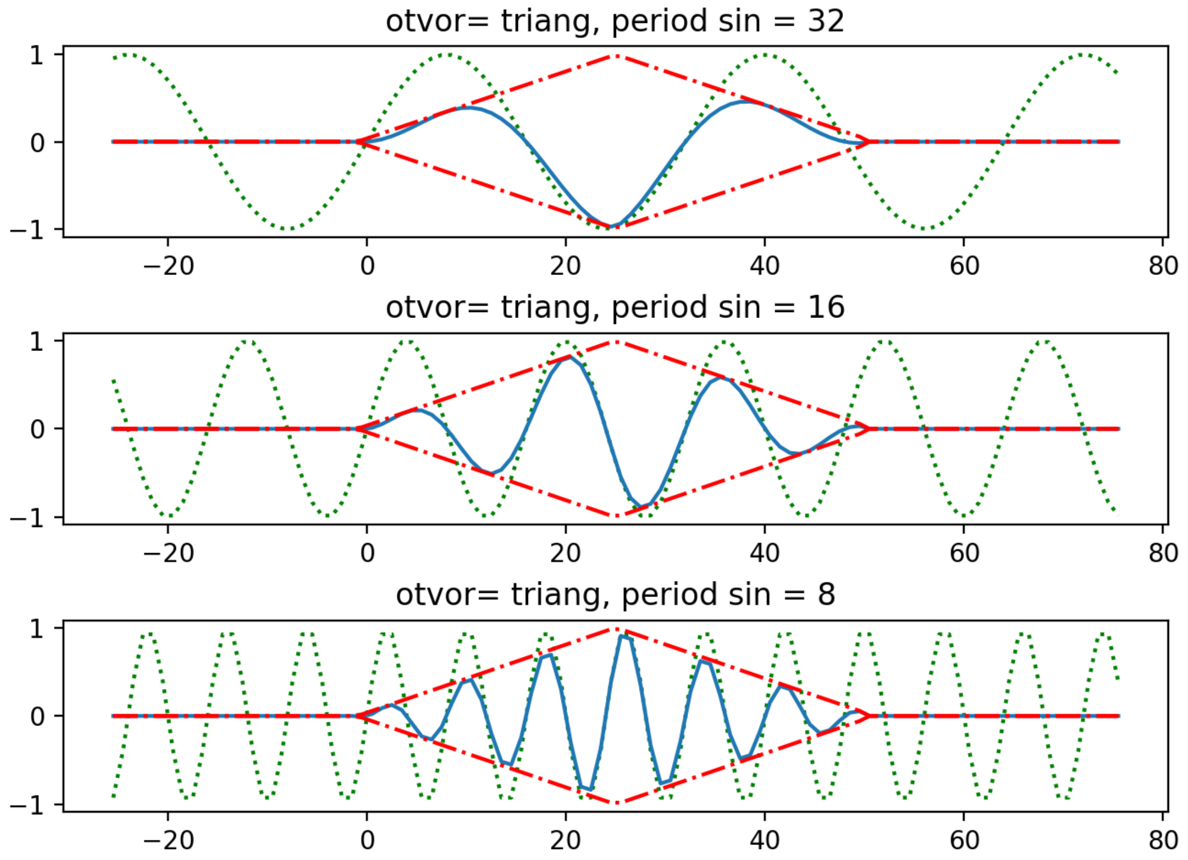
$$x(t) = \frac{1}{2\pi \|g\|^2} \int_{\tau=-\infty}^{\infty} \int_{\omega=-\infty}^{\infty} X(\tau, \omega) g(t-\tau) e^{j\omega t} d\omega \, d\tau$$

- Otvor g mora biti konačne energije

Otvor $g(t)$



Umnožak $g(t-\tau)\sin(\omega t)$



- Umnožak $g(t-\tau)\sin(\omega t)$ je dio $g(t-\tau)e^{j\omega t}$
- Veličina prozora je fiksna
- Period (frekvencija) sinusa se mijenja
- Broj oscilacija = promjenjiv
- uski prozor = „slijep” na niske frekvencije
- široki prozor = izgubljene informacije o brzim promjenama signala

Diskretizacija STFT

- STFT

$$X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)g^*(t-\tau)e^{-j\omega t}dt$$

- Diskretni vremenski pomak $\tau = mT$
- Diskretni frekvencijski pomak $\omega = k\Omega$
- Diskretizirani STFT

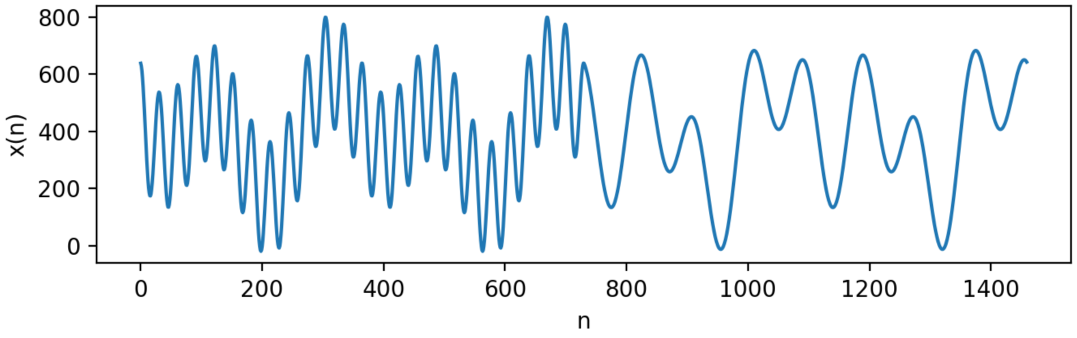
$$X(mT,k\Omega) = \int_{-\infty}^{\infty} x(t)g^*(t-mT)e^{-jk\Omega t}dt$$

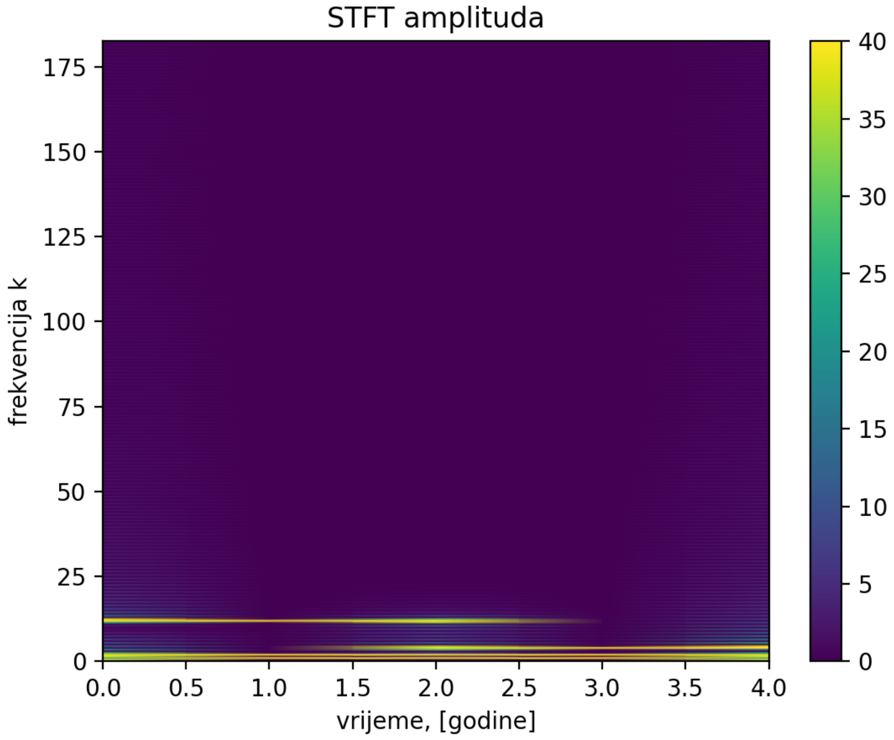
$$X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)g^*(t-\tau)e^{-j\omega t}dt$$

