Viability of Modelling Real World Objects for Educational and Historical Purposes

0724593

Abstract

The simplest way to demonstrate physical laws and properties is often a literal physical demonstration. Equally, the best way to understand the workings of physical devices from the past is through constructing physical copies of them that can then be experimented with. However, some objects can potentially be very big, very dangerous, and not suited to a classroom or academic environment. A trebuchet (a type of medieval catapult) would be a good example. Empirical Modelling can provide the solution to these problems by allowing people to create realistic models of physical items that can then be experimented with, thereby allowing students to experiment with, and learn about, physical laws and properties of these items and allowing historians to experiment with the capabilities of objects from the past, all within a safe, cheap and convenient environment. However, there is only merit in creating models to simulate objects if these simulations are actually accurate and realistic. If the model is not an accurate representation of reality then the knowledge and insight gained from it will not be valid and will lead to users developing an inaccurate mental image of the construct they were studying. Additionally there is only merit in using a model if it actually saves you time and involves a viable workload. If the amount of work involved in creating relatively simplistic models is too high then the potential gain will not be judged worth the effort that must be expended by users. Thus this paper will analyse the viability of modelling real world objects with regards to physical accuracy, ease of model use, and the amount of time realistically required.

1 Introduction

Learning is a complex process, and more often than not the simplest and most effective way for people to learn something is by doing it. Practical interaction can result in students retaining knowledge to a far higher degree than if they had simply been told the knowledge without the aid of the practical interaction[1]. However, classrooms are limited in size. Should a teacher wish to demonstrate a large piece of equipment then they may find it impossible to do so. Equally health and safety is, perhaps somewhat unfortunately, a big issue. Some pieces of equipment may be considered too dangerous for use in a classroom. Additionally some equipment may be prohibitively expensive, especially if each student should be given their own to experiment with.

These issues can be solved via the use of simulations and models. We will take a trebuchet as our example. A trebuchet is a large medieval siege weapon that propels a projectile, usually a large rock, via a sling attached to a throwing arm, moved via a weight on the other end of the arm.

Image 1.1[2]

The physics involved in such a device are fairly complex, and yet a good demonstration of fairly key physical principles. The path of the projectile is governed by the laws of motion, and the force and angle of the release are governed by weights and angles. As such it is clear that the study of a trebuchet would be a valuable classroom exercise. However, a trebuchet is far too large to fit into a classroom. And additionally, even if you were to produce a small model, only 1 pupil could use it at once, and it would probably result in health and safety com-

[1] Evaluation of Practical Learning Programmes, Edge, 2009

[2] Picture of a Trebuchet, real-world-physics-problems.com

plaints. However, if a model of a trebuchet were created then each student could spend some time playing around and experimenting with it, helping them to grasp the basic physics principles it demonstrates. And not only that, but the results from firing the trebuchet would be far more accurately measured too. In a similar way, a historian may wish to examine the mechanics and abilities of a trebuchet in order to gauge its effectiveness in past sieges. Constructing an actual trebuchet would be costly, time consuming and inconvenient, so once again the benefit of a model is clear.

However, whilst clearly there are potential benefits to constructing a model of a physical object, there could also be potential problems. A model is only of use if it accurately simulates reality. If it does not then people will be left with an incorrect impression of how the physical item functions, rendering the exercise pointless. Thus it is important to determine whether models can be made that function in an accurate and realistic manner. Additionally, both teachers and academics such as historians have limited amounts of time available to them. However useful an accurate model might potentially be, they will not create the model if it takes too long to do so, or if it would require expert help.

Thus it is clear that in order to establish the viability of models for use in teaching and by historians we must weight up the potential advantages against the potential problems. To do this I created a trebuchet model. This report will describe the model, the benefits of it, the problems with it, the benefits and problems with the tools used to produce it and then look at existing popular models in common use.

2 The Trebuchet Model

I created my model using EDEN, DoNaLD and SCOUT. When the model loads the user is presented with the image of a trebuchet that is ready to fire. The user need only right click and the trebuchet weight will drop, swinging the arm up and firing the projectile. As par of describing how the model is to be used and modified it is important to explain how the model was constructed and structured.

