

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四次实验报告,本报告采用 Python 仿真同时考虑有限扫描长度和采样间隔误差两因素影响下的傅里叶变换光谱测量系统的光谱测量曲线。实验仿真误差选取是随机、线性或正弦变化的误差,仿真采样间隔为 79.1nm。本仿真以 632.8nm 的 HeNe 激光和 532nm 的 YAG 激光为例。

Index Terms—Optical spectrum, python, fft

I. 实验目的

1. 思考并仿真扫描长度误差对傅里叶变换光谱测量曲线的影响;
2. 思考并仿真采样间隔误差对傅里叶变换光谱测量曲线的影响;
3. 学习利用 Python 仿真不同影响条件下光谱测量曲线。

II. 实验原理

A. 分辨率和分辨能力

光谱仪的光谱分辨率是其定性区分彼此非常接近的两个光谱峰的能力的量度。通常,光谱分辨率由仪器线形状 (ILS) 的半峰全宽 (FWHM) 表示,光谱仪的输出光谱具有纯单色输入辐射。

光谱分辨率 R 由下式定义

$$R = \frac{\sigma_{max}}{\delta\sigma} \quad (1)$$

其中 σ_{max} 是光谱仪设计运行的最大波数。

1) 分辨率与最大 OPD 之间的关系: 根据公式

$$B_c(\sigma) = B_{rel}(\sigma) + iB_{iol}(\sigma) = \int_{-\infty}^{\infty} I(x) \text{rect}(x) e^{-i2\pi\sigma x} dx \quad (2)$$

$$\begin{cases} cB(\sigma) = 2 \left| \int_{-\infty}^{+\infty} I(x) \text{rect}(x) e^{-2\pi\sigma x} dx \right| \\ \phi(\sigma) = \arctan \left[\frac{B_{iol}(\sigma)}{B_{rel}(\sigma)} \right] \end{cases} \quad (\sigma \geq 0) \quad (3)$$

其中矩形函数定义为

$$\text{rect}(x) = \begin{cases} 1 & |x| < L \\ 0 & |x| > L \end{cases}$$

因此,理论上 FTS 的光谱分辨率取决于可移动镜的扫描范围

$$B_e(\sigma) = \begin{cases} lB(\sigma)/2 & (\sigma \geq 0) \\ B(-\sigma)/2 & (\sigma \leq 0) \end{cases} \quad (4)$$

理论上,完美单色线的干涉图可以用来描述

$$I(x) = 2\cos(2\pi\sigma_0 x) \quad (5)$$

将此表达式替换为方程 $B(\sigma)$ 、 $\phi(\sigma)$,考虑到实际上 $\sigma > 0$,可以得到 ILS $B(\sigma)$ 的表达式:

$$B_{ILS}(\sigma) = 2L \frac{\sin[2\pi(\sigma_0 + \sigma)L]}{2\pi(\sigma_0 + \sigma)L} + \frac{\sin[2\pi(\sigma_0 - \sigma)L]}{2\pi(\sigma_0 - \sigma)L} \quad (6)$$

$$\approx 2L \times \frac{\sin[2\pi(\sigma_0 - \sigma)L]}{2\pi(\sigma_0 - \sigma)L} \quad (7)$$

$$= 2L \sin c[2\pi(\sigma_0 - \sigma)L] \quad (8)$$

如图 1所示,光谱从一条线 (δ 函数) 扩展到 sinc 函数形状 (仪器线形)。

$$(\delta\sigma)_{\text{linewidth}} = \frac{1.21}{2L} \quad (9)$$

2) 分离共振: 分辨率可以通过分离图 2所示光谱中相同强度 (或共振) 的两条单色线 (波数和) 来描述。如果两个线峰值之间的下降幅度大于线峰值的 20%,则可以声称解决了两个共振,此时

