GEF model as a subroutine

Karl-Heinz Schmidt
Last update: December 2016

The stand-alone version of the GEF code simulates the spontaneous or neutron-induced fission of a specific nucleus with the Monte-Carlo method.

More complex reaction schemes may be treated by implementing the GEF model in an external nuclear-reaction code and calling GEF as a subroutine when fission occurs. In parallel to the stand-alone version of the GEF code there is also a subroutine version (in FreeBASIC and in Fortran) available that should be suitable for this purpose. The GEF subroutine uses the folding method and, thus, it is better adapted to deterministic nuclear-reaction codes like TALYS or EMPIRE, which are also based on the folding method. The GEF subroutine is automatically produced from the stand-alone version of the GEF code and, thus, it contains the same physics, and it does not require specific maintenance work.

The subroutine is called with a specific fissioning compound nucleus, its excitation energy and angular momentum. It returns a distribution of fission fragments in A, Z, excitation energy and angular momentum prior to prompt-neutron and prompt-gamma emission. Also the fission mode is given on the output. This way, the de-excitation of the fission fragments can be treated consistently with the other processes by the external nuclear-reaction code.

The GEF subroutine is originally written in FreeBASIC[1]. There is a C backend available (www.freebasic.net), which may help for eventually converting the code from FreeBASIC into C. The Fortran version is produced by automatic conversion with a dedicated translator that keeps the same physics and minimizes the maintenance.

The gross structure of this subroutine is shown below. P_Z_CN, P_A_CN, P_E_EXC, and P_J_CN (Z, A, excitation energy and angular momentum of the compound nucleus) are on input. The result of the GEF model is stored in arrays. These arrays must be made accessible to the external nuclear-reaction code.

Gross structure of the GEFSUB routine in FreeBASIC:

```
/' Declarations '/
…
/' Subroutine GEFSUB '/
Sub GEFSUB(P_Z_CN As Integer, P_A_CN As Integer, P_E_EXC As Single,
   P_J_CN As Single)
….
/' Calculations '/
….
/'
 /' ****************** Begin Arrays for GEF Results ********************** '/
 Dim As Integer N_cases           ' Number of cases in NZMkey, Etab, Jtab, and Ytab
 ReDim NZMkey(10000,3) As Integer  ' Key (Mode,N,Z) for E*, spin and yield distr. of fragments
 ReDim Etab(10000,1000) As Single  ' Excitation-energy distribution of fragments (0.1 MeV bins)
 Redim Jtab(10000,100) As Single   ' Spin distribution of fragments
```

1 Note that FreeBASIC is a compiler that produces binary code with similar performance as other compilers like Fortran or C.

```
 Redim Ytab(10000) As Single       ' Yield of fragments
 /' ****************** End Arrays for GEF Results ********************** '/
End Sub
```

Gross structure of the GEFSUB routine in FORTRAN:

```
      SUBROUTINE GEFSUB(P_Z_CN,P_A_CN,P_E_EXC,P_J_CN)
      IMPLICIT NONE
      INTEGER*4 P_Z_CN
      INTEGER*4 P_A_CN
      REAL*4 P_E_EXC
      REAL*4 P_J_CN
C     /' Input parameters: '/
C     /' Atomic number,mass number,excitation energy/MeV,spin/h_bar of CN '/
C     /' Results are stored in external arrays. '/

            … Calculations

C     /' ****************** Begin Module GEFRESULTS ********************** '/
      INTEGER*4  N_cases
C     ' Number of cases in NZMkey,Etab Jtab, and Ytab
      INTEGER*4 , DIMENSION(10000,3) :: NZMkey
C     ' Key (Mode,N,Z) for E*,spin and yield distr. of fragments
      REAL*4 , DIMENSION(10000,1000) :: Etab
C     ' Excitation-energy distribution of fragments (0.1 MeV bins)
      REAL*4 , DIMENSION(10000,100) :: Jtab
C     ' Spin distribution of fragments
      REAL*4 , DIMENSION(10000) :: Ytab
C     ' Yield of fragments
C     /' ****************** End Module GEFRESULTS ********************** '/

      END
```

The module GEFRESULTS provides the properties of the fragments before emission of prompt neutrons and prompt gammas. The fission fragments with a yield above a given threshold (see below) are characterized by a running number K. The array NZMkey contains the key information on the number N of neutrons (NZMkey(K,1)), the number Z of protons (NZMkey(K,2) and the fission mode M (NZMkey(K,3). The yield of a specific fission fragment (specified by N and Z) is given by the sum over Ytab(K) of the contributions of the different fission modes to the production of this fragment. The distribution of the excitation energy above the yrast line (!) is given separately in the array Etab(K) for each fission fragment (specified by N and Z) and its contribution from a specific fission mode M. The same is true for the angular-momentum distribution. The de-excitation of each fragment, specified by N, Z and M should be calculated separately in nested loops over the angular-momentum distribution, provided by Jtab, and the excitation-energy distribution above the yrast line, provided by Etab.

The total kinetic energy is not explicitly given. In the production of a specific fission-fragment pair,

it is obtained by subtracting the rotational energies and the excitation energies above the yrast line of both fragments from the corresponding Q value. This calculation can be performed with the information provided in GEFRESULTS.

The stand-alone version of GEF is a Monte-Carlo code that keeps the correlations between all observables as given by the model. This feature is lost to a great part in the deterministic subroutine version that only provides the excitation-energy distribution and the spin distribution for each nuclide produced in a specific fission mode. On the other hand, the results of the deterministic code version reach to very low cross sections in one calculation step, while the lowest cross section obtained with the Monte-Carlo method is limited by the statistics and, thus, by the computing time.
(The lowest yield to be provided in the Module GEFRESULTS is defined by the variable YMIN in GEFSUBdcl2.for. The value of YMIN can be modified by the user.)

The ZIP file GEF-Fortran-xxx.zip provides the files that are necessary to prepare the Fortran subroutine. The file GEFSUB.FOR is a stand-alone program that can be compiled with g-fortran under Linux. (Only GEFSUB.FOR must be compiled. All other routines and declaration files are included automatically by the pre-processor. The compiler warnings about some unused variables can be ignored.) For demonstration purposes, in the file GEFSUB.FOR the subroutine GEFSUB is called from a provisional main routine for a sample nucleus (e.g. 236U with an excitation energy of 6 MeV).

The function of the code may be tested by checking the results of the calculations. They are stored in the arrays that are declared between the statements “Begin Module GEFRESULTS” and “End Module GEFRESULTS” in GEFSUBdcl2.FOR.