First a SCOUT window is created to contain everything. Then the shape called trebuchet is created, which will contain all the parts that make up the trebuchet. Then within that comes the base that contains three key points/variables. The first of these is bottom_left, so called because it marks the bottom left point of the entire trebuchet. It is from this point that all other points are calculated. Thus, should the user wish to reposition the model they should change the bottom_left value within base, and nothing else.

The second important value within base (its self within trebuchet) is height. This, as one would expect, sets the height of the trebuchet frame. The third important point is pivot. This is the point at the top of the frame about which the throwing arm rotates. It should not be changed, as it is automatically calculated from the height of the trebuchet and width of the base.

The next important variable is within throwarm. It is worth noting that whilst the throwing arm looks like one solid beam it is actually made up of the throwing end and the weighted end. This is done such that the length of each can be separately adjusted. Within throwarm is an angle variable, that controls the angle of the arm.

The projectile is modelled using the basic laws of motion, namely the equation $v = u + at$, or velocity (at any given time) is equal to starting velocity plus acceleration multiplied by time. This is applied separately to the horizontal and vertical plane. Vertically speaking there is no initial velocity of course, as it is released flat, and the acceleration comes from gravity. This value of gravity can be changed to simulate the trebuchet working in other environments. In the horizontal plane acceleration represents air resistance (and is thus deceleration), meaning it should always take the form of a negative value (or zero for no air resistance).

The motion within the simulation is governed by a clock tick. Changing the value of speed will change the speed of the simulation.

3 Usefulness of the Model

The next stage after completing the model was to verify whether or not it would potentially be useful to either students or historians as a valid replacement for an actual physical, real-life model.

3.1 The model irrespective of the tool used to create it

Ignoring temporarily the tkeden environment, I will now look at the model in its self to deduce whether it would be of academic benefit.

My model successfully simulates the flight of a projectile based on the laws of motion. A user could change any of these variables and see the correct adjusted flight path. This is clearly of academic benefit. Equally, whilst the default values are relatively arbitrary, should more specific values for variables be carefully chosen this could thus be used to deduce things about siege weapons.

However, there are some limitations to my model. The height of the trebuchet could be changed, as could the release angle, the weight used, the gravity and air resistance, the length of the throwing arm, the length of the rope between the arm and the projectile sling. But, whilst these variables can be changed, the nature of my model, based on the way I constructed it, is such that some of those changes would not affect the overall result in the way one would expect were they modifying an actual physical trebuchet.

For example, changing the value of gravity or air resistance would accurately change the path of the projectile, as one might expect. However, the centripetal force holding a projectile in a trebuchet's sling before release turns out to be very hard to model. (more on that in section 3.2). As such this means that altering the release angle of the trebuchet will not correctly alter the initial trajectory of the projectile.

Equally, in its current state, the initial velocity of the projectile, whilst alterable, is not initially set by the force exerted on it as a result of the falling weight at the other end of the arm (because this involves a very complex set of mathematical equations that are simply impossible to perform in EDEN on DoNaLD, at least without taking many hundreds of lines of code).

Thus in some cases a user can alter parameters of the model and observe the changes that result. However, in other cases the user can alter parameters of the model and end up with a false, or inaccurate, result.

This highlights a fairly key issue with empirical modelling. Namely that free experimentation with a model is risky. Should someone produce a model, and specify which variables can be changed and experimented with, then all should be well. This clearly provides educational benefit and leaves little room for unexpected side effects. However, one of the potential benefits of empirical modelling is that the user can experiment with the model in unanticipated ways, adapting it to situations unanticipated by the model's creator and exploring non-conventional properties or ways of thinking about things. This is one of the more desirable features of empirical modelling, but it is alas also the most risky. In an ideal world it would be fine. If a model is built perfectly, as an exact representation of reality, with all dependencies exactly correct, then a user may alter anything they like, in ways unanticipated by the creator without any negative side effects. However, if the model was only complete within certain limits, with certain functionality merely simulated in a way that looks right, but technically isn't, then any results from unanticipated changes to the model could essentially be gibberish.