$$(\delta\sigma)_{\text{separation}} = \frac{1.46}{2L} \quad (10)$$

3) 瑞利准则: 瑞利准则分离两个 ILS 的峰值,使得一个共振的最大值落在另一个共振的零点处,因此,仪器线形状决定的分辨率是

$$\delta\sigma = \frac{1}{2L} \quad (11)$$

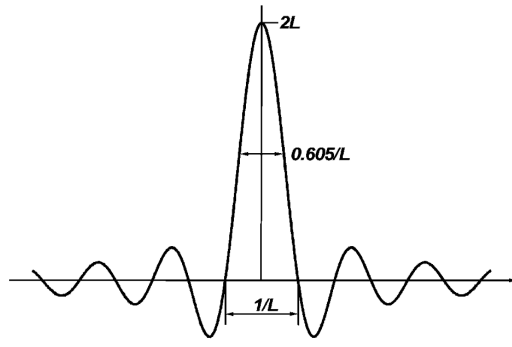


图 1. $B_{115}(\sigma)$

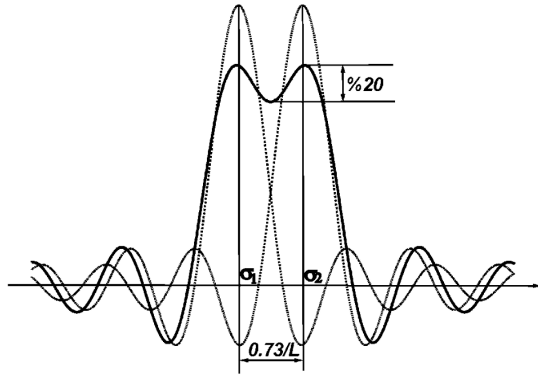


图 2. 分离共振解析结果

B. 分辨率与发散角度之间的关系

图 3 是迈克尔逊干涉仪的等效图，根据图 3 与图 4 可以计算出 OPD 与夹角 θ 之间的关系

$$OPD = 2 \times \frac{x/2}{\cos \theta} - x \tan \theta \sin \theta = x \cos \theta \quad (12)$$

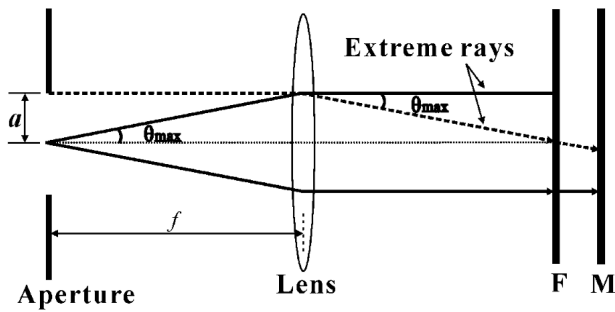


图 3. 迈克尔逊干涉仪的等效图

我们可以得到扩展源的归一化干涉图，对着一个立

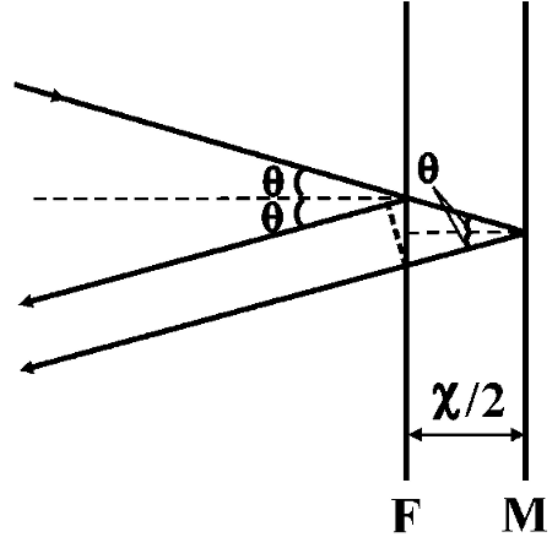


图 4. 迈克尔逊干涉仪接收端放大图

体角 Ω_{max} 是

$$I(x, \Omega_{max}) = \frac{1}{\Omega_{max}} \int_0^\infty B(\sigma) \int_0^{\Omega_{mix}} \cos(2\pi\sigma x \cos \theta) d\Omega d\sigma$$

根据立体角的公式，上述式子可以化简为

$$I(x, \Omega_{max}) = \int_0^\infty B(\sigma) \text{sinc} \frac{\Omega_{max}\sigma x}{2} \cos[2\pi\sigma x - \frac{\Omega_{max}\sigma x}{2}] d\sigma$$

对于单色光源其满足

$$B(\sigma) = \delta(\sigma - \sigma_0)$$

其仪器线轮廓可以写为

$$\begin{cases} B_{Div-1LS}(\sigma) = [\pi/(\sigma_0\Omega_{max})] \text{rect}(\sigma_1, \sigma_2) \\ \quad + [\pi/(\sigma_0\Omega_{max})] \text{rect}(-\sigma_2, -\sigma_1) \\ \sigma_2 = \sigma_0 - \sigma_0\Omega_{max}/2\pi \\ \sigma_1 = \sigma_0 \end{cases}$$

