

2013

The Nucleon-Core Interaction: a Nuclear Physics Simulation Suitable for Classroom Use

Benjamin E. Hogan

Javier E. Hasbun
jhasbun@westga.edu

Follow this and additional works at: <https://digitalcommons.gaacademy.org/gjs>

 Part of the [Physical Sciences and Mathematics Commons](#)

Recommended Citation

Hogan, Benjamin E. and Hasbun, Javier E. (2013) "The Nucleon-Core Interaction: a Nuclear Physics Simulation Suitable for Classroom Use," *Georgia Journal of Science*, Vol. 71, No. 2, Article 5.

Available at: <https://digitalcommons.gaacademy.org/gjs/vol71/iss2/5>

This Research Articles is brought to you for free and open access by Digital Commons @ the Georgia Academy of Science. It has been accepted for inclusion in Georgia Journal of Science by an authorized editor of Digital Commons @ the Georgia Academy of Science.

THE NUCLEON-CORE INTERACTION: A NUCLEAR PHYSICS SIMULATION SUITABLE FOR CLASSROOM USE

Benjamin E. Hogan and Javier E. Hasbun*
Department of Physics
University of West Georgia
Carrollton, GA 30118

*Corresponding Author
E-mail: jhasbun@westga.edu

ABSTRACT

The orbits of a nucleon and its respective parent nucleus about their common center of mass are simulated in an effort to provide a pedagogical approach to the understanding of the structure of atomic nuclei. The nuclear exercise is treated by solving the problem with an effective force on the system's reduced mass. The potential governing the mean field is modeled by the Woods-Saxon form-factor with parameters that enable it to describe experimental findings. The Woods-Saxon potential is preferred over the infinite well and harmonic oscillator methods because both require infinite separation energies of the nucleons. The simulation is created using Easy Java Simulations (EJS) which is part of the Open Source Physics project and a MATLAB version (compatible with Octave) is included in the Appendix.

Key Words: Woods-Saxon Potential, Nucleon-Core Binary System, Shell Model, Easy Java Simulations, MATLAB, Octave

INTRODUCTION

Advancements in computing techniques have made it possible to bring about improvements in the classroom pedagogical approaches. This is particularly true in nuclear physics. The idea that nucleons interact through a nuclear force that is many times stronger and more complicated than the electromagnetic force makes it difficult to carry out simulations of their motion. It is possible, however, that with the availability of a potential that has been painstakingly investigated, and which is a good fit to nuclear interactions, such concepts can be employed in the classroom and simulations can be developed for classroom use.

Here we consider the motion of a single nucleon interacting with the core of a nucleus and develop an approach to perform the simulation of the motion while simplifying the process for classroom accessibility purposes. In studying the shell model of an atomic nucleus, the fundamental assumptions are that the motion of a single nucleon is due to the potential caused by all the other nucleons and that the individual nucleons can orbit as if they were

transparent to each other, *i.e.*, collisions do not occur [2]. If we apply these two properties to the nucleus, a two-body problem emerges, which can be expressed as a one-body problem via the relative coordinate reduced mass concept often studied in classical mechanics [4]. Once knowledge of the relative coordinate versus time is obtained, it is possible to have access to information regarding each individual body's position versus time.

The most commonly used shell model potential in modern day nuclear physics is known as the "Woods-Saxon potential." [1] This potential, parameterized by experimental data, provides the most realistic fit to nuclei across the chart of the nuclides [1]. Using this potential and the concept of reduced mass, we derive the equations of motion. We then solve the resulting equations numerically with MATLAB's ode45 nonstiff differential equation solver. This is a fourth and fifth order Runge-Kutta method which is the first recommended method to try when solving numerical differential equations. We have thus developed a simulation of the nucleon and its parent nucleus orbiting their common center of mass.

The Binary System and the Concept of Reduced Mass

In this section, for pedagogical reasons, we review relative coordinate, reduced mass approach and show our implementation scheme. We consider a nucleus composed of A nucleons; that is, Z protons and $A-Z$ neutrons. One of the nucleons, with mass m_2 , is removed leaving the parent nucleus with $A-1$ nucleons and mass m_1 . We then consider these two particles interacting through a nuclear force. The force itself will later be shown to have origins in a potential of a Woods-Saxon form [2]. The nuclear force is an attractive force which is exerted by each mass onto the other. Considering Figure 1, the coordinate of m_1 is r_1 while the coordinate of m_2 is r_2 .

