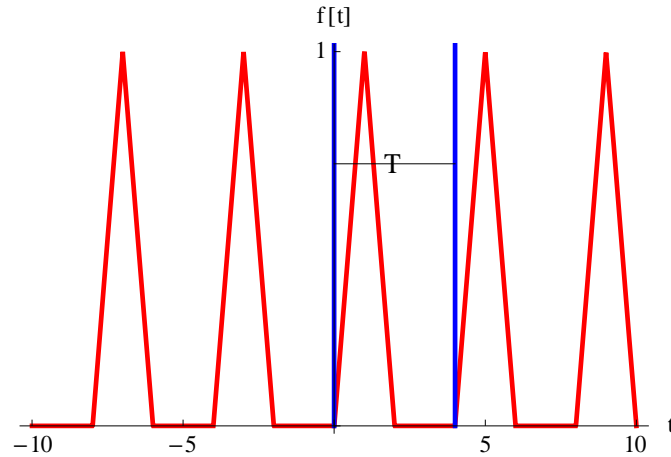


Chapter 11
Fourier series and fast Fourier transform
Masatsugu Sei Suzuki
Department of Physics, SUNY at Binghamton
(Date: October 28, 2010)

11.1 Overview

We consider a periodic function $f(t)$:



$$f(t + T) = f(t),$$

where T is a period.

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

Then $f(t)$ can be described as

$$f(t) = \frac{b_0}{2} + \sum_{n=1}^{\infty} [a_n \sin(n\omega_0 t) + b_n \cos(n\omega_0 t)],$$

where

$$a_n = \frac{2}{T} \int_0^T f(t) \sin(n\omega_0 t) dt,$$

$$b_n = \frac{2}{T} \int_0^T f(t) \cos(n\omega_0 t) dt,$$

and

$$b_0 = \frac{2}{T} \int_0^T f(t) dt .$$

11.2 Form factor of 1D x-ray diffraction

For simplicity, we consider the electron number density $N(x)$ with a period a in the x direction (one dimensional case).

$$N(x) = N(x + a) .$$

Then $n(x)$ may be expressed in terms of the Fourier series.

$$\begin{aligned} N(x) &= \frac{b_0}{2} + \sum_{n=1}^{\infty} \left\{ a_n \sin\left(n \frac{2\pi}{a} x\right) + b_n \cos\left(n \frac{2\pi}{a} x\right) \right\} \\ &= \frac{b_0}{2} + \sum_{n=1}^{\infty} \left[\frac{a_n}{2i} \left\{ \exp\left(i \frac{2\pi n}{a} x\right) - \exp\left(-i \frac{2\pi n}{a} x\right) \right\} + \frac{b_n}{2} \left\{ \exp\left(i \frac{2\pi n}{a} x\right) + \exp\left(-i \frac{2\pi n}{a} x\right) \right\} \right] \\ &= \frac{b_0}{2} + \sum_{n=1}^{\infty} \left[-i \frac{a_n}{2} \left\{ \exp\left(i \frac{2\pi n}{a} x\right) - \exp\left(-i \frac{2\pi n}{a} x\right) \right\} + \frac{b_n}{2} \left\{ \exp\left(i \frac{2\pi n}{a} x\right) + \exp\left(-i \frac{2\pi n}{a} x\right) \right\} \right] \\ &= \frac{b_0}{2} + \sum_{n=1}^{\infty} \left[\left(-i \frac{a_n}{2} + \frac{b_n}{2} \right) \exp\left(i \frac{2\pi n}{a} x\right) + \left(i \frac{a_n}{2} + \frac{b_n}{2} \right) \exp\left(-i \frac{2\pi n}{a} x\right) \right] \end{aligned}$$

where

$$b_0 = \frac{2}{a} \int_0^a N(x) dx$$

$$a_n = \frac{2}{a} \int_0^a N(x) \sin\left(\frac{2\pi n}{a} x\right) dx$$

$$b_n = \frac{2}{a} \int_0^a N(x) \cos\left(\frac{2\pi n}{a} x\right) dx$$

We note that

$$a_{-n} = -a_n$$

$$b_{-n} = b_n$$

$$a_0 = 0$$

and that

$$(-i\frac{a_{-n}}{2} + \frac{b_{-n}}{2}) = i\frac{a_n}{2} + \frac{b_n}{2}$$

Then the expression of $N(x)$ can be rewritten as

$$\begin{aligned} N(x) &= \frac{b_0}{2} + \sum_{n=1}^{\infty} [(-i\frac{a_n}{2} + \frac{b_n}{2}) \exp(i\frac{2\pi n}{a}x) + (i\frac{a_n}{2} + \frac{b_n}{2}) \exp(-i\frac{2\pi n}{a}x)] \\ &= \sum_{n=-\infty}^{\infty} [(-i\frac{a_n}{2} + \frac{b_n}{2}) \exp(i\frac{2\pi n}{a}x)] \end{aligned}$$

where

$$\begin{aligned} g_n &= \frac{2\pi n}{a} \\ N_n &= -i\frac{a_n}{2} + \frac{b_n}{2} = \frac{1}{a} \int_0^a N(x) \exp(i\frac{2\pi n}{a}x) dx \\ N_{-n} &= N_n^* \\ N_0 &= -i\frac{a_0}{2} + \frac{b_0}{2} = \frac{b_0}{2} \end{aligned}$$

For simplicity we use the notation

$$N(x) = \sum_g N_g \exp(igx)$$

where

$$\begin{aligned} N_n &= N_g \\ g_n &= g \end{aligned}$$

The electron number density satisfies the periodic condition,

$$N(x+a) = \sum_g N_g \exp[ig(x+a)] = \exp(iga) \sum_g N_g \exp(igx) = \exp(iga) N(x) = N(x)$$

where

$$\exp(iga) = \exp(i\frac{2\pi n}{a}a) = 1$$

Thus we have

$$N(x) = \sum_g N_g \exp(igx)$$

with

$$N_g = \frac{1}{a} \int_0^a dx N(x) e^{-igx}$$

11.3 Example 1

What is the value of N_g ? We consider the simplest case.

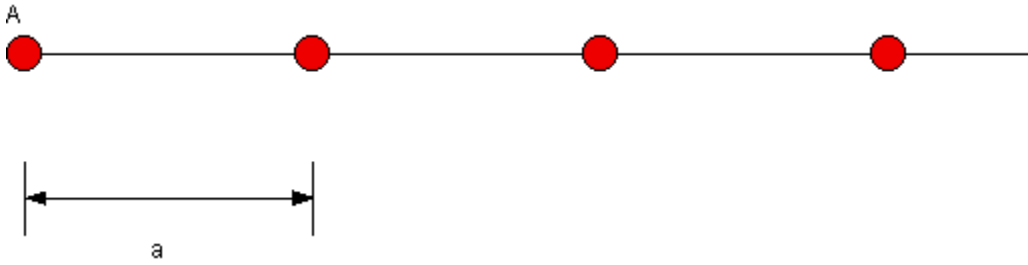


Fig.1 A simple one dimensional array with a lattice constant a .

$$N_g = \frac{1}{a} \int_0^a \delta(x) e^{-igx} dx = \frac{1}{a}$$

where $\delta(x)$ is the Dirac delta function.

11.4 Example 2

There are two atoms in each unit cell with the lattice constant a .

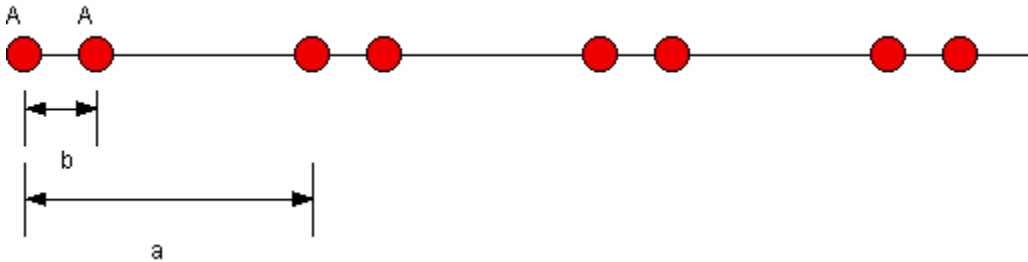
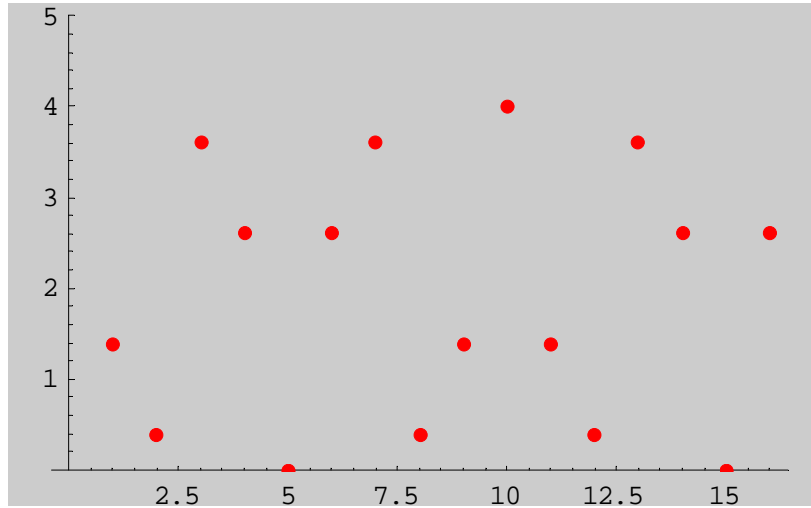


Fig.2 One dimensional array with two atoms per unit cell with a lattice constant a .

$$N_g = \frac{1}{a} \int_0^a [\delta(x) + \delta(x-b)] e^{-igx} dx = \frac{1}{a} (1 + e^{-igb})$$

$$|N_g|^2 = N_g N_g^* = \frac{(1 + e^{-igb})(1 + e^{igb})}{a^2} = \frac{4}{a^2} \cos^2\left(\frac{gb}{2}\right)$$

((Mathematica)): $|N_g|^2$ vs l where $a = 1$ and $b = 0.3$.



-Graphics-

Fig. Intensity $|N_g|^2$ vs Bragg index l for the 1D system shown in Fig.2.

11.5 x-ray diffraction: 3D case

A crystal is invariant under any translation of the form

$$\mathbf{T} = u_1 \mathbf{a}_1 + u_2 \mathbf{a}_2 + u_3 \mathbf{a}_3$$

where u_1, u_2, u_3 are integers and $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ are the periods along the crystal axes.

Any local physical property of the crystal is invariant under \mathbf{T} : charge concentration, electron number, magnetic moment density.

Electron number density $n(\mathbf{r})$ is a periodic function of \mathbf{r} , with periods $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ in the directions of the three axes.

$$n(\mathbf{r} + \mathbf{T}) = n(\mathbf{r})$$

We consider the Fourier series

$$n(\mathbf{r}) = \sum_{\mathbf{G}} n_{\mathbf{G}} \exp(i\mathbf{G} \cdot \mathbf{r})$$

$$n(\mathbf{r} + \mathbf{T}) = \sum_{\mathbf{G}} n_{\mathbf{G}} \exp[i\mathbf{G} \cdot (\mathbf{r} + \mathbf{T})] = \exp[i\mathbf{G} \cdot \mathbf{T}] \sum_{\mathbf{G}} n_{\mathbf{G}} \exp(i\mathbf{G} \cdot \mathbf{r}) = n(\mathbf{r})$$

or

$$\mathbf{G} \cdot \mathbf{T} = 2\pi l$$

where l is an integer.

The extension of the Fourier analysis to periodic function $n(\mathbf{r})$ in the 3D is given by

$$n(\mathbf{r}) = \sum_{\mathbf{G}} n_{\mathbf{G}} \exp(i\mathbf{G} \cdot \mathbf{r})$$

where \mathbf{G} is the reciprocal lattice vectors, and $n_{\mathbf{G}}$ determine the x-ray scattering amplitude.

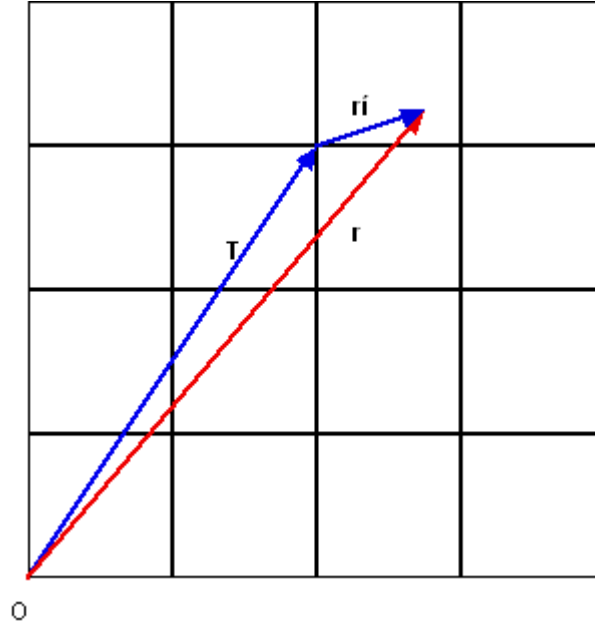
11.6 Derivation of the Fourier component $n_{\mathbf{G}}$

$$\int_V n(\mathbf{r}) e^{-i\mathbf{G} \cdot \mathbf{r}} d\mathbf{r} = \int_V \sum_{\mathbf{G}'} n_{\mathbf{G}'} e^{i\mathbf{G}' \cdot \mathbf{r}} e^{-i\mathbf{G} \cdot \mathbf{r}} d\mathbf{r} = \sum_{\mathbf{G}'} n_{\mathbf{G}'} \int_V e^{i(\mathbf{G}' - \mathbf{G}) \cdot \mathbf{r}} d\mathbf{r} = n_{\mathbf{G}} V$$

or

$$n_{\mathbf{G}} = \frac{1}{V} \int_V n(\mathbf{r}) e^{-i\mathbf{G} \cdot \mathbf{r}} d\mathbf{r}$$

where $V = N V_{\text{cell}}$



System consisting of periodic cells. \mathbf{T} is the translation vector.

$$\mathbf{r} = \mathbf{T} + \mathbf{r}'$$

$$n(\mathbf{r}) = n(\mathbf{T} + \mathbf{r}') = n(\mathbf{r}')$$

$$e^{-i\mathbf{G} \cdot \mathbf{r}} = e^{-i\mathbf{G} \cdot (\mathbf{r}' + \mathbf{T})} = e^{-i\mathbf{G} \cdot \mathbf{r}'}$$

Then we have

$$n_{\mathbf{G}} = \frac{1}{NV_{cell}} \left[N \int_{V_{cell}} n(\mathbf{r}) e^{-i\mathbf{G} \cdot \mathbf{r}} d\mathbf{r} \right] = \frac{1}{V_{cell}} \int_{V_{cell}} n(\mathbf{r}) e^{-i\mathbf{G} \cdot \mathbf{r}} d\mathbf{r}$$

Here we define the structure factor as

$$S_{\mathbf{G}} = \int_{V_{cell}} n(\mathbf{r}) e^{-i\mathbf{G} \cdot \mathbf{r}} d\mathbf{r}$$

or

$$n_{\mathbf{G}} = \frac{1}{V_{cell}} S_{\mathbf{G}}$$

11.7 Definition of reciprocal lattice vectors

Reciprocal lattice vector

$$\mathbf{G} \cdot \mathbf{T} = 2\pi$$

with

$$\mathbf{T} = u_1 \mathbf{a}_1 + u_2 \mathbf{a}_2 + u_3 \mathbf{a}_3$$

We construct the axis vectors, \mathbf{b}_1 , \mathbf{b}_2 , and \mathbf{b}_3 of the reciprocal lattice

$$\mathbf{b}_1 = 2\pi \frac{\mathbf{a}_2 \times \mathbf{a}_3}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]}, \quad \mathbf{b}_2 = 2\pi \frac{\mathbf{a}_3 \times \mathbf{a}_1}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]}, \quad \mathbf{b}_3 = 2\pi \frac{\mathbf{a}_1 \times \mathbf{a}_2}{[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]}$$

where

$$[\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3] = \mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3) = V_c = \text{volume of unit cell}$$

\mathbf{b}_1 , \mathbf{b}_2 , and \mathbf{b}_3 are called the primitive vectors of the reciprocal lattice.

