

Chapter 1

Simulation Pipeline

In this chapter the simulation of a hyperfine spectrum measured at TRIUMF is described in detail. §1 includes little more than a graphical representation of the algorithm followed. Each subsequent section is dedicated to an individual step in the algorithm, detailing the logic and computations required at that step.

1.1 Complete Pipeline

Fig. 1 shows each step taken an atom as it passes through the simulation. The red rounded rectangles represent preliminary steps in the simulation, where initial properties are imparted to the atom. The first preliminary step selects a velocity and ground state for the atom. The following step is the calculation of the likelihood of excitation for each allowed transition. This prepares the atom for what can be considered the main loop of the algorithm. Here, the atom and laser interact as the atom traverses the IR.

The main loop is composed of green ellipses and blue rectangles. Two of the three green ellipses represent regular checks to ensure that the atom is still in the IR. If at any point the atom is found to have moved beyond the IR, then the algorithm moves on to the next atom. The third green ellipse is dedicated to checking if a photon released by the atom would be measured

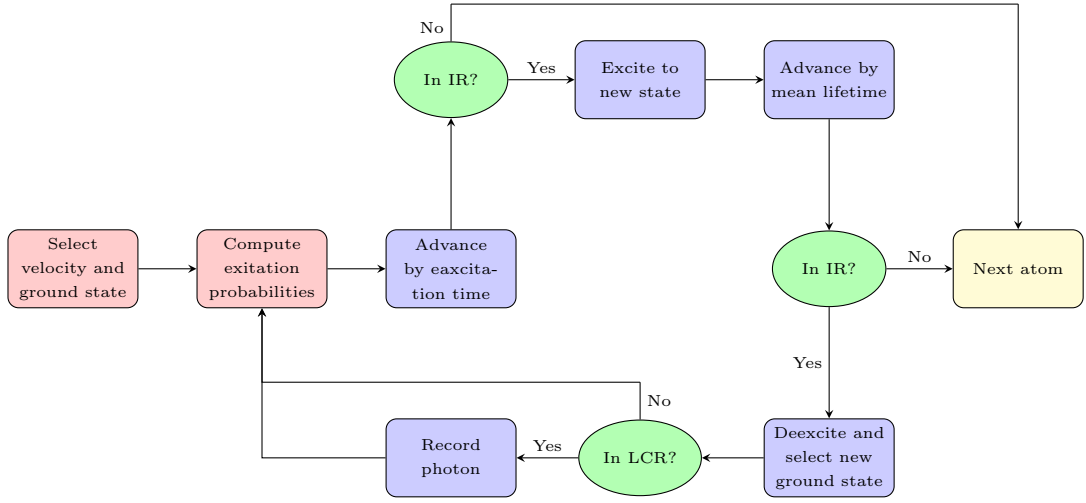


Figure 1.1: Flow chart detailing the steps undertaken by the algorithm to follow the atom as it moves through the IR. The green ellipses and blue rectangles compose the main loop of the simulation, where the atom interacts with the laser and goes through excitation/decay cycles. Finally, the yellow rectangle is the end point of the loop, and occurs when then atom has moved beyond the IR.

by the PMT. The evolution of the state and position of the atom is described by the blue rectangles.

1.2 Preliminary Steps

The first two steps select the initial properties given to the atom, determining how likely the atom is to interact with the laser. The two most important properties are the velocity and the ground state of the atom. Both are chosen through the sampling of their respective probability distributions.

1.2.1 Velocity Selection

The velocity of the atom v_a can be decomposed in to two elements: the mean velocity of the beam, v_m , and the thermal velocity v_T .

$$v_a = v_m + v_T \quad (1.1)$$

The mean velocity is determined by the energy of the beam and the mass of the atom, i.e.

$$v_m = \sqrt{\frac{2E_b}{m_A}} \quad (1.2)$$

where E_b is the energy of the beam and m_A is the mass of an atom with mass number A .

The thermal velocity is selected by sampling the Maxwell-Boltzmann distribution, described in (INSERT REFERENCE TO THEORY SECTION). This is done using the Box-Muller transform, which samples a uniform distribution twice and converts the results into a sample of a normal distribution. A sample v_T , of a Maxwell-Boltzmann distribution for atoms of mass m_A and temperature T is given by

$$v_T = \sqrt{\frac{KT}{2m_A}} \sqrt{-2 \log x_1} \cos(2\pi x_2) \quad (1.3)$$

where $x_1, x_2 \in [0, 1]$ are two randomly generated numbers taken from a uniform distribution.

1.2.2 Ground State Selection

For an atom with a coupled angular momentum state F_g as the ground state, the electron can occupy any of the $2F_g + 1$ projections on the axis of quantization. A particular projection, say $F_i \in [-F_g, -F_g + 1, \dots, F_g - 1, F_g]$, has a probability of being occupied that is proportional to $2F_i + 1$. Taking the probability space occupied by each projection in to account, the probability that an electron occupies the projection F_i , $P(F_i)$, is given by

$$P(F_i) = \frac{2F_i + 1}{\sum_j 2F_j + 1} \quad (1.4)$$

Knowing this, the ground state can be chosen through the generation of a random number $x \in [0, 1]$ from a uniform distribution. Each projection is given a range of numbers from 0 to 1, proportional to Eq. 1.4. If x falls in the assigned range, then that ground state is chosen.

1.2.3 Computation of Excitation Probabilities

With the ground state and the velocity of the atom now selected, the probability that a transition will occur