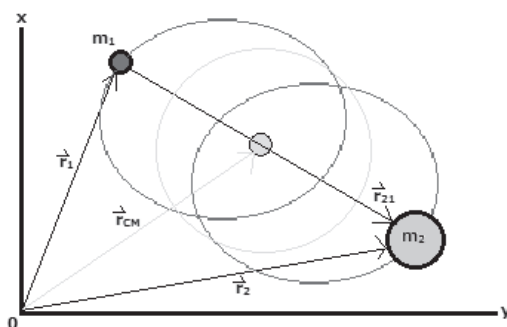


Figure 1. The binary system consisting of two masses, m_1 and m_2 , orbiting about a common center of mass located at r_{CM} .

The force on m_2 due to m_1 is f_{21} , and the force on m_1 due to m_2 is f_{12} . Therefore the force on each mass is, according to Newton's 2nd law is,

$$\mathbf{f}_{12} = m_1 \frac{d^2 \mathbf{r}_1}{dt^2} \quad (1)$$

$$\mathbf{f}_{21} = m_2 \frac{d^2 \mathbf{r}_2}{dt^2} \quad (2)$$

In the binary system, the forces f_{12} and f_{21} point along the vector connecting m_1 and m_2 and, because of Newton's 3rd law, we have that $f_{12} = -f_{21}$, where

$$\mathbf{f}_{12} = f_{12} \hat{\mathbf{r}}_{12} = f_{12} \frac{\mathbf{r}_{12}}{r_{12}} \quad (3)$$

$$\mathbf{f}_{21} = f_{21} \hat{\mathbf{r}}_{21} = f_{21} \frac{\mathbf{r}_{21}}{r_{21}}, \quad (4)$$

and where we have used the definitions

$$\mathbf{r}_{21} = \mathbf{r}_2 - \mathbf{r}_1 \quad (5)$$

$$\mathbf{r}_{12} = \mathbf{r}_1 - \mathbf{r}_2 \quad (6)$$

We see that $r_{12} = -r_{21}$ and $r_{12} = r_{21}$, therefore

$$\mathbf{f}_{12} = -f_{21} \frac{\mathbf{r}_{21}}{r_{21}} \quad (7)$$

Equation (7) permits us to rearrange Equations (3) and (4) as

$$\frac{\mathbf{r}_{21}}{r_{21}} = -\frac{\mathbf{f}_{12}}{f_{12}} = \frac{\mathbf{f}_{21}}{f_{21}} \quad (8)$$

which is consistent with Newton's 3rd law and showing that for the magnitudes $f_{12} = f_{21}$. Since the magnitudes of the forces are equal, we will refer to them as a single central force f later.

Combining Equations (1) and (2) with Equations (3) and (4) gives

$$m_1 \frac{d^2 \mathbf{r}_1}{dt^2} = \frac{f_{12}}{r_{12}} \mathbf{r}_{12} \quad (9)$$

and

$$m_2 \frac{d^2 \mathbf{r}_2}{dt^2} = \frac{f_{21}}{r_{21}} \mathbf{r}_{21}. \quad (10)$$

The center of mass for a system of two particles is

$$\mathbf{r}_{CM} = \frac{m_1 \mathbf{r}_1 + m_2 \mathbf{r}_2}{M}, \quad (11)$$

where $M = m_1 + m_2$; *i.e.*, the total mass. We also define the relative coordinate variable as

$$\mathbf{r} \equiv \mathbf{r}_{21} = \mathbf{r}_2 - \mathbf{r}_1. \quad (12)$$

With the above definitions we can write Equations (11) and (12) in matrix form as

$$\begin{pmatrix} \mathbf{r}_{CM} \\ \mathbf{r} \end{pmatrix} = \begin{pmatrix} m_1/M & m_2/M \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{pmatrix}, \quad (13)$$

which with the help of the inverse of the 2x2 matrix,

$$\begin{pmatrix} m_1/M & m_2/M \\ -1 & 1 \end{pmatrix}^{-1} = \frac{M}{m_1 + m_2} \begin{pmatrix} 1 & -m_2/M \\ 1 & m_1/M \end{pmatrix}$$

allows us to express r_1 and r_2 in terms of r_{CM} and r as

$$\begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = \begin{pmatrix} 1 & -m_2/M \\ 1 & m_1/M \end{pmatrix} \begin{pmatrix} r_{CM} \\ r \end{pmatrix}, \quad (14)$$

or

$$r_1 = r_{CM} - \frac{m_2}{M} r \quad (15)$$

and

$$r_2 = r_{CM} + \frac{m_1}{M} r. \quad (16)$$

We next multiply Equation (9) by m_2 and Equation (10) by m_1 to get

$$m_1 m_2 \frac{d^2 r_1}{dt^2} = m_2 \frac{f_{12}}{r_{12}} r_{12}, \quad (17)$$

$$m_1 m_2 \frac{d^2 r_2}{dt^2} = m_1 \frac{f_{21}}{r_{21}} r_{21}. \quad (18)$$

We then subtract Equation (17) from Equation (18) and obtain

$$m_1 m_2 \frac{d^2}{dt^2} (r_2 - r_1) = m_1 \frac{f_{21}}{r_{21}} r_{21} - m_2 \frac{f_{12}}{r_{12}} r_{12}.$$