Note that

$$\mathbf{a}_1 \cdot \mathbf{b}_1 = 2\pi, \quad \mathbf{a}_2 \cdot \mathbf{b}_2 = 2\pi, \quad \mathbf{a}_3 \cdot \mathbf{b}_3 = 2\pi$$

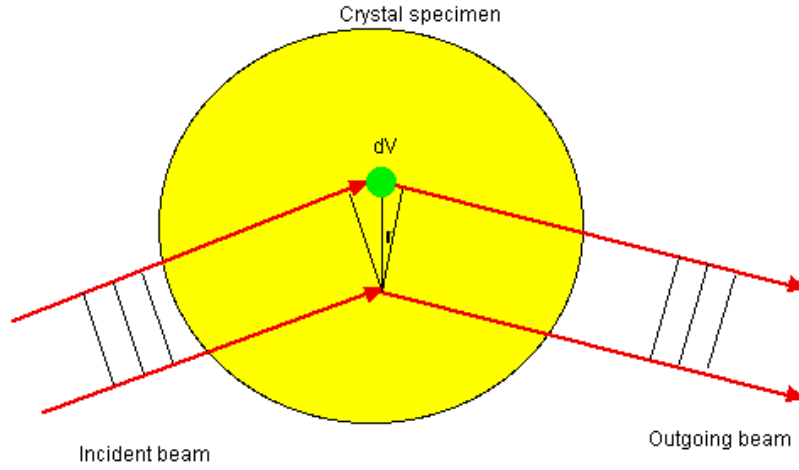
where \mathbf{b}_1 is perpendicular to both \mathbf{a}_2 and \mathbf{a}_3 . \mathbf{b}_2 is perpendicular to both \mathbf{a}_3 and \mathbf{a}_1 . \mathbf{b}_3 is perpendicular to both \mathbf{a}_1 and \mathbf{a}_2 . The reciprocal lattice vector \mathbf{G} is expressed by

$$\mathbf{G} = h\mathbf{b}_1 + k\mathbf{b}_2 + l\mathbf{b}_3.$$

Then we have

$$\mathbf{G} \cdot \mathbf{T} = (h\mathbf{b}_1 + k\mathbf{b}_2 + l\mathbf{b}_3) \cdot (u_1 \mathbf{a}_1 + u_2 \mathbf{a}_2 + u_3 \mathbf{a}_3) = 2\pi(hu_1 + ku_2 + lu_3)$$

11.8 Diffraction condition



The wavevectors of the incoming and outgoing x-rays are \mathbf{k} and \mathbf{k}' . The amplitude of the wave scattered from a volume element is proportional to the local electron concentration $n(\mathbf{r})$. The total amplitude of the scattered wave in the direction of \mathbf{k}' is proportional to the integral over the crystal of $n(\mathbf{r})dV$ times the phase factors $\exp[i(\mathbf{k} - \mathbf{k}') \cdot \mathbf{r}]$,

$$\begin{aligned}
 F &= \int dV \exp(-i\mathbf{k}' \cdot \mathbf{r}) n(\mathbf{r}) \exp(i\mathbf{k} \cdot \mathbf{r}) \\
 &= \int dV \exp(-i\mathbf{k}' \cdot \mathbf{r}) n(\mathbf{r}) \exp(i\mathbf{k} \cdot \mathbf{r}) \\
 &= \sum_G n_G \int dV \exp(-i\mathbf{k}' \cdot \mathbf{r}) \exp(i\mathbf{G} \cdot \mathbf{r}) \exp(i\mathbf{k} \cdot \mathbf{r}) \\
 &= \sum_G n_G \int dV \exp[i(\mathbf{G} - (\mathbf{k}' - \mathbf{k})) \cdot \mathbf{r}] \\
 &= V \sum_G n_G \delta[\mathbf{G} - (\mathbf{k}' - \mathbf{k})]
 \end{aligned}$$

When the scattering vector is equal to a particular reciprocal lattice vector,

$$\Delta\mathbf{k} = \mathbf{k}' - \mathbf{k} = \mathbf{G} \quad (\text{Bragg condition}),$$

the quantity F is equal to

$$F = Vn_G.$$

11.9 Construction of Ewald sphere

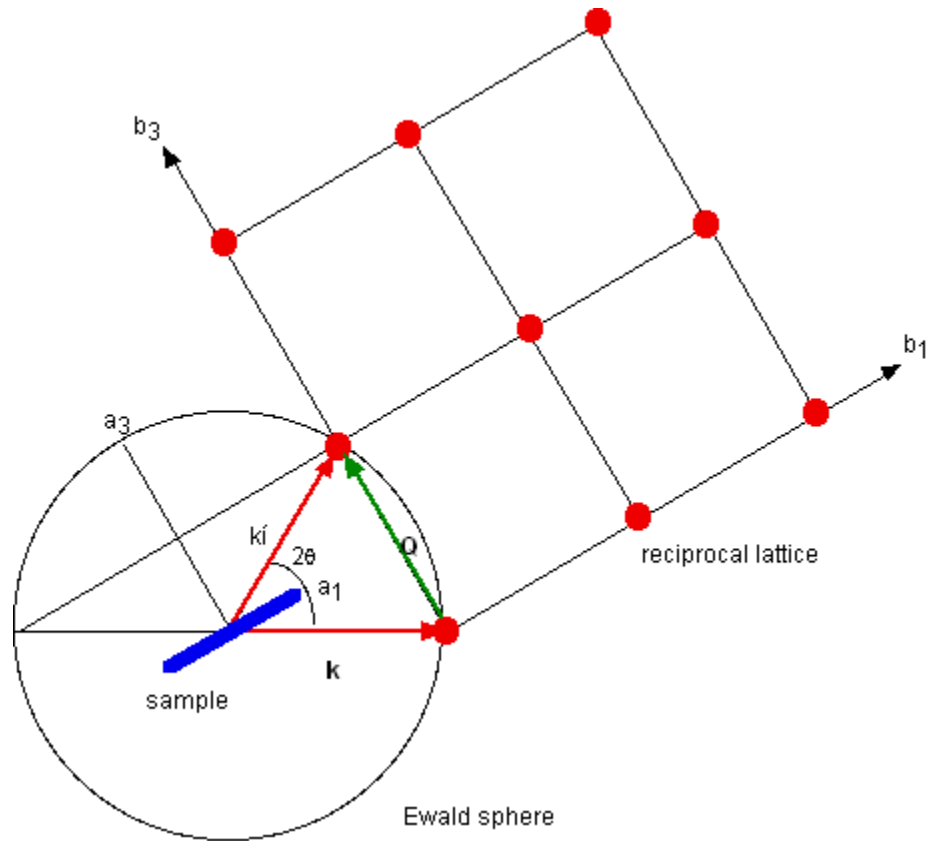


Fig. Ewald sphere. The origin of the reciprocal lattice is located at the end of the wavevector \mathbf{k} of the incident beam.

We draw a sphere of radius $k=2\pi/\lambda$ about the starting point of \mathbf{k} . The origin of the reciprocal lattice plane corresponding to the real space of the sample is at the end point of \mathbf{k} . A diffracted beam will be formed if this sphere intersects any other point in the reciprocal lattice. The Ewald sphere intercepts a point connected with the end of \mathbf{k} by a reciprocal lattice vector \mathbf{G} . This construction is due to Paul Peter Ewald.

11.10 Fourier series of square wave

■

Example-1

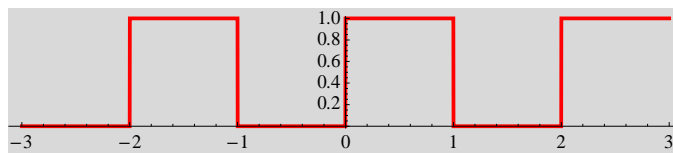
Fourier series of square wave

```
Clear["Global`*"]

f[t_] := 0 /; -1 <= t < 0
f[t_] := 1 /; 0 <= t < 1

extens[t_] := f[t] /; -1 <= t <= 1; extens[t_] := extens[t - 2] /; t > 1;
extens[t_] := extens[t + 2] /; t < -1;

Plot[extens[t], {t, -3, 3}, PlotStyle -> {Red, Thick}, Background -> LightGray,
  AspectRatio -> Automatic]
```



$$f[t] = \frac{b_0}{2} + \text{Sum}[a_n \sin[n \omega_0 t] + b_n \cos[n \omega_0 t], \{n, 1, \infty\}]$$

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

$$T = 2;$$

$$b[0] = \frac{2}{T} \int_0^1 1 \, dt$$

1

$$a[n_] = \frac{2}{T} \int_0^1 \sin\left[n \frac{2\pi}{T} t\right] dt // \text{Simplify}[\#, n \in \text{Integers}] \&$$

$$-\frac{-1 + (-1)^n}{n\pi}$$

$$b[n_] = \frac{2}{2} \int_0^1 \cos\left[n \frac{2\pi}{T} t\right] dt // \text{Simplify}[\#, n \in \text{Integers}] \&$$

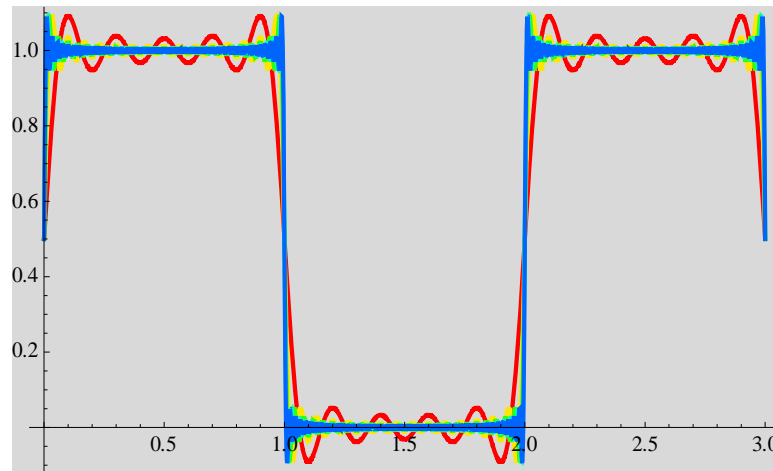
0

$$g[t_, m_] := \frac{b[0]}{2} + \sum_{n=1}^m \left(a[n] \sin\left[n \frac{2\pi}{T} t\right] + b[n] \cos\left[n \frac{2\pi}{T} t\right] \right)$$

$g[t, 10]$

$$\frac{1}{2} + \frac{2 \sin[\pi t]}{\pi} + \frac{2 \sin[3\pi t]}{3\pi} + \frac{2 \sin[5\pi t]}{5\pi} + \frac{2 \sin[7\pi t]}{7\pi} + \frac{2 \sin[9\pi t]}{9\pi}$$

`Plot[Evaluate[Table[g[t, m], {m, 10, 100, 20}], {t, 0, 3}], PlotPoints -> 300,
PlotStyle -> Table[{Hue[0.15 i], Thick}, {i, 0, 5}], Background -> LightGray]`

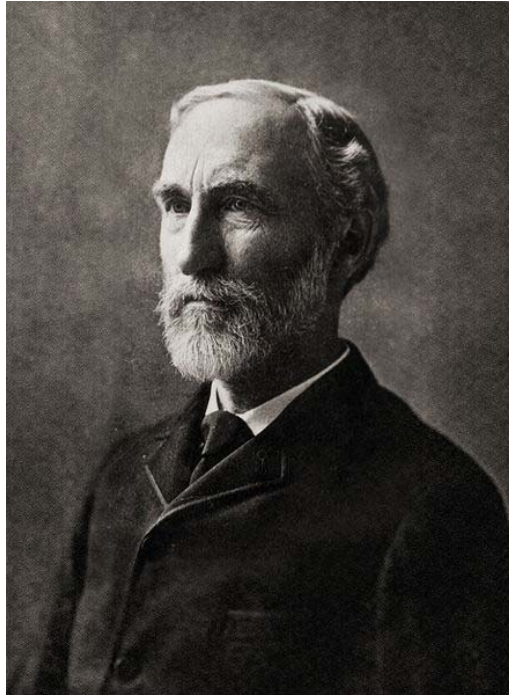


((Note)) Gibbs'sphenomena

The overshoot of the Fourier series at a discontinuity is referred to as the Gibbs phenomenon.

Josiah Willard Gibbs (February 11, 1839 – April 28, 1903) was an American theoretical physicist, chemist, and mathematician. He devised much of the theoretical foundation for chemical thermodynamics as well as physical chemistry. As a mathematician, he invented vector analysis (independently of Oliver Heaviside). Yale University awarded

Gibbs the first American Ph.D. in engineering in 1863, and he spent his entire career at Yale. His thesis was entitled: *On the Form of the Teeth of Wheels in Spur Gearing*.



http://en.wikipedia.org/wiki/Willard_Gibbs

11.11 Fourier series of saw teeth

■

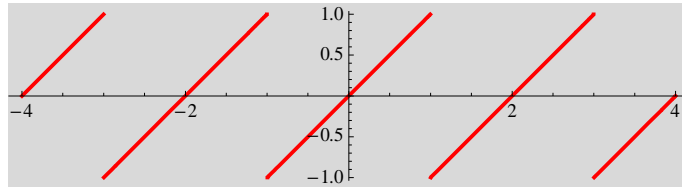
Example-2

Fourier series (Saw-teeth)

```
Clear["Global`*"]
```

```
f[t_] := Mod[t - 1, 2] - 1
```

```
p1 = Plot[f[t], {t, -4, 4}, PlotStyle -> {Red, Thick}, Background -> LightGray,  
  AspectRatio -> Automatic]
```



```
T = 2;
```

$$b[0] = \frac{2}{T} \int_{-1}^1 f[t] \, dt$$

```
0
```

$$a[n_] = \frac{2}{T} \int_{-1}^1 f[t] \sin\left[n \frac{2\pi}{T} t\right] dt // \text{Simplify}[\#, n \in \text{Integers}] \&$$

$$-\frac{2(-1)^n}{n\pi}$$

$$b[n_] = \frac{2}{T} \int_0^2 f[t] \cos\left[n \frac{2\pi}{T} t\right] dt // \text{Simplify}[\#, n \in \text{Integers}] \&$$

```
0
```

```
Table[{n, a[n]}, {n, 1, 10}]
```

```
{ {1, 2/π}, {2, -1/π}, {3, 2/3π}, {4, -1/2π}, {5, 2/5π},  
  {6, -1/3π}, {7, 2/7π}, {8, -1/4π}, {9, 2/9π}, {10, -1/5π} }
```

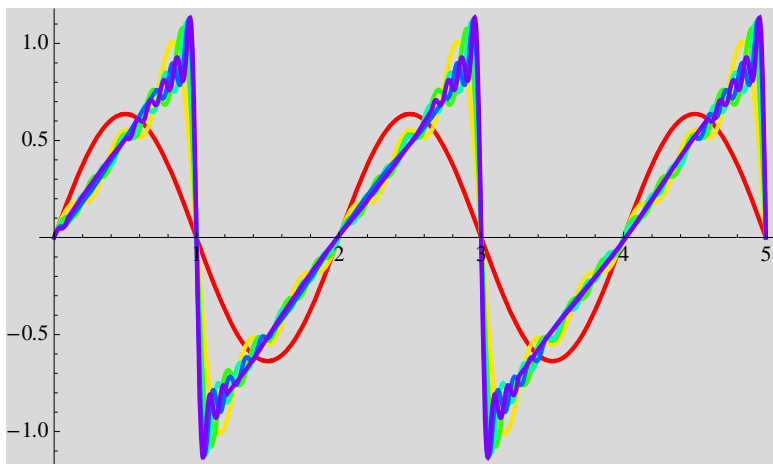
```
F[t_, m_] := Sum[a[n] Sin[n 2 π t], {n, 1, m}]
```

```
F[t, 10]
```

$$\frac{2 \sin[\pi t]}{\pi} - \frac{\sin[2 \pi t]}{\pi} + \frac{2 \sin[3 \pi t]}{3 \pi} - \frac{\sin[4 \pi t]}{2 \pi} + \frac{2 \sin[5 \pi t]}{5 \pi} -$$

$$\frac{\sin[6 \pi t]}{3 \pi} + \frac{2 \sin[7 \pi t]}{7 \pi} - \frac{\sin[8 \pi t]}{4 \pi} + \frac{2 \sin[9 \pi t]}{9 \pi} - \frac{\sin[10 \pi t]}{5 \pi}$$

```
Plot[Evaluate[Table[F[t, m], {m, 1, 21, 4}]], {t, 0, 5},  
PlotStyle -> Table[{Hue[0.15 i], Thick}, {i, 0, 5}], Background -> LightGray]
```



11.12 Fourier series of triangle wave

Example-4

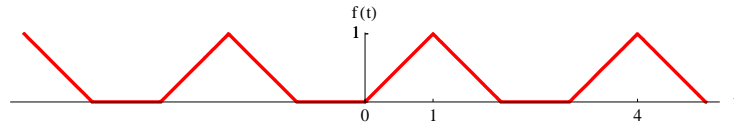
Fourier series

```
Clear["Global`*"]

f[t_] := 0 /; -1 ≤ t ≤ 0; f[t_] := t /; 0 ≤ t ≤ 1; f[t_] := -t + 2 /; 1 ≤ t ≤ 2;

extens[t_] := f[t] /; -1 ≤ t ≤ 2; extens[t_] := extens[t - 3] /; t > 2;
extens[t_] := extens[t + 3] /; t < -1;

p1 = Plot[extens[t], {t, -5, 5}, PlotStyle → {{Red, Thick}},
  AspectRatio → Automatic, AxesLabel → {"t", "f(t)"}, Ticks → {{0, 4, 1}, {0, 1, 1}}]
```



$T = 3;$

$$b[0] = \frac{2}{T} \left(\int_0^1 t \, dt + \int_1^2 (2-t) \, dt \right)$$

$$\frac{2}{3}$$

$$a[n_] = \frac{2}{T} \left(\int_0^1 t \sin\left[n \frac{2\pi}{T} t\right] dt + \int_1^2 (2-t) \sin\left[n \frac{2\pi}{T} t\right] dt \right) //$$

`Simplify[#, n ∈ Integers] &`

$$\frac{12 \cos\left[\frac{n\pi}{3}\right] \sin\left[\frac{n\pi}{3}\right]^3}{n^2 \pi^2}$$

$$b[n_] = \frac{2}{T} \left(\int_0^1 t \cos\left[n \frac{2\pi}{T} t\right] dt + \int_1^2 (2-t) \cos\left[n \frac{2\pi}{T} t\right] dt \right) //$$

`Simplify[#, n ∈ Integers] &`

$$\frac{6 \cos\left[\frac{2n\pi}{3}\right] \sin\left[\frac{n\pi}{3}\right]^2}{n^2 \pi^2}$$

