

One dimensional Scattering

Josh Fernandes
Department of Physics
California State University Long Beach

February 3, 2014

Abstract

The results of a one dimensional scattering equation are plotted.

1 Introduction

This is the first assignment for the class. The purpose of the assignment is to familiarize ourselves with Fortran, Make file, and LaTeX notation. Fortran dates back to 1953, but it has been updated nine times since it's release. This class focuses on Fortran95, but there are more recent versions of Fortran with another update coming in 2015. This paper solves a scattering coefficient equation for several values of Γ . The results are plotted on the same graph, so the effect of Γ on the shape of the curve can be determined.

2 The Fortran95 code

The fortran code solves a one dimensional scattering equation. There is a transmission coefficient that represents a probability of a particle being transmitted. It ranges from 0 to 1. It is centered at some Energy E_0 . The function also depends on a constant denoted by Γ . The formula is

$$T = \frac{\frac{1}{4}\Gamma^2}{(E - E_0)^2 + \frac{1}{4}\Gamma^2}. \quad (1)$$

The Makefile in Listing 1 describes the structure of the code. The codes in `OBJS1` are compiled together. The `.o` files were created from the `.f95` file by using the `F95` command with flags `F95FLAGS`. In building the executable code `LDFLAGS` were used, which contains the `LIB` library. These `LDFLAGS` are optimal for Intel Core2 architecture and uses the Apple's `vecLib` library from Xcode. By typing `make` we can produce the executable file.

Listing 1: Typical Makefile

```

1  OBJS = numtype.o scattering.o
2
3  PROG = scattering
4
5
6  F95 = gfortran
7
8  F95FLAGS = -O3 -funroll-loops -fexternal-blas
9
10 LIBS = -framework vecLib
11
12 LDLFLAGS = $(LIBS)
13
14 all: $(PROG)
15
16 $(PROG): $(OBJS)
17     $(F95) $(LDLFLAGS) -o $@ $(OBJS)
18
19 clean:
20     rm -f $(PROG) *.{o,mod}
21
22 .SUFFIXES: $(SUFFIXES) .f95
23
24 .f95.o:
25     $(F95) $(F95FLAGS) -c $<

```

The Fortran95 code is shown below. The precision is defined in the Module `NumType`. The constants parameters are defined in Module `setup`. The implicit none statement ensures that all the variables has to be defined. `do while-end do` is used to increase the value of `E` while also calculating `T` for the four different Γ values.

Listing 2: Module NumType

```

1
2 module NumType
3
4     save
5     integer, parameter :: dp = kind(1.d0)
6     real(dp), parameter :: pi = 4*atan(1._dp)
7     complex(dp), parameter :: iic = (0._dp,1._dp)
8
9 end module NumType

```

Listing 3: The scattering.f95

```

1
2
3 module setup
4
5     use NumType
6     implicit none
7     real(dp) :: E, dE, E0, gamma(4)
8     REAL, DIMENSION(:, :), ALLOCATABLE :: T
9     integer :: i, j, N, M, DeAllocateStatus,
10     AllocateStatus, Emax
11
12
13 end module setup
14
15 program scattering_path
16
17     use setup
18     implicit none
19
20     gamma = (/ 1.0, 0.5, 0.1, 0.01/)
21
22     E = 0._dp
23     dE = 0.01_dp
24     Emax = 10_dp
25
26     E0 = 5_dp
27     j = 0._dp

```

```

28
29     N = size(gamma)
30     M = (Emax/dE)+1
31
32     ALLOCATE ( T(N, M), STAT = AllocateStatus)
33     IF (AllocateStatus /= 0) STOP
34     "***_Not_enough_memory_***"
35
36     do while (E < Emax)
37         j = j + 1
38         do i=1,4
39             T(i,j) = (.25*(gamma(i)**2))/((E-E0)**2 +
40             .25*(gamma(i)**2))
41         end do
42         print *, E, T(1,j), T(2,j), T(3,j), T(4,j)
43         E = E + DE
44     end do
45
46     DEALLOCATE (T, STAT = DeAllocateStatus)
47 end program scattering_path

```

The code is run by typing `./scattering`. The results are printed on the screen, or the data can redirect to the file 1 by typing `./scattering > 1`. The Plot provides the picture in Fig. 1 and Fig. 2.

