Decay rate calculations of three dimensional metastable states for the undergraduate classroom

P.B. Siegel

California State Polytechnic University Pomona
Pomona, CA 91768
pbsiegel@cpp.edu

July 28, 2025

Abstract

The treatment of Alpha decay using the Gamov-Gurney-Condon method is a staple in introductory modern physics courses. The quantum mechanical tunneling probability times the number of wall collisions per second yields a satisfactory comparison with nuclear alpha decay half-life (or decay rate) data and was one of the earliest successful applications of quantum physics. The next question for the students to consider is how to improve this approximate result. In this article we present a method to calculate the decay rate λ (or half-life τ) for a metastable state from the Schrödinger equation without knowledge of scattering theory. Comparison is made with the Gamov formula for a simple barrier potential which has an analytic solution, and for U^{219} using a numerical method. The calculations are appropriate for the undergraduate Physics curriculum and would fit well into a Computational Physics course.

1 Introduction

The treatment of alpha decay using the method of Gamov[2] and Gurney-Condon[3], which we refer to as Gamov's method, is an important application of quantum mechanical tunneling in introductory modern physics courses [1]. However, Gamov's method is an approximation and the students might wonder how one could calculate the decay rate for alpha decay directly from the Schrödinger equation. In this note, we present a method for calculating the decay rate of a quasi-bound particle "trapped" with angular momentum $l = 0$ inside a spherically symmetric potential well.

In most classroom applications, the kinetic energy of the decaying particle will be small compared to it's rest mass and it will be sufficient to use the non-relativistic Schrödinger equation to analyze the decay. For a symmetric three dimensional potential $V(r)$, where the particle has zero angular momentum, $l = 0$, the particle's wave function $\psi(r)$ can be written as $\psi(r) = u(r)/r$ where $u(r)$ satisfies

$$-\frac{\hbar^2}{2m}\frac{d^2u(r)}{dr^2} + V(r)u(r) = Eu(r) \tag{1}$$

with the boundary condition $u(0) = 0$.

A quasi-bound particle with energy $E > 0$ will decay and be a free particle, since $V(r) \to 0$ as $r \to \infty$. Without significant loss of accuracy, one can set $V(r) = 0$ outside a large distance R from the origin where the particle is not bound. Thus, for $r > R$ the particles wave function will be that of a free particle, which we parameterize as $u(r) = A(E)sin(kr + \delta)$ for $r > R$. Here $E > 0$ is the particle's kinetic energy and $k = \frac{\sqrt{2mE}}{\hbar}$ for a particle with mass m.

For real energy $E > 0$, $A(E)^2 \neq 0$ and the wave function is not normalizable. Our approach to determine the quasi-bound state's energy e_r and width Γ is the following. We start with an energy E below the quasi-bound energy and solve the Schrödinger equation for $A(E)$. Then, we increase E by a small amount, and each time we solve for $A(E)$. $A(E)$ will keep decreasing, a minimum will be reached at e_r, then $A(E)$ will start to increase. The function $|A(E)|^2$ will have an inverse Breit-Wigner (or Lorenzian) form about e_r:

$$|A(E)|^2 \approx |C|^2[(E - e_r)^2 + (\frac{\Gamma}{2})^2] \tag{2}$$

We fit $|A(E)|^2$ with this function to find e_r and Γ. The decay rate λ is related to resonance width Γ via $\lambda = \frac{\Gamma}{\hbar}$. The half-life τ of the quasi-bound particle is $\tau = \frac{ln(2)}{\lambda}$.

Our treatment here is somewhat different than the methods in undergraduate physics texts and in Physics Education Journals. In Physics Education journals, the authors mainly treat one-dimensional metastable states and calculate the transmission T or reflection coefficient R. The transmission coefficient, $T(E)$, will have a Briet-Wigner form about the energy of the metastable state. See Ref. [4] and cited references within for a nice treatment of the one dimensional case. Here we treat the three-dimensional metastable state and the student does not need any knowledge of scattering theory.

2 Theory

For real energy E, $A(E) \neq 0$ and the wave function is not normalizable. However, if we let the energy become complex (e), then there can be an energy where the amplitude $A(e) = 0$. One can determine the complex energy e at which $A(e) = 0$ by calculating the amplitude A for energies E on the real energy axis. Suppose $A(e)$ is a complex analytic function and has a zero at an energy $e_0 = e_r - e_i i$ in the complex plane. For nearly all classroom examples the energy width is very small compared to the resonant energy, i.e. $\frac{e_i}{e_r} <<< 1$. One can expand $A(e)$ about this zero point in a Taylor series:

$$\begin{aligned} A(e) &= 0 + A'(e_0) * (e - E_0) + \cdots \\ &= C(e - (e_r - e_i i)) + \cdots \end{aligned}$$

where C is the complex derivative of A at the energy e_0. The magnitude of $|A|^2$ for energies E on the real energy axis is approximately

$$|A(E)|^2 \approx |C|^2[(E - e_r)^2 + e_i^2] \tag{3}$$

We calculate $A(E)^2$ starting at an energy below the first quasi-bound resonance and increase E until $A(E)^2$ is minimized. Then, using very small intervals ΔE, we determine $A(E + n\Delta E)$ for integer values of n between -50 and $+50$: $-50 < n < +50$. The complex energy parameters e_r and e_i are determined by a fit to the 100 values of $A(E + n\Delta E)$ using gnuplot.