于是可以得到

$$B_{Div-1LS}(\sigma) = [\pi/(\sigma_0\Omega_{max})] \text{rect}(\sigma_1, \sigma_2) \quad (13)$$

$$\bar{\sigma} = \sigma_0 [1 - (\Omega_{max}/4\pi)] \quad (14)$$

总的波数差或分辨率是

$$\delta\sigma = \frac{\sigma_0\Omega_{max}}{2\pi}$$

考虑到可移动镜的扫描长度有限, 实际 FTS 的 ILS 应该是

$$B_{ILS}(\sigma) = B_{Div-ILS}(\sigma) * 2L \sin c[2\pi\sigma L] \quad (15)$$

其中 L 是可移动镜的最大位移, $*$ 是卷积算子。图 4 中两个极端射线的 OPD 之间的差异公式如下:

$$\Delta OPD = 2L - 2L \cos \theta_{\max} \approx L\theta_{\max}^2 = L \frac{a^2}{f^2}$$

其中 a 是入口孔径的半径, f 是准直器的焦距。当 ΔOPD 等于 $\lambda/2$ 时, 两条光线异相, 它们之间会发生破坏性叠加。对于宽带辐射输入, 存在的最短波长决定了 FTS 的 ΔOPD 的最大有效值, 如下式所示:

$$\Delta OPD \leq \frac{\lambda_{\min}}{2} = \frac{1}{2\sigma_{\max}} \quad (16)$$

我们可以得到 a 的值应满足以下不等式, 以获得大于 R 的分辨率:

$$a \leq \frac{f}{\sqrt{R}} \quad (17)$$

C. 傅里叶变换光谱仪的优点

与色散光谱仪相比, FTS 具有许多广泛宣传的优点。然而, 只有吞吐量 (Jacquinot) 和多路复用 (Felgett) 优势是 FTS 操作原理所固有的, 而不是特定的工程设计。

1) 吞吐量或 Jacquinot 优势: 吞吐量优势是 FTS 可以具有大的圆形入口孔, 其面积比分散光谱仪中的狭缝的面积大得多, 以获得相同的分辨率。吞吐量定义为, 其中是限制孔径的面积, 并且是图 5 中所示的准直或聚焦光学器件所对应的立体角。

FTS 的最大吞吐量由下式给出

$$T_{FTS} = A\Omega_s = \frac{\pi}{R} A_{\text{mirror}} \quad (18)$$

其中 A_{mirror} 是镜子的投影面积。

对于光栅光谱仪, 实现分辨率 R 的能量吞吐量受其狭缝区域和准直光学系统的限制, 并由下式给出:

$$T_{\text{grating}} = \frac{l}{f_g R} A_{\text{grating}} \quad (19)$$

其中 f_g 是准直光学系统的焦距, l 是狭缝的高度, A_{grating} 是光栅的投影面积。对于光栅光谱仪, $\frac{l}{f_g R}$ 小于 $1/20$ 。

因此, 对于相同的分辨率和类似的仪器尺寸, FTS 比光栅光谱仪具有高 60 倍的能量收集能力。因此,

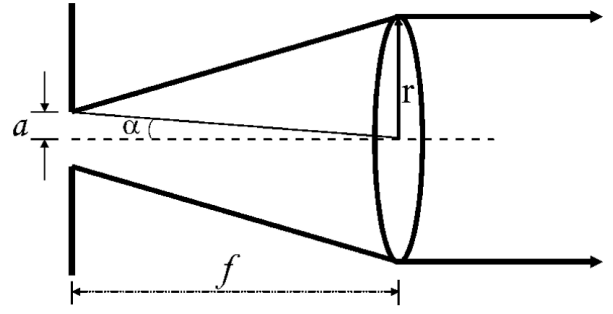


图 5. 孔径和立体角的关系图

Jacquinot 的优势使得 FTS 更适合于弱信号测量, 其中检测器噪声占主导地位, 频谱信噪比 (SNR) 与吞吐量成比例地增加。

2) Multiplex 或 Fellgett 的优势: 多路复用的优点是 FTS 在扫描周期内同时观察来自给定频谱的整个范围的所有频谱信息。因此, 通过使用 FTS 和使用单色仪可获得的 SNR 的比率:

$$\frac{SNR_{FTS}^S}{SNR_{\text{monochromator}}} = N^{\frac{1}{2}} \quad (20)$$

FTS 的多重优势仅存在于红外和远红外信号的测量中, 并且在可见 - 紫外信号的检测中丢失, 因为在红外检测中, 与信号电平无关的检测器噪声占主导地位。量子噪声在可见光 - 紫外信号检测中占主导地位。

3) Connes 的优势: FTS 的波数范围来自 He-Ne 激光条纹, 其作为每次扫描中采样位置的内部参考。这种激光的波数非常准确, 并且非常稳定。因此, 干涉仪的波数校准更精确, 并且具有比分散仪器的校准好得多的长期稳定性。

