6 Electromagnetism

6.1 Diffraction pattern due to a current strip (7 units)

Knowledge of material covered in the Part IB course Electromagnetism is useful as background.

This project investigates the magnetic field generated by an oscillating current. The field is given in terms of an integral whose behaviour is analysed numerically.

1 Theory

Consider an infinite two-dimensional strip of conductive material in the plane $y = 0$ that covers the area defined by $-d < x < d$ and $-\infty < z < \infty$. A time-dependent current flows in the z-direction, and it emits electromagnetic (radio) waves with wavelength λ. We assume that $d = n\lambda/2$ where n is a positive integer. The time-dependent current is independent of x, z, and is given by

$$j_z(t) = j_0 e^{i\omega t}.$$

where j_0 is a parameter and $\omega = 2\pi c/\lambda$. In the following, all length scales are normalised so that $\lambda = 1$, hence for example $d = n/2$.

Now consider the component of the magnetic field in the x-direction. It is independent of z. For this particular form of $j_z(t)$, it can be derived from Maxwell's equations of electromagnetism as $H_x(x, y, t) = j_z(t)h_x(x, y)$ with

$$h_x(x, y) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{2\pi i u x} A(u, y) \mathrm{d}u \tag{1}$$

where

$$A(u, y) = \frac{\sin(n\pi u)}{u} \times \begin{cases} \exp\left(2\pi i y\sqrt{1-u^2}\right), & |u| \leqslant 1 \\ \exp\left(-2\pi|y|\sqrt{u^2-1}\right), & |u| > 1 \end{cases} \tag{2}$$

To avoid ambiguity, it is convenient to specify $A(0, y) = \lim_{u\to 0} A(u, y)$.

It can be shown that for large y, the complex modulus of the magnetic field asymptotically approaches

$$|h_x| \simeq \left|\frac{\sin n\pi v}{2\pi v}\right| \sqrt{\frac{v(1-v^2)}{x}}\,, \tag{3}$$

where

$$v = \frac{x}{\sqrt{x^2+y^2}}\,.$$

This project investigates numerical approximations to $h_x(x, y)$, as defined in (1).

2 Numerical method

The right-hand side of (1) is a Fourier integral. Numerical estimation of this function has some tricky features: for example, if x is large then the integrand oscillates rapidly in u. This project uses a specialised method for integrals of this type, called the fast Fourier transform (FFT). It is a very efficient method, in particular it allows simultaneous estimation of $h_x(x, y)$ at N distinct values of x.

 © University of Cambridge

To apply the method, note first that A decays rapidly for large u, so it is reasonable to introduce a (large) parameter U and approximate $h_x(x,y)$ as

$$h_x(x,y) \approx \frac{1}{2\pi}\int_{-U}^{U} e^{2\pi iux} A(u,y)\mathrm{d}u \tag{4}$$

This approximation is accurate for sufficiently large U.

Now define a periodic function A^{per} with period $2U$ by taking $A^{\mathrm{per}}(u,y) = A(u,y)$ for $|u| \leqslant U$ and $A^{\mathrm{per}}(u+2mU,y) = A^{\mathrm{per}}(u,y)$ for any integer m. The integral in (4) is unchanged on replacing A by A^{per}. The domain of integration can then be replaced by $[0,2U]$, and it is natural to estimate the integral by a (Riemann) sum. Define

$$\hat{h}_x(x,y) = \frac{\Delta u}{2\pi}\sum_{k=0}^{N-1} e^{2\pi ikx\Delta u} A^{\mathrm{per}}(k\Delta u,y) \tag{5}$$

with $\Delta u = 2U/N$.

Under certain conditions, this allows $h_x(x,y)$ to be approximated by $\hat{h}_x(x,y)$, but the accuracy of this approximation requires some care. For example $\hat{h}_x$ exhibits rapid oscillations as a function of x, which are not present in h_x. Also, the right hand side of (5) can be recognised as a Fourier series (or discrete Fourier transform, DFT). Hence $\hat{h}_x(x,y)$ is periodic in x, specifically $\hat{h}_x(x,y) = \hat{h}_x(x+2mX,y)$ with $X = 1/(2\Delta u)$. However, h_x is not periodic.