To relate e_i to Γ, we use the property that at the energy $e_r - e_i i$, the amplitude $A(e)$ is zero for $r > R$ and the complete wave function for the three dimensional case is

$$\begin{aligned} u(r,t) &= u(r)e^{-i\frac{e}{\hbar}t} \\ &= u(r)e^{-i\frac{(e_r - e_i i)}{\hbar}t} \\ &= u(r)e^{-i\frac{e_r}{\hbar}t}e^{-\frac{e_i}{\hbar}t} \end{aligned}$$

The absolute value of $u(r,t)$ squared is

$$u(r,t)u^*(r,t) = u(r)^* u(r)e^{-2\frac{e_i}{\hbar}t} \tag{4}$$

Thus, the quasi-bound state decays away with a decay constant $\lambda = 2\frac{e_i}{\hbar}$. The decay width $\Gamma = \hbar\lambda = 2e_i$. That is, $e_i = \frac{\Gamma}{2}$. The half-life of the decay, $\tau_{\frac{1}{2}}$, is given by $\tau_{\frac{1}{2}} = \frac{ln2}{\Gamma}$.

We also note that the phase shift δ is the $l = 0$ scattering phase shift, δ_0, for alpha-nucleus scattering. At the resonance energy, $\delta_0 = \frac{\pi}{2}$, and the energy dependence of δ_0 is given by $\delta_0(E) = tan^{-1}(\frac{\Gamma/2}{e_r - E})$. One could calculate $\delta_0(E)$ and fit this function with $tan^{-1}(\frac{\Gamma/2}{e_r - E})$ to determine e_r and Γ as done in Ref.[5]. However, we find it easier for undergraduate students to calculate and fit $A(E)$, since they do not need to have covered scattering theory.

3 Examples

We present two examples that would be appropriate for undergraduate students. The first is a simple piecewise step potential that has an analytic solution for $u(r)$. The second is a more realistic potential that is a fair approximation to the potential that an alpha particle experiences in and outside a nucleus. In this case, the Schrödinger Equation is solved numerically to determine $u(r)$.

In each case, the method we use to solve for e_r and Γ is the following. We choose a starting energy E below e_r and solve for $A(E)$. We choose an initial energy increment ΔE and solve for $A(E + \Delta E)$. Then, the energy is increased repeatedly by ΔE and the amplitude A is determined. At first $A(E + i\Delta E))$ will decrease. When $A(E + i\Delta E)$ starts increasing, the energy

increment ΔE is replaced by $-\frac{\Delta E}{2}$ and the process is repeated. After many iterations, an energy E_0 is reached which will be very close to e_r.

To get an energy step size about which to calculate $A(E)$ near E_0, we first determine the Gamov width, *gwidth*, which is defined as the tunneling probability times the number of wall collisions, or hit rate, per second. The computer program then produces a file that lists $A(E)$ for values E_i equal to $E_i = E_0 + i * gwidth$ for integer values of i between -50 to $+50$: $-50 < i < +50$. Finally, to determine e_r and Γ, a gnuplot program is run that fits the data $A(E)$ with the inverse Lorenzian function..

3.1 Piece-wise Constant Potential

The following potential is a simple example for which $u(r)$ can be determined analytically. There are three regions where the potential is constant:

$$\begin{aligned} V_I(r) &= 0 \qquad for \qquad 0 < r < a \\ V_{II}(r) &= V_0 \qquad for \qquad a \leq r \leq b \\ V_{III}(r) &= 0 \qquad for \qquad r > b \end{aligned}$$

Note that in region I the wave function $u_I(0) = 0$ since $u(r) = r\psi(r)$ and $\psi(r)$ is finite at the origin. This is equivalent to having an infinite barrier at $r = 0$, i.e. $V_I(0) = \infty$. Therefore the wave function $u_I(r)$ is proportional to $sin(kr)$ in region I, and we take the proportionality constant to be one. Since the potential is constant in each region, the Schrödinger equation is easy to solve with solutions:

$$\begin{aligned} u_I(r) &= sin(kr) \\ u_{II}(r) &= Be^{-k'r} + Ce^{k'r} \\ u_{III}(r) &= A\ sin(kr + \delta) \end{aligned}$$

where $k = \frac{\sqrt{2mE}}{\hbar}$ and $k' = \frac{\sqrt{2m(V_0 - E)}}{\hbar}$.

At $r = a$ and $r = b$ the wave function and it's derivative must be continuous. These requirments will yield four equations to solve for the four parameters. We are only interested in the parameter A.

