

An Educational Model for Newtonian Mechanics

0216631

Abstract

Although Newtonian Mechanics is known to not be an entirely accurate model describing the motion of bodies, it is still widely used due to its relative simplicity and adequate level of accuracy for many applications. This report discusses the issues surrounding the development of a model as an educational aid for learning Newtonian Mechanics, including the characteristics a successful model would possess.

1- Introduction

The inspiration for the model described in this report arose from an incident that occurred while I was studying Advanced Level Physics. During the first class about Newtonian Mechanics the teacher explicitly outlined that the vertical and horizontal components of a bodies' motion were independent. We were then confronted with a device that would simultaneously launch a ball sideways and let an identical ball drop straight downwards, and asked to predict which ball would reach the ground first. Enlightened by the teacher's previous remarks the class correctly agreed that both balls would reach the ground simultaneously. I agreed with this deduction yet my intuition led me to believe that somehow the launched ball would somehow take longer to reach the ground. The experiment was performed and the predicted outcome was observed- both balls hit the ground at exactly the same time. Witnessing that single demonstration was enough to reform my very understanding of the motion of bodies under gravity. The goal of the project was to produce an EM-model that could replace the device from the story, and allow users to gain knowledge and understanding about Newtonian Mechanics by interacting with it..

2- Models for Education

2.1- Learning with Models

The first stage of the learning process is based around interaction with artefacts, leading to recognition of correlations between artefacts and in turn the identification of dependencies in a given system or situation [1]. The focus of this report will be on the development of a model that can provide interactions and experience to aid this initial stage of the

learning process in the context of attempting to learn about aspects of Newtonian Mechanics.

2.2- Benefits of Educational Models

There are many potential benefits to using a model as an educational aid. In the context of teaching Newtonian Mechanics any action or experiment performed in the natural world is implicitly restricted by the laws of nature. Adding to this are more practical issues such as cost, availability of equipment and time constraints; for example it is unlikely that many people would be able to perform experiments in locations with different levels of gravity (such as the surface of the moon) despite it being possible to do so. However an EM model is free of these constraints, and allows for a range of experiments to be performed (and experience gained) in situations and under conditions that otherwise would not be possible- such as under different gravitational conditions. In the same manner that a mathematician can use abstract concepts such as 0 and complex numbers to solve problems, a model can be used to interact with an environment where laws defining behaviour can be completely subverted (for example constructing a physical simulation that doesn't include gravitational forces). It follows that an educational model could be extremely useful for experimentation.

2.3 – Uses of Educational Models

There are several ways a model could be used as an educational aid, described here as *observation*, *interaction* and *redefinition*. Observation is defined here as use of a model in a purely passive sense, analogous to being shown a demonstration, and represents the lowest quality of experience that can be gained from a model. Interaction is use of a

model where the user has some kind of input to the model, and can observe the effects that the input has on the state of the model. Redefinition represents the experience where the user is granted free reign over the model- interaction and observation are coupled with the ability to explore and alter the functionality of the model itself. This kind of experience with a model would be the ‘richest’ experience with an educational model a user could have; be redefining and altering the model the user gains an insight into the model that goes beyond simple observation:

...the Newtonian World must enter the student, for it is a conceptual world which must be recreated in the mind of anyone who would know it. Each student must literally reinvent the Newtonian World in his/her own mind to understand it.

[2]

Therefore in the context of a Newtonian Mechanics model, it is essential that a user has the opportunity to access, observe, and alter the definitions of a model to fully comprehend it.

A successful educational model should be able to be utilised in all of these ways.

2.4- Desired Features of Educational Models

In order for the model to be of maximum use to the user, there are several features the model should ideally possess:

- Correctness – the model should convey or reproduce experiences about the entity, system or object it is modelling as accurately as possible. Although useful experience could be gained from interaction with an ‘incorrect’ model (such as contrasting experience with the model to ‘real-world’ experiences) this should be a result of the redefinition by users themselves rather than being subjected to a questionable model in the first instance. Note that this is not devaluing *simplistic* models, merely erroneous ones.
- Accessibility – the model should be as easy to use as possible. In the case of educational models, the purpose of the model is to enhance the users understanding or experience of a concept, idea, theory or situation. The use of the model is a means to an end, and the educational topic should not be overshadowed by details of the model itself. Furthermore, a frustrating experience

with a model could be counter-productive if it disrupts the users learning process.

- Attractiveness – the model should be engaging for the user. This does not necessarily refer to features such as ‘pretty’ graphical user interfaces (although they would not be a negative feature), but to the overall encounter the user has with the model; if it can be made enjoyable then it is more likely the user will benefit from their encounter with the model.