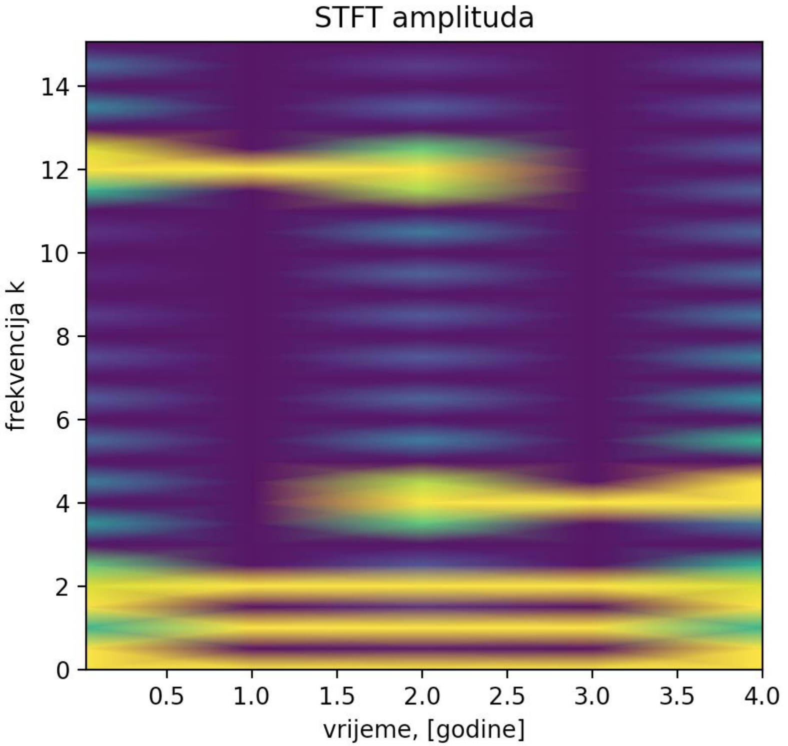
Signal koji se mijenja u vremenu

$$x_1(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}12n\right), \text{za } n \in \langle 0,2T]$$

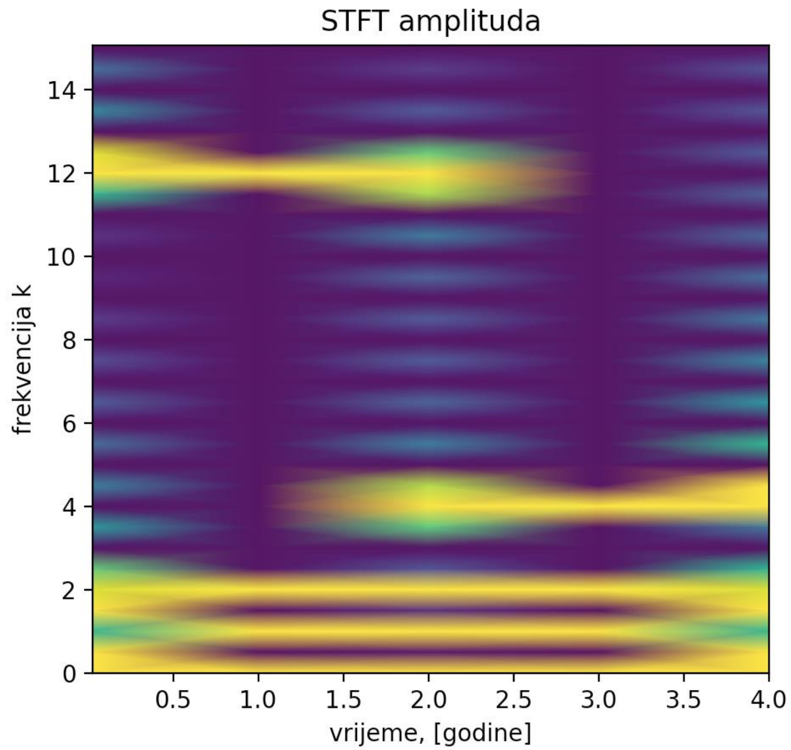
$$x_2(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}4n\right), \text{za } n \in \langle 2T,4T]$$