At $r = a$ we have:

$$\begin{aligned} sin(ka) &= Be^{-k'a} + Ce^{k'a} \\ k\ cos(ka) &= k'\ (-Be^{-k'a} + Ce^{k'a}) \end{aligned}$$

and at $r = b$ the continuity conditions are

$$\begin{aligned} A\ sin(kb+\delta) &= Be^{-k'b} + Ce^{k'b} \\ Ak\ cos(kb+\delta) &= k'\ (-Be^{-k'b} + Ce^{k'b}) \end{aligned}$$

After some algebra one obtains:

$$\begin{aligned} A\ sin(kb+\delta) &= \frac{e^{k'(a-b)}}{2}(sin(ka) - \frac{k}{k'}cos(ka)) \\ &\quad + \frac{e^{k'(b-a)}}{2}(sin(ka) + \frac{k}{k'}cos(ka)) \\ &\quad and \\ A\ cos(kb+\delta) &= \frac{e^{k'(a-b)}}{2}(cos(ka) - \frac{k'}{k}sin(ka)) \\ &\quad + \frac{e^{k'(b-a)}}{2}(\frac{k'}{k}sin(ka) + cos(ka)) \end{aligned}$$

One can determine $|A|^2$ by squaring and adding the right sides of the above equations, since $(A\ sin(kb+\delta))^2 + (A\ cos(kb+\delta))^2 = A^2$.

We have written a computer program that computes $|A|^2$ as a function of the energy E of the quasi-bound particle. The program is listed and explained in Appendix I. The $|A(E)|^2$ are fit to an inverse Lorenzian function by the gnuplot program "barrierfit.p", which is listed in Appendix II. A graph of the amplitude squared, $|A(E)|^2$, versus energy around the resonance energy e_r is shown in Fig. 1. The horizontal axis is in units of $(gwidth)/10$. The resulting resonance width is $\Gamma = 2.3 * (gwidth) \approx 9.2 \times 10^{-7}\ MeV$.

Comparing the value using the Schrödinger Eq. with that of the approximate Gamov tunneling calculation, we see that the Gamov calculation is only off by a factor of 2.3. The ratio $\frac{\Gamma}{e_0} \approx 4 \times 10^{-7}$, which is easily computed using long double precision.

It is interesting to fix V_0 and a and increase the tunnel length b to see when long double precision fails. We increased b from 10 fm to 18 fm and

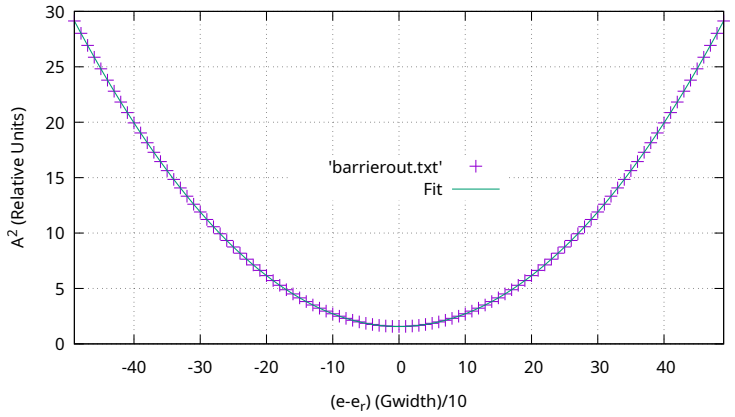

Figure 1: A graph of A^2 versus energy. The parameters of the barrier potential are: $a = 4\ fm$, $b = 10\ fm$ and $V_0 = 10\ MeV$. The data are fit to the function $C((e - e_r)^2 + (\frac{\Gamma}{2})^2)$. The energy of the metastable state is $e_r \approx 2.2817286\cdots\ MeV$ with a width of $\Gamma \approx 9.2 \times 10^{-7}\ MeV$, resulting in $\frac{\Gamma}{e_r} \approx 4 \times 10^{-7}$.

the quasi-bound energy e_0 was unchanged to within 8 significant figures. The width Γ decreased down to $10^{-15}\ MeV$, but remained equal to $2.4 * (gwidth)$. At $b = 19\ fm$ the calculation started to become inaccurate, i.e. when $\frac{\Gamma}{e_0} < 10^{-15}$.

3.2 Alpha Decay Potential

The potential of the previous section was simple and enabled a calculation of the amplitude $A(E)$ without using numerical methods. In this section we determine how the Gamov tunneling calculation compares to the calculation using the Schrödinger equation for the potential felt by an alpha particle trapped in a heavy nucleus. In a heavy nucleus, neutrons and protons can cluster into an alpha particle. The formed alpha will be attracted to the other neutrons and protons in the nucleus via the strong interaction, but repelled by the other protons in the nucleus due to the electrical force. The size of the nucleus we call c. We choose the potential for the alpha to be the sum of the electric potential of a uniformly charged sphere of radius c, regions I and II, plus a simple attractive square well potential, region I ($r \leq c$), of depth V_0 for the strong nuclear interaction:

$$
\begin{aligned}
V_I(r) &= 2(Z-2)q^2\frac{3c^2-r^2}{2c^3} - V_0 \qquad for \qquad 0 < r \leq c \\
V_{II}(r) &= \frac{2(Z-2)q^2}{r} \qquad for \qquad c < r \leq R \\
V_{III}(r) &= 0 \qquad for \qquad r > R
\end{aligned}
$$