3- Description of the Model

3.1- Concept & Goals of the Model

The concept was to produce a model that could be used as an educational aid for the teaching of Newtonian Mechanics. As Newtonian Mechanics itself is a model of the behaviour of bodies in the natural world, it is a good candidate for an educational model (in the sense that the laws defining Newtonian Mechanics are well understood and can be successfully implemented or. the model could be implemented *correctly*).

Before development of the model began, several criteria were outlined relating to the previous discussion of educational models that the model should aim to satisfy. It should be possible to use the model in a completely open fashion – that is it allows observation, interaction and redefinition, so that it is fully customisable. Additionally the model should be as easy and intuitive to use as possible.

3.2- Research and Other Models

Other existing models were examined to research how interaction with the model was achieved, and how these models made use of the visualisation tools. Models such as the 3D Room Viewer [3] and the 3D Oxo Model [4] make valuable use of a three-dimensional Sasami display to visualise the model, yet interaction with the models is still limited to text input in the tkeden tool, or by ‘static’ mouse events via a Scout interface (usually by clicking on buttons). In order to make the model as accessible as possible, a more natural, dynamic method of interaction would be preferred, similar to the way the Planimeter model [6] can be manipulated. Ideally interaction directly with the Sasami display would be allowed, but currently this isn’t a facility available within Sasami.

3.3- Overview of the Model

The model is comprised of three distinct elements:

1. The model's 'engine'
2. The SCOUT window interface (with DONALD animated interfaces)
3. The Sasami 3D Visualisation

The engine is responsible for the movement and calculations involved within the model, and can be redefined utilising the tkeden interpreter. The SCOUT window interface allows visualisation of the model as well as dynamic interaction and manipulation of elements within the model. The Sasami display is a three-dimensional visualisation of the model.

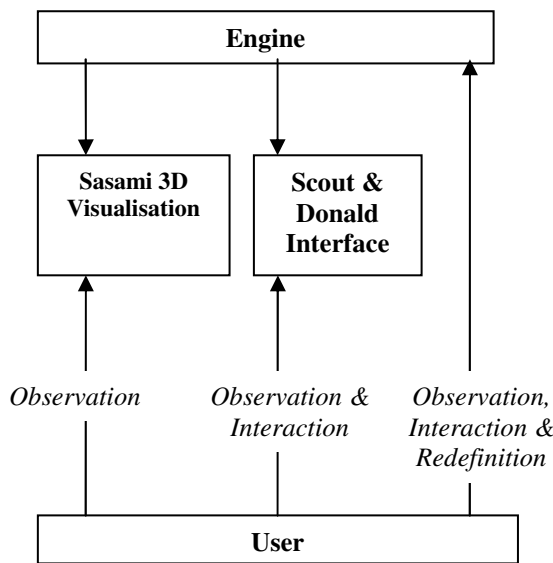


Figure 3.2 – An outline of the model, and the actions possible to the user.

3.4- The Model 'Engine'

The engine of the model is responsible for the movement of elements in the environment and the associated calculations required. Using the *eden clocks* timing features the positions of the elements are calculated and updated at specific time intervals. To allow the model of Newtonian physics to be expanded and customised, effects upon elements within the model are expressed as procedures, and stored in a list. This list is iterated over every environment cycle, allowing each procedure to act upon the environment.

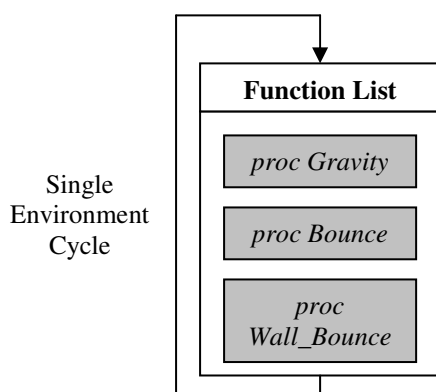


Figure 3.4- An outline of how behaviour within the model is calculated. The functions within the list are executed every environment clock cycle, and represent the physical laws of the environment within the model.

As a user can freely redefine and add functions to the function list, the entire behaviour within the model can be customised and altered. During the development of the model many different procedures were written to enable different behaviours within the environment, such as making the bodies within the model bounce, and to prevent them from leaving the 'arena' –an area designated for movement. This functions can be included and removed from the function list as required.(Refer to Appendix for examples.)

In an earlier version of the model, all the behaviour within the environment was explicitly coded via a series of definitions, building a dependency tree to automatically calculate the positions of elements within the environment. While this worked well for the gravitational forces and simple collision detection initially featured in the model, it proved to be too difficult to redefine in order to specify more complex behaviour. In order to keep the model accessible, the approach described in Figure 3.4 was used.