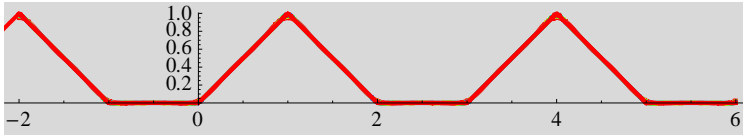
$$g[t_, m_] := \frac{b[0]}{2} + \sum_{n=1}^m \left(a[n] \sin\left[n \frac{2\pi}{T} t\right] + b[n] \cos\left[n \frac{2\pi}{T} t\right] \right)$$

`g[t, 10]`

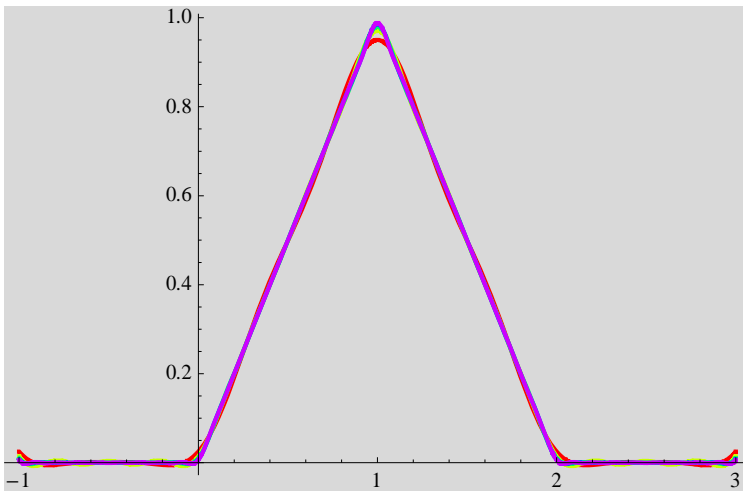
$$\begin{aligned} & \frac{1}{3} - \frac{9 \cos\left[\frac{2\pi t}{3}\right]}{4 \pi^2} - \frac{9 \cos\left[\frac{4\pi t}{3}\right]}{16 \pi^2} - \frac{9 \cos\left[\frac{8\pi t}{3}\right]}{64 \pi^2} - \frac{9 \cos\left[\frac{10\pi t}{3}\right]}{100 \pi^2} - \frac{9 \cos\left[\frac{14\pi t}{3}\right]}{196 \pi^2} - \\ & \frac{9 \cos\left[\frac{16\pi t}{3}\right]}{256 \pi^2} - \frac{9 \cos\left[\frac{20\pi t}{3}\right]}{400 \pi^2} + \frac{9 \sqrt{3} \sin\left[\frac{2\pi t}{3}\right]}{4 \pi^2} - \frac{9 \sqrt{3} \sin\left[\frac{4\pi t}{3}\right]}{16 \pi^2} + \frac{9 \sqrt{3} \sin\left[\frac{8\pi t}{3}\right]}{64 \pi^2} - \\ & \frac{9 \sqrt{3} \sin\left[\frac{10\pi t}{3}\right]}{100 \pi^2} + \frac{9 \sqrt{3} \sin\left[\frac{14\pi t}{3}\right]}{196 \pi^2} - \frac{9 \sqrt{3} \sin\left[\frac{16\pi t}{3}\right]}{256 \pi^2} + \frac{9 \sqrt{3} \sin\left[\frac{20\pi t}{3}\right]}{400 \pi^2} \end{aligned}$$

```
g1 = Plot[Evaluate[Table[g[t, m], {m, 5, 25, 5}]], {t, -2, 6}, PlotPoints -> 300,
  PlotStyle -> Table[{Hue[0.2 i], Thick}, {i, 0, 3}], Background -> LightGray,
  AspectRatio -> Automatic];
```

`Show[g1, p1]`



```
g2 = Plot[Evaluate[Table[g[t, m], {m, 5, 25, 5}]], {t, -1, 3}, PlotPoints -> 300,
  PlotStyle -> Table[{Hue[0.2 i], Thick}, {i, 0, 5}], Background -> LightGray]
```



11.13 Fast Fourier transform (FFT)

The Fast Fourier Transform (FFT) is just a clever computer algorithm to perform a discrete Fourier transform (DFT) efficiently on a list of numbers. It computes exactly the

formulas given above, no more, no less. One might expect an algorithm for a DFT to take a time proportional to N^2 for lists of length N , since the direct matrix multiplication would involve N^2 multiplications and almost N^2 additions. But in fact the FFT's running time is only proportional to $N \log N$, which grows with N *much* less rapidly than N^2 . For large N the difference is dramatic, and the FFT makes possible lots of computations that would be effectively unthinkable without it.

We consider a series

$$\{x_0, x_1, \dots, x_{N-1}\}$$

where

$$x_n = x[n\Delta]. \quad (n = 0, 1, 2, \dots, N-1).$$

The sampling time for the observation of x_n is Δ , where $T = N\Delta$ is the total time for the observation.

((Fourier transform))

The Fourier transform is defined as

$$\begin{aligned} X_k &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \exp(i\omega_k t_n) \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \exp(i \frac{2\pi k}{N\Delta} n\Delta), \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n [\exp(i \frac{2\pi}{N} k)]^n \end{aligned}$$

where

$$t_n = n\Delta, \quad \omega_k = k \left(\frac{2\pi}{T} \right) = k \left(\frac{2\pi}{N\Delta} \right),$$

and $T = N\Delta$.

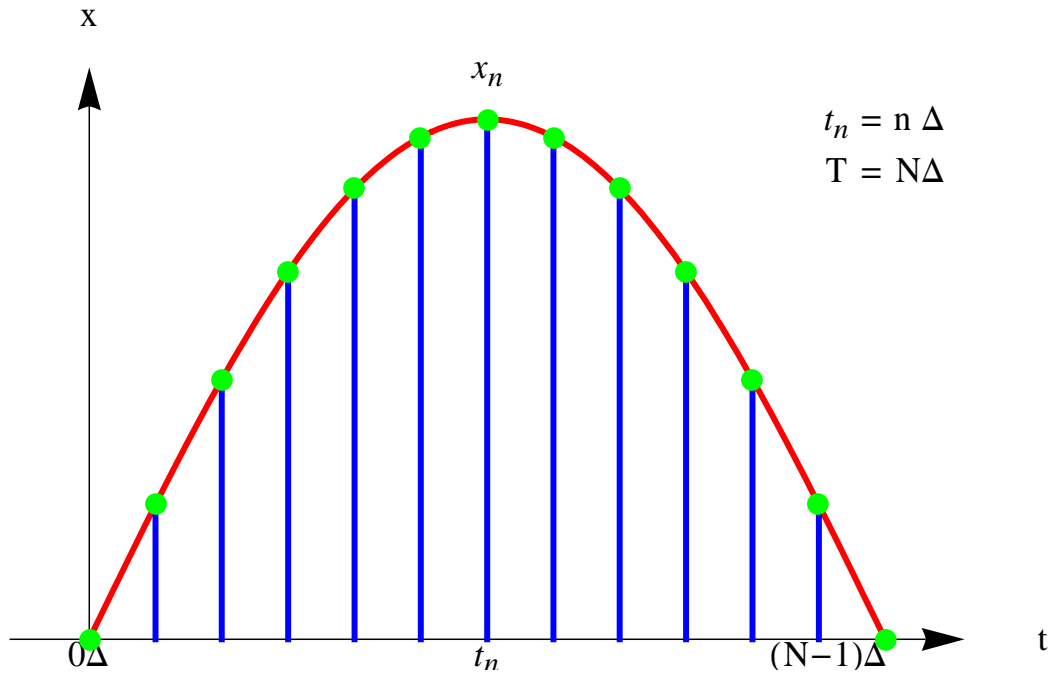


Fig. The data of x_n vs t_n . $t_n = n\Delta$. x_n is the value of x at $t = t_n$.

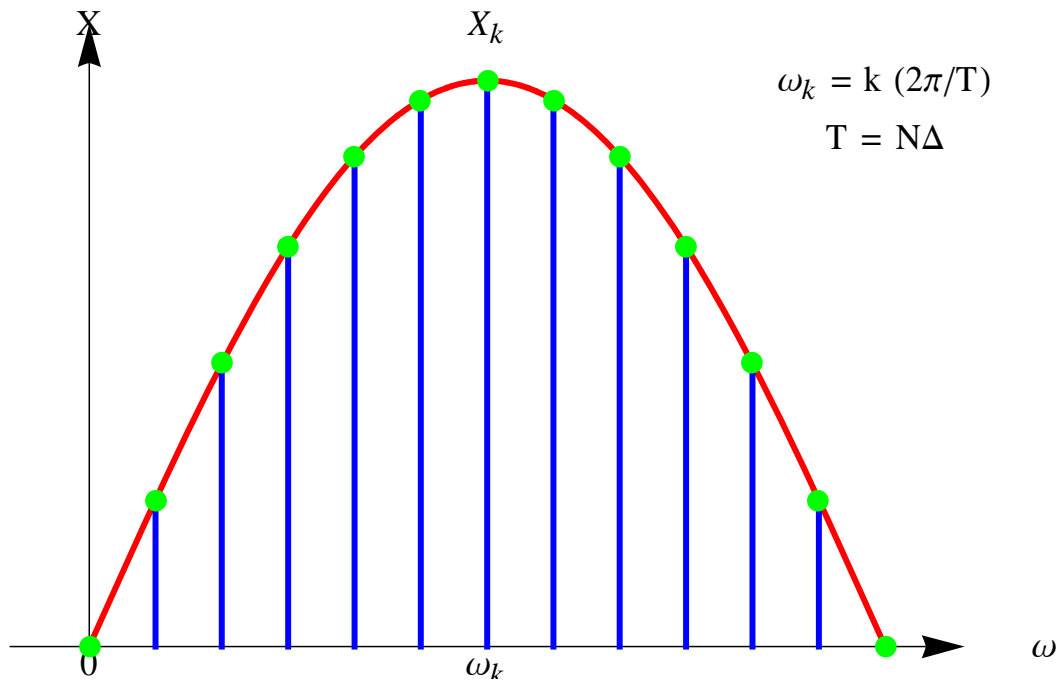


Fig. Fourier transform (X_k vs ω_k). $\omega_k = k (2\pi/T)$. $T = N\Delta$. X_k is the Fourier transform of x_n vs t_n .

((Inverse Fourier transform))

$$X_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \exp\left[\frac{2\pi i}{N} k n\right].$$

The angular frequency corresponding to X_k is

$$\omega_k = \frac{2\pi k}{N\Delta}, \quad (k = 0, 1, 2, \dots, N-1).$$

The inverse Fourier transform is defined as

$$x_n = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_k \exp\left[-\frac{2\pi i}{N} n k\right].$$

Note that

$$\begin{aligned} x_n &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \exp\left[-\frac{2\pi i}{N} n k\right] \frac{1}{\sqrt{N}} \sum_{n'=0}^{N-1} x_{n'} \exp\left[\frac{2\pi i}{N} k n'\right] \\ &= \frac{1}{N} \sum_{n'=0}^{N-1} x_{n'} \sum_{k=0}^{N-1} \exp\left[\frac{2\pi i}{N} k n'\right] \exp\left[-\frac{2\pi i}{N} n k\right] \\ &= \frac{1}{N} \sum_{n'=0}^{N-1} x_{n'} \sum_{k=0}^{N-1} \exp\left[\frac{2\pi i}{N} k n' - \frac{2\pi i}{N} n k\right] \\ &= \frac{1}{N} \sum_{n'=0}^{N-1} x_{n'} \sum_{k=0}^{N-1} \exp\left[\frac{2\pi i}{N} k (n' - n)\right] \end{aligned}$$

Here we use

$$\sum_{k=0}^{N-1} \exp\left[\frac{2\pi i}{N} k (n' - n)\right] = N \delta_{n', n}$$

((Essence of the FFT))

1. When one look at the matrix, there are only N different possible values, not N^2 , so one could certainly save some time by computing them all first and resusing them as appropriate.

$$X_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n \exp\left[\frac{2\pi i}{N} k n\right]$$

(i) $N = 2^4$

Suppose for example that $N = 16 = 2^4$. Then we can write any integer m in the range $[0, 15]$ as

$$m = 2^3 a + 2^2 b + 2^1 c + 2^0 d$$

where a , b , c , and d are each either 0 or 1. This is just the binary decomposition.

$$\begin{aligned} \exp\left(\frac{2\pi i}{N}m\right) &= \exp\left[\frac{2\pi i}{2^4}(2^3 a + 2^2 b + 2c + d)\right] \\ &= \exp\left[2\pi i\left(\frac{a}{2} + \frac{b}{2^2} + \frac{c}{2^3} + \frac{d}{2^4}\right)\right] \\ &= \exp\left[\pi i a + \frac{\pi i}{2} b + \frac{\pi i}{4} c + \frac{\pi i}{8} d\right] \\ &= [\exp(\pi i)]^a [\exp(\frac{\pi i}{2})]^b [\exp(\frac{\pi i}{4})]^c [\exp(\frac{\pi i}{8})]^d \\ &= A^a B^b C^c D^d \end{aligned}$$

Once we calculate A , B , C , and D , we can get all the required exponentials (16) by multiplying them together in the right combinations.

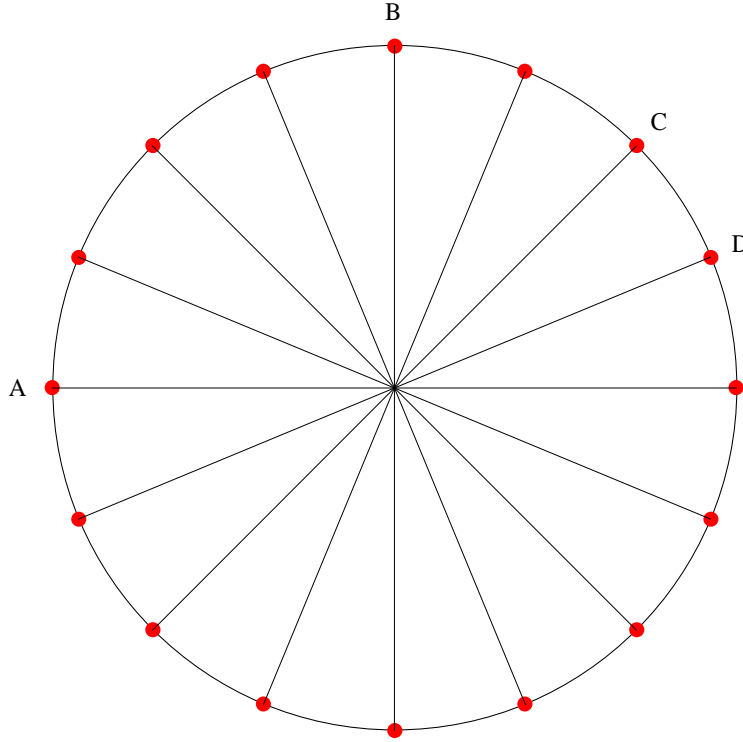


Fig. The unit circle in the complex plane for $N = 2^4$. The points A, B, C, and D are shown in the figure.

(ii) $N = 2^3$.

Suppose for example that $N = 2^3$. Then we can write any integer m in the range $[0, 7]$ as

$$\begin{aligned}
 m &= 2^2 a + 2^1 b + c \\
 \exp\left(\frac{2\pi i}{N} m\right) &= \exp\left[\frac{2\pi i}{2^3} (2^2 a + 2^1 b + c)\right] \\
 &= \exp\left[\pi i a + \frac{\pi i}{2} b + \frac{\pi i}{4} c\right] \\
 &= [\exp(\pi i)]^a [\exp(\frac{\pi i}{2})]^b [\exp(\frac{\pi i}{4})]^c \\
 &= A^a B^b C^c
 \end{aligned}$$

where a and b are each either 0 or 1. Once we calculate A , B , and C , we can get all the required exponentials (8) by multiplying them together in the right combinations.

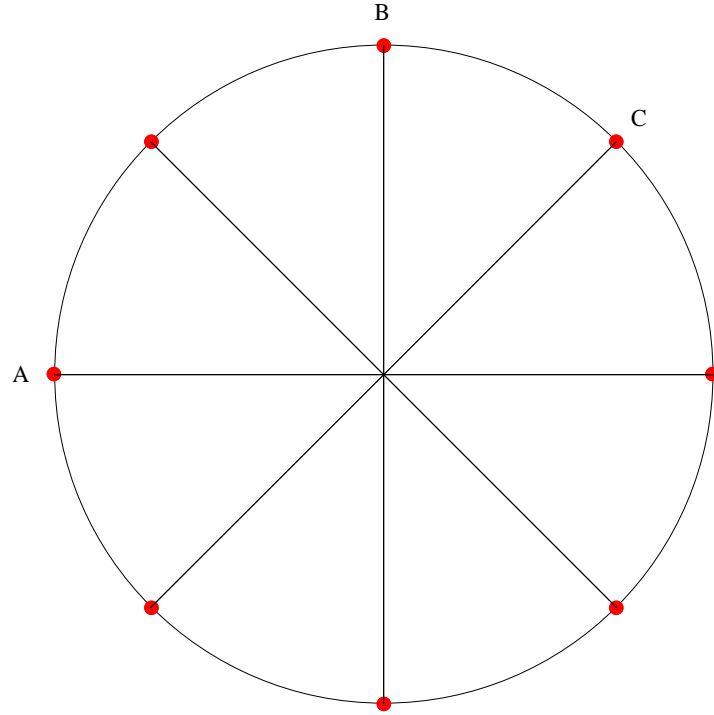


Fig. The unit circle in the complex plane for $N = 2^3$. The points A , B , and C are shown in the figure.