(19)

Using the fact that $r = r_2 - r_1 = r_{21} = -r_{12}$ so that $r_{21} = r_{12} = r$ and that the magnitudes $f_{21} = f_{12} = f$, we can rewrite Equation (19) as

$$m_1 m_2 \frac{d^2 r}{dt^2} = \frac{f}{r} r (m_1 + m_2). \quad (20)$$

or

$$\mu \frac{d^2 r}{dt^2} = f \hat{r} \quad (21)$$

where we have used $r = r\hat{r}$ and the definition for the reduced mass

$$\frac{m_1 m_2}{m_1 + m_2} = \mu. \quad (22)$$

Solving the single Equation (21) for the motion of the reduced mass μ with relative coordinate r is equivalent to solving Equations (9) and (10) separately for the motion of m_1 and m_2 with respective coordinates r_1 and r_2 . This is so because, given that the center of mass of the system, once r is known we can use Equations (15) and (16) to obtain the separate coordinates.

In order to develop a program code for classroom simulation purposes, we will work with the following two-dimensional Cartesian coordinate system:

$$\mathbf{r} = x\hat{i} + y\hat{j}, \quad (23)$$

$$r = \sqrt{x^2 + y^2}, \quad (24)$$

$$\hat{r} = \frac{\mathbf{r}}{r} = \frac{x}{\sqrt{x^2 + y^2}} \hat{i} + \frac{y}{\sqrt{x^2 + y^2}} \hat{j}; \quad (25)$$

so that Equation (21) becomes

$$\frac{d^2}{dt^2} (x\hat{i} + y\hat{j}) = \frac{f}{\mu} \left(\frac{x}{\sqrt{x^2 + y^2}} \hat{i} + \frac{y}{\sqrt{x^2 + y^2}} \hat{j} \right), \quad (26)$$

or for each coordinate

$$\frac{d^2 x}{dt^2} = \frac{f}{\mu} \frac{x}{\sqrt{x^2 + y^2}}, \quad (27)$$

$$\frac{d^2 y}{dt^2} = \frac{f}{\mu} \frac{y}{\sqrt{x^2 + y^2}} \quad (28)$$

We will be seeking the numerical solutions for $x(t)$ and $y(t)$ in these differential equations. From these, with $r_{CM} = x_{CM}\hat{i} + y_{CM}\hat{j}$, we will obtain the individual coordinates of Equations (15) and (16) as follows.

$$x_1 = x_{CM} - \frac{m_2}{m_1 + m_2} x \quad (29)$$

$$y_1 = y_{CM} - \frac{m_2}{m_1 + m_2} y \quad (30)$$

$$x_2 = x_{CM} + \frac{m_1}{m_1 + m_2} x \quad (31)$$

$$y_2 = y_{CM} + \frac{m_1}{m_1 + m_2} y \quad (32)$$

To summarize, our approach is to solve Equations (27) and (28) numerically as a function of time. From these solutions, we will apply Equations (29-32) to obtain the motion for the binary system's individual masses m_1 and m_2 . Before we can effect this process, however, we need to develop the force f that's responsible for the nucleon-nucleus core interaction. This brings us directly to the Woods-Saxon potential.

The Woods-Saxon Form Factor

The much stronger nuclear force overcomes the repulsive Coulomb force between the constituent protons and thus the Coulomb interaction is ignored. The protons are accompanied by a number of neutrally charged particles – the neutrons. Unlike the gravitational and Coulomb forces, the strong force does not have a definitely well-known simple form. The Woods-Saxon potential models the nuclear mean field of the atomic nucleus, which is governed by the Fermi-function form factor [1]; that is,

$$f(r, R, a) = \left[1 + \exp\left(\frac{r-R}{a}\right) \right]^{-1} \quad (33)$$

where the size R and diffuseness of the surface a are fixed parameters. The total nuclear potential is given by

$$V(r) = -V_0 f(r, R, a), \quad (34)$$

where V_0 is the total potential strength and the minus sign indicates its attractive nature. Since

$$V(r) = - \int f(r) \cdot dr \quad (35)$$

we know that

$$f(r) = - \frac{dV(r)}{dr}, \quad (36)$$

which, with the use of Equations (33) and (34) into Equation (36), yields the Woods-Saxon nuclear force

$$f(r) = - \frac{V_0}{a} \left[1 + \exp\left(\frac{r-R}{a}\right) \right]^{-2} \exp\left(\frac{r-R}{a}\right). \quad (37)$$