It is unrealistic to assume that models will be entirely accurate (and in many cases actually rather undesirable. After all, empirical modelling isn't about creating an exact model of something, but more about creating something the user will interpret as the right thing. It is about how the user construes the model, based on their own frame of reference that counts). As such a model modified in unexpected ways may not function desirably or usefully. Occasionally these unexpected side effects may be interesting in themselves. But certainly from an educational point of view, and from the point of view of accurate history, there can be no room for doubt, and no room for potentially incorrect results or conclusions being drawn. As such it seems that a model is only valid if along with it is supplies a set of instructions as to what the user can and should feel able to change whilst still obtaining valid results.

3.2 The model with regards to tkeden, EDEN, DoNaLD and SCOUT

There are several issues with EDEN and DoNaLD that should be highlighted. The first is simply that they are two different notations that must be used with each other. Both of them are missing key bits of functionality, with this problem being solved by switching between the two. However, whilst this might provide a somewhat hacked together solution it is hardly ideal.

In some ways it is simple syntactic differences between the languages that are tiresome. EDEN considers a boolean variable to be 1 or 0, whereas DoNaLD considers it true or false. Equally in DoNaLD a # denotes a comment, whereas in EDEN you would use */. Now whilst this may not be a massive issue once you have become used to the different notations it is certainly unnecessary (and makes it impossible to easily develop an IDE for EDEN and DoNaLD, which is something that would be highly useful. It is of course worth noting that in some ways cadence combines elements of EDEN and DoNaLD, but cadence relies on other programs and languages just as much.

More concerning are incompatibilities in the interpretation of variables and types between EDEN and DoNaLD. For example in DoNaLD you can define a point (based on x and y coordinates). However, whilst EDEN can access these variables it sees them as a list (a list of values, ie, the x and y values) rather than truly understanding what a point is. This removes the ability to, within EDEN, subtract one point from another for example. This was a significant issue at times.

Another possibly petty complaint with DoNaLD is that the coordinates system originates from the wrong place. Whilst traditionally mathematically the bottom left corner is considered (0,0) this is not the case with computers. For legacy reasons relating to windows scrolling downwards creating new lines as they go computers generally consider the top left corner to be coordinate (0,0).However, DoNaLD does not follow the traditional convention and had (0,0) in the bottom left. If tkeden was simply used to create a model that the users then interfaces with via a menu interface and buttons then this would be fine. However, a feature of this system is that the user can type into tkeden, in either EDEN or DoNaLD format, and access and change any part of the model. This is part of the power of the empirical modelling system, that the user can change the system as they see fit. But it relies on them having the correct frame of reference, or they will construe things incorrectly. If the user believes the coordinates start from the top left as is more computationally normal then they may edit values in a way other than they actually intended. Thus whilst this seems like a small point it highlights the fact that, when you give the uses complete power over a model, you must make sure that there is no ambiguity in the way that the system works.

4 A Look at More Popular Models

Many, many models exist that are extremely popular, and serve a variety of purposes. Perhaps the most popular models are games. Any computer game is essentially a model of something physical. In the case of early games like pong it was a very crude model of table tennis. In the case of modern games we have fairly accurate models of cars and race tracks, weapons and virtual environments in which characters can interact with one another. There are entire simulated worlds and communities, such as World of Warcraft.

However, these models all have one thing in common, in that they can only be accessed in the ways expected and specified by the creator of the game. You cannot change anything they didn't want you to be able to change, and are highly limited in ways you can mess with the environment. Ina game of pong for example you cannot alter the length of your paddle.

This is all very well for gaming and entertainment, but possesses less educational value. Learning to drive a car on the computer is very different from learning to drive one in reality. Though of course it could be argued that these games lead on to the first training simulators. BSM (British School of Motoring) now offer the use of realistic driving simulators, and most military pilots are trained on simulators before being allowed in an actual fighter plane. These models are highly accurate, and possess massive educational value. However, they do not allow the user to modify things. A user of a realistic flight simulator may be able to safely train in the use of a fighter jet, but they cannot experiment to see what happens if you double the strength of gravity, for example.