(iii) $N = 2^2$.

Suppose for example that $N = 2^2$. Then we can write any integer m in the range $[0, 3]$ as

$$m = 2a + b$$

$$\begin{aligned}\exp\left(\frac{2\pi i}{N}m\right) &= \exp\left[\frac{2\pi i}{2^2}(2^1a + b)\right] \\ &= \exp\left[\left(\pi i a + \frac{\pi i}{2}b\right)\right] \\ &= [\exp(\pi i)]^a [\exp(\frac{\pi i}{2})]^b \\ &= A^a B^b\end{aligned}$$

where a and b are each either 0 or 1. Once we calculate A and B , we can get all the required exponentials (4) by multiplying them together in the right combinations.

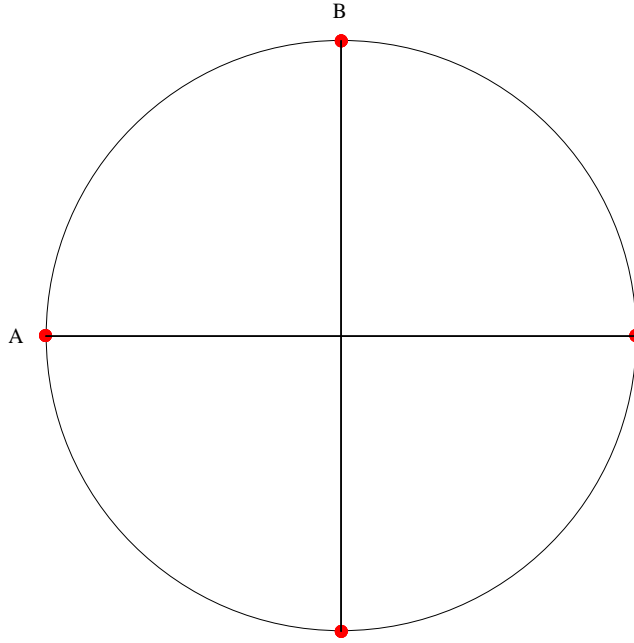


Fig. The unit circle in the complex plane for $N = 2^2$. The points A and B are shown in the figure.

$$\begin{aligned}X_0 &= \frac{1}{2} \sum_{n=0}^3 x_n \exp\left[\frac{2\pi i}{4}0\right]^n = \frac{1}{2}(x_0 + x_1 + x_2 + x_3) \\ X_1 &= \frac{1}{2} \sum_{n=0}^3 x_n \exp\left[\frac{2\pi i}{4}n\right] = \frac{1}{2}(x_0 + Bx_1 + Ax_2 + ABx_3) \\ X_2 &= \frac{1}{2} \sum_{n=0}^3 x_n \exp\left[\frac{2\pi i}{4}2n\right] = \frac{1}{2}(x_0 + Ax_1 + x_2 + Ax_3) \\ X_3 &= \frac{1}{2} \sum_{n=0}^3 x_n \exp\left[\frac{2\pi i}{4}3n\right] = \frac{1}{2}(x_0 + ABx_1 + Ax_2 + Bx_3)\end{aligned}$$

or

$$\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & B & A & AB \\ 1 & A & 1 & A \\ 1 & AB & A & B \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

11.14 Fourier[list]

((Mathematica))

Here we show how to calculate the FFT by using Mathematica.

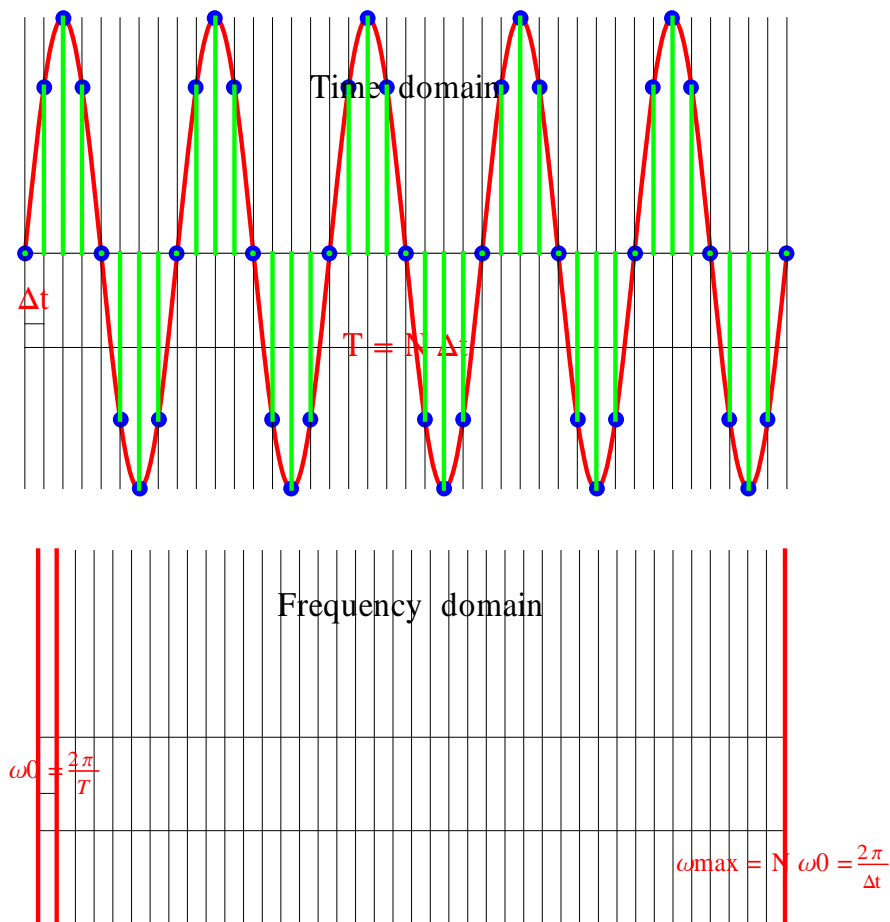
Method of Fast Fourier Transform

1. Give the number of divisions, N .
2. Give the upper limit of the time domain, T .
3. Calculate the distance of each division, $\Delta = T/N$
4. Make the list of the value of $f(t)$ at $t = n \Delta$, where $n = 0, 1, 2, 3, \dots, N$.
5. Use the program "Fourier", for the FFT .
6. Calculate the division of the Frequency domain, $\omega_0 = 2\pi/T$
7. Calculate the upper limit of frequency domain, $\omega_{\max} = 2\pi/\Delta$
8. One get the Fourier spectrum as a function of k in the units of ω_0 ; $\omega = k \omega_0$ with $k = 0, 1, \dots, N$.

Fourier[list]

to calculate the FFT.

to find the discrete Fourier transform of a list of complex numbers.



11.15 Example

■

Example-1

FFT (Fast Fourier transform)

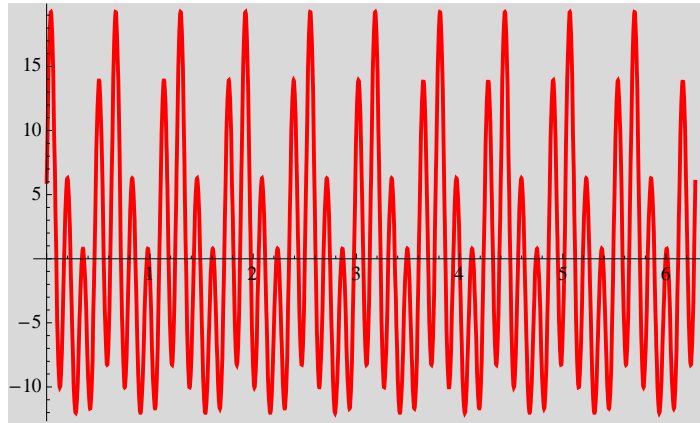
```
Clear["Global`*"]
```

```
N1 = 2048; T = 10  $\pi$ ;  $\Delta = \frac{T}{N1}$ ;
```

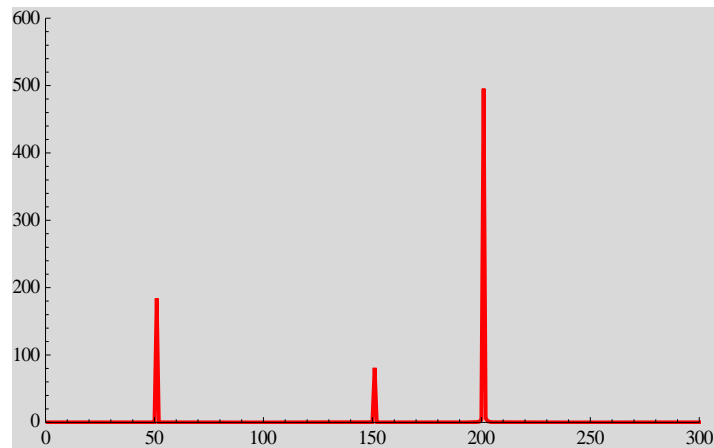
```
 $\omega 0 = \frac{2 \pi}{T}$ ;
```

```
f1[t_] = 6 Cos[10 t] + 4 Sin[30 t] + 10 Sin[40 t];
```

```
Plot[f1[t], {t, 0, 2  $\pi$ }, PlotStyle -> {Red, Thick}, Background -> LightGray]
```



```
list1 = Table[f1[ $\frac{T}{N1} n$ ], {n, 0, 2048}] // N; list2 = Fourier[list1] // Chop;  
ListPlot[Abs[list2]2/100, Joined -> True, PlotRange -> {{0, 300}, {0, 600}},  
PlotStyle -> {Red, Thick}, Background -> LightGray]
```



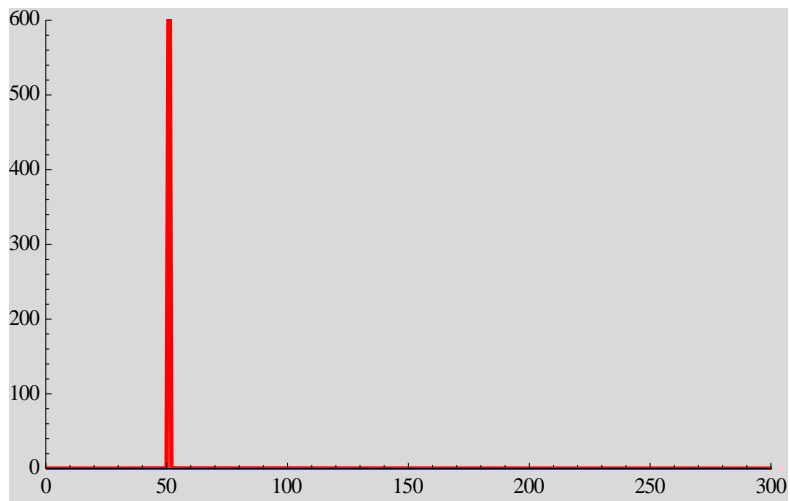
11.16 Example

```
Clear["Global`*"]
```

```
f2[t_] = 13 Sin[10 t] ; list1 = Table[f2[ $\frac{10 \pi}{2048} n$ ], {n, 0, 2048}] // N;
```

```
list2 = Fourier[list1] // Chop;
```

```
ListPlot[Abs[list2]2/100, Joined → True, PlotRange → {{0, 300}, {0, 600}},  
PlotStyle → {Red, Thick}, Background → LightGray]
```



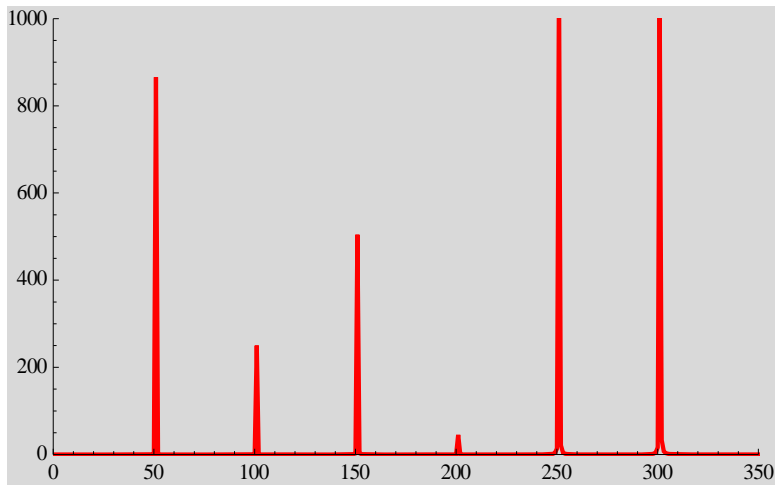
11.17 Example

```

Clear["Global`*"]

f5[t_] = 13 Sin[10 t] + 7 Sin[20 t] + 10 Sin[30 t] + 3 Sin[40 t] +
  15 Sin[50 t] + 15 Sin[60 t]; list1 = Table[f5[ $\frac{10 \pi}{2048} n$ ], {n, 0, 2048}] // N;
list2 = Fourier[list1] // Chop;
ListPlot[Abs[list2]^2/100, Joined -> True, PlotRange -> {{0, 350}, {0, 1000}},
  PlotStyle -> {Red, Thick}, Background -> LightGray]

```



11.18 Beat frequencies

■

Beat which has the combinations of two waves with almost the same frequencies.

Method of Fast Fourier Transform

1. Give the number of divisions, N.
2. Give the upper limit of the time domain, T.
3. Calculate the distance of each division, $\Delta = T/N$
4. Make the list of the value of $f(t)$ at $t = k\Delta$, where $k = 1, 2, 3, \dots, N$.
5. Use the program "Fourier, for the FFT .
6. Calculate the division of the Frequency domain, $\omega_0 = 2\pi/T$
7. Calculate the upper limit of the frequency domain, $\omega_{\max} = 2\pi/\Delta$
8. One gets the Fourier spectrum as a function of k (0, 1, ..., N) in the units of ω_0 .

```
Clear["Global`*"]
```

```
N1 = 2048; T = 10  $\pi$ ;  $\Delta = \frac{T}{N1}$ 
```

$$\frac{5 \pi}{1024}$$

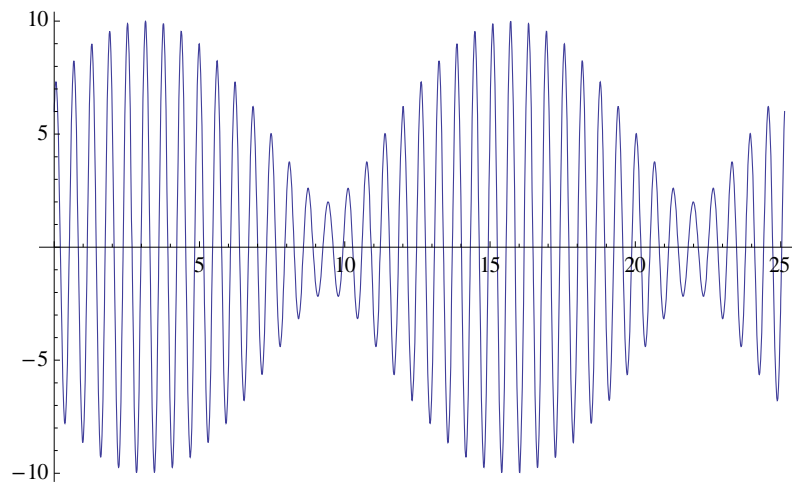
$$\omega_0 = \frac{2 \pi}{T}$$

$$\frac{1}{5}$$

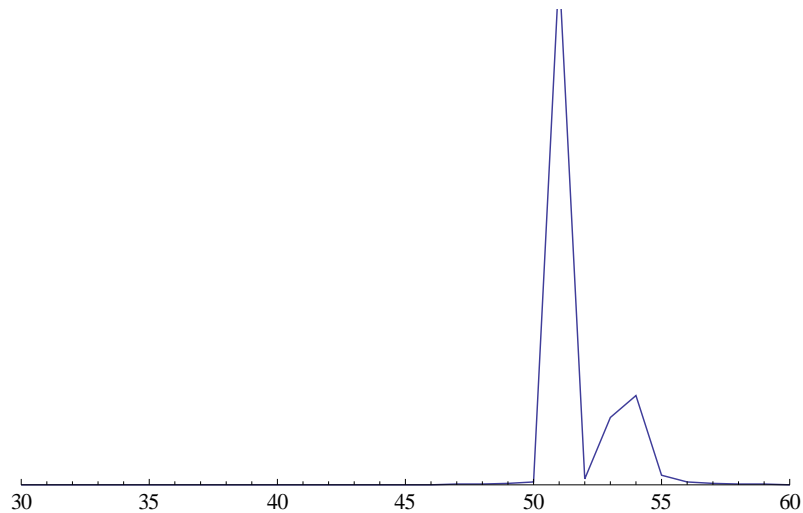
```
g1[t_] = 6 Cos[10 t] + 4 Sin[10.5 t]
```

```
6 Cos[10 t] + 4 Sin[10.5 t]
```

```
Plot[g1[t], {t, 0, 8  $\pi$ }]
```



```
list1 = Table[g1[ $\frac{T}{N1} n$ ], {n, 0, 2048}] // N; list2 = Fourier[list1] // Chop;  
ListPlot[Abs[list2]2/100, Joined → True, PlotRange → {{30, 60}, {0, 200}}]
```



11.19 FFT for simple pendulum

Pendulum

$$\frac{d^2 \theta}{dt^2} = -\frac{g}{l} \sin \theta$$

with the initial condition

$$\theta(t=0)=\theta_0 \text{ and } \theta'(t=0)=v_0$$

$$\epsilon = \frac{g}{l}$$

For convenience, the differential equation is separated into two parts,

$$v = \frac{d\theta}{dt}, \quad \frac{dv}{dt} = -\epsilon \sin \theta$$

with the initial conditions $v(t=0) = v_0$ and $\theta(t=0)=\theta_0$

First we solve this differential equation using numerical method (*Mathematica*, NDSolve)

Second, we analyze the Fourier component by using FFT (*Mathematica* , Fourier).

