A Monte Carlo Event Generator for Radiative Capture Reactions

Chris Ruiz - June 2003

1 Intentions

A Monte Carlo event generator for the type of capture reactions investigated using DRAGON is being developed in conjunction with a modification of the present GEANT DRAGON simulation. The GEANT simulation describes the geometry of DRAGON, and tracks particles through the separator. However, the reaction event generation is user defined. At present, the event generator in the GEANT user routine works well only for narrow resonances, since it assumes that the resonant reaction occurs near the centre of the gas target. In general, this is not the case, since for broad and continous reaction cross-sections, reactions may occurs over the entire length of the target. It was my intention to develop an event generator which more closely maps the properties of the distribution of reactions through the target when cross-sections other than narrow resonances are involved. This could eventually be implemented in GEANT instead of the existing user routine *or*, it could be used as a consistency check to test any new algorithm inserted into the current GEANT user routine. However, the intention was also to have a more immediate Monte Carlo program which may help explain the type of recoil event distributions and gamma-ray distributions we see in the data from the ^{13}C(p,γ)^{14}N experiment, for example.

2 Methodology

2.1 Gas Target

The gas target density distribution may not be constant over the length of the target, and will extend outside the actual target box. To start with, I have used the model of a gas density which is nearly constant over the internal region of the target, and has Gaussian edges, so that the density drops to half its maximum value at the entrance and exit collimators to the target box. This is achieved by convoluting a gaussian with a top-hat function over the distance of the target (z-direction) (see Figure 1.a). I arbitrarily chose the spread of the Gaussian used to be 10% of the target length to begin with.

2.2 Distribution of centre-of-mass energies

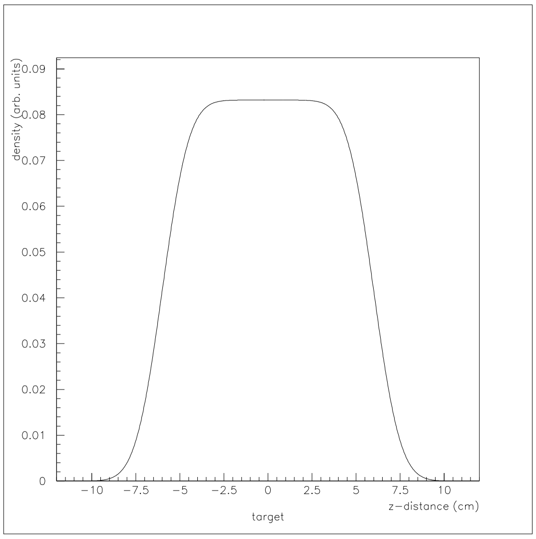
Consider a monoenergetic beam (no width) entering the gas target with energy E_i. If no energy loss straggling occured, the beam would exit the target with some energy E_o. A mapping can be made between energy and z, and assuming the energy loss is linear through the target, we can set the mapping so that E_i occurs at the entrance fwhm of the target (where z = -targetlength/2) and that E_o occurs at the rear fwhm of the target (where z = +targetlength/2). In reality, the beam has a spread, and energy loss straggling occurs through the target. The beam is modelled as a Gaussian, with spread roughly 0.2%. To get the distribution of centre-of-mass energies, I convolute the beam gaussian through the target distribution (which has been transformed from a function of z to a function of mean centre-of-mass energy at that z), this is shown in Figure 1.b. One could increase the spread of the beam Gaussian as it runs through the target to simulate the straggling, but as this is small for the current case I am running, I have neglected this.

