

Using graphical design modelling tools to understand phenomena

Victor Kabukoba - 1060672

In the following paper, empirical modelling is analysed and the suitability and purpose for applying it within the domain of graphical design is assessed. The background to design and research is reviewed and compared and contrasted to the field of empirical modelling. A model developed using the empirical modelling tool tkEden is introduced and reviewed to demonstrate the tool's suitability for this task.

1 Introduction

Vision is the most fundamental sense humans use to learn, the evidence of this is clear [x]. Utilising techniques that maximise modelling the real life visual observations and understandings could only increase the ability to learn and understand scenarios and phenomena which humans see.

1.1 Motivations to apply empirical methods

Throughout history people have attempted to apply various make sense of their surroundings. The scientific or inductive method of investigating our surroundings was famously applied by the mathematician and physicist Galileo. Upon discussing the theory of the tidal waves on earth, he applied an inductive methodology. He took one of the plausible theories which would explain the tidal periods and on which superficially made sense. By applying its principles to other highly related scenarios one could see whether the original theory was credible. In the case of for example, varying basin depths for bodies of water Galileo used an example of lakes and pools lacking tidal periods as example of the theory failing to apply as a reason to disregard it [2].

The interest in work applying inductive reasoning is when compared to deductive reasoning it can be suggested the inductive method is more intuitive and natural, it lends itself well to the human mind. Simply put applying a form of inductive model building can enable a user to build a model based on ones' experiences and observations.

1.2 Methodology of building Empirical Models.

On the whole empirical modelling (E.M) takes a different approach to the process of not just building programs and models but also what we know as computing. The paper

The Foundations of Computing by Brian Cantwell Smith [5] demonstrates the will to direct computing into what can be shown in the real world. The paper highlights a empirical criteria as one of the possible ways to define computing.

One way of interpreting the paper is to view the idea of symbolic representations or computing not being able to fulfil the way we humans think but that focusing on what we humans experience in reality. Whilst on the surface the ideas in the paper may be difficult to apply in order to build programs which

carry a function for building models which are based on the observers experiences is natural.

In order to build an empirical model the three components can be identified which work mutually to build a model. A empirical model comprises of three components.

1. Modeller/Observer. This is the role the person who effectively builds the model. The observer observes whatever they wish to, whether that be audibly, logically, mathematically and visually.

2. Referent. This is what the observer is observing and attempting to model. As long as it persists in a form it can be modelled

3. The construal is the personal 'model' the observer is constructing. Its contents, designs are influenced by the observers own perceptions, experiences and interests. It may not reflect the referent exactly in the eyes of another observer but the point is that it is constructed in the eyes of the observer/modeller.

Within the construal observables are identified, simply objects which the observer has identified. Connections between observables can be made by identifying the dependencies between them. Dependencies define how changes to one object can effect others which can be interpreted by our understanding of the referent.

Within the model there can be certain objects which can affect the environment through actions, these are the agents within the model.

Between all three components information travels in a bi-directional manner. The interpretations made by the observer change with experimentation as the observer observes to the referent and changes their understanding as they begin to understand and interpret their own creation. The construal's faithfulness or efficacy is closely related to the observers own interpretation.

This is a fairly contrasting approach to building a model based using traditional computational techniques. Which can involve more distinct and certainly chronologically varying steps. For example designing and algorithm which satisfies the truths of

what the observer understands may well be the initial step. Translating this into a high level programming language and testing are just some of the steps involved. What is produced is a circumscribed system which conforms to an abstract specification of what the model should do or how it should do it.

Whilst empirical modelling seems the more natural approach which can be analogised with an explorer attempting to discover the properties of a discovered mysterious artefact the question is how is this applicable in reality, what systems are in place to build this construal?

1.3 Interactive model building.

Many tools to construct models are available, in order to perform empirical modelling the tool *tkEden* developed at the University of Warwick can be used. A fundamental aspect in building the construal is the idea that it is an interactive process. In practice the observer makes a change to the code for the construal and observes the change in the construal. The construal is compared to the referent and more changes can be made, the process is repeatable, until the observer is satisfied with the construal.

Interactivity can be achieved through making changes to the graphical representation of a construal. The dependencies may still be implemented in the code but they can also be identified via the graphical representation.

In *Evaluating definitive principles for interactive graphics* by M.Beynon (1989) [1], it is implied that an object orientated approach fulfils some of the identified characteristics of interactive graphics. Where we see an object which is made up of smaller constituent parts but represented in the program as a object as a whole. The relationships of these objects can be described and consequently the relationship of their constituent parts also.