Substituting this force, into Equations (27) and (28), we find the Cartesian form for each component

$$\frac{d^2x}{dt^2} = -\frac{V_0}{\mu a} \left[1 + \exp\left(\frac{\sqrt{x^2+y^2}-R}{a}\right) \right]^{-2} \exp\left(\frac{\sqrt{x^2+y^2}-R}{a}\right) \frac{x}{\sqrt{x^2+y^2}}, \quad (38)$$

$$\frac{d^2y}{dt^2} = -\frac{V_0}{\mu a} \left[1 + \exp\left(\frac{\sqrt{x^2+y^2}-R}{a}\right) \right]^{-2} \exp\left(\frac{\sqrt{x^2+y^2}-R}{a}\right) \frac{y}{\sqrt{x^2+y^2}}. \quad (39)$$

The universal values of the parameters R , a , and V_0 provided by Schwierz *et. al.* [1] are

$$V_0 = 52.06 \text{ MeV}, \quad (40)$$

$$a = 0.662 \text{ fm}, \quad (41)$$

$$R = R_0 A^{1/3}, \quad (42)$$

where

$$R_0 = 1.26 \text{ fm}. \quad (43)$$

In calculations of this type and to ensure proper functionality of the simulating program, all quantities should be expressed in appropriate dimensions. We define our distances in units of $a_0=1\text{fm}$; that is,

$$x = \bar{x}a_0 \quad y = \bar{y}a_0 \quad a = \bar{a}a_0 \quad \text{and} \quad R = \bar{R}a_0, \quad (44)$$

with the barred quantities being dimensionless throughout this paper. Time will be expressed in units of τ and energy in units of $E_b=1\text{MeV}$, to write

$$t = \bar{t} \tau \quad \text{and} \quad V_0 = \bar{V}_0 E_b. \quad (45)$$

The units of mass will be given in terms of the proton mass $m_0 = 938\text{MeV}$, to have

$$m_1 = \bar{m}_1 m_0 \quad m_2 = \bar{m}_2 m_0, \quad (46)$$

and

$$\mu = \frac{m_1 m_2}{m_1 + m_2} = \frac{\bar{m}_1 \bar{m}_2}{\bar{m}_1 + \bar{m}_2} m_0 = \bar{\mu} m_0. \quad (47)$$

Since we are dealing with a nucleon-core interaction where the nucleon mass is m_{nucleon} , the values of \bar{m}_1 and \bar{m}_2 are

$$\bar{m}_1 = \frac{m_{\text{nucleon}}}{m_0}, \quad \text{and} \quad \bar{m}_2 = \frac{m_c}{m_0}. \quad (48)$$

By making these replacements into Equations (38) and (39) and grouping units we have

$$\frac{d^2\bar{x}}{d\bar{t}^2} = -\left(\frac{E_b \tau^2}{m_0 a_0^2}\right) \frac{V_0}{\bar{\mu} \bar{a}} \left[1 + \exp\left(\frac{\sqrt{\bar{x}^2+\bar{y}^2}-\bar{R}}{\bar{a}}\right) \right]^{-2} \exp\left(\frac{\sqrt{\bar{x}^2+\bar{y}^2}-\bar{R}}{\bar{a}}\right) \frac{\bar{x}}{\sqrt{\bar{x}^2+\bar{y}^2}}, \quad (49)$$

$$\frac{d^2\bar{y}}{d\bar{t}^2} = -\left(\frac{E_b \tau^2}{m_0 a_0^2}\right) \frac{V_0}{\bar{\mu} \bar{a}} \left[1 + \exp\left(\frac{\sqrt{\bar{x}^2+\bar{y}^2}-\bar{R}}{\bar{a}}\right) \right]^{-2} \exp\left(\frac{\sqrt{\bar{x}^2+\bar{y}^2}-\bar{R}}{\bar{a}}\right) \frac{\bar{y}}{\sqrt{\bar{x}^2+\bar{y}^2}}. \quad (50)$$

Arbitrarily we next let

$$\left(\frac{E_b \tau^2}{m_0 a_0^2}\right) = 1, \quad (51)$$

so that Equations (49) and (50) yield the final equations that will be used in the simulation; that is,

$$\frac{d^2 \bar{x}}{dt^2} = -\frac{V_0}{\mu \bar{a}} \left[1 + \exp\left(\frac{\sqrt{\bar{x}^2 + \bar{y}^2} - R}{\bar{a}}\right) \right]^{-2} \exp\left(\frac{\sqrt{\bar{x}^2 + \bar{y}^2} - R}{\bar{a}}\right) \frac{\bar{x}}{\sqrt{\bar{x}^2 + \bar{y}^2}}, \quad (52)$$

$$\frac{d^2 \bar{y}}{dt^2} = -\frac{V_0}{\mu \bar{a}} \left[1 + \exp\left(\frac{\sqrt{\bar{x}^2 + \bar{y}^2} - R}{\bar{a}}\right) \right]^{-2} \exp\left(\frac{\sqrt{\bar{x}^2 + \bar{y}^2} - R}{\bar{a}}\right) \frac{\bar{y}}{\sqrt{\bar{x}^2 + \bar{y}^2}}. \quad (53)$$