2.3 Cross-section and Yield

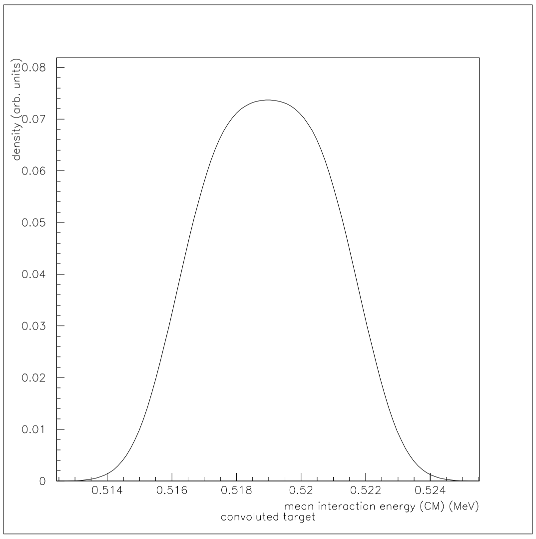
The previous method gives us the distribution of centre-of-mass energies. The total reaction cross-section is a function of energy, so the direct multiplication of the cross-section by the energy distribution gives us the relative reaction probability as a function of energy (which I call the *Yield Function*). Figures 1.c and 1.d show the cross-section for the $^{13}C(p,\gamma)^{14}N$ reaction and the corresponding Yield Function respectively.

2.4 Reaction Depth

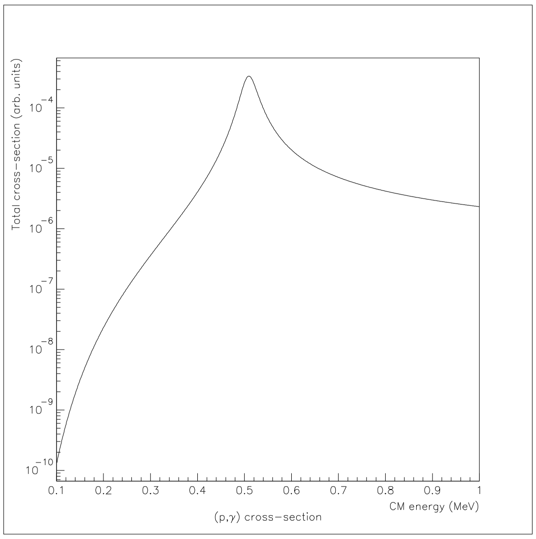
Although z can be transformed into centre-of-mass energy, this is only the mean centre-of-mass energy. The reverse mapping would then say that for a given centre-of-mass energy, the z-value is the mean z at which that energy occurs within the target. We also have range straggling, so that a given energy occurs over a range of z-positions, with some spread, related to the energy straggling.



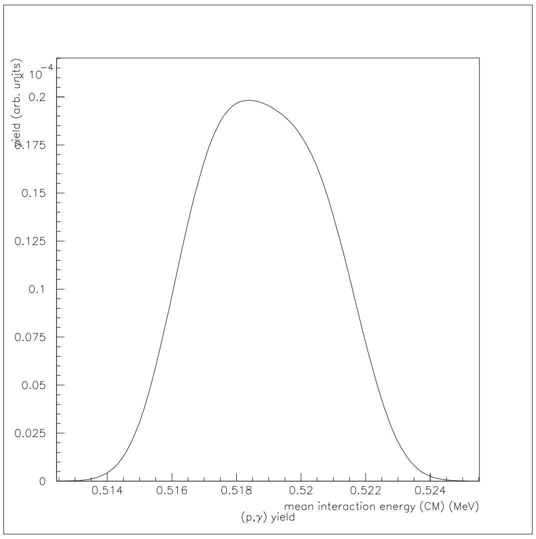
(a)



(b)



(c)



(d)

Figure 1: Target density distribution, cross-section, and related functions used in the Monte Carlo

2.5 Generation of Events

The Yield Function described above can be used to generate events weighted to this distribution, and generate a depth in the target corresponding to this, with range straggling taken into account. This is the basic event generation method in the program, and subsequently the kinematics and decays are taken into account. I will now proceed with a description of the program.

3 The Program

3.1 Initialisation

The program reads in from input files describing the physical properties of the target and projectile, such as mass and charge. It also reads in the beam energy, the beam spread, and the mean energy loss through the target, as well as the cross-section properties such as resonance energy and width. A separate file reads in the recoil properties, such as the level scheme, branching ratios, spin-parities etc. HBOOK functions are created to be used later in the program, eg, such as the Yield Function and detector resolution.

