

# 傅立叶变换光谱测量技术实验报告

黄润华

Ocean University of China  
email@noreply.com

杨超

Ocean University of China  
email@somewhere.com

摘要—本实验报告为傅立叶变换光谱测量技术第五次实验报告,本报告采用 Python 仿真动镜倾斜误差与采样间隔误差两个因素同时影响下的傅里叶变换光谱测量系统的光谱测量曲线。误差可以是随机、线性或正弦变化的。以 632.8nm 的 HeNe 激光和 532nm 的 VAG 激光为例。采样间隔 79.1nm。本仿真以 632.8nm 的 HeNe 激光为例。

Index Terms—Optical spectrum, python, fft

## I. 实验目的

1. 思考动镜倾斜误差与采样间隔误差两个因素同时影响下的傅里叶变换光谱测量曲线形状;
2. 学习掌握 Python 绘图库基础知识。

## II. 实验原理

### A. 单色线的干涉方程

理论上单色线的干涉图可以用公式 (1) 来描述

$$I(x) = 2\cos(2\pi\sigma_0 x) \quad (1)$$

### B. 采样间隔误差对干涉方程的影响

人为进行动镜的平移过程通常掺杂着过程噪声。过程噪声的形式多种多样,常见的简单过程噪声类型为正弦噪声、随机噪声与线性噪声。然而,外部环境是一个不可预测的变量,在不同时刻的噪声类型通常是不相关且随机的,这种随机性带给光谱测量的影响是复杂性的。在操作不当的情况下,动镜移动的距离受到过程噪声的影响较大时,得到的傅立叶变换光谱测量曲线具备较低的可信度同时带来了阅读上的复杂性。

动镜移动时过程噪声带来的采样间隔误差是不可避免的,采样间隔误差的直接影响主要体现在对公式 (1) 中扫描长度  $x$  的影响:

$$I(x) = \cos(2\pi\sigma_0(x + \varepsilon_s)) \quad (2)$$

采样过程通常会选取  $2^{12}$  到  $2^{20}$  个采样点,每次采样均有可能存在一定的采样误差,采样误差经过多次迭

代后对最终傅立叶光谱测量曲线带来的影响是可观且巨大的。

本次仿真模拟的采样间隔误差均为简单采样间隔误差,即代表每次采样时的误差类型为单一的正弦噪声误差、随机噪声误差或线性噪声误差。同时假设采样过程均为独立事件,前后采样过程噪声互不相干且独立。

正弦形式的采样间隔误差表达式如下:

$$\varepsilon_s = \frac{\lambda_0}{16} \sin(2\pi\sigma_\varepsilon x) \quad (3)$$

其中波数 (Wave number) 数值为  $\sigma_\varepsilon = 0.4 \times 10^4 m^{-1}$ 。

随机形式的采样间隔误差表达式可以采用均匀分布来表示:

$$\varepsilon_s \sim U(-\frac{\lambda_0}{5}, \frac{\lambda_0}{5}) \quad (4)$$

线性形式的采样间隔误差表达式可以用正比例函数来表示:

$$\varepsilon_s(k) = k, \quad k \in [0, 79.1 \times 10^9 m] \quad (5)$$

### C. 动镜倾斜误差对干涉方程的影响

动镜倾斜误差带来的影响从干涉图上看为在干涉图上添加一个特定的噪声:

$$I = \cos(2\pi\sigma_0 x) + \varepsilon_m \quad (6)$$

本次仿真模拟的动镜倾斜误差均为简单动镜倾斜误差,即代表动镜倾斜误差类型为单一的正弦噪声误差、随机噪声误差或线性噪声误差。

正弦形式的动镜倾斜误差表达式如下:

$$\varepsilon_m = \sin(2\pi\frac{\sigma_0}{16}x) \quad (7)$$

随机形式的动镜倾斜误差表达式可以采用均匀分布来表示:

$$\varepsilon_m \sim U(-1, 1) \quad (8)$$

线性形式的动镜倾斜误差表达式可以用正比例函数来表示：

$$\varepsilon_m(k) = k, \quad k \in [0, 1] \quad (9)$$

#### D. 综合误差对干涉方程的影响

由式 (2) 与式 (6) 可以得到采样干涉误差与动镜倾斜误差两因素同时影响下的干涉图方程：

$$I = \cos[2\pi\sigma_0(x + \varepsilon_s)] + \varepsilon_m \quad (10)$$

### III. 实验内容

仿真动镜倾斜误差因素影响下的傅里叶变换光谱测量系统的光谱测量曲线。误差可以是随机、线性或正弦变化的。以 632.8nm 的 HeNe 激光和 532nm 的 VAG 激光为例。采样间隔 79.1nm。