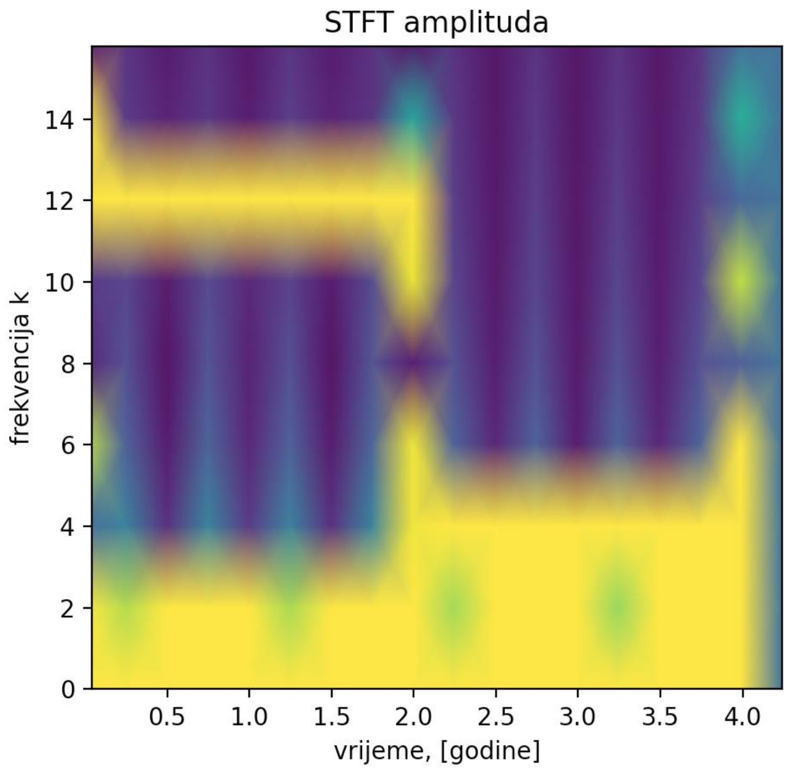




Utjecaj širine otvora



Širina pravokutnog otvora = 2T
Frekvencije možemo očitati precizno



Širina pravokutnog otvora = T/2
Trenutke u vremenu možemo očitati precizno

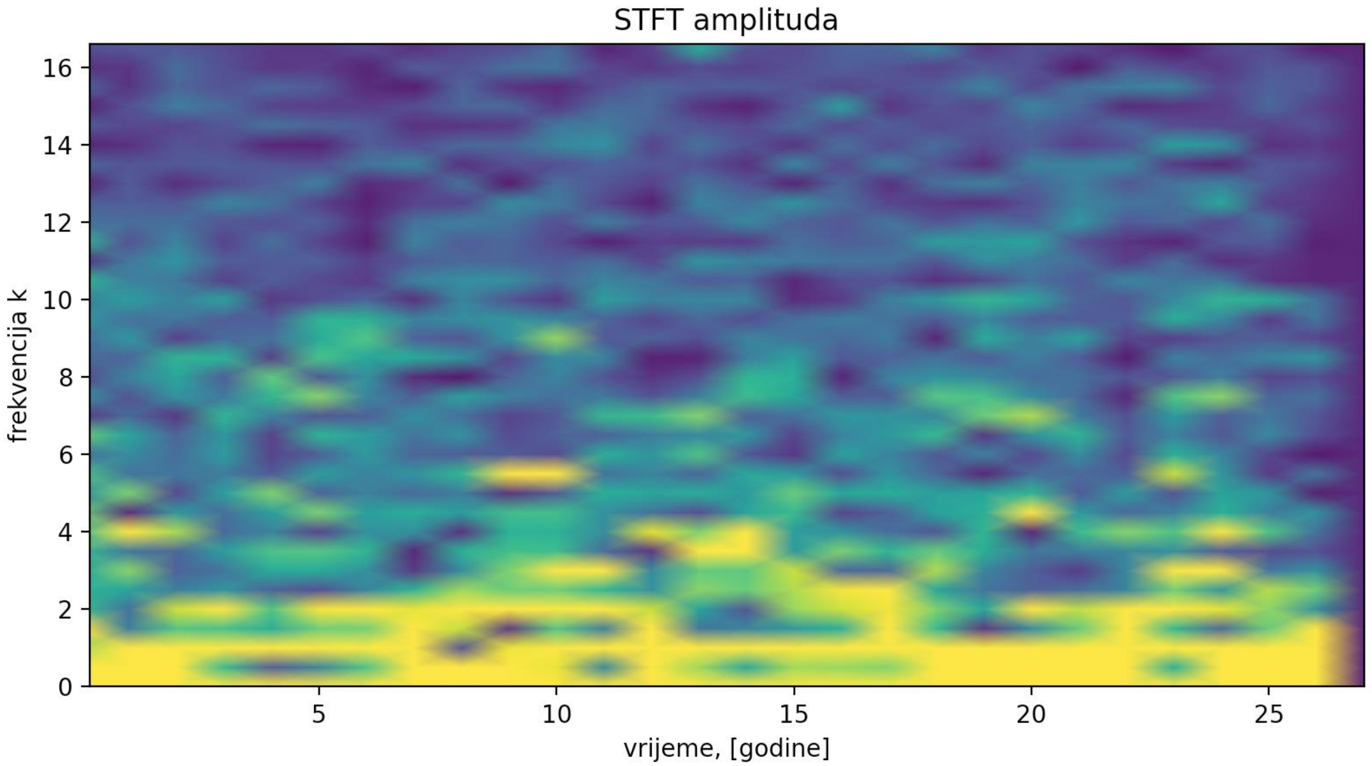
Primjer: složeni signal u Pythonu

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import signal


T=365 # broj dana u godini - broj uzoraka signala
n = np.arange(0,2*T,1)  # vektor vremena: od 0 do 2T, korak = 1

x1 = 100*np.cos(2*np.pi*n/T) # kosinus frekvencije w = 2 pi / T
x2 = 150*np.cos(2*np.pi*n*2/T-T/6) # kosinus frekvencije w = 2 pi / (T/2)
x3 = 200*np.cos(2*np.pi*n*12/T) # kosinus frekvencije w = 2 pi / (T/12)
xd1 = 400 + x1 + x2 + x3 # prvi dio signala

x4 = 200*np.cos(2*np.pi*n*4/T)  # kosinus frekvencije w = 2 pi / (T/4)
xd2 = 400 + x1 + x2 + x4 # drugi dio signala

n = np.arange(0,4*T,1) # ukupni vektor vremena
x = np.append([xd1],[xd2]) # spajanje dva niza: xd2 nakon xd1

plt.figure(1)
plt.plot(n,x)  # signal ima mnogo uzoraka -> crtati kao da je kontinuirani
plt.xlabel('t')
plt.ylabel('x(t)')
plt.show()
```

Primjer: STFT u Pythonu

```
fs = 365 # frekvencija ocitavanja signala x
window = 'boxcar' # otvor koji ce se koristiti - boxcar je pravokutni otvor
f, t, Zxx = signal.stft(x, fs, window, nperseg=2*T) # STFT, nperseg = duljina otvora
# izlazi: f = niz ocitanih frekvencija, t = niz vremena, Zxx = STFT od x