III. 实验内容

仿真同时考虑有限扫描长度和采样间隔误差两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、线性或正选变化的。以 632.8nm 的 He-Ne 激光和 532nm 的 YAG 激光为例。(采样间隔为 79.1nm)

IV. 实验结果

本实验报告以 632.8nm 的波长为例。

632.8nm - Comparison between the original waveform and the waveform after adding three kinds of noise

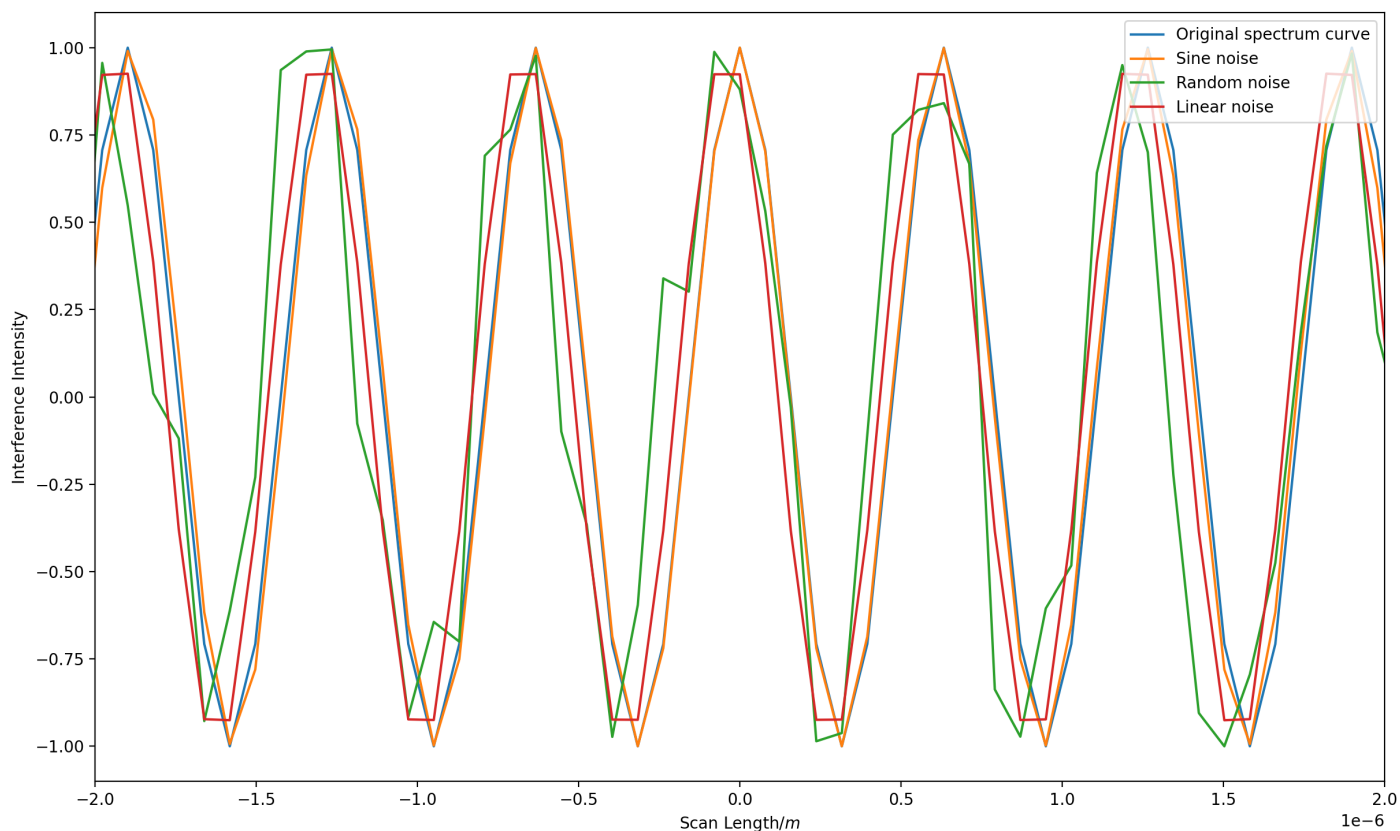


图 6. 原始信号的干涉图与叠加三种不同噪声后干涉图的比较

A. 三种不同的叠加噪声

图片 7、图 8与图 9显示了本次仿真的三种不同的叠加噪声，其中图 7为叠加的正弦噪声，图 8为叠加的随机噪声，图 9为叠加的线性噪声。

B. 噪声叠加结果

图片 10、图 11与图 12显示了三种不同噪声的叠加结果，其中图 10叠加了正弦噪声，图 11叠加了随机噪声，图 12叠加了线性噪声。每张图片第一行代表未叠加噪声的干涉图与光谱曲线图，第二行代表添加部分噪声后的干涉图与光谱曲线图；左半部分代表无噪声干扰与叠加噪声干扰后的干涉图，右半部分代表无噪声干扰与叠加噪声干扰后的光谱曲线图。