The branching ratios are turned into limits to be used in the choice of the decay route. For each initial state, labelled by index i, a variable is defined describing the fractional probability that it decays to a final state j, and obviously the sum of these variables over all possible final states equals 1.

3.2 Monte Carlo

It is here that the Monte Carlo really begins. The beam energy is near the resonance energy, so the initial state formed is the state corresponding to the resonance (this doesn't do direct capture yet, where we would be between resonances). A random reaction energy is chosen from the Yield Function using the HBOOK routine HRNDM1. The z-value is then calculated (and a weighted random added for the straggling). Random numbers are generated and a decay route picked from them, giving the energies of each gamma-ray in the decay process.

A random emission direction is chosen for each gamma-ray by choosing a random azimuthal angle between $\phi = [0, 2\pi]$, and a random cosine of the polar angle between -1 and 1 (if you simply choose a random θ then more events get bunched up at the forward and backward directions due to the projection into three dimensions, whereas the above method gives a true random direction). A subroutine adds up the gamma-ray vectors in three dimensions, giving a resultant vector which is used to calculate the recoil "kick".

All events are then transformed into the laboratory coordinate system using the relatavistic Lorentz equations. This then gives new gamma-ray energies (which are Doppler-shifted) and new recoil energies (which are confined to a cone in the beam direction). The vector directions of each event are stored.

The collection of variables calculated during each event in the program is stored in an ntuple file which can then be read by PAW++.

I will now illustrate the capabilities of the program using a specific example.

Note that the simulation does **not** include: Beam straggling; beam multiple scattering; recoil energy loss out of the target; recoil multiple scattering out of the target; initial non-straight beam direction. Several of these effects will result in a smearing of the recoil spectrum.

3.3 Simulation of ^{13}C(p,γ)^{14}N

The 511 keV resonance in ^{13}C(p,γ)^{14}N populates a state in ^{14}N at 8.062 MeV which decays primarily to the ground state. The resonance is $\ell = 0$, so the angular distribution of the decay is isotropic. All subsequent decays are assumed isotropic (correlations not added).

Figure 2.a shows a 3-dimensional scatter plot of the resultant gamma-ray decay vectors in the centre-of-mass frame, projected onto a unit sphere in the CM. This shows the isotropic nature of the decays. Figure 2.b shows the relative intensities of the decays in the CM frame. The solid line are the ground state decays; the dashed lines are the decays from a two-step cascade; and the dotted lines are the decays from a three-step cascade. Figure 2.c shows these lines once transformed into the lab, and therefore Doppler shifted.

Figure 2.d shows the distribution of recoil cone angles. The solid line is the ground state distribution, the dashed line the two-step cascade and the dotted line the three-step cascade. It can be seen that for the ground state decay, most events lie on the outside of the cone, which is an effect of projecting the isotropic CM sphere distribution into the lab via the Lorentz boost, ie. purely geometrical and kinematic. The cascade distributions are

smeared out, as the kick the recoil recieves is always on average less than the maximum kick they could recieve, due to the random nature of the isotropic vector addition in the CM frame. It can be seen that the maximum cone angle is around 20 mrad.