The potential is set to zero for $r > R$, where R is very large and outside the "tunnel region". For $r > R$, the potential is zero, and the wave function $u_{III}(r)$ equals $A\ sin(kr + \delta)$. Here Z represents the number of protons in the nucleus, q is the electron's charge, c is the radius of the nucleus, and V_0 is the strength of the attractive strong potential. The above form for the alpha-nucleus potential is similar to the one used in Ref. [5]. Instead of a Woods-Saxon potential for the strong nuclear interaction we have chosen a simple attractive square well.

There is no analytic solution for this potential, so numerical methods must be used. The half-life for alpha decay can be very long, and hence the ratio $\frac{\Gamma}{e_r}$ can be very small. In a 64 bit compiler, a floating point number in long double precision has 17 significant decimal digits. Therefore to examine the amplitude as a function of energy near a metastable resonance the ratio $\frac{\Gamma}{e_r}$ needs to be greater than around 10^{-17}. For the alpha potential, a larger value of e_r will yield a larger decay width. Thus, for a numerical calculation it is best to try alpha decays that have a large energy. The largest energies for alpha decays are around 10 MeV, so decays with a width Γ of greater than $10^{-16} MeV$ should be feasible. A resonance width of 10^{-16} MeV corresponds to a half-life of around 4.5 μs. Searching the uranium isotopes, we see that U^{219} is a good candidate to consider. U^{219} has an energy of 9.94 MeV above the ground state and a half-life of 42 μs [6].

We have used the Euler algorithm, a method taught in undergraduate Computational Physics courses [7], to determine $u(r)$ for the three regions. If students want to calculate longer lived uranium isotopes, one can use quadruple precision which represents floats with 128 bits and therefore $\frac{\Gamma}{e_0}$ can be as small as 10^{-34}. The challenge one has is to determine $|A(E)|^2$ around the resonance energy, e_r, for values of $E \approx e_r(1 \pm 10^{-17})$. In Appendix III, we list and discuss the c program we used.

We show the results for U^{219} for three different values of V_0 in Figs. 2-4. In each case we have chosen the nuclear radius to be $c = 1.3A^{1/3}$ and the horizontal axis is in units of $gwidth/10$. In Fig. 2, the potential is

$V_0 = 111.0\ MeV$. For this value of V_0, the first $l = 0$ metastable state has an energy of $e_r = 11.27\ MeV$. The width is 1.74 $gwidth \approx 7.22 \times 10^{-14}\ MeV$. The ratio $\frac{\Gamma}{e_r} \approx 6.4 \times 10^{-15}$ and the graph is very smooth.

In Fig. 3, the potential strength V_0 is 112 MeV. Here the first metastable state occurs at $e_r = 10.32\ MeV$ and a width of 0.84 $gwidth \approx 4.41 \times 10^{-16}\ MeV$. The ratio $\frac{\Gamma}{e_r} \approx 4.3 \times 10^{-17}$ and the fit is fair. The data however are starting to deviate from a smooth curve.

Finally, in Fig. 4 the potential V_0 is set to 112.4 MeV. The first metastable state occurs at $e_r = 9.94\ MeV$, which agrees with the experimental value. The width Γ equals 1.9 $gwidth \approx 1.43 \times 10^{-16}\ MeV$. The ratio $\frac{\Gamma}{e_r} \approx 1.4 \times 10^{-17}$ and the fit is barely acceptable.

A decay width of $\Gamma = 1.43 \times 10^{-16}\ MeV$ results in a half-life of $\tau = \frac{ln2\,\hbar}{\Gamma} \approx 3.2\mu s$. This calculated value compares favorably with the experimental value for the half-life of U^{219} which is 42 μs [6]. The experimental value for τ should be longer than the calculated one done here, since we have assumed that an alpha particle has already been formed. There is a preformation factor, P, which needs to be considered. P represents the fraction of the time an alpha is formed in a nucleus in the metastable state [8]. In our case $P \approx \frac{3.2}{42} \approx 0.07$. From the literature, the preformation factor can range from 0.2 to 0.02.

4 Summary

We have presented a method, appropriate for the undergraduate classroom, for determining the energy and decay rate of a particle in a three dimensional, $l = 0$, metastable state. The method involves the calculation of the amplitude of the wave function, $u(r)$ at large r, as a function of real energy near the energy of the metastable state. The method can be used without knowledge of scattering theory. Two examples were presented, one with an analytic solution and one with a numerical solution. The half-life example of U^{219} gives students the experience in analyzing data with a model that requires numerical methods which are covered at the undergraduate level.