### IV. 实验结果

本实验报告以 632.8nm 的波长为例。

### V. 结语

本实验完成了利用 Python 仿真动镜倾斜误差因素影响下的傅里叶变换光谱测量系统的光谱测量曲线。

### 附录

实验代码（以随机误差噪声为例）

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.fftpack import fft
from pylab import *
import random
import matplotlib.gridspec as gridspec

# 本程序模拟动镜倾斜误差影响下的傅里叶
# 变换光谱测量系统的光谱测量曲线
# 程序具体参数如下：
# - 采样间隔选取 79.1nm
# - 干涉图波长选取 632.8nm
# - 干涉图采样点数选取 215
# - 本程序只叠加一种噪声——随机噪声

def GetFFT(I0, Iw0, n0):
    Y0 = 2*abs(fft(I0, n0))
```

```
Y0_max = max(Y0);
Y0 = Y0/Y0_max
Y0 = Y0[:int(n0/2)]

Yw0 = 2*abs(fft(Iw0, n0))
Yw0_max = max(Yw0);
Yw0 = Yw0/Yw0_max
Yw0 = Yw0[:int(n0/2)]
return Y0, Yw0

# 波长为632.8nm
laimda0 = 632.8*10**(-9)
# 79.1nm的采样间隔
i = 79.1*10**(-9)
# 中心点的采样频
sigma0 = 1/laimda0
p1 = (-1)*(2**14)*79.1*10**(-9)
# 2**n个点
p2 = (2**14-1)*79.1*10**(-9)
# 无补零
x0 = np.arange(p1, p2, i)
n0 = n1 = 2**((int(np.log2(len(x0)))+1)
print("n0 length = %d" %n0)

I0 = np.cos(2*np.pi*sigma0*x0)

# 设置频谱图的横坐标
fs_0 = 1/i*np.arange(n0/2)/n0

# 此部分为叠加随机噪声
noise_rand = [random.uniform(-1, 1) for
_ in range(len(x0))]
# noise_rand = [random.uniform(0,
laimda0/5) for _ in range(len(x0))]

Iw1 = I0 + noise_rand

Y0, Yw1 = GetFFT(I0, Iw1, n0)

# 设置频谱图的横坐标
fs_1 = 1/i*np.arange(n0/2)/n0
```

```

best_Y0_average_range = 0.5*np.ones((
    Yw1.size , 1))

figure(1)

gs = gridspec.GridSpec(3, 6)
gs.update(wspace=0.5, hspace=0.3)

subplot(gs[0, :6])
plt.plot(x0, noise_rand)
plt.xlim(-0.00005, 0.00005)
plt.title("Waveform of random noise")

subplot(gs[1, 0:3])
s1, = plt.plot(x0, I0)
plt.legend(handles=[s1], labels=['Sample
    number =  $2^{15}$ '], loc='upper
    right')
plt.xlim(-6*(10**(-6)), 6*(10**(-6)))
plt.title("$I_0$ Original Interferogram
    ")

subplot(gs[1, 3:6])
s1, = plt.plot(fs_1, Y0, marker='o', ms
    =5)
p1, = plt.plot(fs_0,
    best_Y0_average_range)
plt.legend(handles=[p1], labels=['FWHM =
    385.8  $\text{m}^{-1}$ '], loc='upper right'
    )
plt.title("$I_0$ Original spectrum
    curve")
plt.xlim(1.5758*(10**6),
    1.58622*(10**6))

subplot(gs[2, 0:3])
plt.plot(x0, Iw1)
plt.xlim(-6*(10**(-6)), 6*(10**(-6)))

```

```

plt.title("Interferogram of $I_0$ after
    superimposing random noise")

subplot(gs[2, 3:6])
plt.plot(fs_1, Yw1, marker='o', ms=5)
p2, = plt.plot(fs_0,
    best_Y0_average_range)
plt.legend(handles=[p2], labels=['FWHM =
    386.9  $\text{m}^{-1}$ '], loc='upper right'
    )
plt.title("After superimposing random
    noise")
plt.xlim(1.5758*(10**6),
    1.58622*(10**6))

plt.suptitle("632.8nm – Comparison of
    the original signal and the signal
    after adding random noise", fontsize
    = 20)

plt.show()

```