1. We choose the $N (= 2^n)$ data, typically the data between 0 and $T (=t_{\max})$.
2. The minimum time division is $\Delta = \frac{T}{N}$, which corresponds to the maximum $\omega_{\max} = \frac{2\pi}{\Delta} = \frac{2\pi}{T} N$
3. The Fourier spectrum is plotted as a function of the channel number scaled by $\frac{2\pi}{T}$.

11.20 Example

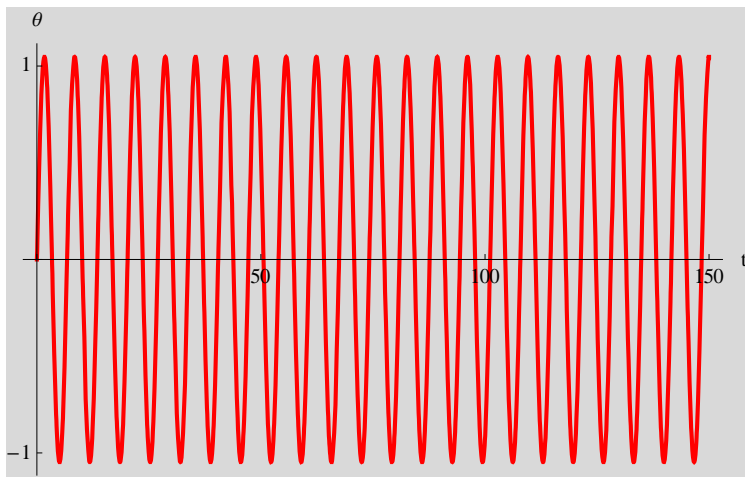
```

Clear["Global`*"]

pentime[{ $\theta_0$ _,  $v_0$ _},  $\epsilon$ _,  $t_{max}$ _,  $opts$ _] :=
Module[{numsol, numgraph},
  numsol = NDSolve[{ $v'$ [ $t$ ] +  $\epsilon$  Sin[ $\theta$ [ $t$ ]] == 0,  $\theta'$ [ $t$ ] ==  $v$ [ $t$ ],  $\theta$ [0] ==  $\theta_0$ ,
     $v$ [0] ==  $v_0$ }, { $\theta$ [ $t$ ],  $v$ [ $t$ ]}, { $t$ , 0,  $t_{max}$ }}];
  numgraph = Plot[Evaluate[ $\theta$ [ $t$ ] /. numsol[[1]]], { $t$ , 0,  $t_{max}$ },
     $opts$ , DisplayFunction -> Identity]]

penlist = pentime[{0, 1}, 1, 150, PlotStyle -> {Red, Thick},
  AxesLabel -> {" $t$ ", " $\theta$ "}, Background -> LightGray, PlotRange -> All,
  Ticks -> {Range[0, 200, 50], Range[-5, 5]},
  DisplayFunction -> Identity];
Show[penlist, DisplayFunction -> $DisplayFunction]

```

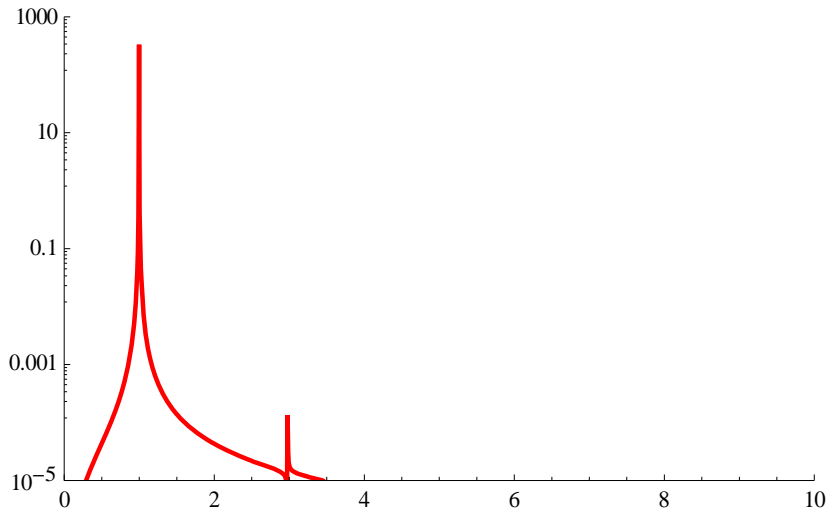



```

pentime[{ $\theta_0$ _,  $v_0$ _},  $\epsilon$ _,  $t_{max}$ _,  $N1$ _] :=
Module[{numsol, numtable},
  numsol = NDSolve[{ $v'$ [ $t$ ] +  $\epsilon$  Sin[ $\theta$ [ $t$ ]] == 0,  $\theta'$ [ $t$ ] ==  $v$ [ $t$ ],  $\theta$ [0] ==  $\theta_0$ ,
     $v$ [0] ==  $v_0$ }, { $\theta$ [ $t$ ],  $v$ [ $t$ ]}, { $t$ , 0,  $t_{max}$ }}];
  numtable = Table[{ $t$ , Evaluate[ $\theta$ [ $t$ ] /. numsol[[1]]]},
    { $t$ , 0,  $t_{max}$ ,  $\frac{t_{max}}{N1}$ }]]];

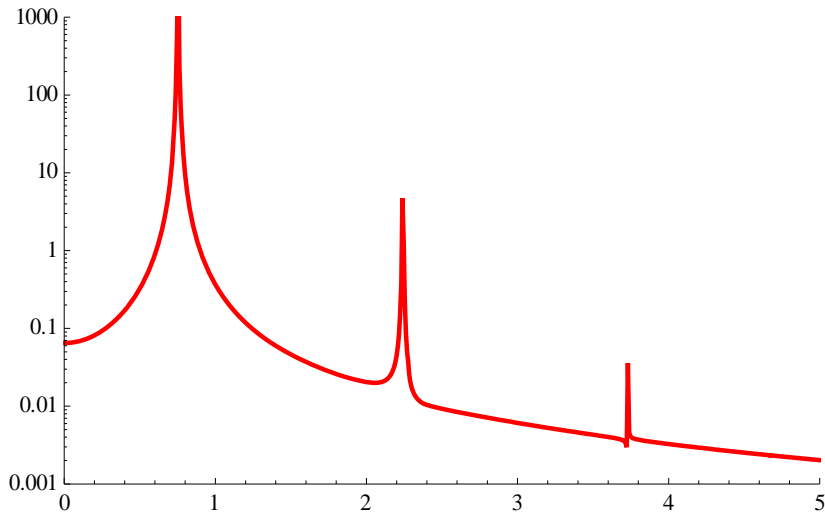
eq1 = pentime[{0, 0.4}, 1, 800, 8192]; d1 = Dimensions[eq1][[1]];
list1 = Table[eq1[[ $i$ , 2]], { $i$ , 1, d1 - 1}]; list2 = Fourier[list1] // Chop;
list3 = Table[{ $\frac{2\pi}{800} n$ , Abs[list2[[ $n$ ]]]^2}, { $n$ , 0, 8192}];
ListLogPlot[list3, Joined → True,
  PlotRange → {{0, 10}, {0.00001, 1000}}, PlotStyle → {Thick, Red}]

```



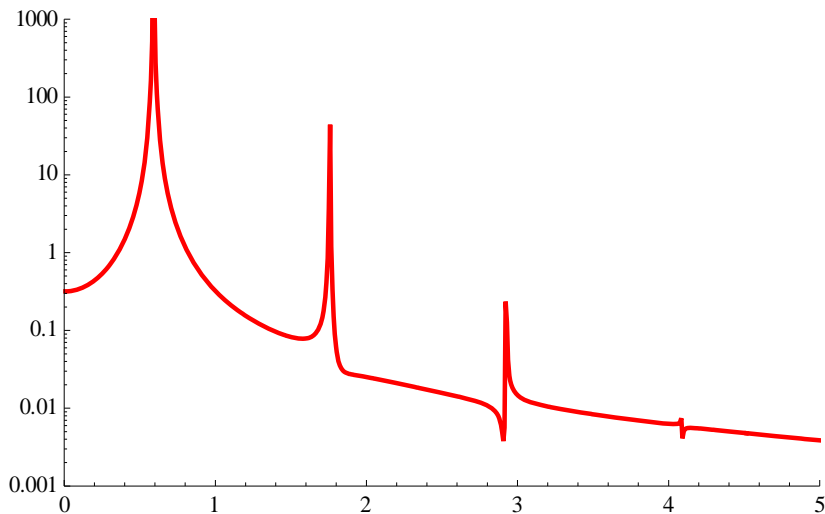
11.21 Example

```
eq1 = pentime[{0, 1.7}, 1, 800, 8192]; d1 = Dimensions[eq1][[1]];
list1 = Table[eq1[[i, 2]], {i, 1, d1 - 1}; list2 = Fourier[list1] // Chop;
list3 = Table[{ $\frac{2\pi}{800}n$ , Abs[list2[[n]]]^2}, {n, 0, 8192}];
ListLogPlot[list3, Joined → True, PlotRange → {{0, 5}, {0.001, 1000}},
  PlotStyle → {Thick, Red}]
```



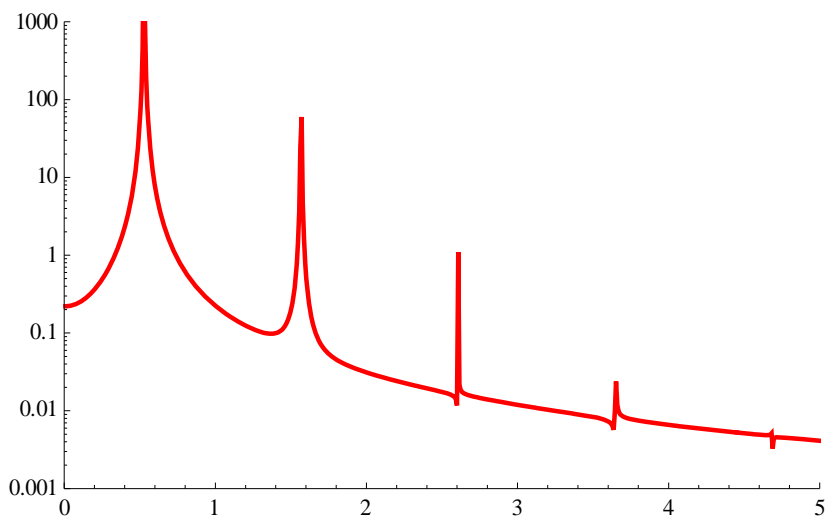
11.22 Example

```
eq1 = pentime[{0, 1.92}, 1, 800, 8192]; d1 = Dimensions[eq1][[1]];
list1 = Table[eq1[[i, 2]], {i, 1, d1 - 1}; list2 = Fourier[list1] // Chop;
list3 = Table[{ $\frac{2\pi}{800}n$ , Abs[list2[[n]]]^2}, {n, 0, 8192}];
ListLogPlot[list3, Joined → True, PlotRange → {{0, 5}, {0.001, 1000}},
  PlotStyle → {Thick, Red}]
```



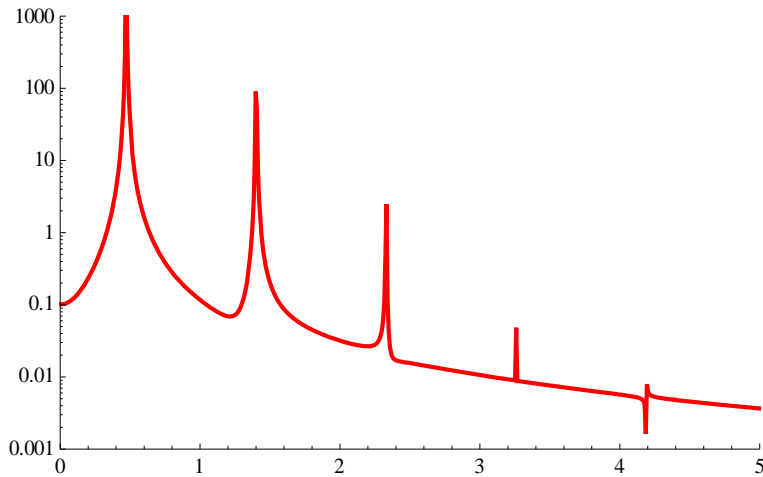
11.23 Example

```
eq1 = pentime[{0, 1.96}, 1, 800, 8192]; d1 = Dimensions[eq1][[1]];
list1 = Table[eq1[[i, 2]], {i, 1, d1 - 1}; list2 = Fourier[list1] // Chop;
list3 = Table[{ $\frac{2\pi}{800} n$ , Abs[list2[[n]]]^2}, {n, 0, 8192}];
ListLogPlot[list3, Joined → True, PlotRange → {{0, 5}, {0.001, 1000}},
  PlotStyle → {Thick, Red}]
```



11.24 Example

```
eq1 = pentime[{0, 1.981}, 1, 800, 8192]; d1 = Dimensions[eq1][[1]];
list1 = Table[eq1[[i, 2]], {i, 1, d1 - 1}; list2 = Fourier[list1] // Chop;
list3 = Table[{ $\frac{2\pi}{800} n$ , Abs[list2[[n]]]^2}, {n, 0, 8192}];
ListLogPlot[list3, Joined → True, PlotRange → {{0, 5}, {0.001, 1000}},
  PlotStyle → {Thick, Red}]
```



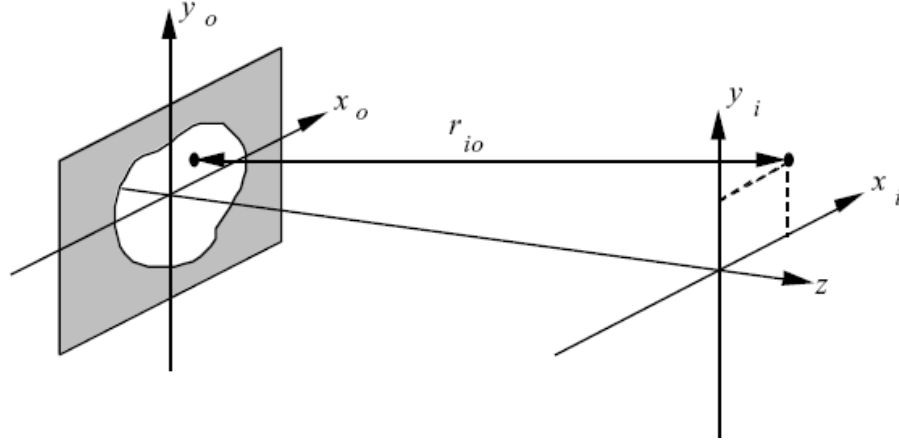
11.25 Fraunhofer diffraction: FFT in two dimensions

Joseph von Fraunhofer (6 March 1787 – 7 June 1826) was a German optician. He is known for the discovery of the dark absorption lines known as Fraunhofer lines in the Sun's spectrum, and for making excellent optical glass and achromatic telescope objectives.



http://en.wikipedia.org/wiki/Joseph_von_Fraunhofer

11.26 Principle of Fraunhofer diffraction¹



We assume that the aperture acts as the source of a field $E_0(x_0, y_0)$. The field $E_i(x_i, y_i)$ on the screen is given by¹

$$E_i(x_i, y_i) = \frac{\exp(ikz)}{i\lambda z} \iint E_0 \exp\left\{\frac{ik}{2z}[(x_i - x_0)^2 + (y_i - y_0)^2]\right\} dx_0 dy_0. \quad (1)$$

where the integral is a sum of the spherical wave of the field E_0 originating from the object points (x_0, y_0) within the aperture, and k is the wave number ($2\pi/\lambda$). So the surface integral runs over the aperture. Note that the spherical wave diverging from the object point (x_0, y_0) is described by $\exp(ikr_{io})/r_{io}$.

This is known as the Fresnel approximation to the scalar diffraction theory. It is useful when z is very large compared to a wavelength. A further rearrangement of the Fresnel diffraction expression will be computationally convenient.

$$E_i(x_i, y_i) = \frac{\exp(ikz)}{i\lambda z} \iint E_0 \exp\left\{\frac{ik}{2z}[(x_i^2 + x_0^2 - 2x_i x_0) + (y_i^2 + y_0^2 - 2y_i y_0)]\right\} dx_0 dy_0, \quad (2)$$

or

$$E_i(x_i, y_i) = \frac{\exp(ikz)}{i\lambda z} \exp\left[\frac{ik}{2z}(x_i^2 + y_i^2)\right] \iint E_0(x_0, y_0) \exp\left[\frac{ik}{2z}(x_0^2 + y_0^2)\right] \exp\left[-\frac{ik}{z}(x_i x_0 + y_i y_0)\right] dx_0 dy_0$$

If $z \gg k(x_0^2 + y_0^2)_{\max}$, this equation is approximated as

$$E_i(x_i, y_i) = \frac{\exp(ikz)}{i\lambda z} \exp\left[\frac{ik}{2z}(x_i^2 + y_i^2)\right] \iint E_0(x_0, y_0) \exp\left[-\frac{ik}{z}(x_i x_0 + y_i y_0)\right] dx_0 dy_0. \quad (3)$$

This infinite-distance limit is called the Fraunhofer regime and is the case usually considered in elementary treatments. This is simply the Fourier transform of the aperture illumination. We usually want to know the optical intensity, which is proportional to $|E_i|^2$, so the phase factor in front is irrelevant.

$$\left| \iint E_0(x_0, y_0) \exp[-(k_x x_0 + k_y y_0)] dx_0 dy_0 \right|^2, \quad (4)$$

where $k_x = \frac{kx_i}{2\pi z}$ and $k_y = \frac{ky_i}{2\pi z}$ are wave numbers (so called spatial frequencies), with units of $(\text{length})^{-1}$. So the diffraction pattern is simply the absolute square of the 2D Fourier transform of the aperture. Note that $(x_i, y_i) = \frac{2\pi z}{k}(k_x, k_y)$.