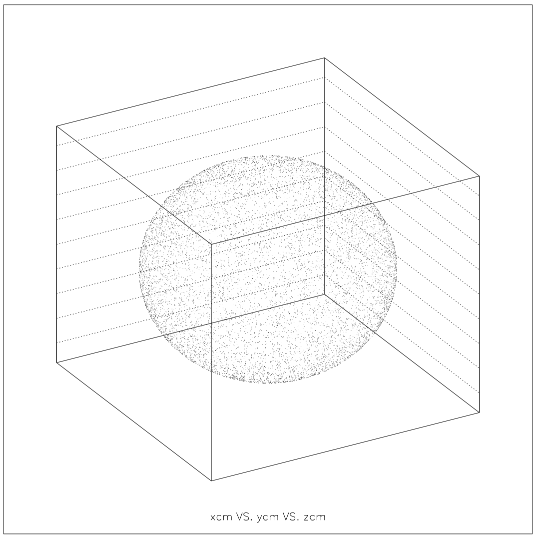
Figure 3.a shows the detected recoil spectrum, where a cut at 19 mrad has been made (all events with cone angles greater than 19 mrad are rejected). The spectrum prior to the cut was a single peak, so it is events at the mean energy which are cut out by cone angle acceptance (ones that lie on the outside of the cone and are therefore associated with 90° decays in the CM system). An important observation is that cone angle acceptance on its own cannot result in a large asymmetrical deficiency in the recoil spectrum. Referring the Figure 2.d, it can also be understood that the recoil cone angle cut also removes mostly ground state decays.

Figure 3.b shows the recoil energy spectrum from a simulation of the $^{12}C(\alpha,\gamma)^{16}O$ reaction, where the relevant resonance has a quadrupole-type gamma-ray angular distribution, resulting in two peaks. Using the same arguments as above, it is hard to see how a large asymmetric cut could arise in such a spectrum, as acceptance cuts will in general be asymmetric.

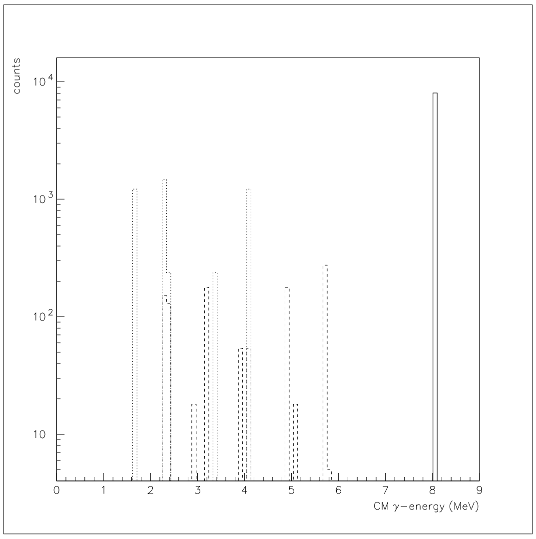
4 Conclusions

The simulation detailed here, although incomplete in many ways, is a useful indicator of expected recoil and gamma-ray event distributions for any radiative capture reaction. This simulation will be used to compare with GEANT simulations during the development stage of a new event generator within the existing GEANT simulation framework.

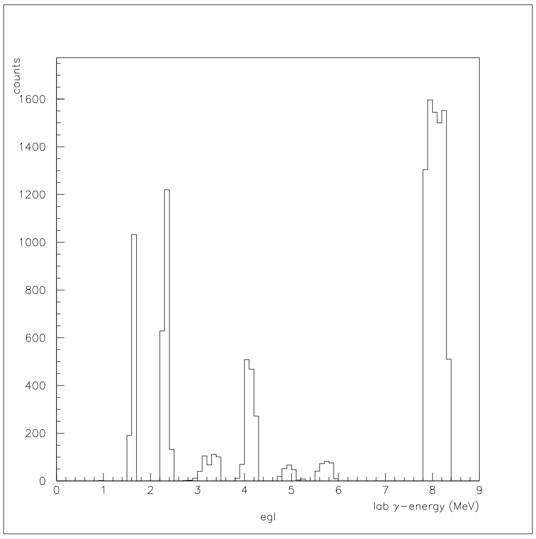
Because of the booking of the simulation data into ntuples, a powerful teaching tool is available, enabling anyone to plot any kinematic variable against any other, with any amount of logical conditions, eg: plotting the recoil energy versus the recoil angle from either the cascades or ground state decays. It is hoped that the eventual full GEANT simulation will be able to use ntuples in the same way.