plt.pcolormesh(t, f, np.abs(Zxx), vmin=0, vmax=np.abs(Zxx.max())/10, shading='gouraud')
# 2D slika apsolutnih vrijednosti Zxx, na apscisi je vrijeme t, na ordinati su frekvencije f,
# abs(Zxx) je prikazana bojama, vmin i vmax odreduju raspon boja,
# shading = nacin bojanja s interpolacijom
plt.colorbar() # raspon boja i veza s abs(Zxx)
plt.title('STFT amplituda')
plt.ylabel('frekvencija k')
plt.xlabel('vrijeme, [godine]')
plt.show()
```

Kobaš, STFT



- Žuto – veća amplituda (vodostaj)
- Zanimljive frekvencije:
  - k=1 → 1 godina
  - k=2 → 1/2 godine
  - k=4 → 3 mjeseca
  - k=12 → 1 mjesec

Kontinuirana valićna transformacija (CWT)

- CTFT

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$$

- STFT

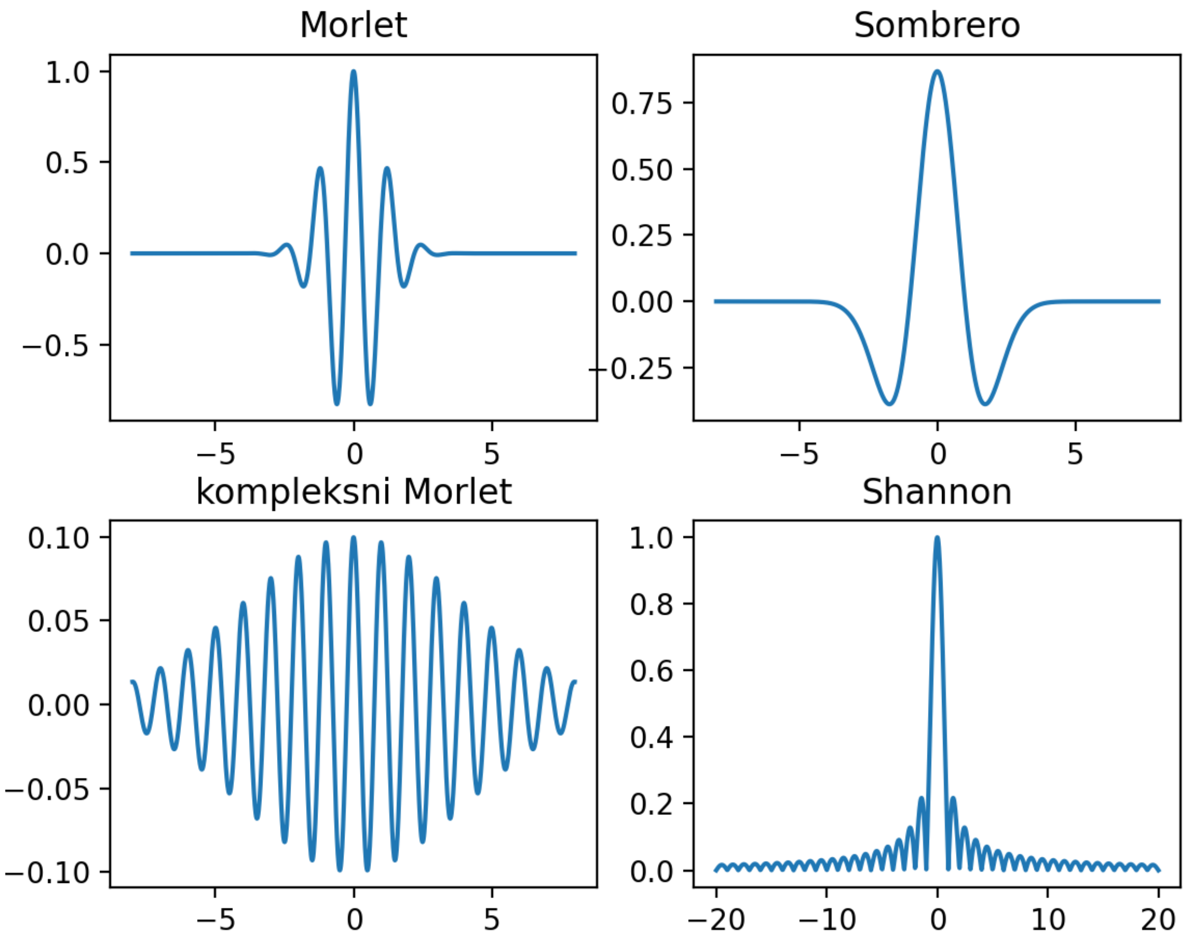
$$X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)g^*(t-\tau)e^{-j\omega t}dt$$

- zamijenimo $g(t-\tau)e^{j\omega t} \rightarrow \frac{1}{\sqrt{|a|}}\Psi\left(\frac{t-\tau}{a}\right)$
- $\Psi(t)$ lokalna analizirajuća funkcija željenih svojstava, τ pomak, a skala
- CWT

$$X(\tau,a) = \frac{1}{\sqrt{|a|}}\int_{-\infty}^{\infty} x(t)\Psi^*\left(\frac{t-\tau}{a}\right)dt$$