11.27 Definition of Fourier Transform

The *amplitude* in the diffraction pattern is simply the 2D Fourier transform of the aperture, besides scaling. We will look at intensity, obtained by taking the absolute value of the amplitude. This would be called the power spectrum if the independent variable were time instead of the space.

The conventional mathematical representation of an image is a function of two *spatial* variables, $f(x, y)$. The function at a particular location, (x, y) , is the intensity at that point. The term *transform* means an alternative mathematical representation of the image.

A *Fourier Transform* uses a series of complex exponentials (sinusoids) with different frequencies to represent an image. The Fourier Transform has applications in image processing, feature recognition, signal processing etc.

If $f(x, y)$ are spatial variables in the continuous domain the 2D Fourier Transform is defined as:

$$F(k_x, k_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} dx dy f(x, y) \exp[-i2\pi(k_x x + k_y y)], \quad (5)$$

where $f(x, y)$ is the light intensity at point (x, y) . (k_x, k_y) are the horizontal and vertical spatial frequencies respectively. Inversely, the Fourier Transform can be transformed back to the spatial domain by the inverse Fourier transform:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} dk_x dk_y F(k_x, k_y) \exp[i2\pi(k_x x + k_y y)]. \quad (6)$$

11.28 Numerical calculation using Fast Fourier transform (FFT)

In numerical calculation we use the program of the FFT (fast Fourier transform) of the Mathematica.

3.1 Fourier and inverse Fourier transform

There are two kinds of Fourier transforms in Mathematica 5.2.

(a) Fourier transform

```
Fourier[{x1, x2, ..., xn}]  
produces the list { , , ..., } that is the discrete Fourier transform of the list  
{ , , ..., }, defined by  
Multidimensional data can be transformed by  
Fourier[{ { , , ..., },  
          { , , ..., }, ...,  
          { , , ..., } }].
```

If we input exact data, the data will be numericalized to machine precision.

(b) Inverse Fourier Transform

```
InverseFourier[{y1, y2, ..., yn}]  
produces the list { , , ..., } that is the discrete inverse Fourier transform of { ,  
, ..., }, defined by  
InverseFourier[{ { , , ..., },  
                 { , , ..., }, ...,  
                 { , , ..., } }].
```

11.29 Power spectrum

The following discussion is based on the note written by FFT program (Richard G. Palmer)² and the Mathematica programs (Graphics) written by Micahel Trott.³

(a) Shape of the Aperture:

We represent apertures (or *masks*) by arrays of 0's (opaque) and 1's (transparent). We can use **ListDensityPlot[]** to plot them, with the **Mesh→False** option to suppress the grid lines.

(b) Power spectrum of the FFT

Fourier[] from which the FFT of the aperture can be calculated.

Abs[]^2 from which the power spectrum is calculated.

(c) The center position

We need "frequencies" (here called spatial frequencies, or wave numbers, or just k 's) running from $-K$ to K , whereas the FFT gives 0 to $2K$. That applies in both the k_x and k_y direction ($k_x = -K$ to K , $k_y = -K$ to K). We can do both rotations at once with a `RotateLeft[array, {m, n}]`, where $\{m, n\}$ is the center position.

(d) Erasing of the large intensity

There is a problem. The diffraction bands are far too bright. That is because `ListDensityPlot[]` has truncated some of the larger values, just like `Plot[]` and `ListPlot[]` do. `PlotRange->All` tells it to behave, making pure white represent the largest value:

11.30 Power spectrum of small square aperture

Here we show the power spectrum of small square aperture which is calculated using Mathematica.

((Mathematica)) I


```

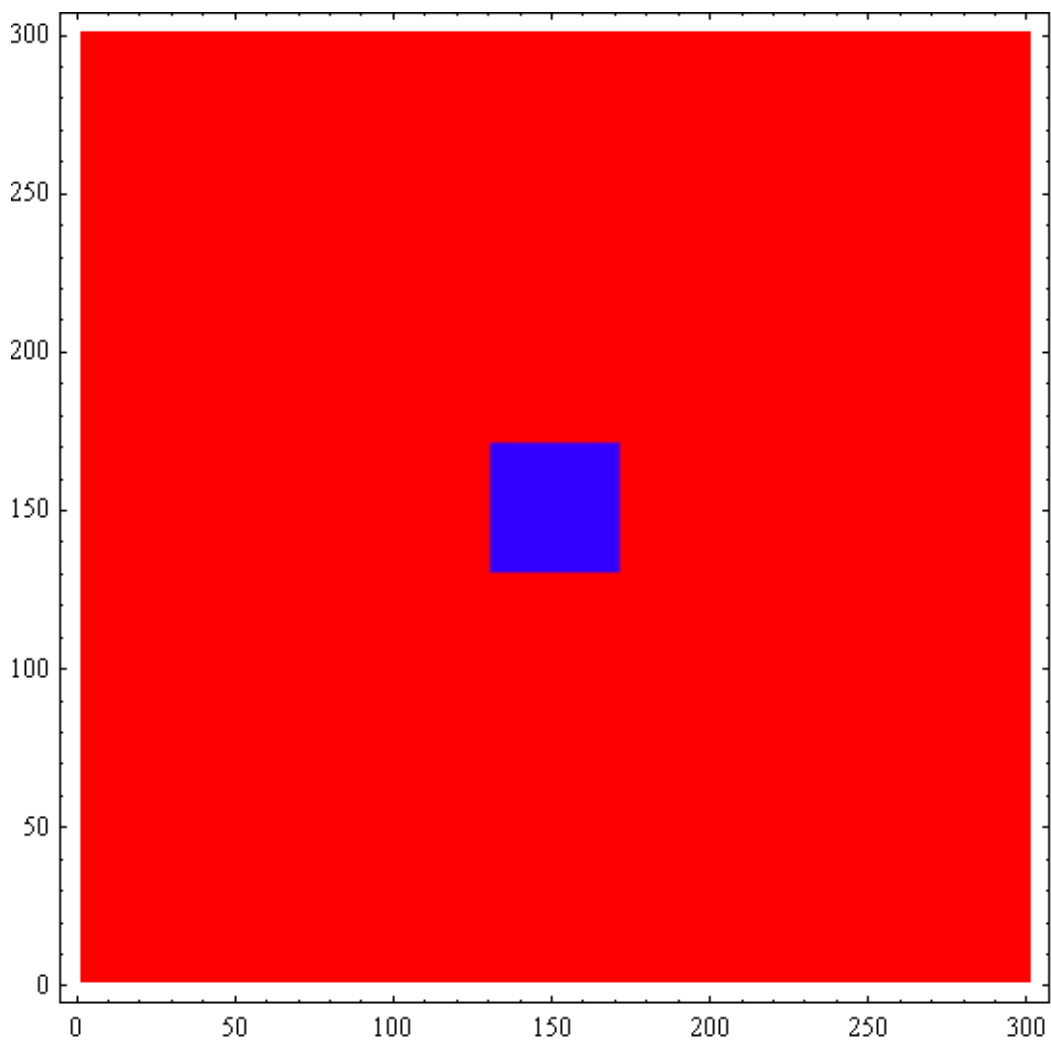
Clear["Global`*"]

n = 300;

square =
  Table[If[n/2 - 20 ≤ x ≤ n/2 + 20 && n/2 - 20 ≤ y ≤ n/2 + 20,
    1.0, 0.0], {y, 0, n}, {x, 0, n}];

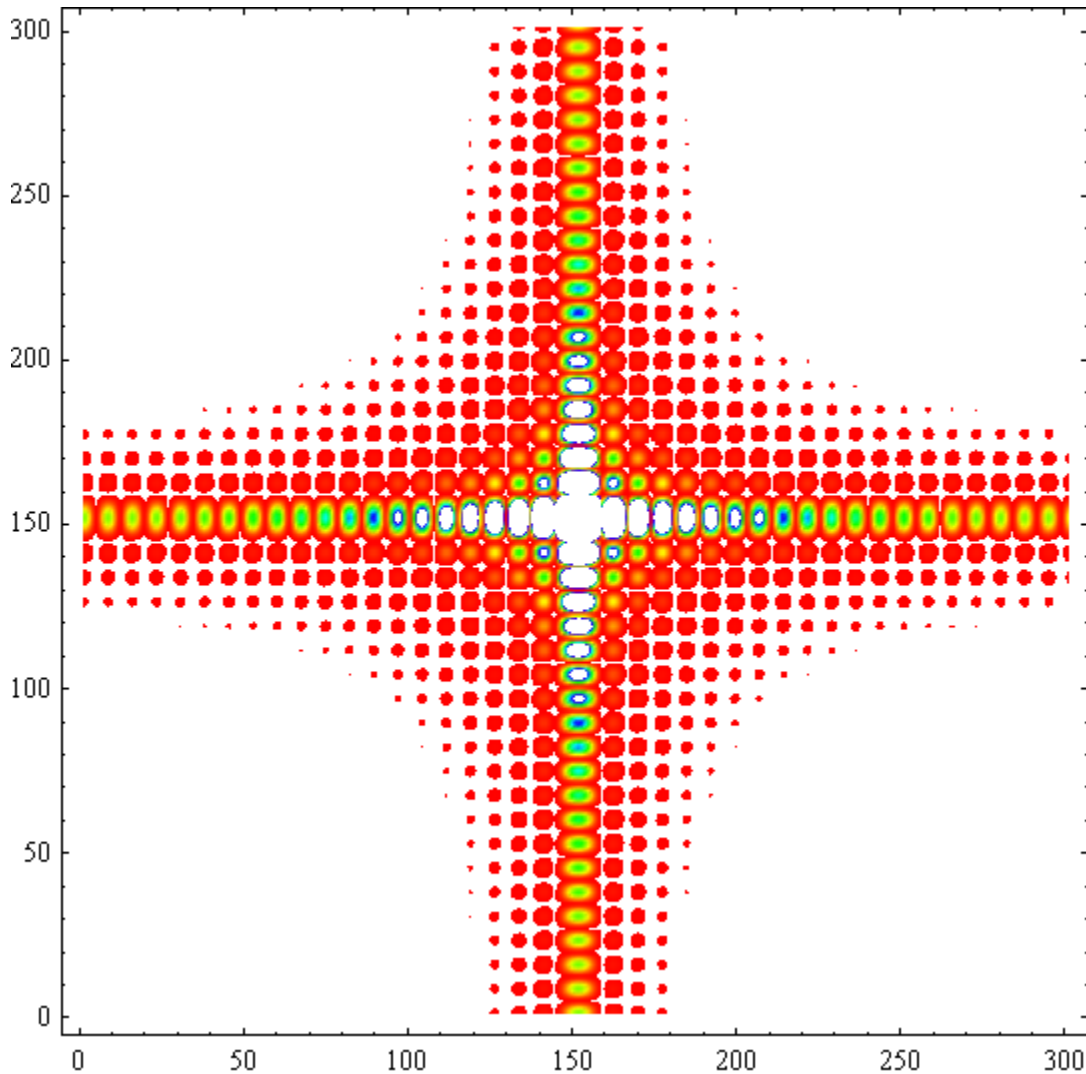
ListDensityPlot[square, Mesh → False,
  ColorFunction → (Hue[0.7 #] &)]

```



```
fftsq = Fourier[square];
pssq = Abs[fftsq] ^ 2;

ldp = ListDensityPlot[RotateLeft[pssq, {n/2, n/2}],
  Mesh → False, ColorFunction → (Hue[0.7 #] &),
  PlotRange → {0.0001, 0.05}]
```



11.31 Power spectrum of small circular hole aperture

((Mathematica))

```

Clear["Global`*"]

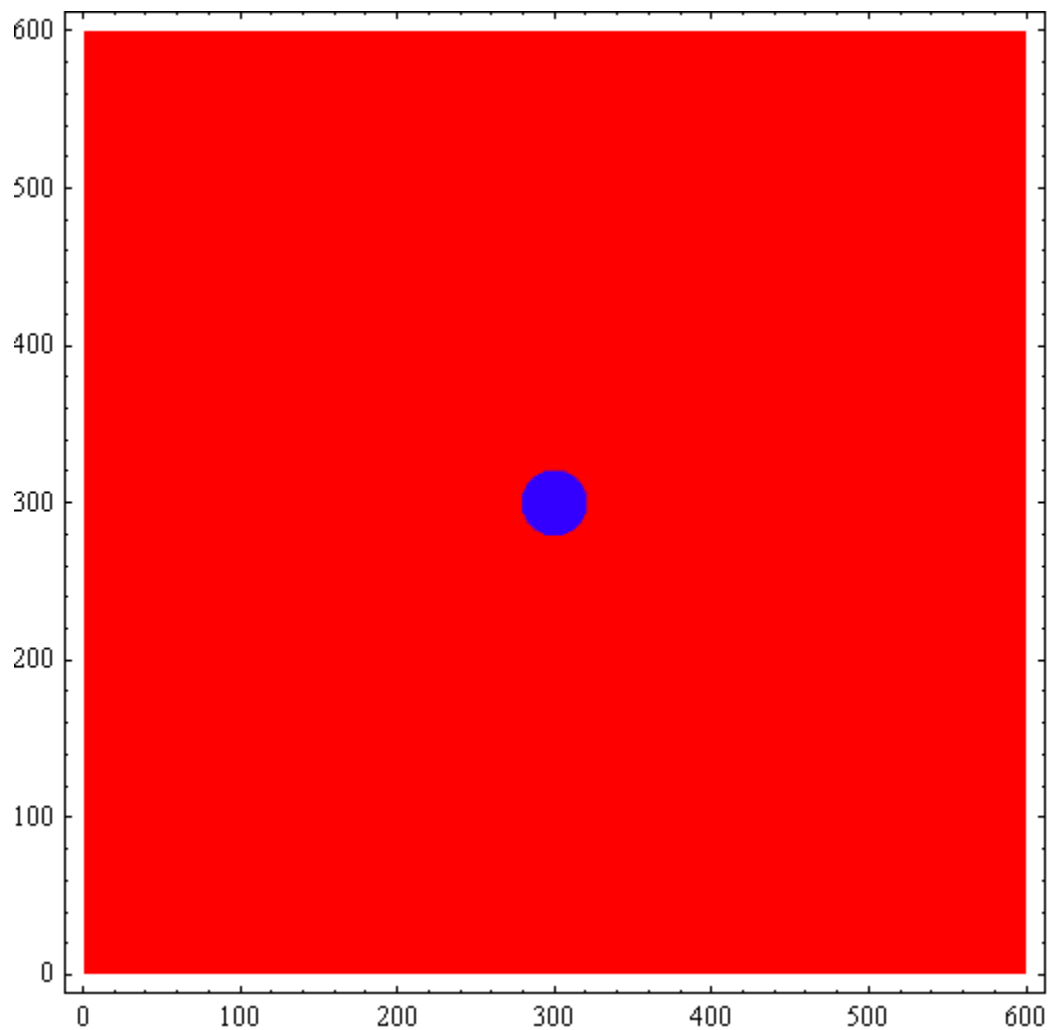
n = 600; (* n x n *)

r = 0.07; (* Radius *)

rr = (r*n*0.5)^2;

circ = Table[If[(x - n/2 + 1)^2 + (y - n/2 + 1)^2 < rr,
    1.0, 0.0], {y, 0, n - 1}, {x, 0, n - 1}];
ListDensityPlot[circ, Mesh → False,
    ColorFunction → (Hue[0.7 #] &)]

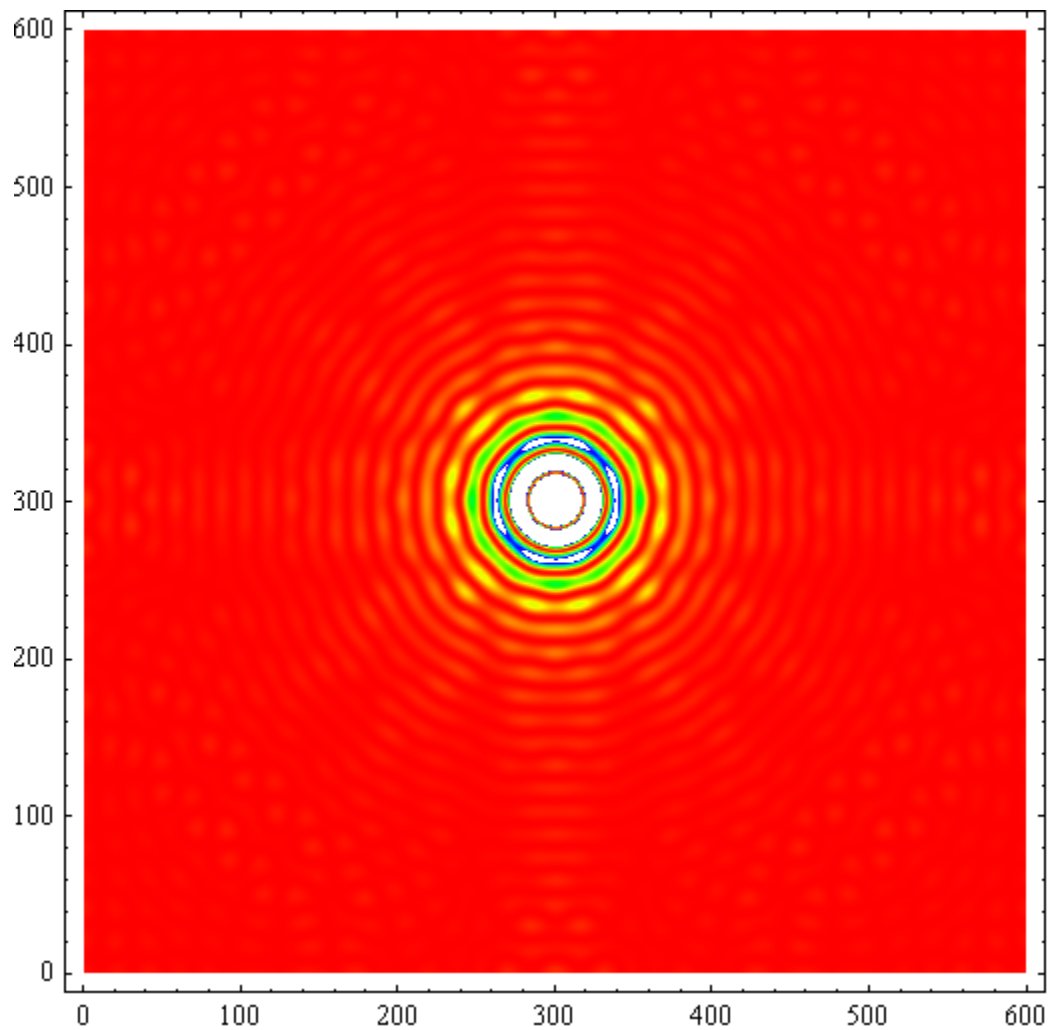
```



```

pscirc = Abs[Fourier[circ]]^2;
ListDensityPlot[RotateLeft[pscirc, {n/2, n/2}],
  Mesh → False, PlotRange → {0, 0.02},
  ColorFunction → (Hue[0.7 #] &)]