Thus this is where we must differentiate between two different types of models. The type that are truly interactive, and the time that are limited to a set interface.

There are of course many existing educational models, such as the mathematics ITP measuring scales simulator[3], which allows children to get to grips with basic weights and measures. Once again though we see that the most useful and popular models are the ones that are not fully customisable and modifiable, unlike models created in EDEN, which can of course be changed in any way at any time whilst running.

It is also worth considering existing historical models. In most cases these are limited to simulations of historical environments (like a 3D model of the Titanic you can walk around). There are few models available for historical machines and devices. This could possibly be because and modern historical items still physically exist in museums, and any older items are not well enough understood to produce accurate models. However, this is where tools like EDEN would be ideal, as a full understanding of the device being modelled is not required when you start. You can experiment as you go, changing things moving things about as you go. Thus it could be argued that modelling in the tkeden style is ideally suited to use by historians.

5 Conclusions

It seems to be clear that models of physical items can certainly be created that would have educational benefit and that would also be beneficial to historians and scholars.

However, it is also clear that the issue is not simple. Whilst there are benefits to using a model rather than an actual physical model there are also several quite problematic disadvantages. It largely seems it depends what the focus of the educational or historical model is intended to be.

For users who desire a polished and accurate model, that will give correct answers at all times and that will only allowed them to interact in a set and controlled way, then models certainly already exist, and are fairly simplistic to produce. For models of this type EDEN and DoNaLD (nor Cadence) are arguably not the best tools to use. They encourage the user to modify the model in unexpected ways, which is nor desirable. Instead programs such

[3] http://nationalstrategies.standards.dcsf.gov.uk/node/47789

as Java may be far more suitable, or other programs more specifically tailored to simulations, such as the Unity engine.

For users who wish to be able to experiment with a model based on their own personal frame of reference and interpretation of the construct, who wish to change things in ways that are perhaps not realistic or expected, then models created in EDEN and DoNaLD are far more suitable. The tkeden interface is perfectly suited to such situations and interactions.

However, as is so often the case, with increased abilities comes increased risk or errors and increased required expertise. In these situations the user must also be careful to make sure they understand the way in which the model was originally created. The designer of the model may have had to inaccurately simulate some features of the model, in ways that are not an issue if the model is used as the creator intended, but that may be an issue if other things are changed.

In order to minimise this as an issue it is important that the creator of a model properly sets up variables to reference one another, avoiding hard coding in any values where possible, such that if one value is changed then everything else that relates to it correctly updates also.

It is worth noting that the EDEN and DoNaLD interfaces are in some ways problematic, which may be a problem if models are made for educational purposes aimed at a younger audience, as younger users may be more easily confused.

Thus is seems to be undeniably the case that modelling is a powerful and useful tool for both educational and historical purposes. However, it should not be assumed that all types of model are suitable for all purposes and styles of user, and it is highly important to remember that you can only stretch a model within certain limits. Using a model in unanticipated ways and in an experimental manner can be a powerful tool indeed, but like most powerful tools can result in unfortunate results if the system is not fully understood before use. This perhaps highlights the importance of empirical modelling, and emphasises why it a discipline that users should be introduced to as early as possible.

References

Evaluation of Practical Learning Programmes, Edge, 2009, Accessed in PDF format on 25/1/11 at

http://www.edge.co.uk/media/uploads/Downloadable/ba5ed34d-916a-49fd-9c78-e573e1e27377.pdf

Image 1.1, Picture of Trebuchet, Real World Physics Problems, 2009-11, real-world-physics-problems.com, accessed at

http://www.real-world-physics-problems.com/trebuchet-physics.html

Trebuchet Mechanics, Donald B Siano, March 28, 2001, accessed at http://www.algobeautytreb.com/trebmath35.pdf