- * konjugacija → nije potrebna za realne valiće
- dvije dimenzije: vremenski pomak τ i skala a
- ovisi o izabranom valiću $\Psi(t)$

Valić $\Psi(t)$



- Morlet – `morl`

$$\Psi(t) = e^{-\frac{t^2}{2}} \cos 5t$$

- Sombrero (Mexican hat) – `mexh`

$$\Psi(t) = \frac{2}{\sqrt{3\sqrt[4]{\pi}}}(1 - t^2)e^{-\frac{t^2}{2}}$$

- Kompleksni Morlet – `cmorB-f`

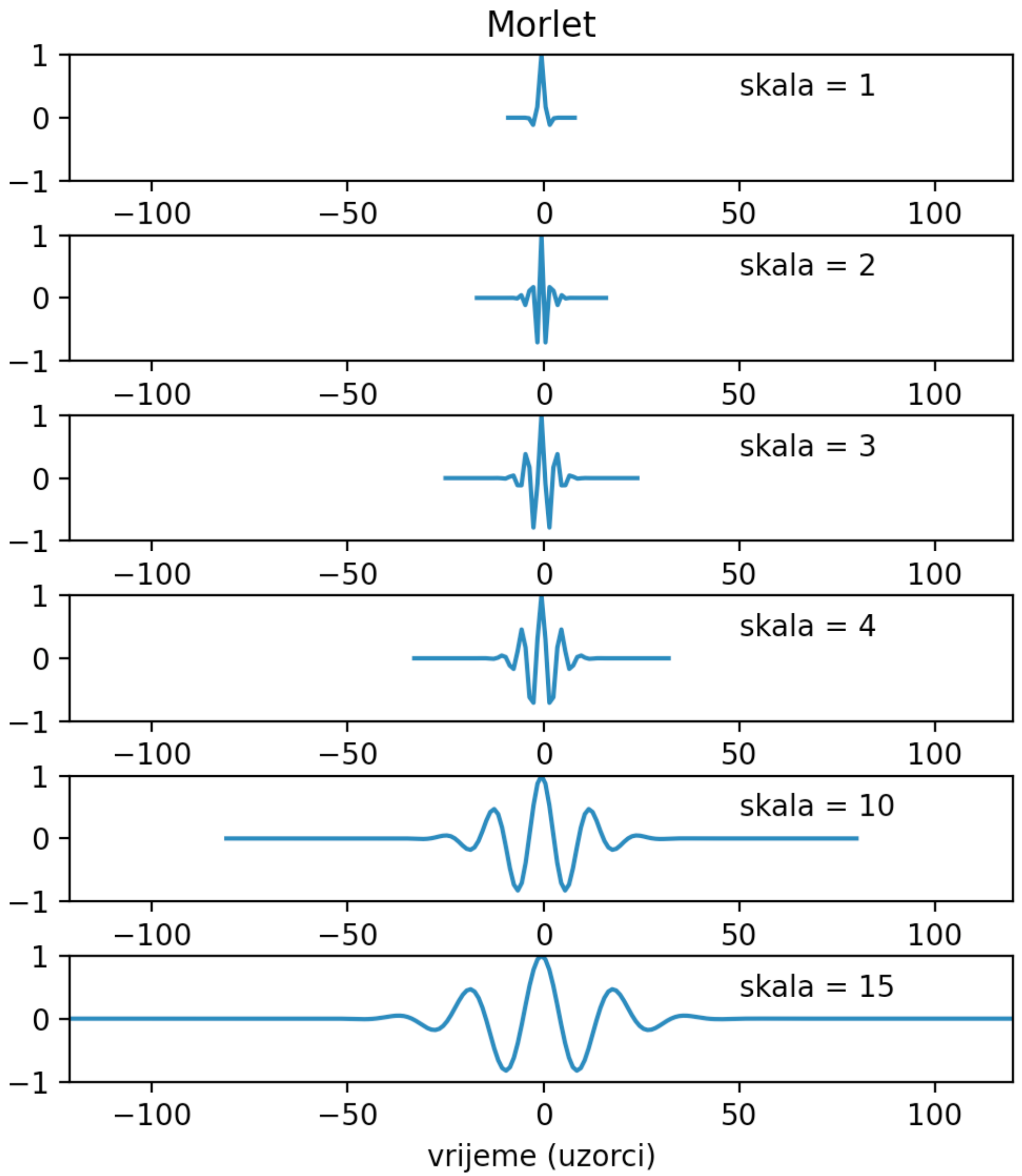
$$\Psi(t) = \frac{1}{\sqrt{\pi B}} e^{-\frac{t^2}{B}} e^{-j2\pi f t}$$

- Shannon – `shanB-f`

$$\Psi(t) = \sqrt{B}\,\frac{\sin(\pi B t)}{\pi B t} e^{-j2\pi f t}$$

Skala

- valić bez konstante: $\Psi\left(\frac{t-\tau}{a}\right)$
- skala a → obratno proporcionalna frekvenciji
- određuje koliko ćemo valić stegnuti ili rastegnuti
- broj valića se ne mijenja
- valić s konstantom: $\frac{1}{\sqrt{|a|}}\Psi\left(\frac{t-\tau}{a}\right)$
  → mijenjanjem skale mijenja se amplituda valića



Inverzna kontinuirana valićna transformacija (ICWT)

- CWT

$$X(\tau, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\Psi^*\left(\frac{t-\tau}{a}\right) dt$$

- ICWT