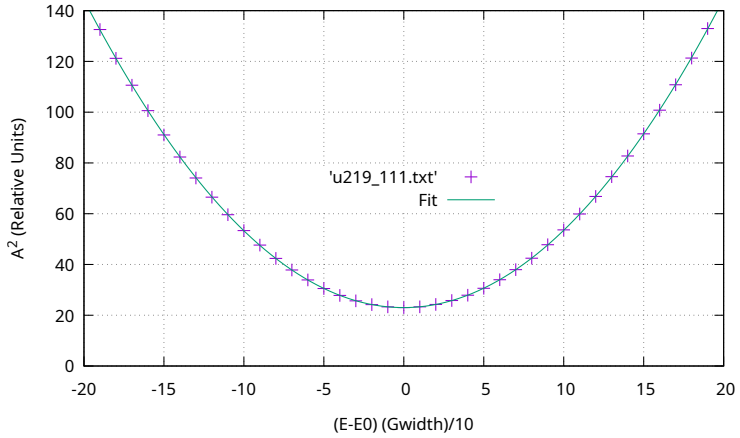

Figure 2: A graph of A^2 versus energy for U^{219}. The parameters of the square well potential are: $c = 1.3\ fm$ and $V_0 = 111\ MeV$. The data are fit to the function $C((e - e_r)^2 + (\frac{\Gamma}{2})^2)$. The energy of the metastable state is $e_r \approx 11.2\ MeV$ with a width of $\Gamma \approx 7.2 \times 10^{-14}\ MeV$, resulting in $\frac{\Gamma}{e_r} \approx 6.4 \times 10^{-15}$.

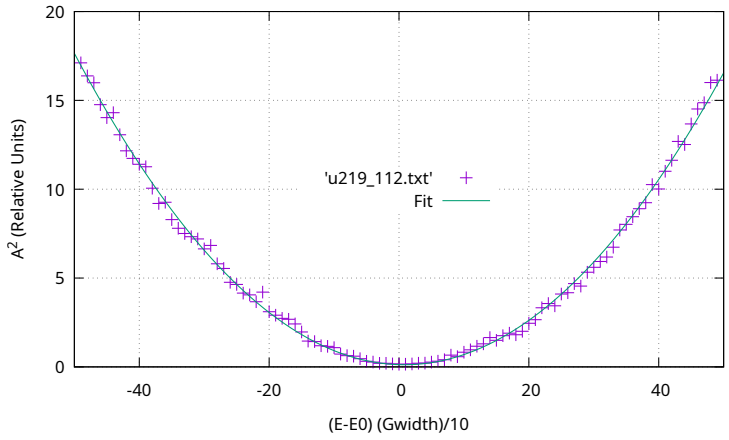

Figure 3: A graph of A^2 versus energy for U^{219}. The parameters of the square well potential are: $c = 1.3\ fm$ and $V_0 = 112\ MeV$. The data are fit to the function $C((e - e_r)^2 + (\frac{\Gamma}{2})^2)$. The energy of the metastable state is $e_r \approx 10.3\ MeV$ with a width of $\Gamma \approx 4.4 \times 10^{-16}\ MeV$, resulting in $\frac{\Gamma}{e_r} \approx 4.3 \times 10^{-17}$.

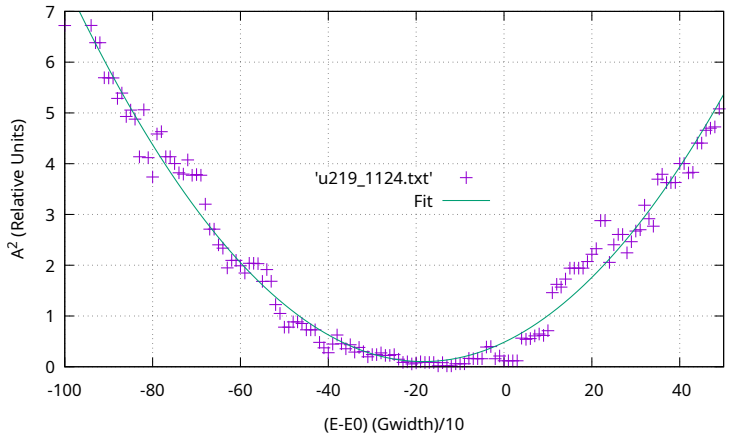

Figure 4: A graph of A^2 versus energy for U^{219}. The parameters of the square well potential are: $c = 1.3\ fm$ and $V_0 = 112.4\ MeV$. The data are fit to the function $C((e - e_r)^2 + (\frac{\Gamma}{2})^2)$. The energy of the metastable state is $e_r \approx 9.94\ MeV$ with a width of $\Gamma \approx 1.43 \times 10^{-16}\ MeV$, resulting in $\frac{\Gamma}{e_r} \approx 1.4 \times 10^{-17}$ and a half-life of $3.2\mu\ sec$.