To understand the relation of $\hat{h}_x$ to h_x, define a periodic function h_x^{per} by taking $h^{\mathrm{per}}(x,y) = h(x,y)$ for $|x| \leqslant X$ and $h_x^{\mathrm{per}}(x+2mX,y) = h_x^{\mathrm{per}}(x,y)$ for any integer m. Define also $\Delta x = 2X/N$. Then for integer m and sufficiently large values of N and U, one has

$$\hat{h}_x(m\Delta x,y,t) \approx h_x^{\mathrm{per}}(m\Delta x,y,t)\ . \tag{6}$$

Under these conditions, h_x can be approximated by $\hat{h}_x$ as long as $|x| \leqslant X$ and $x = m\Delta x$. This construction relies on the fact that $\Delta x\Delta u = 1/N$ so that the exponential factors in (5) are the Nth roots of unity.

The FFT method is an efficient algorithm for computing sums of the form (5), for $x = m\Delta x$ and $m = 0,1,2,\ldots,N-1$. This allows accurate estimation of $h_x^{\mathrm{per}}(m\Delta x,y,t)$ for $x \in [0,2X]$ and hence of h_x. The method is described in the Appendix. For cases where N is an integer power of 2, the FFT is much faster than computing the sum (5) individually for each value of m in turn. For this project, it is not necessary to understand any of the details, you only need to invoke an FFT routine to compute the relevant quantities. You may use a Matlab routine such as `fft` or `ifft`, or an equivalent routine in any other language, or you may write your own (but you should not compute (5) directly).

Finally, note that we have defined the method by taking N and U as parameters, from which $\Delta u, \Delta x, X$ are derived. From a practical point of view it is more natural to take N and X as parameters, from which one may derive U and the other relevant quantities.

3 Numerical work

Programming Task: Given values of n, y, N, X, write a program to compute (5) by FFT, for $x = m\Delta x$ and $m = 0,1,\ldots,N-1$. It is sufficient to restrict to $N = 2^p$ for integer p. The program should also use (6) to estimate the real and imaginary parts of h_x for $x \in [-X,X]$. Also estimate its complex modulus $|h_x|$. It will be necessary to plot these estimates.

 © University of Cambridge

Question 1 Take

$$n = 2\,, \qquad y = 0.2\,, \qquad X = 5\,, \qquad N = 256\ .$$

Plot your estimates of the real and imaginary parts of h_x, and its modulus, for $|x| < X$. Derive the relationships between $h_x(x, y)$ and $h_x(-x, y)$ and $h_x(x, -y)$. Verify that your results are consistent with these relationships.

Question 2 Keeping $n = 2$ and $y = 0.2$, compute estimates of $h_x(x, y)$ for $|x| < 5$, using different values of X and N (always with $X \geqslant 5$). Analyse the behaviour of your estimates, as N and X are varied.

Note: In this question and throughout this project, you should provide graphs that illustrate clearly the effect of the parameters on your results. Note that large numbers of graphs are *very unlikely to be effective* in communicating this information.

Question 3 For $n = 2$, produce a single graph that shows $|h_x|$ as a function of x for for $y = 0.12, 0.6, 1, 6, 12$. Fix $N = 256$ and choose suitable values of X (dependent on y). Justify the values that you have chosen. Are there some values of y for which larger (or smaller) values of N would be appropriate?

Compare your numerical results for large y with the asymptotic formula (3). This comparison must be presented in a way that illustrates clearly any differences between the numerical estimates and the asymptotic formula. It may be useful to consider additional values of y, as well as those listed above.

Question 4 Perform a similar analysis to question 3 but now for $n = 3, 4$. Justify your choices of N, X. Combining these results with those of question 3, discuss how the approximation of h by $\hat{h}$ depends on both n, y and N, X.

Question 5 Comment on the physical significance of your results. In particular, how do your results demonstrate the phenomenon of diffraction?