$$x(t) = \frac{1}{C} \int_{\tau=-\infty}^{\infty} \int_{a=-\infty}^{\infty} X(\tau, a) \frac{1}{\sqrt{|a|}} \Psi\left(\frac{t-\tau}{a}\right) da\, d\tau$$

- Uz $C = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{\omega} d\omega,\ 0 < C < \infty,\ \int_{-\infty}^{\infty} \Psi(t) dt\ = 0$

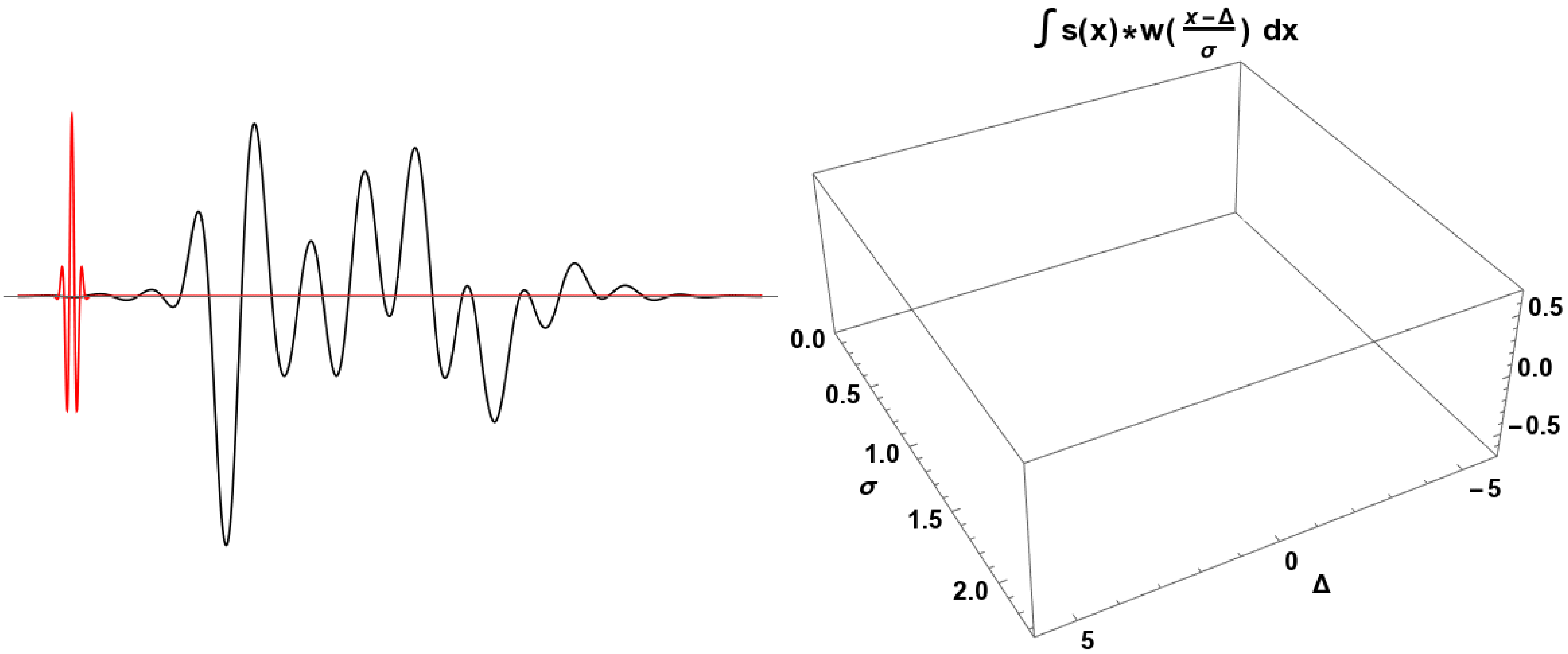
Diskretizacija CWT

- CWT

$$X(\tau,a)=\frac{1}{\sqrt{|a|}}\int_{-\infty}^{\infty}x(t)\Psi^*\left(\frac{t-\tau}{a}\right)dt$$

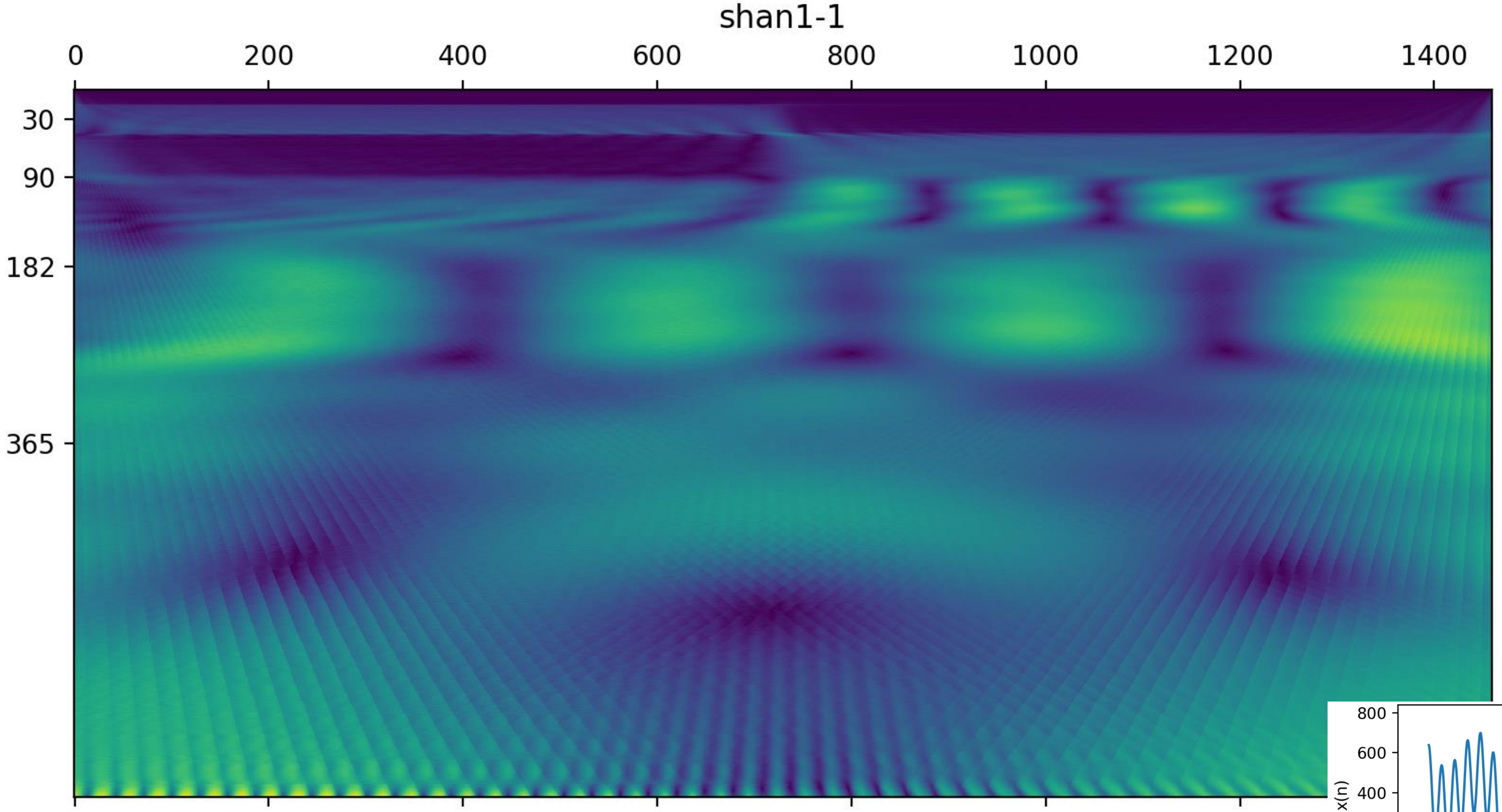
- Logaritamska podjela u skali $a=a_0^k$, najčešći izbor $a_0=2$
- Diskretni pomak ovisi o skali $\tau=mTa=mTa_0^k$
- Valić $\Psi_{m,k}=\frac{1}{\sqrt{a_0^k}}\Psi\left(\frac{t}{a_0^k}-mT\right)$
- Diskretizirani CWT

$$X(m,k)=\frac{1}{\sqrt{a_0^k}}\int_{-\infty}^{\infty}x(t)\Psi^*\left(\frac{t}{a_0^k}-mT\right)dt$$

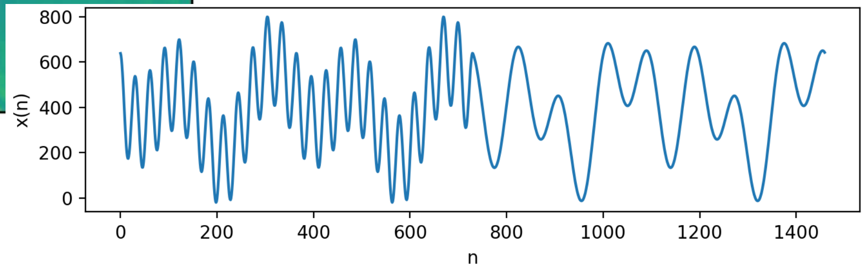


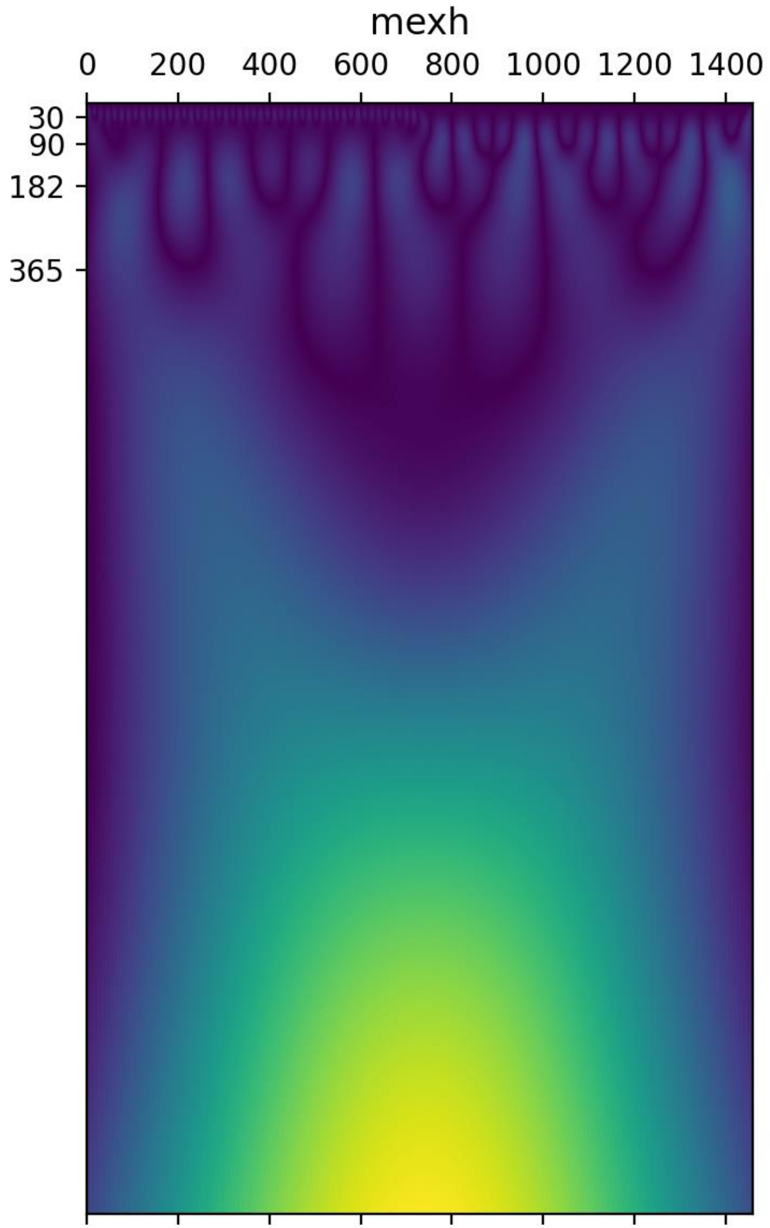
$\int s(x) * w(\frac{x-\Delta}{\sigma})\, dx$
0.0
0.5
1.0
σ
1.5
2.0
5
0
Δ
−5
0.5
0.0
−0.5

Signal koji se mijenja u vremenu, CWT

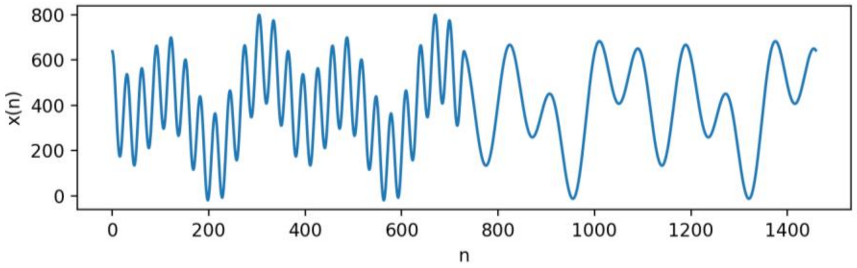


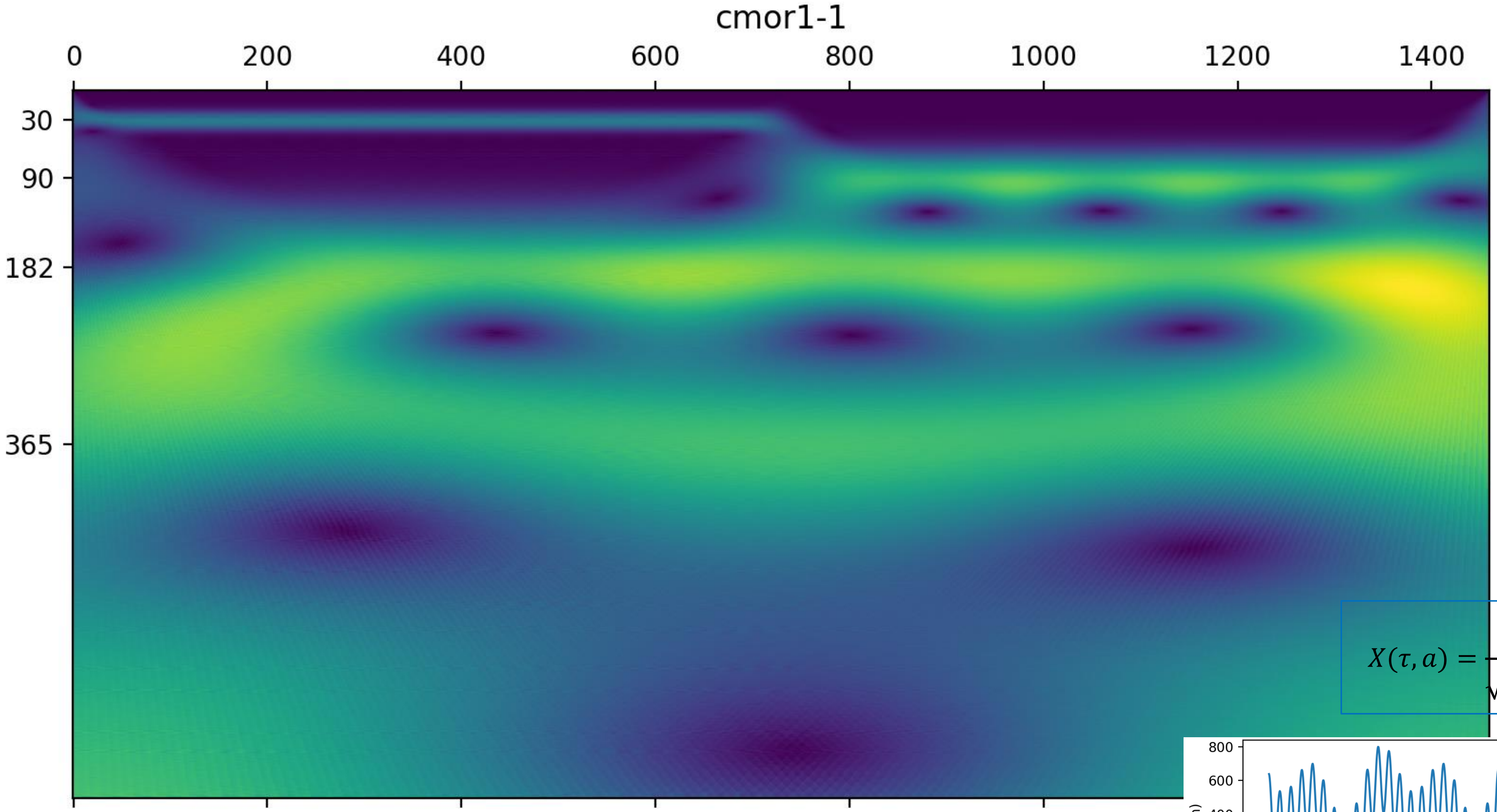