References

[1] Robert Eisberg and Robert Resnick, Quantum Physics of Atoms, Molecules, Solids, Nuclei, and Particles, 2nd Edition, 1985, John Willey and Sons.

[2] Gamow, G. Zur Quantentheorie des Atomkernes. Z. Physik 51, 204–212 (1928).

[3] Wave Mechanics and Radioactive Disintegration, R. GURNEY and E. CONDON, Nature volume 122, page 439 (1928).

[4] John Eric Goff and Don C. Colladay,"Metastable states in a 1D Quantum system", Am. J. Phys. **91**, 993-998 (2023), and References in this article.

[5] D. Duarte and P.B. Siegel, "A Potential Model for Alpha Decay", Am. J. Phys. **78**, 949-953 (2010).

[6] Brookhaven National Lab nuclear data center: www.nndc.bnl.gov/nudat3.

[7] Harvey Gould and Jan Tobochnik, An Introduction to Computer Simulation Methods: Applications to Physical Systems, 3rd Edition, 2006, Gould and Toba, Pearson Education.

[8] S.M.S. Ahmed, R. Yahaya, S. Radiman and M.S. Yasir, "Alpha-Cluster Preformation Factors in Alpha-decay for Even-Even Heavy Nuclei using the Cluster-Formation Model", J. Phys. G: Nucl. Part. Phys. **40**, (2013).

5 Appendix I

Below, we list the c program that we used to produce a file of $A(E)$ to be analysed by a gnuplot fitting program.

We used values of $a = 4\ fm$, $b = 10\ fm$ and $V_0 = 10\ MeV$ for an alpha particle, mass $mc^2 = 3727\ MeV$. As a starting energy we used $E = 0.1\ MeV$, and a starting energy increment $\Delta E = 0.03\ MeV$. After many iterations, the amplitude $A(E)$ has a minimum at $e_0 \approx 2.2817286\cdots\ MeV$. The gamov decay rate is defined as the tunneling probability $tprob = e^{-2k'(b-a)}$ times the "hitrate" $hrate = \sqrt{2e/m}/(2a)$. The resonance width from the Gamov approximation $gwidth = (tprob * hrate * \hbar)$ is determined. Then, a file "barrierout.txt" is produced which lists $A(E)$ for energies $E = e_0 + i\ gwidth/10$ for values of $-50 \leq i \leq +50$.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <math.h>

int i;
long double a, b, e, e0, dele, v0, k, kp;
long double mc2, hc, amp2, psi1, psi1p;
long double amp2scale, test, tprob, hfreq, decaywidth, halflife;
FILE *out1;

int main()
{
```

```
 out1=fopen("barrierout.txt","w");
 a=4.0;
 b=10.0;
 mc2=3727.0;
 hc=197.33;
 e=hc*(2.0*3.14/a)*(2.0*3.14/a)/8.0/mc2;
 dele=0.01;
 v0=10;
//Find Amp at starting energy e for test value
 k=sqrtl(2.0*mc2*e/hc/hc);
 kp=sqrtl(2.0*mc2*(v0-e)/hc/hc);
 psi1=expl(kp*(a-b))*(sin(k*a)-(k/kp)*cos(k*a));
 psi1=psi1+expl(kp*(b-a))*(sin(k*a)+(k/kp)*cos(k*a));
 psi1p=expl(kp*(a-b))*(cos(k*a)-(kp/k)*sin(k*a));
 psi1p=psi1p+expl(kp*(b-a))*((kp/k)*sin(k*a)+cos(k*a));
 test=psi1*psi1+psi1p*psi1p;
//Find the quasi-bound state energy
 for(i=1;i<1000;i++)
 {
 e=e+dele;
 k=sqrtl(2.0*mc2*e/hc/hc);
 kp=sqrtl(2.0*mc2*(v0-e)/hc/hc);
 psi1=expl(kp*(a-b))*(sin(k*a)-(k/kp)*cos(k*a));
 psi1=psi1+expl(kp*(b-a))*(sin(k*a)+(k/kp)*cos(k*a));
 psi1p=expl(kp*(a-b))*(cos(k*a)-(kp/k)*sin(k*a));
 psi1p=psi1p+expl(kp*(b-a))*((kp/k)*sin(k*a)+cos(k*a));
 amp2=psi1*psi1+psi1p*psi1p;
 if(amp2>test) dele=-dele/2.0;
 test=amp2;
 }

 e0=e;
 hfreq=sqrt(2.0*e/mc2)*hc/2.0/a;
 tprob=expl(-2.0*kp*(b-a));
 decaywidth=hfreq*tprob;
 printf("e0 = %.10Lf k = %Lf  kp = %Lf  dele = %Le\n",e0, k, kp, dele);
 printf("Tunnel Prob =   %.10Le hfreq = %Lf   decaywidth = %.10Le \n",tprob, hfreq
 halflife=log(2.0)*(6.58*pow(10,-22))/decaywidth;
```