图片 6将叠加三种不同噪声后的干涉图放在了一起，可以看到，叠加了线性噪声与随机噪声后，原始信

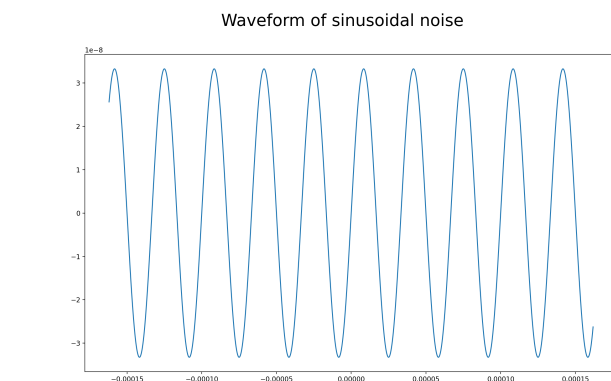


图 7. 叠加的正弦噪声

号的干涉图变化较大；叠加正弦噪声后的干涉图与原始信号干涉图差别只在幅值上，大体上与原始信号相符合。上述仿真说明线性噪声与随机信号噪声对原始光的

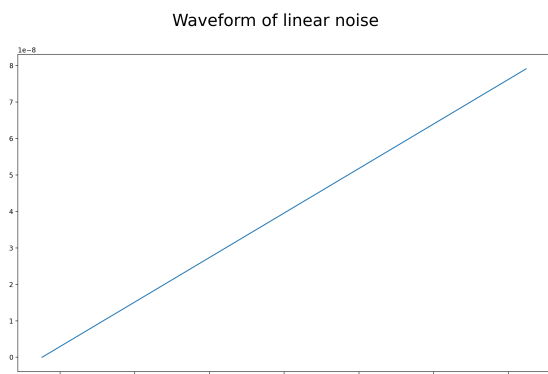


图 8. 叠加的随机噪声

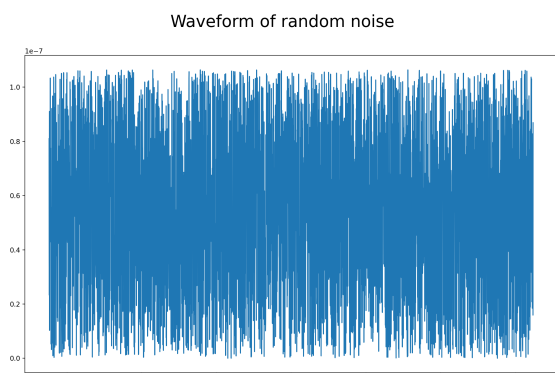


图 9. 叠加的线性噪声

632.8nm - Comparison of the original signal and the signal after adding sinusoidal noise

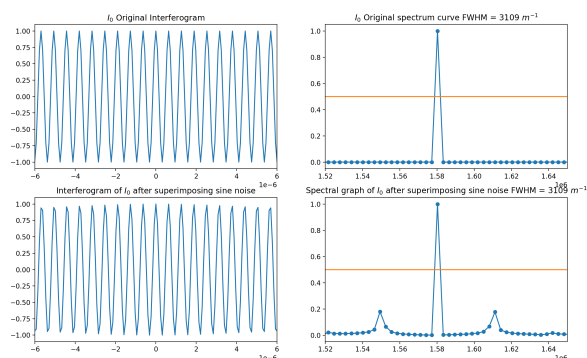


图 10. 正弦噪声叠加结果 (图片第一行代表未叠加噪声的干涉图与光谱曲线图, 第二行代表添加正弦噪声后的干涉图与光谱曲线图; 左半部分代表无噪声干扰与叠加正弦噪声干扰后的干涉图, 右半部分代表无噪声干扰与叠加正弦噪声干扰后的光谱曲线图。)

632.8nm - Comparison of the original signal and the signal after adding linear noise