$$X(\tau, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\Psi^*\left(\frac{t-\tau}{a}\right) dt$$



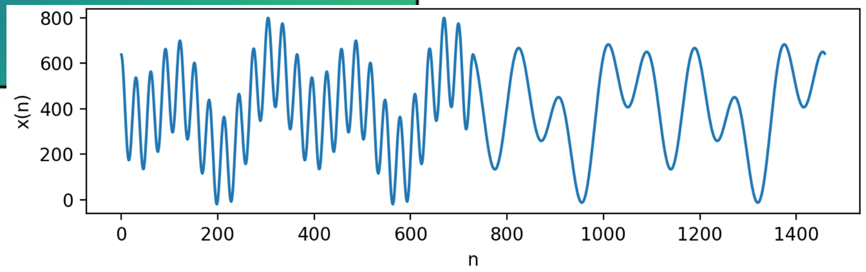


$$X(\tau,a)=\frac{1}{\sqrt{|a|}}\int_{-\infty}^{\infty}x(t)\Psi^{*}\left(\frac{t-\tau}{a}\right)dt$$



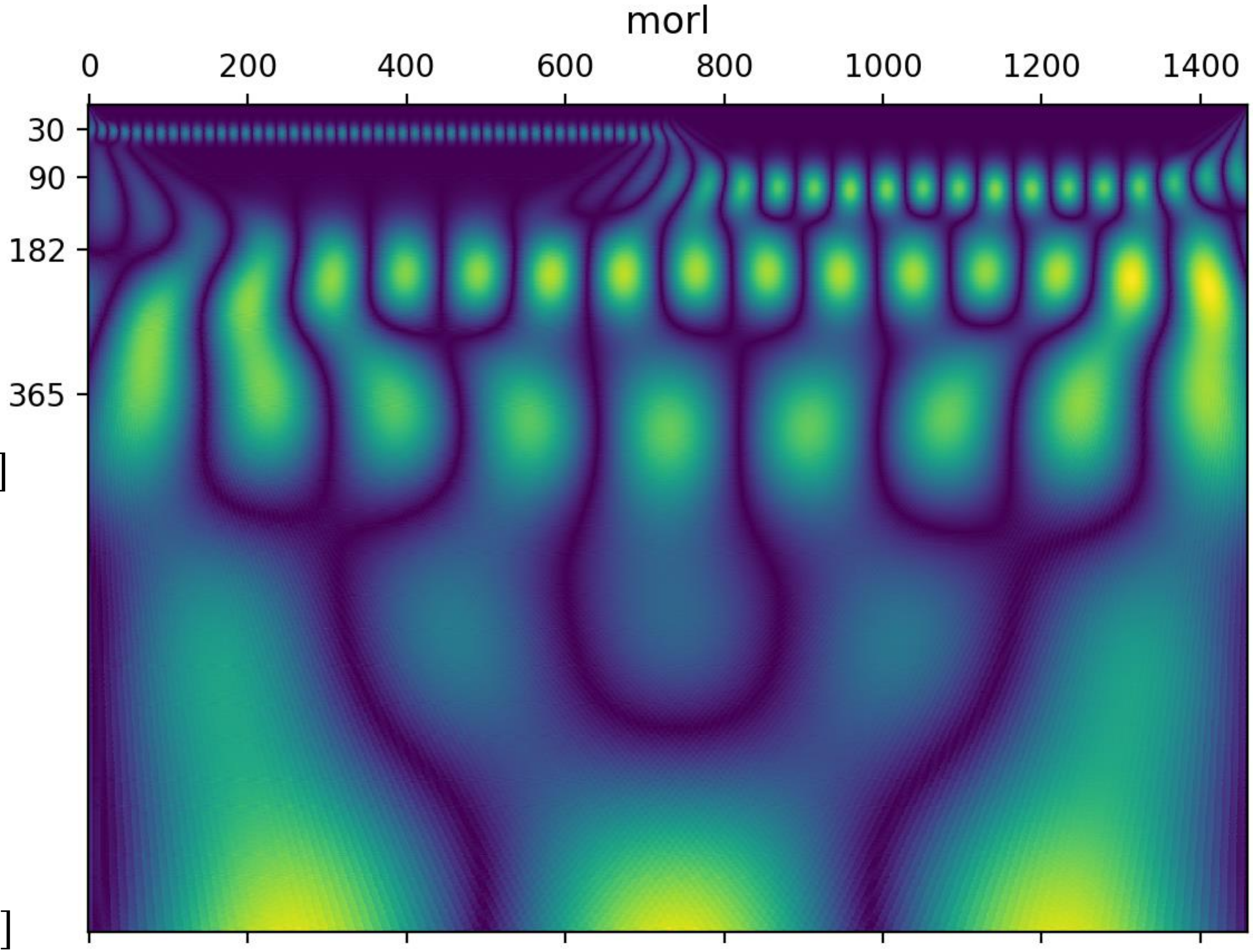


$$X(\tau, a) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t) \Psi^{*}\left(\frac{t-\tau}{a}\right) dt$$

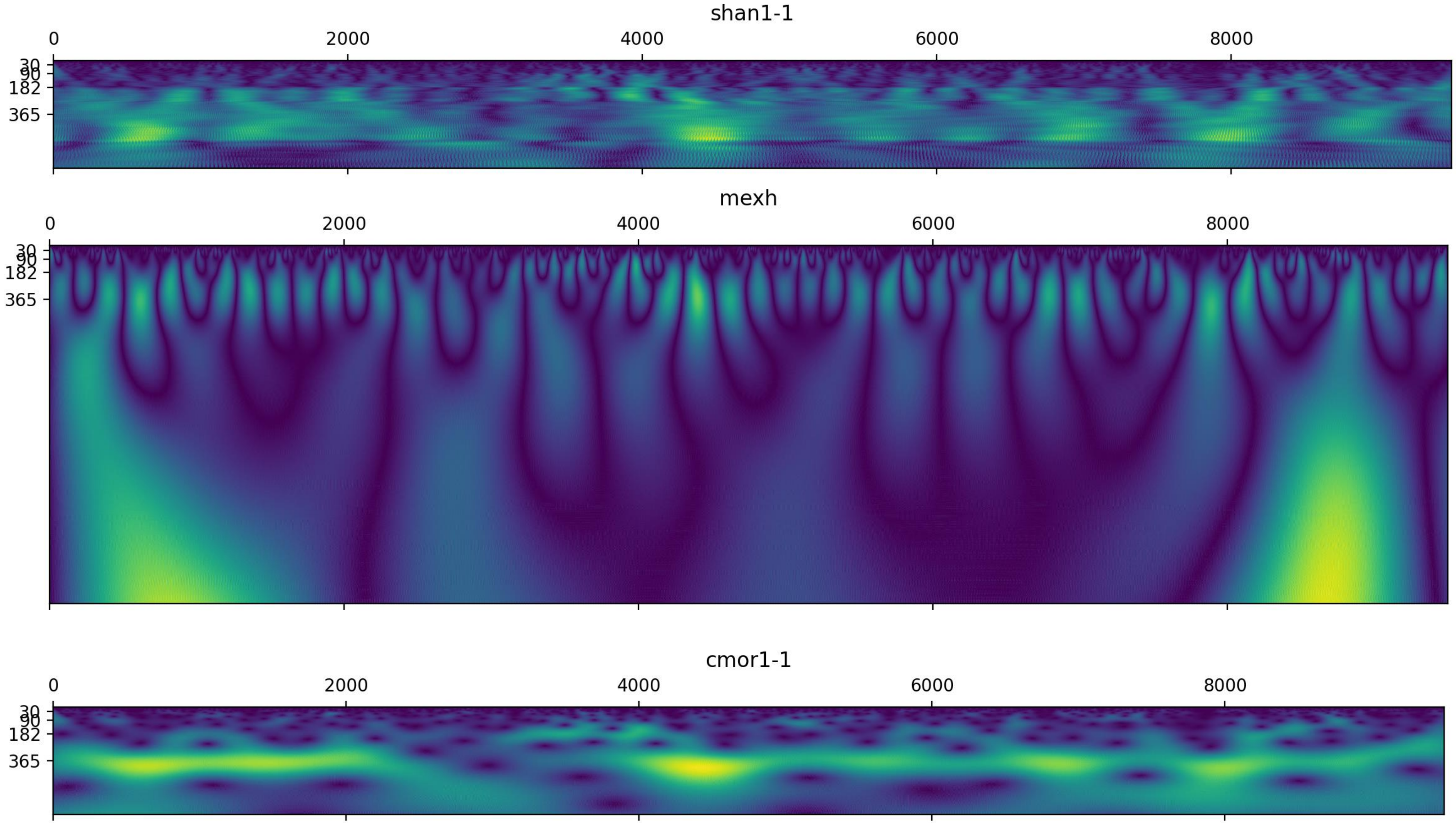


$$x_1(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}12n\right), \text{za } n \in \langle 0, 2T]$$

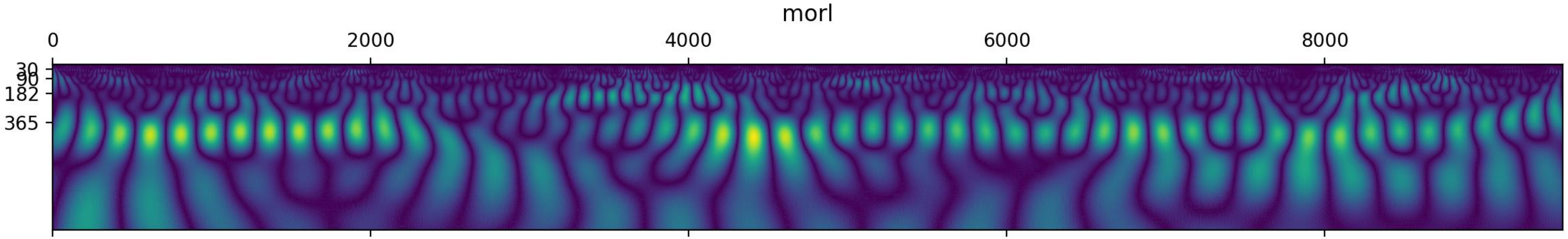
$$x_2(n) = 400 + 100\cos\left(\frac{2\pi}{365}n\right) + 150\cos\left(\frac{2\pi}{365}2n - \frac{365}{6}\right) + 200\cos\left(\frac{2\pi}{365}4n\right), \text{za } n \in \langle 2T, 4T]$$



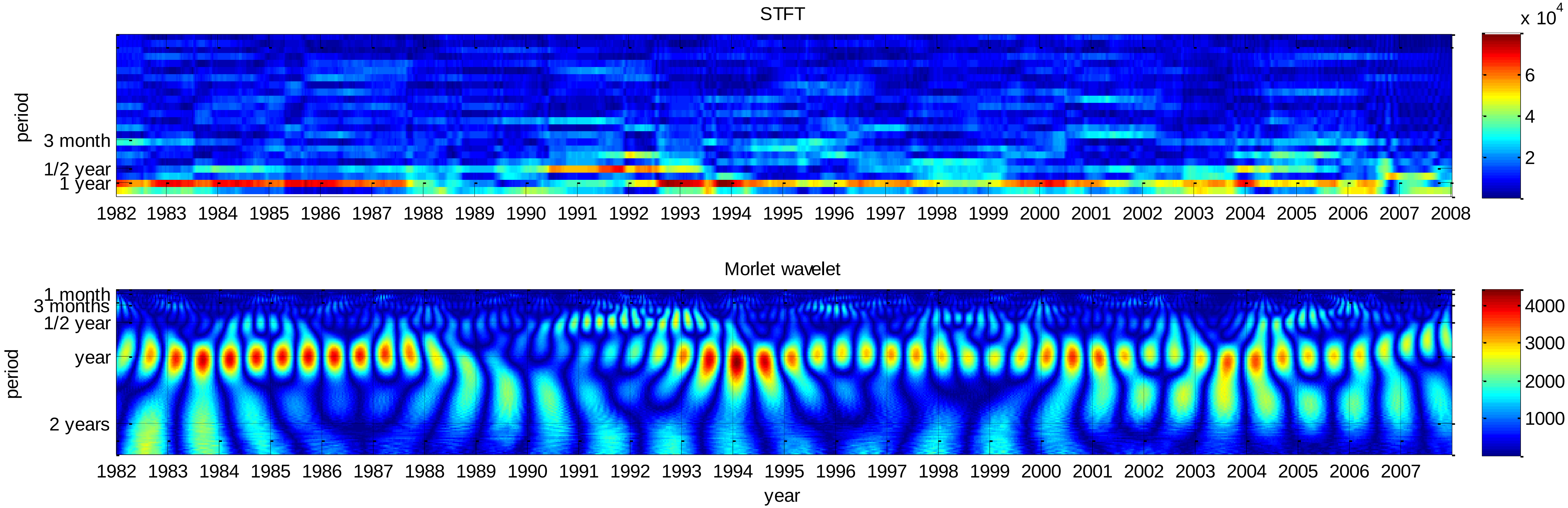
Kobaš i CWT

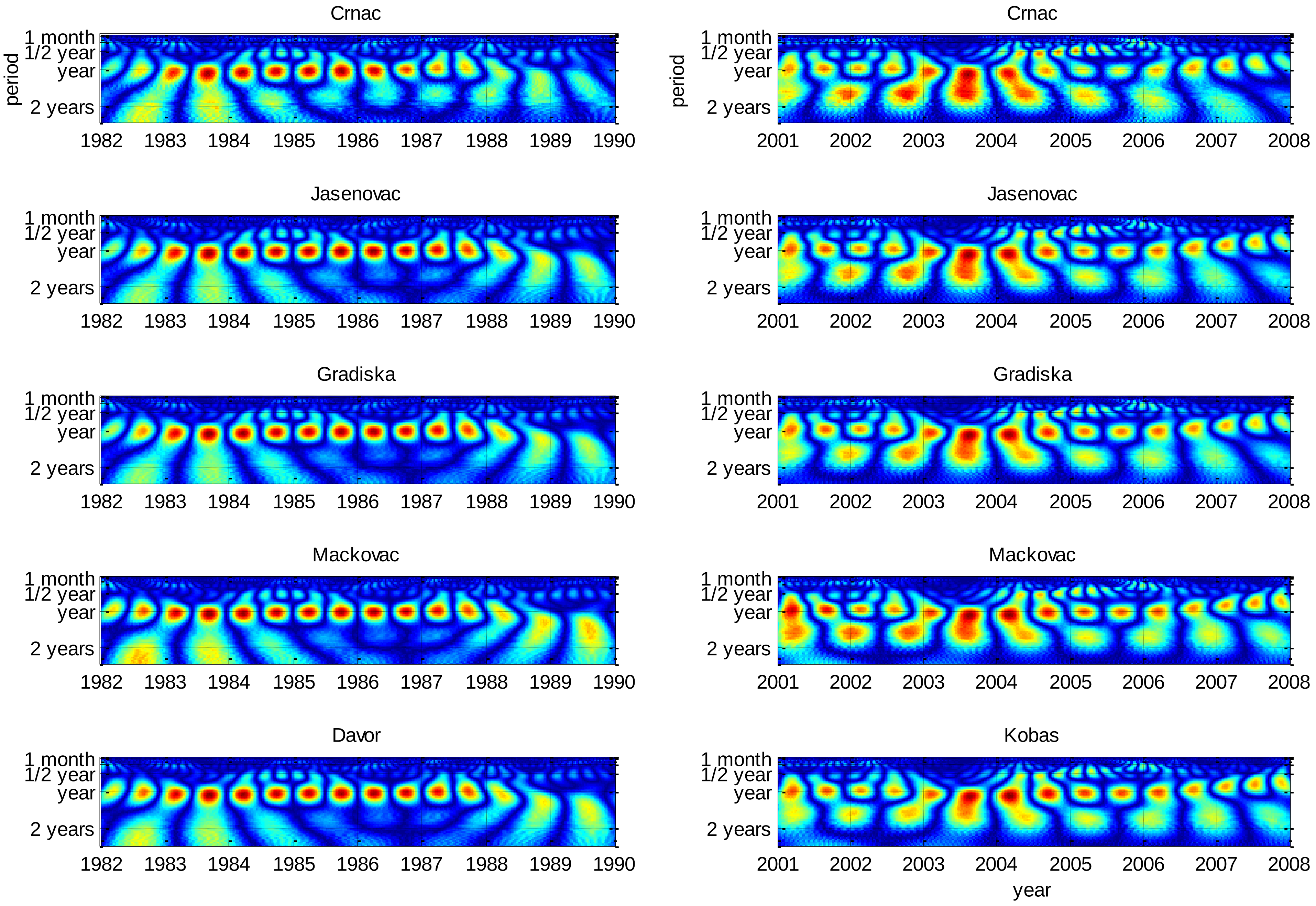

shan1-1
0
2000
4000
6000
8000
30
90
182
365
mexh
0
2000
4000
6000
8000
30
90
182
365
cmor1-1
0
2000
4000
6000
8000
30
90
182
365

Kobaš, CWT



Kobaš, STFT ili CWT




Crnac
Jasenovac
Gradiska
Mackovac
Davor
Kobas
period