```
 printf("halflife = %.10Le sec \n",halflife);
//Write the data file barrierout.txt for gnuplot to use
 for(i=-50;i<50;i++)
  {
   e=e0+i*decaywidth/10.0;
   k=sqrtl(2.0*mc2*e/hc/hc);
   kp=sqrtl(2.0*mc2*(v0-e)/hc/hc);
   psi1=expl(kp*(a-b))*(sin(k*a)-(k/kp)*cos(k*a));
   psi1=psi1+expl(kp*(b-a))*(sin(k*a)+(k/kp)*cos(k*a));
   psi1p=expl(kp*(a-b))*(cos(k*a)-(kp/k)*sin(k*a));
   psi1p=psi1p+expl(kp*(b-a))*((kp/k)*sin(k*a)+cos(k*a));
   amp2=psi1*psi1+psi1p*psi1p;
   amp2scale=amp2*exp(14);
   fprintf(out1,"%d  %.8Lf\n",i,amp2scale);
  }
 fclose(out1);
 return(0);
}
```

6 Appendix II

The following is a program for gnuplot that fits the data in the file barrierout.txt

```
set term pdf
set out 'barrierfitout.pdf'

set grid
set key center
set xlabel '(e-e_r) (Gwidth)/10 MeV'
set ylabel 'A^2 (Relative Units)'
set xrange [-49:49]
set yrange [0.0:30.0]
```

```
e=0.1
del=10
f(x) = c*((x-e)*(x-e)+del*del/2.0/2.0)
fit f(x) "barrierout.txt" u 1:2 via c, e, del
plot 'barrierout.txt', f(x) t 'Fit'
```

7 Appendix III

The following is a c program that produces a file of $|A|^2$ versus E for the alpha potential.

Subroutine **ecalc**

For an alpha of energy e, the "tunnel region" is from the nuclear radius $r = c$ to $rmax = \frac{2(Z-2)q^2}{e}$. We choose for the nuclear radius a value $c = 1.3A^{1/3}\ fm$, which for U^{219} is 7.83 fm. For an energy of $e = 10.1\ MeV$, $rmax = 25.59\ fm$. We first obtain an accurate value for e_r by choosing a value for r near the end of the "tunnel region", $r = 20\ fm$, and searching for the energy where $psi(r = 20\ fm) = 0$.

To obtain values for $|A|^2$ in region III, we chose $R = rmax + 20 \approx 45\ fm$. So, for values of $r > 45\ fm$, the potential $V_{III}(r) = 0$, and $psi(r) = A\ sin(kr + \delta)$. We calculate $|A|^2(e)$ at indices $rcal1$ and $rcal2$. The energy is increased until a minimum is found for $|A|^2$, which gives a very accurate value for e_r.

Subroutine **gfac**

This subroutine calculates the Gamov width $gwidth$.

Subroutine **psicalc**

This subroutine produces the file ampout.txt which lists the values of $|A|^2$ for energies near the resonance energy e_r.

```
//Program for calculating Alpha decay using a simple square model

#include <stdio.h>
#include <math.h>
#include <stdlib.h>

//Defne constants
```

```
long double pi = 3.14159265, hc = 197.33, fsc = 1.44, mc2 = 3727.0,aa,bb;
long double psi[10000],r[10000],rmax,delr,sum,sqrint,xrint,e0,trans,hfreq;
long double del,fac,e,test0,test1,dele,z,c,rmaxint,rminint,xk;
long double r1,r2,gwidth,e0,v0,r0,anuc,phase;
int i,n,j,icheck,imax,ical1,ical2;
FILE *out1;
char fileout[20];

void ecalc();
void gfac();
void psicalc();

void ecalc()
{
   imax=2000;
   e=2.0;
   psi[0] = 0.0;
   psi[1] = 1.0;
   del = 0.01;
//First try to get a starting test0
    for (i=1; i<imax; i++)
    {
     r[i] = i*del;
     fac = (z-2.0)*2.0*fsc/r[i];
     if (r[i]<c) fac=2.0*fsc*(z-2.0)*(3.0/c-r[i]*r[i]/c/c/c)/2.0+v0;
     psi[i+1] = 2*psi[i] - psi[i-1] - del*del*psi[i]*2*mc2/hc/hc*(e-fac);
    }
  test0=psi[imax-1];
//Loop to get estimate e0
dele=1.0;
for (j=1; j<200; j++)
  {
   e = e+dele;
   psi[0] = 0.0;
   psi[1] = 1.0;
    for (i=1; i<imax; i++)
    {
     r[i] = i*del;
```