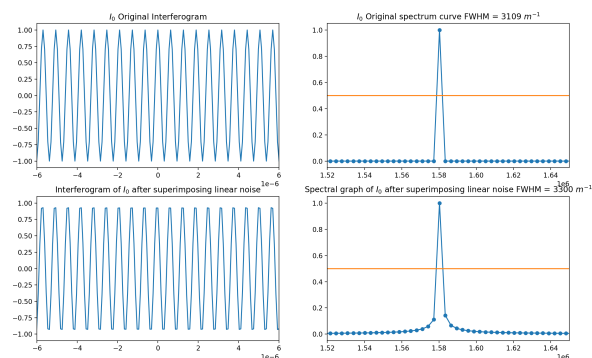


图 11. 随机噪声叠加结果 (图片第一行代表未叠加噪声的干涉图与光谱曲线图, 第二行代表添加随机噪声后的干涉图与光谱曲线图; 左半部分代表无噪声干扰与叠加随机噪声干扰后的干涉图, 右半部分代表无噪声干扰与叠加随机噪声干扰后的光谱曲线图。)

632.8nm - Comparison of the original signal and the signal after adding random noise

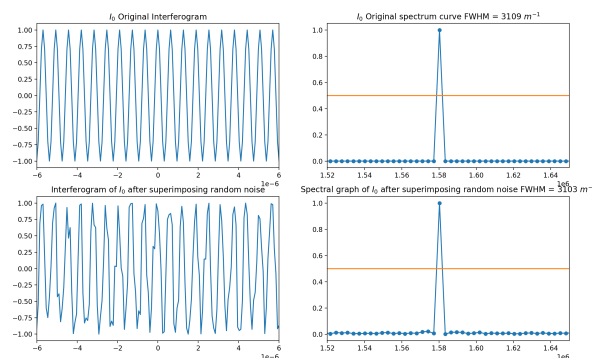


图 12. 线性噪声叠加结果 (图片第一行代表未叠加噪声的干涉图与光谱曲线图, 第二行代表添加线性噪声后的干涉图与光谱曲线图; 左半部分代表无噪声干扰与叠加线性噪声干扰后的干涉图, 右半部分代表无噪声干扰与叠加线性噪声干扰后的光谱曲线图。)

干涉图影响较大, 叠加正弦噪声对原始光的干涉图影响较小。

图片 13 将叠加三种不同噪声后的光谱测量曲线放在了一起, 可以看到, 叠加了噪声后, 原始信号周围出现了小部分的旁瓣, 叠加正弦噪声后旁瓣峰值较高, 这是由于正弦噪声信号的频谱存在两个幅值不为零的频率点。同时, 叠加线性噪声后的光谱测量曲线的半峰全宽与原始信号相比较为宽, 而叠加正弦噪声信号与随机信号后光谱测量曲线的半峰全宽与原始信号相比较无明显变化, 此现象说明线性噪声会使得原始波形的波形分辨率下降, 从频谱上分析即为使得原始光谱测量曲线的半峰全宽变宽。

632.8nm - Comparison between the original waveform and the spectrum measurement curve after adding three kinds of noise