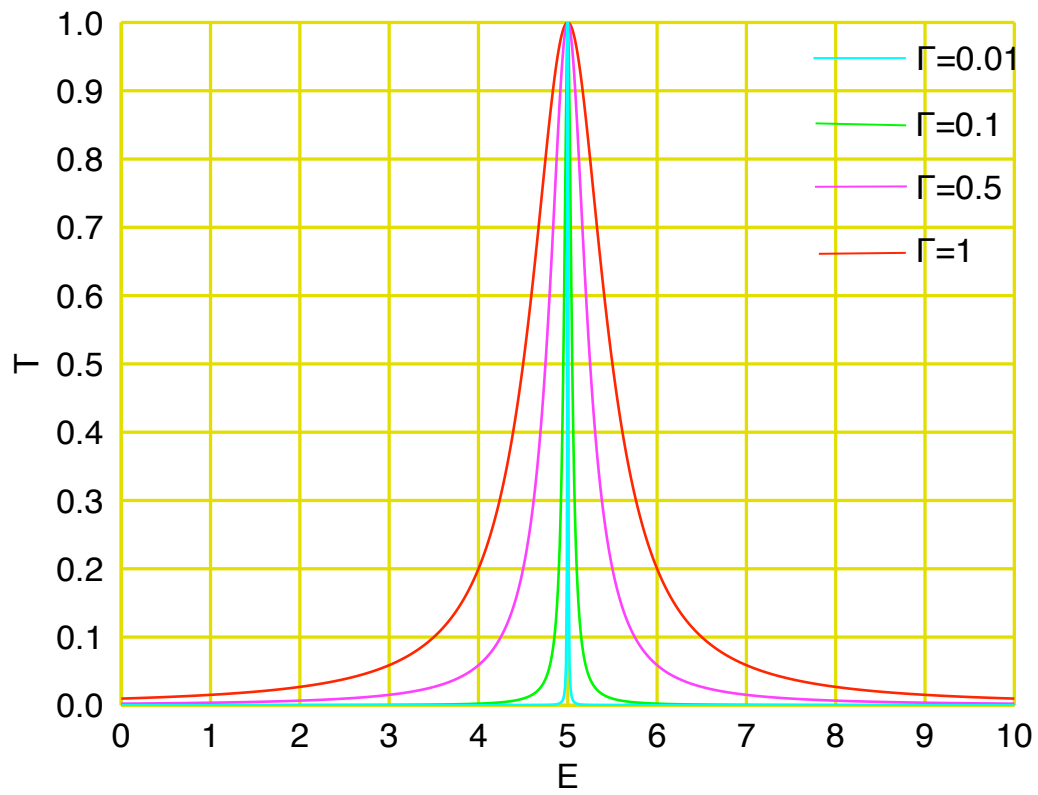


Figure 1: Results of the `scattering.f95` code plotted on a linear scale.

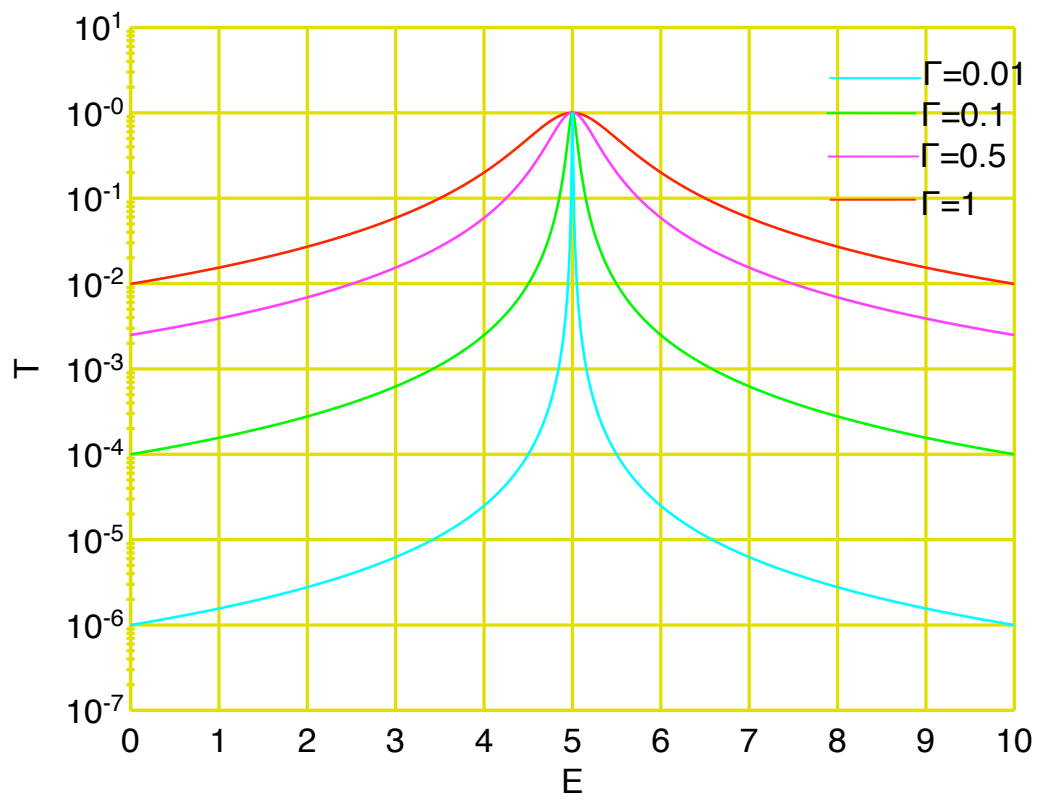


Figure 2: Results of the `scattering.f95` code plotted on a log scale.

3 Summary and conclusions

The first Fortran95 code [1] for the class was written. The transmission coefficient as a function of E with four different Γ values was plotted. The numerical results agree with our prior knowledge and expectations.

References

- [1] M. Metcalf, J. Reid and M. Cohen, *Fortran 95/2003 explained*. Oxford University Press, 2004.