Gratings.c: Equations explained

Yuri R. Tonin, Lucas M. Rios, Lucas N. Abrantes

June, 2016

1 Introduction

The Gratings.c file defines 3 different functions: get_initial_intensity(), intensity_after_1st_grating and intensity_after_2nd_grating. These were previously called gp0(), gp1() and gp2(). All the calculations contained in these functions come from the physical theory presented at the paper McMorran and Cronin (2008). The equations will be explained in greater detail in the sections ahead.

2 get_initial_intensity()

The function 'gp0' computes the intensity profile before the beam hits any grating, that is, it computes the beam profile after propagating a certain distance in free space. The equations used in get_initial_intensity() can be found in section **II.A** of McMorran and Croni (2008). More specifically, the loop inside get_initial_intensity() refers to calculating equation (11) in the paper.

3 intensity_after_1st_grating()

Function intensity_after_1st_grating() computes the intensity profile along the z-axis after the beam hits the first grating. To perform this, it uses Fourier components of the beam. These equations are presented in section **II.B** in the paper. Equation (15) is the one being calculated in the 3 nested for loops we find in intensity_after_1st_grating(). The reason for using the 2 inner loops is the fact that equation (15) has a sum over 2 different indexes m' and m.

As explained right after equation (13) in the paper, these indexes are the Fourier components of the complex transmission function that describes the gratings. See appendix A for more details on this.

The outer for loop is responsible for calculating the intensity at different points in the x direction.

In our program, a few changes were made in the equations at intensity_after_1st_grating(). We found a few algebra mistakes that happened when the person translated the

IgorPRO code (from Dr. McMorran's thesis) to the C language. These changes were commited to gitHub and can be found here and here.

4 intensity_after_2nd_grating()

Function intensity_after_2nd_grating() computes the intensity profile after the beam hits the second grating. The calculations can be found at section **II.C** in the paper and are analog to the ones in intensity_after_1st_grating(), but now we have a sum over 4 indexes: a'_m, a_m,b'_n and b_n (see equation 18a in the paper). These indices correspond to variables n1, n2, m1 and m2 in our C code.

The loops compute equation (18a) through (18e) in the paper. The original IgorPRO and the translated C code did not present equations (18a-18e) in the exact same form as in the paper. We tried to rewrite the code equations to match the ones in the paper. By doing these we found a few terms missing in the code equations (see gitHub commit). One troubling thing was the presence in the original code of a term "-2*pi*dn*G2_x/d2" (see line 271 in the old code in the commit mentioned above); we could not find the origin of this term in the papers; we believe this was a term put by hand in the equation since we found the following comment in front of "G2_x" in the Misc.h filw: "50 nm. Initial lateral offset of G2.".

We rewrote the intensity_after_2nd_grating() equations located inside the loop to look exactly the same like the equations in the paper (see gitHub commit). We compiled and ran the code. The output image changed almost nothing from the previous version: visually, you could not tell any differences; however, the values of "mean x", "mean y", "stdev x" and "stdev y" shown in the final plot slightly changed. Therefore, we may conclude that the algebra differences were not that relevant to the final values. We are still going to check this with Dr. McMorran.

Please, see appendix B for a list of things that still need to be understood in the Gratings.c file.

Appendix A Fourier transform

The inverse Fourier transform of $g(t)$ is given by

$$g(t) = \int_{-\infty}^{+\infty} G(f) e^{-i2\pi f t} df \tag{1}$$

that is, to construct the function $g(t)$ we sum over all its frequencies components, where $G(f)$ are called the Fourier components of $g(t)$.

According to our current understanding (we shall confirm that with Dr. McMorran), coefficients a'_m and a_m in equation (13) in the paper are the analog to $G(f)$ in the equation above, while $t(\rho)$ is the analog to $g(t)$ above.

One of the Fourier transforms that seem to appear in the code is the rectangular function

$$g(t - t_0) = A * rect\left(\frac{t - t_0}{T}\right) \tag{2}$$

which is a rectangle of height A and width T centered at t_0. The Fourier components are

$$G(f) = AT\frac{\sin(\pi fT)}{\pi fT}e^{-i2\pi ft_0} \tag{3}$$

Appendix B What we still need to understand

- In the paper, equations (12), (15) and (18a) present coefficients (beam waist parameters) before the big sum. These coefficient were not present in the IgorPRO code, neither in any version of the C code.
- We understand that "ReT" and "ImT" are altered inside of the PhaseShifts files. But we don't understand exatcly what they mean nor why they are used in the "coef" variables in the Granting.c file (see next item).
- The 'coef' variables both, in intensity_after_1st_grating() and intensity_after_2nd_grating(), they use the arrays "ReT" and "ImT". These arrays have 41 elements each. However, inside the for loops, the argument "ReT" and "ImT" only takes values from 15 through 25, i.e., on the 11 central terms of the array are using for computing the intensities. Why?