```



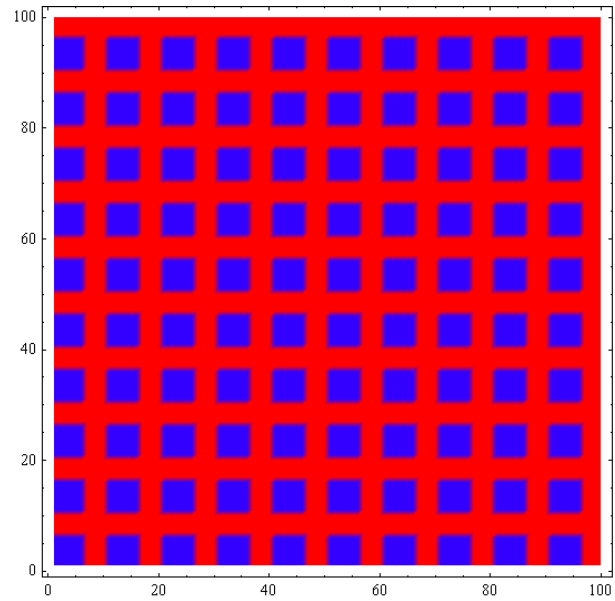
11.32 Power spectrum of square lattice

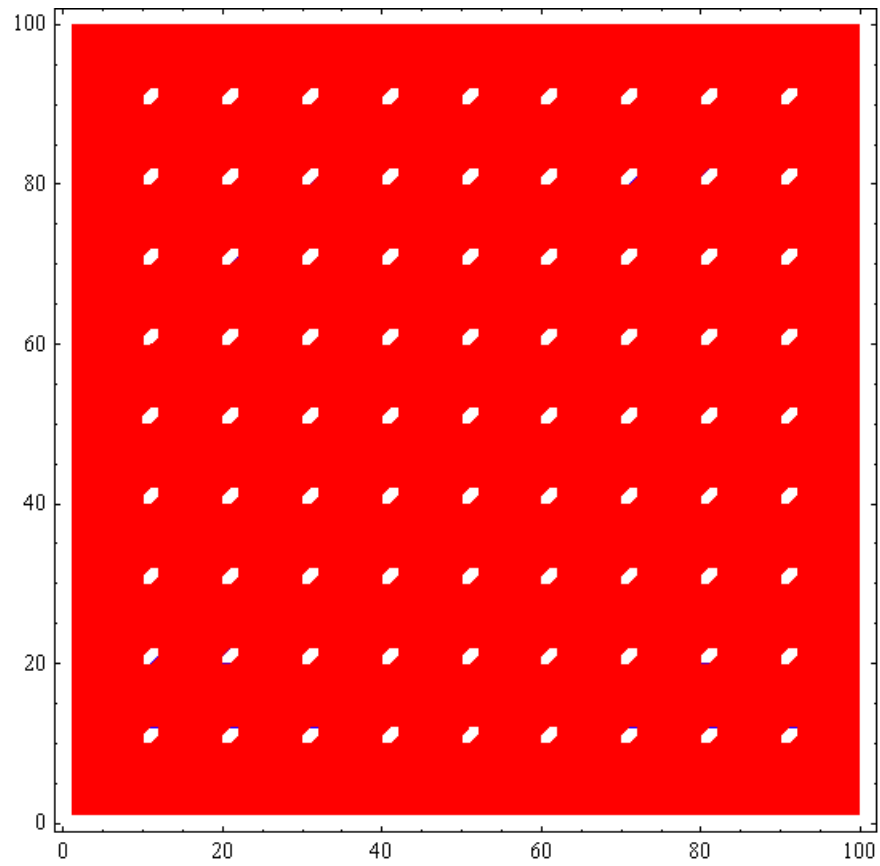
((Mathematica))

```

n = 100; (* n x n , n is even*);
doubleslit1 =
  Table[If[5 < x < 10 || 15 < x < 20 || 25 < x < 30 || 35 < x < 40 || 45 < x < 50 || 55 < x < 60 ||
    65 < x < 70 || 75 < x < 80 || 85 < x < 90 || 95 < x < 100 || 5 < y < 10 || 15 < y < 20 || 25 < y < 30 ||
    35 < y < 40 || 45 < y < 50 || 55 < y < 60 || 65 < y < 70 || 75 < y < 80 || 85 < y < 90 || 95 < y < 100, 0, 1],
    {y, 0, n-1}, {x, 0, n-1}];
ListDensityPlot[doubleslit1, Mesh → False, ColorFunction → (Hue[0.7 #] &)]
ps1 = Abs[Fourier[doubleslit1]]^2; dh1 = RotateLeft[ps1, {n/2, n/2}];
ListDensityPlot[dh1, Mesh → False, PlotRange → {0, 0.01}, ColorFunction → (Hue[0.7 #] &)]

```





11.33 Power spectrum of a circular ring

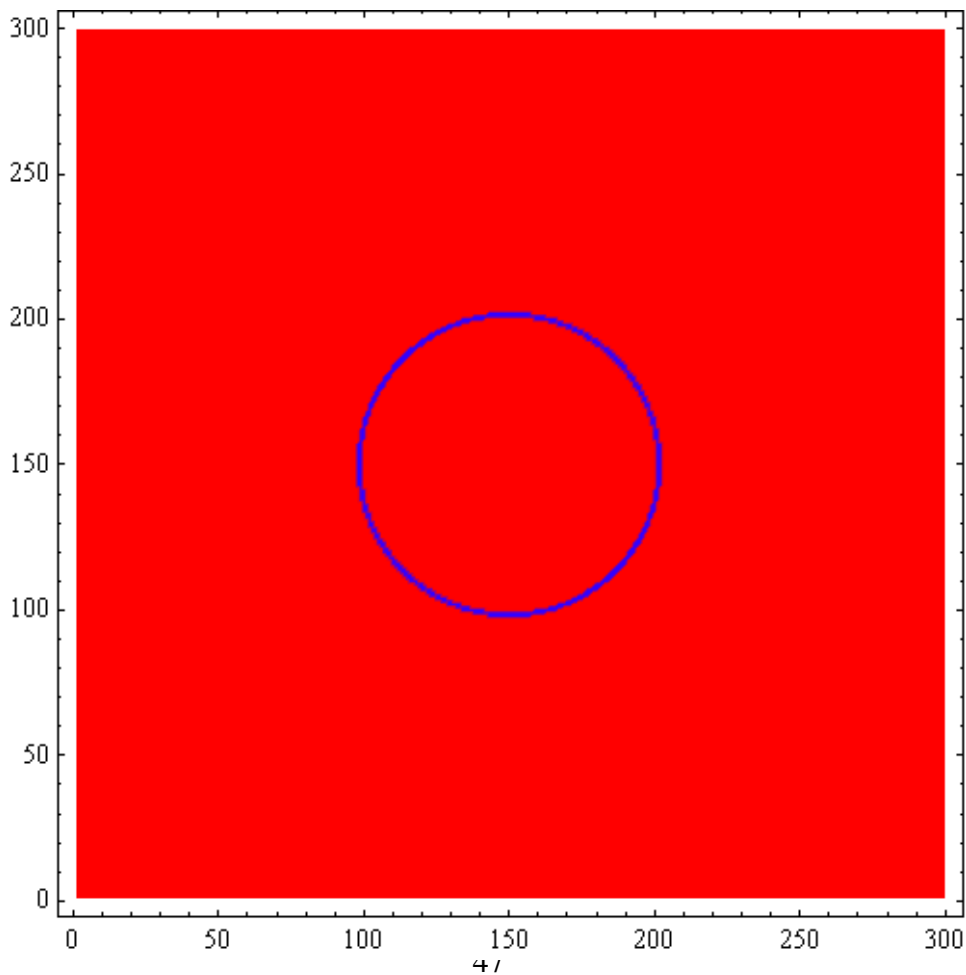
```

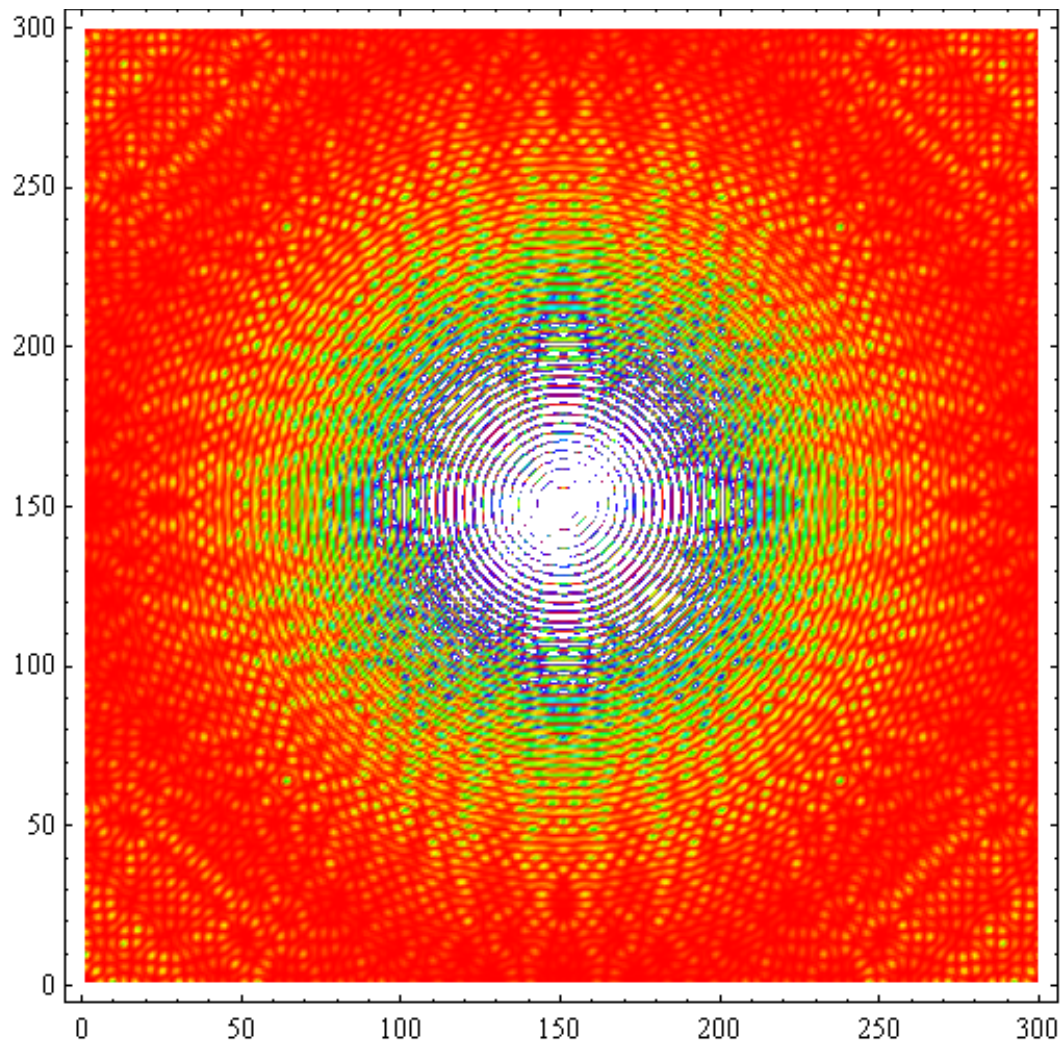
Clear["Global`*"]

n = 300; (* n x n *)
r = 0.35; (* Radius *)
rr = (r * n * 0.5) ^ 2; rrm = (r * (n - 10) * 0.5) ^ 2
circ =
  Table[If[rrm < (x - n / 2 + 1) ^ 2 + (y - n / 2 + 1) ^ 2 < rr,
    1.0, 0.0], {y, 0, n - 1}, {x, 0, n - 1}];
ListDensityPlot[circ, Mesh → False,
  ColorFunction → (Hue[0.7 #] &)]
pscirc = Abs[Fourier[circ]] ^ 2;
ListDensityPlot[RotateLeft[pscirc, {n / 2, n / 2}],
  Mesh → False, PlotRange → {0, 0.03},
  ColorFunction → (Hue[0.7 #] &)]

```

2575.56





11.34 Double-slit diffraction

Fig.1a Shape of the aperture (double slits).

Fig.1b Its power spectrum.

Fig.1c Intensity distribution of the power spectrum along the k_y axis with $k_x = n/2$.

Fig.1d Intensity distribution of the power spectrum along the k_x axis with $k_y = n/2$.