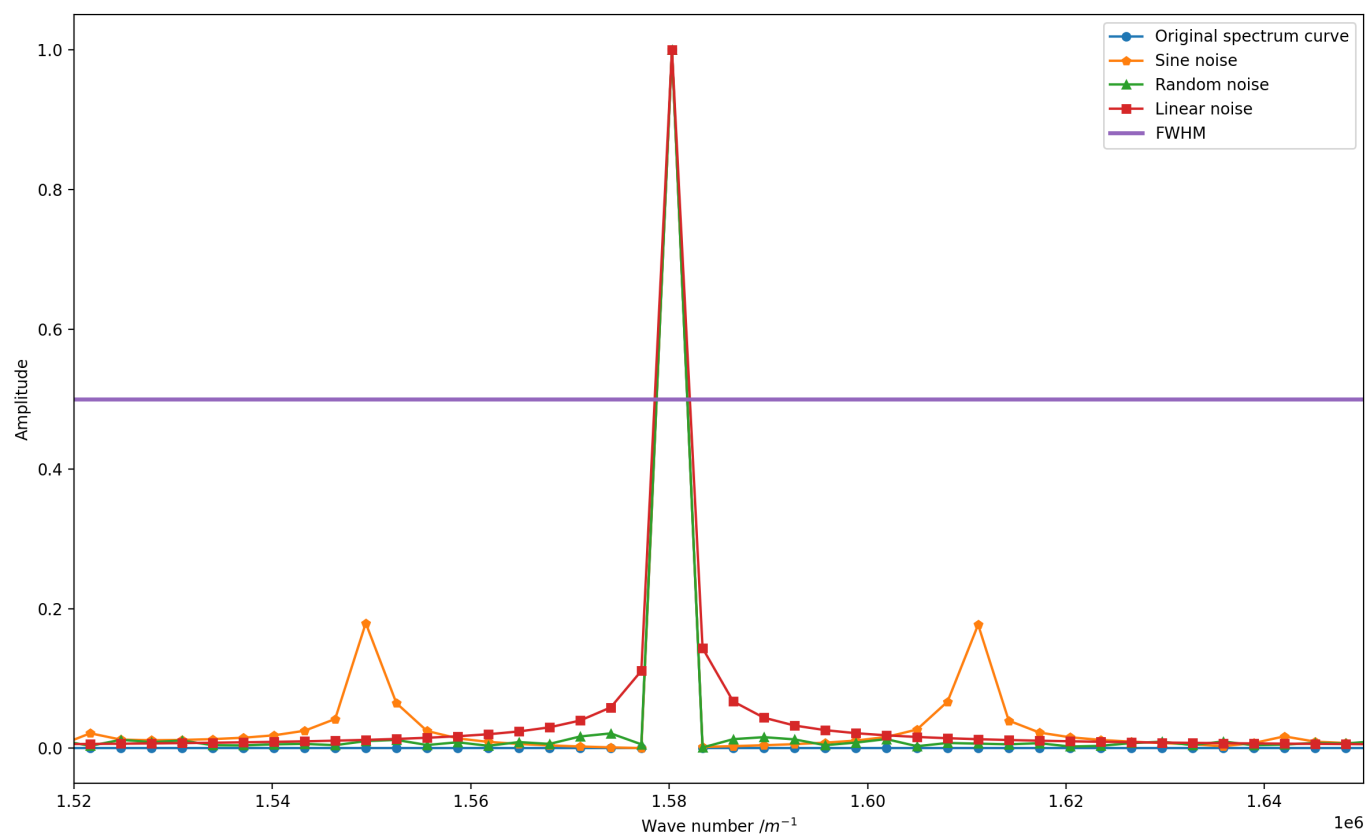


图 13. 原始信号的干涉图与叠加三种不同噪声后光谱测量曲线的比较

V. 结语

本实验完成了利用 Python 仿真同时考虑有限扫描长度和采样间隔误差两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

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

# 波长为532nm
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**11)*79.1*10**(-9)
# 2**n个点
p2 = (2**11-1)*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)

#####
# 此部分为叠加正弦噪声
#####
```

```

noise_sin1 = laimda0/16*np.sin(2*np
    .pi*(3*10**4)*x0)

I0 = np.cos(2*np.pi*sigma0*x0)
Iw0 = np.cos(2*np.pi*sigma0*(x0 +
    noise_sin1))

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)]

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

best_Y0_average_range = 0.5*np.ones
    ((Y0.size , 1))

figure(1)
plt.plot(x0, noise_sin1)
plt.suptitle("Waveform of
    sinusoidal noise", fontsize =
    25)

figure(2)
subplot(2,2,1)
plt.plot(x0, I0)
plt.xlim(-6*(10**(-6)), 6*(10**(-6))
    ))
plt.title("$I_0$ Original
    Interferogram")

subplot(2,2,2)
plt.plot(fs_0, Y0, marker='o', ms
    =5)

```

```

plt.plot(fs_0,best_Y0_average_range
    )
plt.title("$I_0$ Original spectrum
    curve FWHM = 3109 $m^{-1}$")
plt.xlim(1.52*(10**6), 1.65*(10**6)
    )

subplot(2,2,3)
plt.plot(x0, Iw0)
plt.xlim(-6*(10**(-6)), 6*(10**(-6))
    ))
plt.title("Interferogram of $I_0$
    after superimposing sine noise")

subplot(2,2,4)
plt.plot(fs_0, Yw0, marker='o', ms
    =5)
plt.plot(fs_0,best_Y0_average_range
    )
plt.title("Spectral graph of $I_0$
    after superimposing sine noise
    FWHM = 3109 $m^{-1}$")
plt.xlim(1.52*(10**6), 1.65*(10**6)
    )

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

#####
# 此部分为叠加随机噪声
#####
noise_rand = laimda0/5*(rand(1,len(
    x0)))
Iw1 = np.cos(2*np.pi*sigma0*(x0 +
    noise_rand[0]))

Yw1 = 2*abs(fft(Iw1,n0))
Yw1_max=max(Yw1);
Yw1 = Yw1/Yw1_max

```

```

Yw1 = Yw1[:int(n0/2)]