It is useful to know the time scale of the simulation. We can solve (51) for τ to find

$$\tau = \sqrt{\frac{m_0 a_0^2}{E_b}} = \sqrt{\frac{(\frac{938 \text{ MeV}}{c^2})(1 \text{ fm}^2)}{1 \text{ MeV}}} = \sqrt{938 \left(\frac{\text{fm}}{c}\right)^2} = 30.6 \frac{\text{fm}}{c} = 1.02 \times 10^{-22} \text{ s}. \quad (54)$$

From this, we see that each second of simulation time is approximately equivalent to about 10^{-22} seconds in real time. This is indeed the reason why dimensionless units are used.

Initial Conditions

To get the simulation process started, we need the initial relative coordinate r and the relative velocity $\dot{r} = v$ at $t=0$. The relative and the center of mass coordinates are given by Equations (11) and (12); that is,

$$\mathbf{r}_{CM} = \frac{m_1 \mathbf{r}_1 + m_2 \mathbf{r}_2}{M}, \quad \mathbf{r} \equiv \mathbf{r}_{21} = \mathbf{r}_2 - \mathbf{r}_1. \quad (55)$$

We also need the velocities

$$\frac{d}{dt} \mathbf{r} = \frac{d}{dt} (\mathbf{r}_2 - \mathbf{r}_1) = \mathbf{v} = \mathbf{v}_2 - \mathbf{v}_1, \quad (56)$$

Similarly, for the center of mass velocity ($d \mathbf{r}_{CM} / dt$),

$$\mathbf{v}_{CM} = \frac{m_1 \mathbf{v}_1 + m_2 \mathbf{v}_2}{M}. \quad (57)$$

With these, if we pick the initial coordinates $r_{1,0}$ and $r_{2,0}$, then the relative coordinate and center of mass positions at $t=0$ are

$$\mathbf{r}_0 = \mathbf{r}_{2,0} - \mathbf{r}_{1,0}, \quad (58)$$

$$\mathbf{r}_{CM,0} = \frac{m_1 \mathbf{r}_{1,0} + m_2 \mathbf{r}_{2,0}}{M}. \quad (59)$$

Similarly, letting the initial particles' velocities be $v_{1,0}$ and $v_{2,0}$, we have for the relative and center of mass velocities

$$\mathbf{v}_0 = \mathbf{v}_{2,0} - \mathbf{v}_{1,0}, \quad (60)$$

$$\mathbf{v}_{CM,0} = \frac{m_1 \mathbf{v}_{1,0} + m_2 \mathbf{v}_{2,0}}{M}. \quad (61)$$

We further let the center of mass be located at the origin with zero velocity; that is,

$$\mathbf{r}_{CM,0} = 0, \quad \mathbf{v}_{CM,0} = 0. \quad (62)$$

With this understanding, Equation (59) gives

$$m_1 \mathbf{r}_{1,0} = -m_2 \mathbf{r}_{2,0}, \quad (63)$$

and Equation (61) gives

$$m_1 \mathbf{v}_{1,0} = -m_2 \mathbf{v}_{2,0}. \quad (64)$$

We see that Equation (63) yields the initial position of m_1 as

$$\mathbf{r}_{1,0} = -\frac{m_2}{m_1} \mathbf{r}_{2,0} \quad (65)$$

or in Cartesian coordinates

$$x_{1,0} = -\frac{m_2}{m_1} x_{2,0}, \quad y_{1,0} = -\frac{m_2}{m_1} y_{2,0}. \quad (66)$$

Likewise, from Equation (64) we have

$$v_{x1,0} = -\frac{m_2}{m_1} v_{x2,0}, \quad v_{y1,0} = -\frac{m_2}{m_1} v_{y2,0}. \quad (67)$$