1 month
1/2 year
year
2 years
1982
1983
1984
1985
1986
1987
1988
1989
1990
2001
2002
2003
2004
2005
2006
2007
2008
year

PyWavelets

- Biblioteka s funkcijama za valićnu transformaciju:
  - CWT, DWT, SWT
  - 1D, 2D i nD
- Kompatibilno s Matlab Wavelt Toolbox

```
import pywt
```

C

D

S

Primjer: CWT u Pythonu

```
import numpy as np
import matplotlib.pyplot as plt
import pywt

print(pywt.wavelist(kind='continuous'))  # popis obitelji kontinuiranih valica

valic = 'morl'  # izabrani valic

w = pywt.ContinuousWavelet(valic) # kreiranje objekta w koji sadrzi sve o valicu morl
#print(w) # ispis informacije o obitelji valica
psi, t = w.wavefun(level=10) # vrijednosti Valicne funkcije Psi

plt.figure(1)
plt.plot(t,psi)
plt.title(valic)

# generiranje liste sa zanimljivim skalama u rasponu 0.1 do 2T, uz korak "step"
step = 0.7
skala = np.arange(0.1,2*T,step)

# racunanje CWT signala x, uz skalu "skala" i izabrani valic
coef, freqs = pywt.cwt(x,skala,valic)

plt.matshow(abs(coef)) # coef je matrica vrijednosti CWT
plt.yticks([T//12, T//4, T//2, T]) # oznake na y-osi
plt.title(valic)

plt.show()
```