3.5- The SCOUT window Interface

The SCOUT window interface allows the user to interact with the model in an intuitive dynamic fashion as well as act as a visualisation of the model. This interface utilises SCOUT windowing definitions and Donald definitions to provide a visualisation which is updated using dependency. The user can reposition elements in the environment by clicking in the relevant plane, and 'throw' elements by dragging them in their relevant plane. (Refer to Appendix for examples.) This approach to interaction is intended to allow a rich experience with the model, as by using the mouse to 'throw' bodies in the model environment is more akin to the natural experience of dropping or throwing something in the real world. This approach aims to make the model more *accessible* and *attractive* to the user.

4- Conclusions & Further Work

The model as it stands defines an arena, in which a body moves under the influence of environmental forces. The model can be observed utilising the Sasami 3D interface, SCOUT window, or by using the tkeden interpreter to observe the state of the model. The user can interact with the model in a natural,

intuitive, and dynamic fashion utilising the SCOUT window, or again via the tkeden interpreter, and the entire model can be redefined by adding procedures and altering the value of the function list to represent the 'physical laws' currently action within the environment. The model is accessible and attractive, and correct regarding the subsection of Newtonian Mechanics that was implemented.

Currently the model features only a single body under the influence of the environment- due to the open design of the model's engine it could easily be extended to support multiple bodies. Additionally, by adding more behaviour defining procedures into the model (as described in figure 3.4), features such as inter-body collision detection and conservation of momentum could be implemented. It can be said that the model is fully customisable. With the further work described here, the model produced could stand as a replacement for the device mentioned in section 1, and therefore be a worthwhile educational aid.

References

- [1] *Empirical Modelling for Educational Technology*, W.M Beynon.
- [2] *Modelling Games In The Newtonian World*, David Hestenes, Arizona State University
- [3] *3D Room Viewer Model*, Ben Carter, 1999.
<http://empublic.dcs.warwick.ac.uk/projects/room3dsasamiCarter1999/>
- [4] *3D OXO*, Chris Roe, 2001.
<http://empublic.dcs.warwick.ac.uk/projects/3doxoRoe2001/>
- [5] *Planimeter Model*, Charles Care, 2005.

Appendix

(Included as a guide only, not to be included in page count)

Examples of using the Model

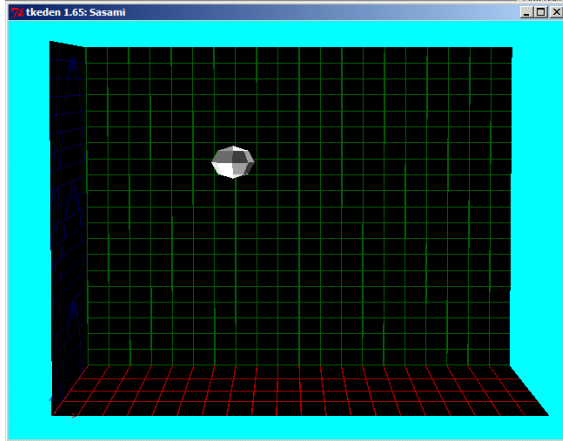


Figure i : The 3D Sasami visualisation.

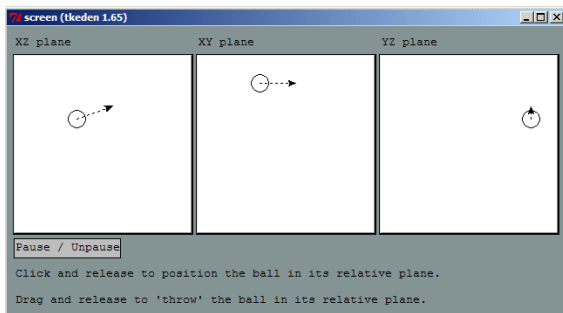


Figure ii : The Scout & Donald Interface. The arrows represent the vector of movement for the ball within the specified plane.

- 1) By clicking and releasing on one of the planes, the ball will be moved to that relative position.
- 2) By clicking and dragging on the ball in the plane, the ball will be 'thrown' in the model in that plane.
- 3) Pressing the 'Pause / Unpause' button will stop and restart the simulation. (Note: Throwing the ball will automatically restart the simulation)

Examples of extending / customising the model

- 1) As all of the environment properties are defined as procedures, to redefine the strength of gravity to 5m/s^2 you would enter the following Eden definition into the tkeden interpreter:

```
%eden
proc gravity
{
    vector = addLists(vector, [0,0,-
        0.5*increment]);
}
```

Figure iii : an eden definition to redefine the effect of gravity.

Note: increment is merely a multiplier to ensure smooth movement within the model.

- 2) To alter the actual physical laws governing the system, the `functionList` can be redefined. For example, the bounce of the ball can be replaced with the definitions for collision detection included in `model.e` with the following eden definition:

```
%eden
functionList = [arena_bounds,
    floor_collision, gravity];
```

Figure iv : an eden definition redefining functionList- in this case the 'normal' bouncing ball rules are replaced with straight collision detection with the edges of the arena.