Equations (66) and (67) consist of our initial conditions for the masses. We essentially choose $x_{2,0}$, $y_{2,0}$, $v_{x2,0}$, and $v_{y2,0}$, then use the formulas above to solve for the unknowns. These are then inserted into the relative coordinate formulas of Equations (58) and (60) giving an initial position and velocity for the reduced mass orbit of Equations (52) and (53); that is,

$$v_{x,0} = v_{x2,0} - v_{x1,0}, \quad v_{y,0} = v_{y2,0} - v_{y1,0}, \quad (68)$$

$$x_0 = x_{2,0} - x_{1,0}, \quad y_0 = y_{2,0} - y_{1,0}. \quad (69)$$

Simulations and Results

The actual simulation involves the numerical solutions of the Equations (52) and (53). The sample nucleus treated in our code is helium, but other nuclei can be considered equally well by changing the value of the nucleus' mass as well as the revolving particle's mass (neutron or proton). As mentioned before, the nucleus is considered as a system of two particles. One of the particles is a nucleon (m_1) that's considered to be interacting with the second particle; i.e., the remaining core (m_2). In our simulation the revolving particle is a neutron, thus the remaining core consists of two protons and a neutron. We have carried out the solutions in Java (using EJS) and MATLAB or Octave. The implementation is such that one writes

$$\frac{d\bar{x}}{dt} = \bar{v}_x, \quad \frac{d\bar{v}_x}{dt} = \bar{a}_x = \frac{\bar{f}}{\mu\bar{r}} \bar{x}, \quad \frac{d\bar{y}}{dt} = \bar{v}_y, \quad \frac{d\bar{v}_y}{dt} = \bar{a}_y = \frac{\bar{f}}{\mu\bar{r}} \bar{y}, \quad (70)$$

where $\bar{f} = -\frac{\gamma_0}{\bar{a}} \left[1 + \exp\left(\frac{\sqrt{\bar{x}^2 + \bar{y}^2} - \bar{R}}{\bar{a}}\right) \right]^{-2} \exp\left(\frac{\sqrt{\bar{x}^2 + \bar{y}^2} - \bar{R}}{\bar{a}}\right)$, and $\bar{r} = \sqrt{\bar{x}^2 + \bar{y}^2}$. So that we actually solve four first order differential equations with the initial conditions outlined in the previous sections. A plot of \bar{f} versus \bar{r} is shown in Figure 2.

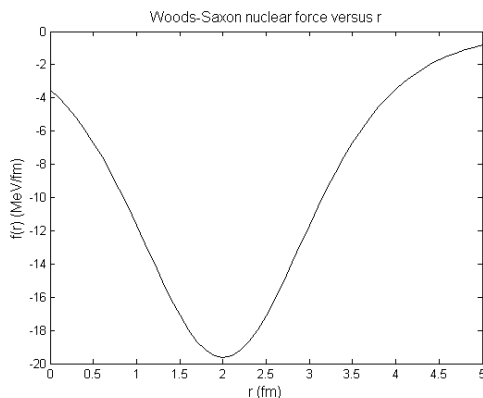


Figure 2. The Woods-Saxon force versus position.

The solution of these equations yields the sought reduced mass coordinates $\bar{x}(\bar{t})$ and $\bar{y}(\bar{t})$, which we plot below in Figure 3. Included in the plot are the coordinate positions r_1 and r_2 for the masses m_1 and m_2 , respectively, using the dimensionless units discussed previously. The figure's legend shows the coordinates plotted. The dotted black curve represents the coordinates of the reduced mass versus time. The middle sized orbit represents the motion of m_1 (the neutron) while the smaller sized orbit refers to the motion of m_2 (the remaining core). The center of mass is located at the origin. For colors refer to the code provided in the appendix or the website of reference [5].

To produce Figure 3, in addition to the conditions mentioned before, the initial values of $x_2 = 0$, $y_2 = 1.0$, $v_{y2} = 0$, and $v_{x2} = 0.085$ were also used. The two masses are seen to orbit about their common center of mass, from which the larger mass deviates less, producing a smaller orbit. The number of revolutions of either mass before returning to their respective starting positions is determined by the initial velocities. A larger tangential velocity will result in fewer revolutions.