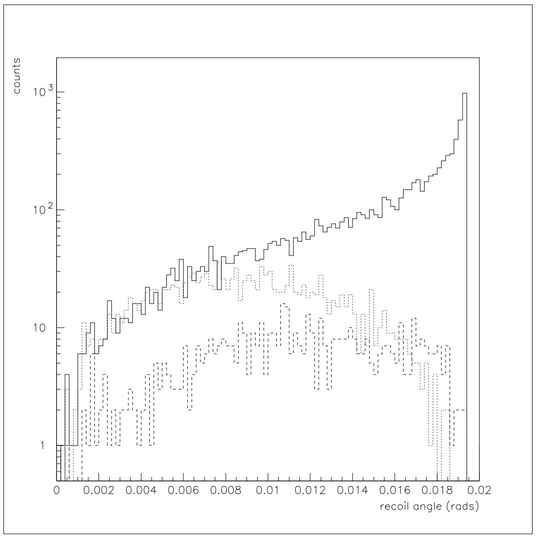
(a) γ-ray event distribution in CM frame



(b) Relative decay intensities in CM frame

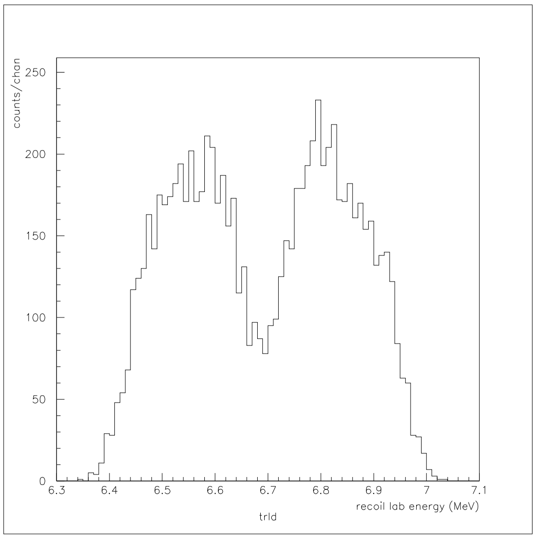


(c) Resulting Doppler-shifted γ-energies in lab frame

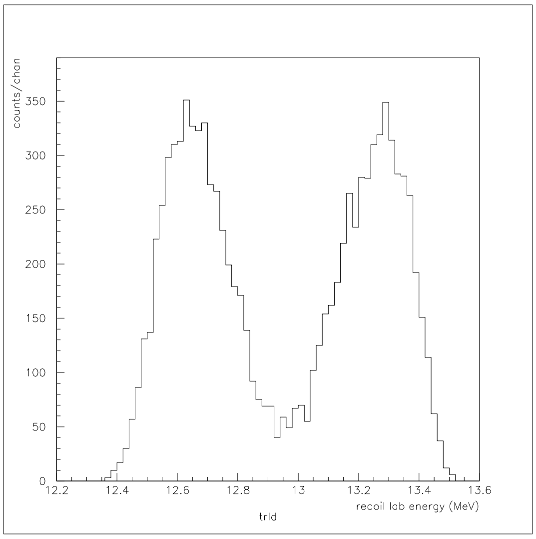


(d) Distibution of recoil cone angles for ground state and cascade decays

Figure 2:



(a) ^{13}C(p,γ)^{14}N recoil energy spectrum with 19 mrad cone angle cut



(b) ^{12}C$(\alpha,\gamma)^{16}$O recoil energy spectrum, with m=0 quadrupole angular distribution

Figure 3:

4.1 Aside: Recoil Energy Distribution

The Lorentz boost transformation for a recoil particle is given by

$$E_R = \gamma\{E_R^* + V p_\gamma^* \cos\theta_\gamma^*\}$$

where E_R is the total lab recoil relatavistic energy, γ is the relatavistic gamma-factor, E_R^* is the total CM recoil relatavistic energy, V is the velocity of the CM system, p_γ^* is the CM momentum of the gamma-ray (or resultant addition of cascade momenta), and θ_γ^* is the gamma-ray emission angle in the CM frame.

The only angular term shows that the distribution of recoil energies maps the distribution of gamma-ray decay cosines directly, before experimental smearing effects are taken into account.