((Mathematica))


```

Clear["Global`*"]

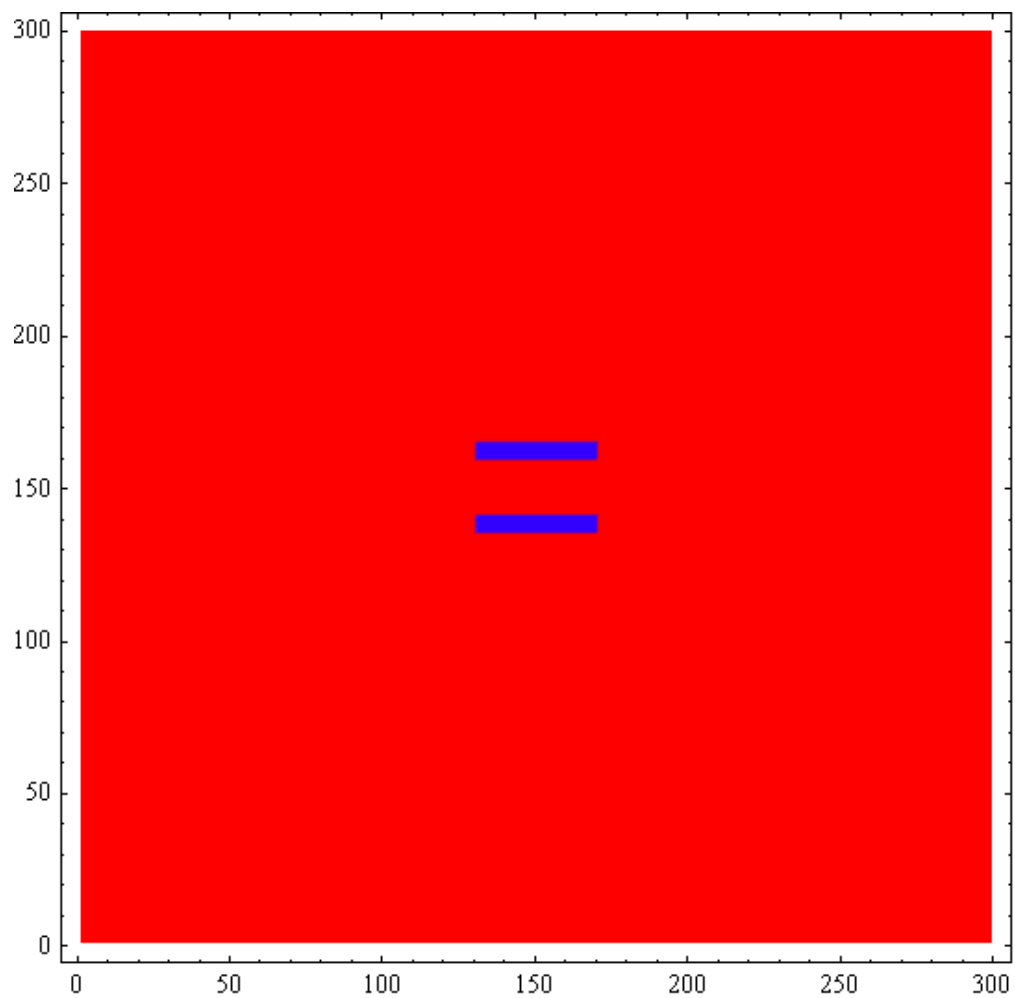
n = 300;

doubleslit1 =
  Table[If[ $\left(\frac{n}{2} - 15 \leq y < \frac{n}{2} - 9 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) \ || \$   

 $\left(\frac{n}{2} + 9 \leq y < \frac{n}{2} + 15 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right)$ , 1., 0.],
    {y, 0, n - 1}, {x, 0, n - 1}];

ListDensityPlot[doubleslit1, Mesh → False,
  ColorFunction → (Hue[0.7 #1] &)]

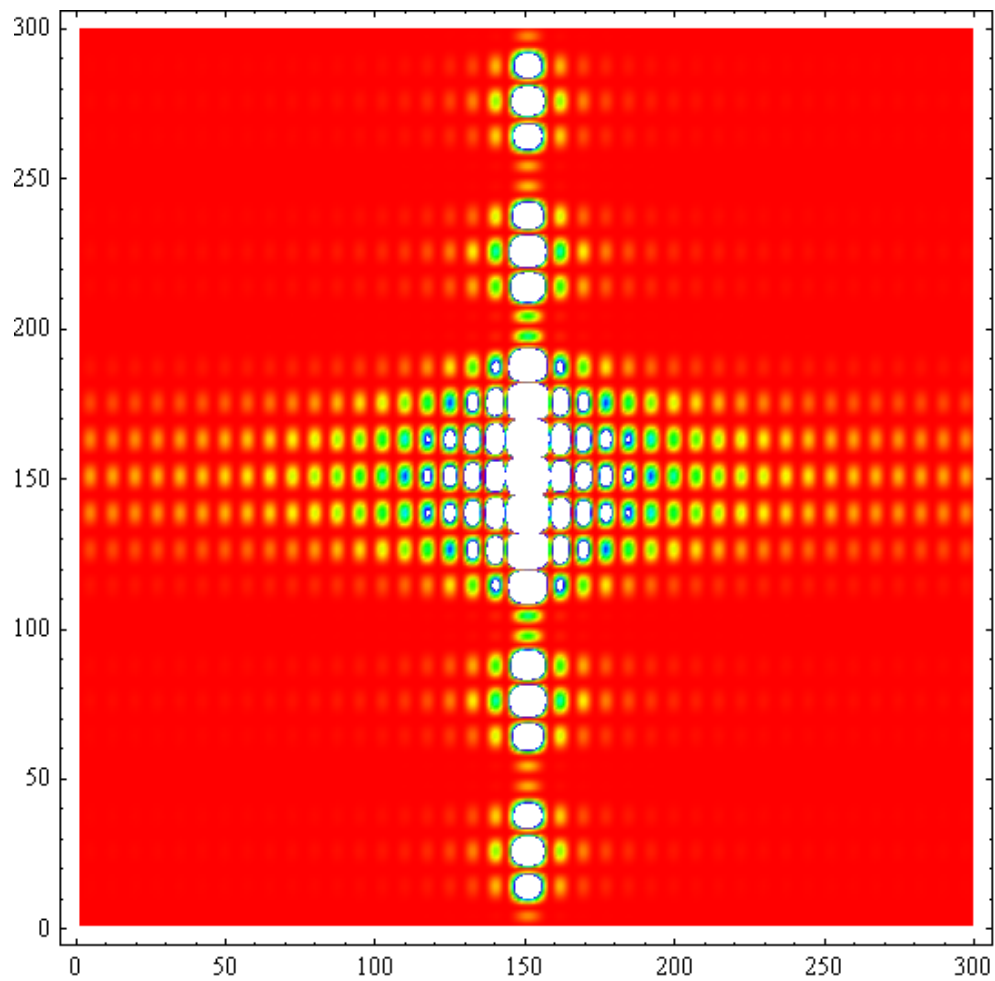
```



```

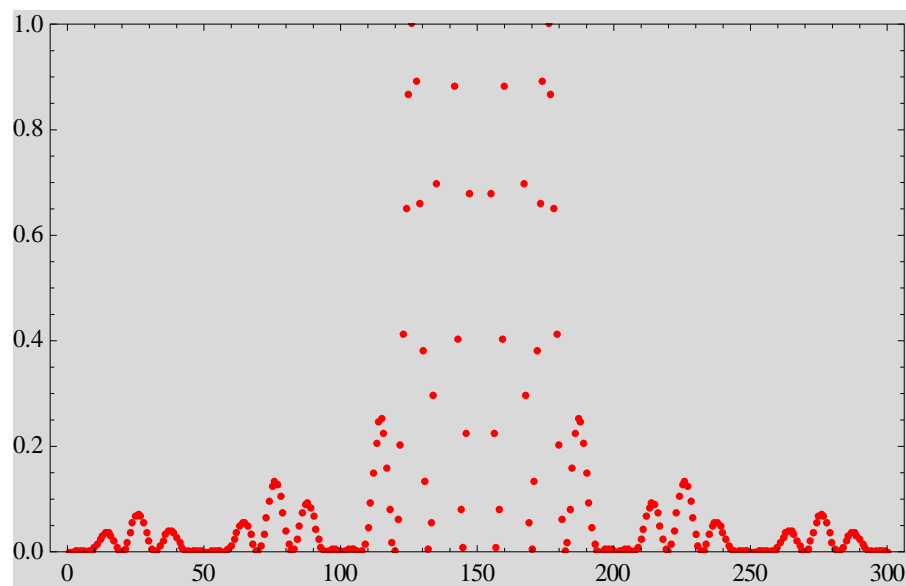
ps1 = Abs[Fourier[doubleslit1]]2;
dh1 = RotateLeft[ps1, { $\frac{n}{2}$ ,  $\frac{n}{2}$ }];
ListDensityPlot[dh1, Mesh → False,
  PlotRange → {0, 0.01}, ColorFunction → (Hue[0.7 #1] &)]

```



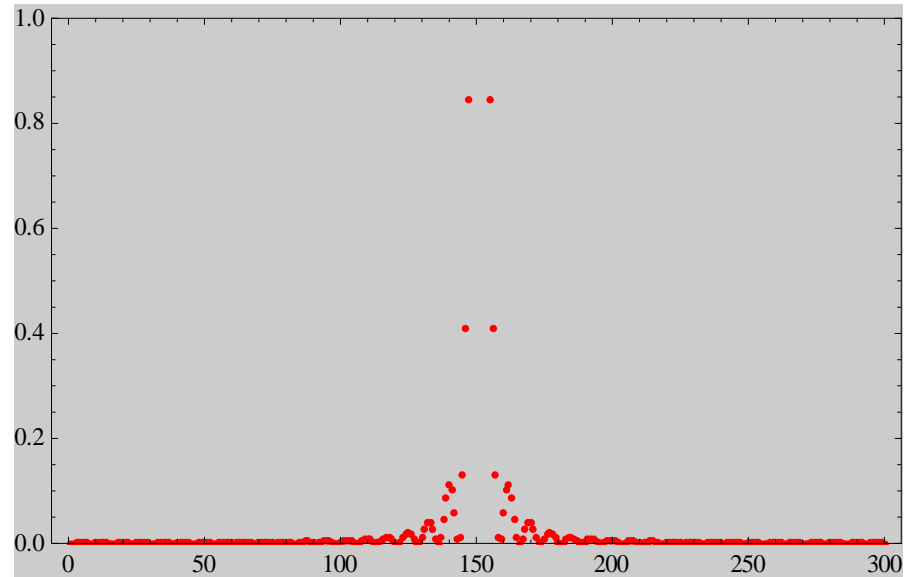
`dh1[[m,n/2]]` means that the scan is made along the y axis at $x = n/2$

```
ListPlot[Evaluate[Table[dh1[[m, n/2]], {m, 1, n}]],  
PlotRange -> {0, 1}, Prolog -> AbsolutePointSize[4],  
PlotStyle -> Hue[0], Background -> LightGray,  
Frame -> True]
```



$dh1[[n/2,m]]$ means that the scan is made along the x axis at $y = n/2$

```
ListPlot[Evaluate[Table[dh1[[n/2,m]], {m, 1, n}]],
PlotRange -> {0, 1}, Prolog -> AbsolutePointSize[4],
PlotStyle -> Hue[0], Background -> GrayLevel[0.8],
Frame -> True]
```



11.36. Power spectrum of the diffraction grating

Fig.1a Shape of the aperture (diffraction grating)

Fig.2b Power spectrum

Fig.2c Intensity distribution of the power spectrum along the k_y axis with $k_x = n/2$,

Fig.2d Intensity distribution of the power spectrum along the k_x axis with $k_y = n/2$,

```

Clear["Global`*"]

n = 300;

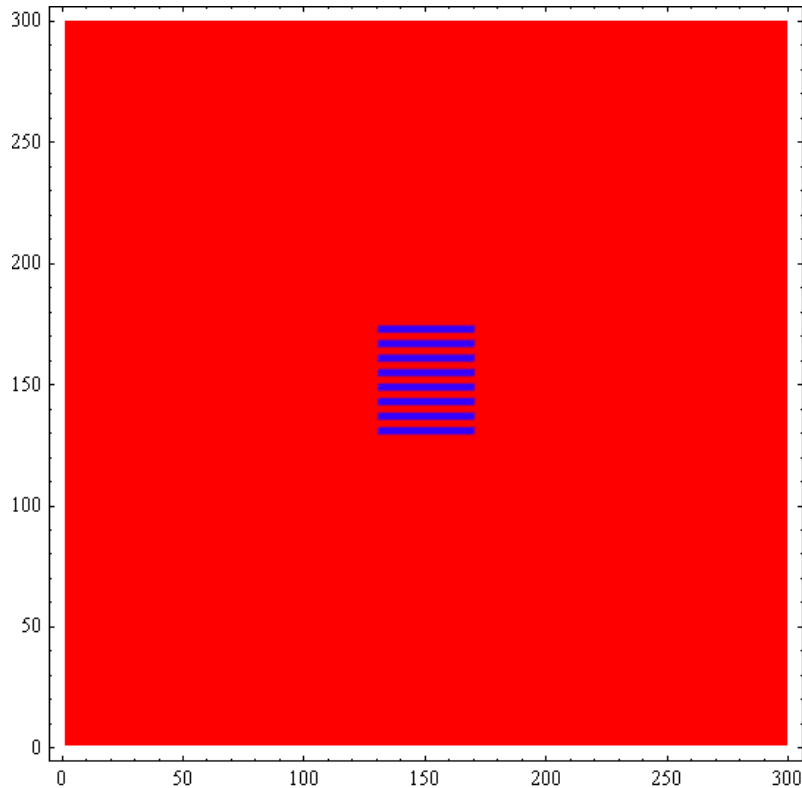
slit1 =
Table[If[ $\left(\frac{n}{2} - 21 \leq y < \frac{n}{2} - 18 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} - 15 \leq y < \frac{n}{2} - 12 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} - 9 \leq y < \frac{n}{2} - 6 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} - 3 \leq y < \frac{n}{2} \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} + 3 \leq y < \frac{n}{2} + 6 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} + 9 \leq y < \frac{n}{2} + 12 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} + 15 \leq y < \frac{n}{2} + 18 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right) ||$ 
 $\left(\frac{n}{2} + 21 \leq y < \frac{n}{2} + 24 \ \&\& \ \frac{n}{2} - 20 \leq x < \frac{n}{2} + 20\right), 1., 0.],$ 
{y, 0, n - 1}, {x, 0, n - 1}];

```

```

ListDensityPlot[slit1, Mesh → False,
ColorFunction → (Hue[0.7 #1] &)]

```



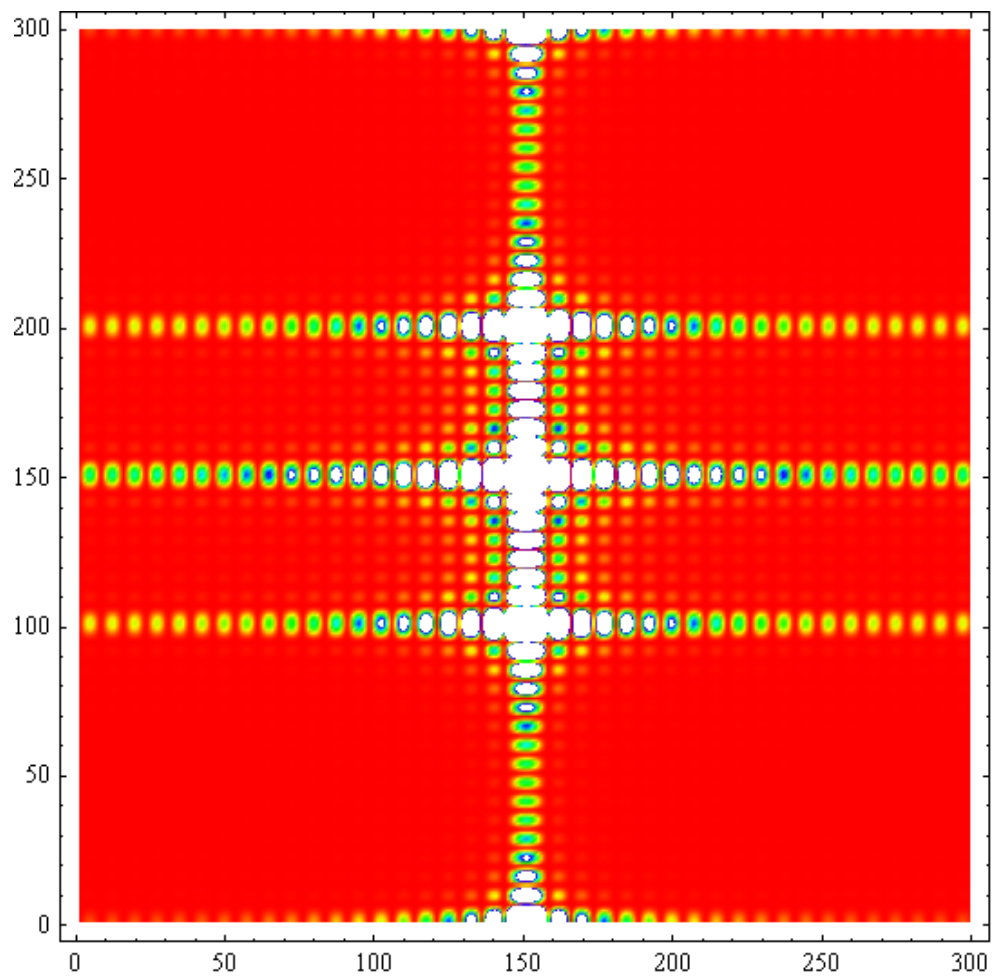
```

ps1 = Abs[Fourier[slit1]]2;

dh1 = RotateLeft[ps1, { $\frac{n}{2}$ ,  $\frac{n}{2}$ }];

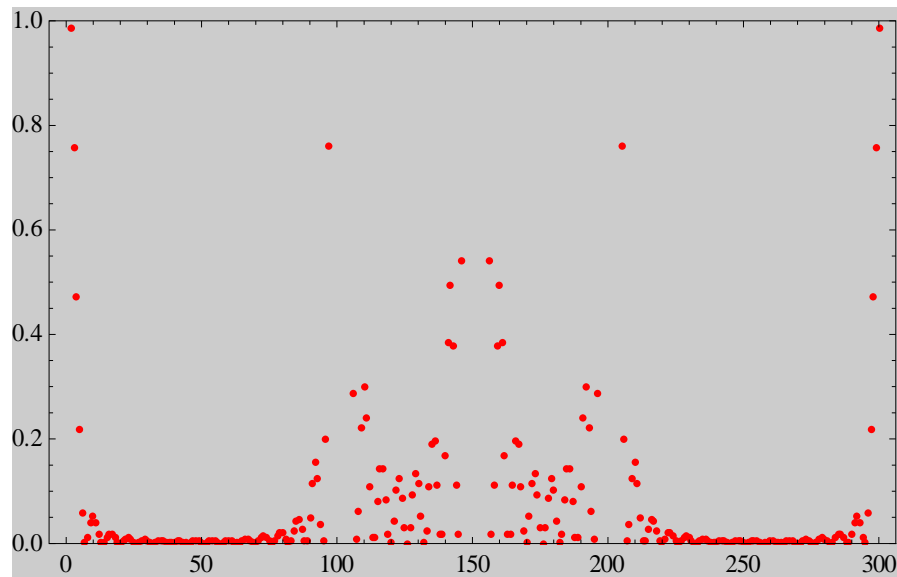
ListDensityPlot[dh1, Mesh → False,
  PlotRange → {0, 0.01}, ColorFunction → (Hue[0.7 #1] &)]

```



$dh1[[m, n/2]]$ means that the scan is made along the y axis at $x = n/2$

```
ListPlot[Evaluate[Table[dh1[[m, n/2]], {m, 1, n}],
  PlotRange -> {0, 1}, Prolog -> AbsolutePointSize[4],
  PlotStyle -> Hue[0], Background -> GrayLevel[0.8],
  Frame -> True]
```



$dh1[[n/2, m]]$ means that the scan is made along the x axis at $y = n/2$

```
ListPlot[Evaluate[Table[dh1[[n/2, m]], {m, 1, n}],
  PlotRange -> {0, 0.5}, Prolog -> AbsolutePointSize[4],
  PlotStyle -> Hue[0], Background -> GrayLevel[0.8],
  Frame -> True]
```