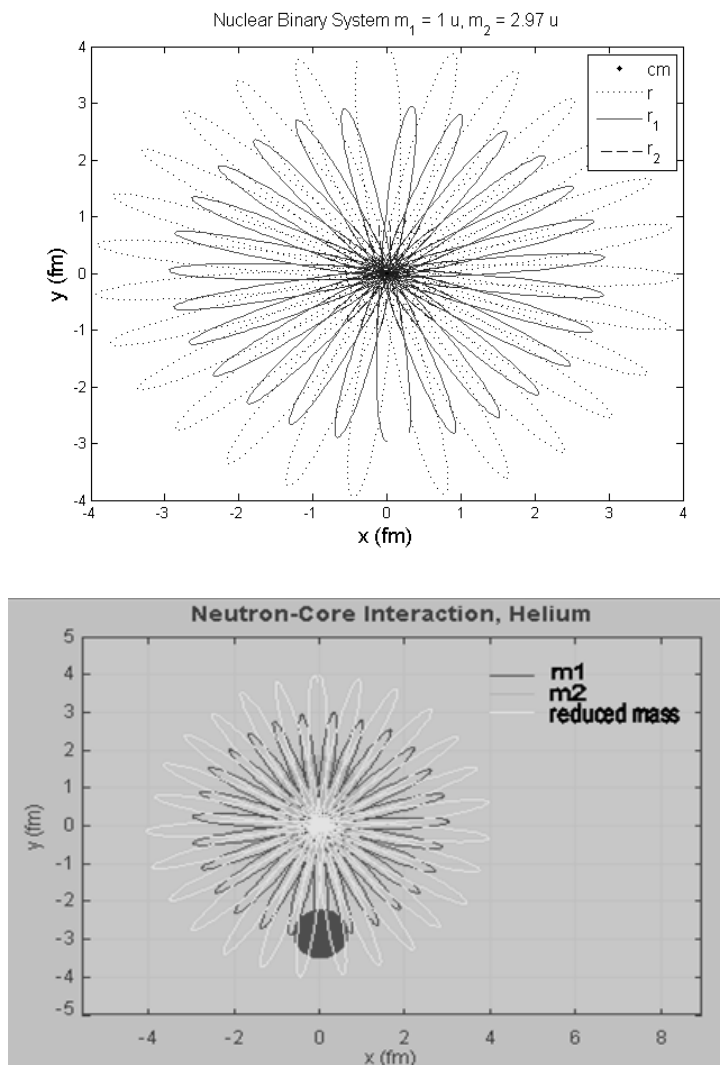


Figure 3. The results of the simulation for r , r_1 , and r_2 as a function of time. They have been plotted on an x-y plane. The panel on the top was produced by the MATLAB code shown in the Appendix. The panel on the bottom was produced using EJS [3,5]. In both panels, the larger orbit is that of the reduced mass. The middle sized orbit is that of the neutron, and the smaller sized orbit is the remaining nucleus. The large (cyan in the simulation of reference [5]) object in the top panel is the nucleus and the smaller red object (reference [5]) is the neutron.

The MATLAB/Octave code used to carry out the simulation is reproduced in the Appendix and the results are shown on the left panel of the figure. The EJS [3,5] code was used to create the figure's right panel. While the right panel of Figure 3 only shows the resulting plot, the EJS simulation contains a panel with which the user can change the atomic mass number and mass in order to analyze the orbitals of other atoms. The MATLAB simulation can be altered in the same way by simply changing the same values in the code.

CONCLUSION

In this paper, the orbits of a nucleon and its respective parent nucleus about their common center of mass has been simulated in an effort to provide a pedagogical approach to the understanding of the structure of atomic nuclei. We have treated the problem by modeling the interaction between the particles using an effective force derived from a Woods-Saxon potential. The parameters used in the simulation are those proposed in a recent study [1]. We have presented full details of the simulation to serve teachers and students alike. The approach is suitable for classroom use for those interested in learning more about nuclear physics. We have developed Java code that will be made accessible in the future [5] and which runs using the EJS open source project [3]. A suitable version of the simulation has also been developed using MATLAB [6] code, which is compatible with Octave [7]. This has been included in the Appendix.

ACKNOWLEDGEMENTS

We thank Francisco Esquembre [3] and the Open Source Physics project for the use of the EJS software. The OSP project can be found at <http://www.opensourcephysics.org/>.

REFERENCES

1. Schwierz N, Wiedenhöver I, and Volya A: "Parameterization of the Woods-Saxon Potential for Shell-Model Calculations." Cornell University Library, p5, 2007. The manuscript is found here: <http://arxiv.org/pdf/0709.3525v1>.
2. Krane KS: Introductory Nuclear Physics. John Wiley & Sons, NY, p118-121, 1988.
3. Esquembre, F: Easy Java Simulations." 2011. <http://fem.um.es/Ejs>. See also <http://www.opensourcephysics.org/>
4. Hasbun, Javier E: Classical Mechanics with MATLAB Applications. Jones and Bartlett, Sudbury, MA, 2009.
5. The EJS open source code to run the Java version of the simulation will be available in the future through this website: <http://www.westga.edu/~jhasbun/osp/osp.htm> (item 25); or directly here: http://www.westga.edu/~jhasbun/osp/nuclear_binary_system_EJS.html.

6. MATLAB. Natick, MA: The Mathworks Inc., 2012. Available at <http://www.mathworks.com>.
- 7 GNU Octave. John W Eaton, 1998-2012. Available at <http://www.gnu.org/software/octave/>.