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

figure(3)
plt.plot(x0, noise_rand[0])
plt.suptitle("Waveform of random
             noise", fontsize = 25)

figure(4)
subplot(2,2,1)
plt.plot(x0, I0)
plt.xlim(-6*(10**(-6)), 6*(10**(-6))
        ))
plt.title("$I_0$ Original
           Interferogram")

subplot(2,2,2)
plt.plot(fs_1, Y0, marker='o', ms
        =5)
plt.plot(fs_0, best_Y0_average_range
        )
plt.title("$I_0$ Original spectrum
           curve FWHM = 3109 $m^{-1}$")
plt.xlim(1.52*(10**6), 1.65*(10**6)
        )

subplot(2,2,3)
plt.plot(x0, Iw1)
plt.xlim(-6*(10**(-6)), 6*(10**(-6))
        ))
plt.title("Interferogram of $I_0$
           after superimposing random noise
           ")

subplot(2,2,4)
plt.plot(fs_1, Yw1, marker='o', ms
        =5)

```

```

plt.plot(fs_0, best_Y0_average_range
        )
plt.title("Spectral graph of $I_0$
           after superimposing random noise
           FWHM = 3103 $m^{-1}$")
plt.xlim(1.52*(10**6), 1.65*(10**6)
        )

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

#####
# 此部分为叠加线性噪声
#####
noise_linear = linspace(0,i,len(x0)
                        )
Iw2 = np.cos(2*np.pi*sigma0*(x0 +
                             noise_linear))

Yw2 = 2*abs(fft(Iw2,n0))
Yw2_max=max(Yw2);
Yw2 = Yw2/Yw2_max
Yw2 = Yw2[:int(n0/2)]

# 设置频谱图的横坐标
fs_2 = 1/i*np.arange(n0/2)/n0

figure(5)
plt.plot(x0, noise_linear)
plt.suptitle("Waveform of linear
             noise", fontsize = 25)

figure(6)
subplot(2,2,1)
plt.plot(x0, I0)
plt.xlim(-6*(10**(-6)), 6*(10**(-6))
        ))
plt.title("$I_0$ Original
           Interferogram")

```



```

subplot(2,2,2)
plt.plot(fs_1, Y0, marker='o', ms
         =5)
plt.plot(fs_0, best_Y0_average_range
         )
plt.title("$I_0$ Original spectrum
         curve FWHM = 3109  $m^{-1}$ ")
plt.xlim(1.52*(10**6), 1.65*(10**6)
         )

subplot(2,2,3)
plt.plot(x0, Iw2)
plt.xlim(-6*(10**(-6)), 6*(10**(-6)
         ))
plt.title("Interferogram of $I_0$
         after superimposing linear noise
         ")

subplot(2,2,4)
plt.plot(fs_2, Yw2, marker='o', ms
         =5)
plt.plot(fs_0, best_Y0_average_range
         )
plt.title("Spectral graph of $I_0$
         after superimposing linear noise
         FWHM = 3300  $m^{-1}$ ")
plt.xlim(1.52*(10**6), 1.65*(10**6)
         )

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

figure(7)
l1, = plt.plot(x0, I0)
l2, = plt.plot(x0, Iw0)
l3, = plt.plot(x0, Iw1)
l4, = plt.plot(x0, Iw2)

```

```

plt.legend(handles=[l1, l2, l3, l4
                  ], labels=['Original spectrum
                              curve', 'Sine noise', 'Random
                              noise', 'Linear noise'], loc='
                              upper right')

plt.suptitle("632.8nm – Comparison
         between the original waveform
         and \n the waveform after adding
         three kinds of noise", fontsize
         = 20)
plt.xlim(-2*(10**(-6)), 2*(10**(-6)
         ))
plt.xlabel("Scan Length$/m$")
plt.ylabel("Interference Intensity"
         )

figure(8)
s1, = plt.plot(fs_0, Y0, marker='o'
               , ms=5)
s2, = plt.plot(fs_0, Yw0, marker='p'
               , ms=5)
s3, = plt.plot(fs_1, Yw1, marker='^'
               , ms=5)
s4, = plt.plot(fs_2, Yw2, marker='s'
               , ms=5)
s5, = plt.plot(fs_0,
               best_Y0_average_range, linewidth
               = 2.5)
plt.legend(handles=[s1, s2, s3, s4,
                  s5], labels=['Original spectrum
                              curve', 'Sine noise', 'Random
                              noise', 'Linear noise', 'FWHM'],
          loc='upper right')

plt.suptitle("632.8nm – Comparison
         between the original waveform
         and the spectrum measurement
         curve \n after adding three
         kinds of noise", fontsize = 20)
plt.xlim(1.52*(10**6), 1.65*(10**6)
         )
plt.xlabel("Wave number  $/m^{-1}$ ")

```

```
plt.ylabel("Amplitude")
```

```
plt.show()
```