Appendix: The Fast Fourier Transform

Given a vector of complex numbers $\mu = (\mu_0, \mu_1, \ldots, \mu_{N-1})$, define

$$\lambda_r = \sum_{k=0}^{N-1} \mu_k e^{-2\pi i k r/N}\ . \tag{7}$$

The FFT is an efficient (fast) method of evaluating the vector $\lambda = (\lambda_0, \lambda_1, \ldots, \lambda_{N-1})$, which is the discrete Fourier transform. The same algorithm can also be used to evaluate similar vectors where the factor $e^{-2\pi i k r/N}$ in the definition of λ_r is replaced by $e^{2\pi i k r/N}$, this is sometimes called the inverse FFT.

 © University of Cambridge

Note that (7) corresponds to multiplication of the vector μ by a particular $N \times N$ matrix that we denote by $\Omega^{(N)}$. Its elements are taken from the set of Nth roots of unity. It follows that λ can be computed using approximately N^2 multiplication operations. (There would be a similar number of addition operations, it is assumed here that the multiplication operations take the greater part of the computational effort.) If $N = 2^p$ for integer p, the FFT can compute λ much more quickly, it requires approximately $(N/2)\log_2 N$ multiplication operations.

To see this, divide μ into even and odd subsequences, that is $\mu^{\mathrm{E}} = (\mu_0, \mu_2, \ldots, \mu_{N-2})$ and $\mu^{\mathrm{O}} = (\mu_1, \mu_3, \ldots, \mu_{N-1})$. Their Fourier transforms are given by matrix multiplication as

$$\lambda^{\mathrm{E}} = \Omega^{(N/2)}\mu^{\mathrm{E}}, \qquad \lambda^{\mathrm{O}} = \Omega^{(N/2)}\mu^{\mathrm{O}} \,. \tag{8}$$

Then it may be shown that

$$\left.\begin{aligned} \lambda_r &= \lambda_r^E + e^{2\pi i r/N}\lambda_r^O \\ \lambda_{r+N/2} &= \lambda_r^E - e^{2\pi i r/N}\lambda_r^O \end{aligned}\right\} \qquad r = 0, 1, \ldots, \tfrac{N}{2} - 1 \tag{9}$$

Hence if λ^{E} and λ^{O} are known, it requires $(N/2)$ multiplications to evaluate λ.

Moreover, since λ^{E} is itself the Fourier transform of a particular sequence μ^{E}, it can be estimated efficiently by further splitting μ^{E} into even and odd subsequences. For $N = 2^p$, this decomposition is repeated p times, leading to an FFT in p stages.

In stage 1, each element μ_k of μ is treated as a sequence $\mu^{(k,1)}$ of length 1. Their Fourier transforms are simply $\lambda_0^{(k,1)} = \mu_0^{(k,1)}$. These sequences are labelled as even/odd, and are combined in pairs using a rule similar to (9), which generates $N/2$ sequences each of length 2. These are denoted as $\lambda^{(k,2)}$ for $k = 0, 1, 2, \ldots, (N/2) - 1$. In stage 2, these new sequences are again labelled as even/odd and combined in pairs using the generalised (9), to obtain $N/4$ sequences of length 4, denoted by $\lambda^{(k,4)}$ for $k = 0, 1, 2, \ldots, (N/4) - 1$. The procedure repeats until stage p ends with a single sequence $\lambda^{(0,2^p)}$ of length 2^p.

The detailed rules that explain how the sequences are combined can be found in the original paper [1] or in standard textbooks such as [2]. These are chosen such that $\lambda^{(0,2^p)} = \lambda$, the vector of interest.

For efficiency, the key point is that each step requires $N/2$ multiplication operations and there are $p = \log_2 N$ stages. Hence the algorithm only requires $(N/2)\log_2 N$ multiplication operations, as advertised above.

References

[1] JW Cooley, and JW Tukey, *An algorithm for the machine calculation of complex Fourier series,* Math. Comput. 19: 297-301 (1965)

[2] TH Cormen, CE Leiserson, RL Rivest, and C Stein, Chapter 30 of *Introduction to Algorithms*, MIT press, 3rd edition (2009)

 © University of Cambridge