APPENDIX

This appendix contains the MATLAB/Octave code “nuclear_binary_system.m” that reproduces the text’s Woods-Saxon nucleon-nucleus-core force shown in Figure 2 as well as the simulation results shown in Figure 3.

```
%nuclear_binary_system_matlab_octave.m - by Javier E. Hasbun and
%Benjamin E. Hogan (2012)
%Program to simulate the nucleon-core motion as a binary system
%The code is setup to run compatibly with Matlab as well as Octave.
%Matlab (commercial) is found here http://www.mathworks.com
%Octave (open source) is found here http://www.gnu.org/software/octave/
```

```
function nuclear_binary_system_matlab_octave
clear;
global a V0 R mu
mp=1.00727647; %proton mass in atomic mass units
mn=1.00866492; %neutron mass in atomic mass units
m1 = mn/mp; %m1 in units of one proton (the neutron here)
A= 4; %atomic mass number (A=4 => helium)
m2 = 4.002603-mn; %core: mass of He nucleus minus one neutron
m2 = m2/mp; %m2 in “proton units”
M = m1+m2; %total mass
V0= 52.06; %potential of the system (units of Eb=1 MeV)
a= 0.662; %skin thickness of the nucleus (units of fm)
R= 1.26*A^(1/3); %radius of the nucleus (units of fm)
mu= m1*m2/M; %reduced mass (units of mp)
%center of mass set at zero
CMx = 0;
CMy = 0;
%initial positions for m1 and m2
x2 = 0;
x1 = (-m2/m1)*x2;
y2=1.0;
y1 = (-m2/m1)*y2;
vy2 = 0;
vy1 = (-m2/m1)*vy2;
vx2 = 0.085;
vx1 = (-m2/m1)*vx2;
%initial conditions for the reduce mass orbit
vx0 = vx2-vx1;
```

```

vy0 = vy2-vy1;
x0 = x2-x1;
y0 = y2-y1;
ic=[x0;vx0;y0;vy0]; %initial conditions
tmax=50; %max simulation time in units of tau (tau_0~1e-20 s)
%Use Runge-Kutta (4,5) formula with the following settings
opt=odeset('AbsTol',1.e-7,'RelTol',1.e-4,'InitialStep',3,'MaxStep',5);
[t,w]=ode45(@nuclear_binary_der,[0.0,tmax],ic,opt);
%the calculated reduced mass coordinates and velocities
x=w(:,1);
vx=w(:,2);
y=w(:,3);
vy=w(:,4);
[nr,nc]=size(w); %get rows and columns of r
%Uncommenting the lines in the following loop allow real time view
%but works best only in MATLAB.
% figure
% for i=1:nr
% clf;
% plot(CMx,CMy,'.')% center of mass
% hold on
% plot(x,y,'k:')
% plot(x(i),y(i),'k.','MarkerSize',20)
% pause(0.5)
% end
figure
plot(CMx,CMy,'k.')% center of mass
hold on
plot(x,y,'k:')
x1 = CMx - m2*x/M;
y1 = CMy - m2*y/M;
x2 = CMx + m1*x/M;
y2 = CMy + m1*y/M;
plot(x1,y1,'r-')
plot(x2,y2,'b--')
legend('cm','r','r_1','r_2')
xlabel('x (fm)','FontSize',14)
ylabel('y (fm)','FontSize',14)
str=cat(2,'Nuclear Binary System m_1 = ',num2str(m1,3),' u, m_2 = ',...
num2str(m2,3),' u');
title(str,'FontSize',12)

%Plot of the Woods-Saxon nuclear force
figure
r=0:0.1:5;

```

```

plot(r,woods_saxon(r,a,V0,R),'k')
xlabel('r (fm)', 'FontSize', 12)
ylabel('f(r) (MeV/fm)', 'FontSize', 12)
title('Woods-Saxon nuclear force versus r', 'FontSize', 12)

```

```

%nuclear_binary_der: returns the derivatives for the binary mass system
function derivs = nuclear_binary_der(t,w)

```

```

%This calculates the derivatives:

```

```

%vx=dx/dt, ax=dvx/dt, vy=dy/dt and ay=dvy/dt

```

```

%Entries in the vector of dependent variables are:

```

```

%w(1,2,3,4)=x,vx,y,vy

```

```

global a V0 R mu

```

```

x=w(1);

```

```

vx=w(2);

```

```

y=w(3);

```

```

vy=w(4);

```

```

r=sqrt(x.^2+y.^2);

```

```

fw=woods_saxon(r,a,V0,R)/mu;

```

```

derivs = [vx;fw.*x./r;vy;fw.*y./r]; %contains the derivatives

```

```

function fws=woods_saxon(r,a,V0,R)

```

```

%The Woods-Saxon nuclear force

```

```

ep=exp((r-R)/a);

```

```

eps=a*(1+ep).^2;

```

```

fws=-V0*ep./eps;

```