```
     fac = (z-2.0)*2.0*fsc/r[i];
     if (r[i]<c) fac=2.0*fsc*(z-2.0)*(3.0/c-r[i]*r[i]/c/c/c)/2.0+v0;
     psi[i+1] = 2*psi[i] - psi[i-1] - del*del*psi[i]*2*mc2/hc/hc*(e-fac);
    }
  test1=psi[imax-1];
   if (test1*test0<0) dele = -dele/2.0;
  test0 = test1;
   }
e0=e;
printf("Quasi-bound energy = %25.24Lf  +/-  dele = %Le \n", e0, dele);
printf("psi[imax] = %25.24Lf   psi[imax-1]  = %25.24Lf \n", psi[imax-1],psi[imax-2
//Fine search for accurate value of e0
 rmax=2.0*(z-2.0)*fsc/e;
 del=0.01;
 gfac();
 dele=gwidth*100.0;
 icheck=floor(rmax/del)+2000;
 imax=icheck+400;
 test0=0.0;
 printf("imax = %d    icheck = %d \n",imax,icheck);
  for (j=1;j<200;j++)
  {
   e=e+dele;
   psi[0] = 0.0;
   psi[1] = 1.0;
    for (i=1; i<imax; i++)
    {
     r[i] = i*del;
     fac = (z-2.0)*2.0*fsc/r[i];
     if (r[i]<c) fac=2.0*fsc*(z-2.0)*(3.0/c-r[i]*r[i]/c/c/c)/2.0+v0;
     if (i>icheck) fac=0.0;
     psi[i+1] = 2*psi[i] - psi[i-1] - del*del*psi[i]*2*mc2/hc/hc*(e-fac);
    }
  xk=sqrtl(e*2.0*mc2/hc/hc);
  ical1=icheck+100;
  ical2=icheck+110;
  r1=ical1*del;
  r2=ical2*del;
```

```
  aa=psi[ical1]*cos(xk*r2)-psi[ical2]*cos(xk*r1);
  bb=psi[ical1]*sin(xk*r2)-psi[ical2]*sin(xk*r1);
  test1=(aa*aa+bb*bb);
  if(test1>test0) dele=-dele/2;
  test0=test1;
 }
 e0=e;
printf("Quasi-bound energy = %25.24Lf  +/-  dele = %Le \n", e0, dele);
return;
}

void gfac()
{
 sum=0.0;
 rminint=c;
 rmaxint=(z-2.0)*2.0*fsc/e;
 printf("rmin= %Le   rmax= %Le \n",rminint,rmaxint);
 delr=(rmaxint-rminint)/10000.0;
 for (j=0;j<10000;j++)
 {
  xrint=rminint+j*delr;
  sqrint = (z-2.0)*2.0*fsc/xrint-e;
  if (sqrint<0)
   {
    sqrint=0;
   }
  sum=sum+sqrtl(sqrint*2.0*mc2/hc/hc);
 }
 sum=sum*2.0*delr;
 trans = exp(-sum);
 hfreq=sqrt(2.0*(e-v0)/mc2)*hc/(2.0*c);
 gwidth=trans*hfreq;
 printf("e = %Le  Trans Prob =  %Le  freq = %Le  gwidth = %Le \n", e, trans, hfreq
}

// Calculate A^2 as a function of energy near resonance energy
void psicalc()
 {
```

```
   rmax=2.0*(z-2.0)*fsc/e;
   del=0.01;
   icheck=floor(rmax/del)+2000;
   imax=icheck+400;
   printf("imax = %d    icheck = %d \n",imax,icheck);
   dele=gwidth/10;
  for (j=-50;j<50;j++)
  {
   e=e0+j*dele;
   psi[0] = 0.0;
   psi[1] = 1.0;
    for (i=1; i<imax; i++)
    {
     r[i] = i*del;
     fac = (z-2.0)*2.0*fsc/r[i];
     if (r[i]<c) fac=2.0*fsc*(z-2.0)*(3.0/c-r[i]*r[i]/c/c/c)/2.0+v0;
     if (i>icheck) fac=0.0;
     psi[i+1] = 2*psi[i] - psi[i-1] - del*del*psi[i]*2*mc2/hc/hc*(e-fac);
    }
  xk=sqrtl(e*2.0*mc2/hc/hc);
  ical1=icheck+100;
  ical2=icheck+110;
  r1=ical1*del;
  r2=ical2*del;
  aa=psi[ical1]*cos(xk*r2)-psi[ical2]*cos(xk*r1);
  bb=psi[ical1]*sin(xk*r2)-psi[ical2]*sin(xk*r1);
  phase=atanl(aa/bb);
  test0=(aa*aa+bb*bb)*expl(32);
//print data to file ampout.txt for gnuplot
  fprintf(out1,"%d   %25.24Lf   \n",j, test0);
  }
 }

//Main Loop
int main()
{
 out1=fopen("ampout.txt","w");
//I used r0=1.3 and A=219
```

```
 printf("Input the value of r0 and A:   ");
 scanf("%Lf %Lf", &r0, &anuc);
 c=r0*pow(anuc,0.3333);
 printf("The radius is %Lf \n",c);
//I used z=92 for U
 printf("Input the value of z:   ");
 scanf("%Lf", &z);
//I used -111 to -112.4
 printf("Input the value of V0:   ");
 scanf("%Lf", &v0);

 del=0.01;
 dele = 0.1;
 ecalc();
 printf("The energy is %25.24Lf  \n",e);
 gfac();
 psicalc();
 return (1);
}
```