The degree of interactivity required is an important aspect to consider. In *Evaluating definitive principles for interactive graphics* (1989) an engineering and constructive perspective is taken.

"The demands that a user would ideally wish to make upon a design are very great:

- relationships should be easily perceptible, and conveniently modified.
- it should be possible to record partial information about relationships conveniently.
- relationships should be expressible at many levels of abstraction."
- it should be possible to accommodate a temporary failure to meet constraints at a later stage and conveniently make consequent changes retrospectively." *M.Beynon (1989) [1]*.

The construction or engineering project envisaged is for a complex building project. The interesting theme raised in above excerpt is how relationships or in *tkEden* terms *dependencies* need to be conceivable and flexible. These are crucial features that the empirical modelling tool *tkEden* provides.

In engineering design it can be said that there are two main methods. One which is conceptual, and involves thinking about all the possibilities whether proven logical or otherwise, very much reflecting the inductive reasoning used by Galileo [2]. The other is a practical design approach which bases itself in proven designs and case studies, which to degree is at odds with the deductive approach [3].

It is common in engineering design to also to solve mechanical problems by using physical assumptions and mathematical approximations. This is possibly where a feature such as "...accommodate a temporary failure to meet constraints at a later stage and conveniently make consequent changes retrospectively." stated by M.Beynon (1989) [1] can be become useful for a designer.

What can be deduced from these sources is that there is a opportunity for a software tool to allow a non computer scientist to work with a modelling tool build a model and identified relationships and gather information back. In order to broaden the use of this tool certain delivery methods can be used which allow a user work with the tool without extensive programming experience. An emphasis on building a graphical representations of components

of model can go a long way to meeting the needs of a wide range of designers and their experience.

1.4 Graphical Modelling

Visual stimulation has been shown to aid the learning process of users, especially in subjects who lack experience in the subject matter [7].

A most comprehensive example was shown by work by W.Sung and S.Ou presented in the paper *Learning Computer Graphics Using Virtual Reality Technologies Based on Constructivism - Case Study of the WebDeGrator System (20022 [8])*. The work involved applying a virtual reality system which presents to the user a 3D world in which they can interact with the system to learn computer graphics principles. The idea is that the users actions are represented in real-time, and their actions are more obvious in the 3D world. The basis of their work was research done in increase in user understanding when shown 3D objects.

The topics users were asked to learn in the system were described as complex ideas involving " Bezier B-Spline and NURBS". The described complexity of the work can be compared to that of an engineer working on a complex building as discussed in section 1.3. The article attempts to convince the user of the need for this system by citing work by " Jonassen (1996)" concerning constructivism. It explains that formal symbolic representations usually presented in programming relies on previous experience and knowledge. The stimulation and immersion provided in a visual and especially virtual reality environment reduces the need for previous experience and knowledge and relates far more to the real world. This touches on points made by Brian Cantwell Smith [5].

The work done in developing the *tkEden* empirical modelling tool was started far earlier than the applications described by W.Sung and S.Ou [8]. The early work in developing the graphics component for *tkEden*, *DoNaLD* had a high degree of interactivity in mind. *DoNaLD* consists of definitive notations which define how different variables relate with each other in the form of equations. These equations represent graphics components in sun-rock.v1 EM model manner in which the changes in the equations

are expressed in the graphical representation shown to the user. The system can run effectively in a real-time manner such that any change in the equations is reflected immediately making the dependencies between variables not just shown in code but graphically also.

The graphical components in Dare made up of points and lines and shapes which make up more complex components.

tkEden and *DoNaLD* allow a users to not only deduce the observables and dependencies and clearly the construal, but the context can be deduced from the code.

1.5 Example Empirical Model.

In this section a relatively simple empirical model is described which was developed using *tkEden*.

The motivation and basis of the model was rather inspired by the work of Galileo, as he attempted to understand whether the planets or the moon affected the tide on earth, as well as his work on analysing the planets with his telescope [2].

At the time writing of this article, human knowledge of the earth's solar system is far advanced to that of the time of Galileo [2]. Despite this the motion of massive bodies such as the planets, stars and meteorites is still of educational and recreational interest.

The model is a simple representation of solar system with a solitary star and another massive body, possibly a planet or huge meteorites orbiting the sun.

The model utilises a few equations of motion especially equation 1.1 a adaptation of Newton's 2nd Law of motion, to model specially the attractive gravitational forces between two bodies.

$$g = \frac{Gm_e}{R^2} \quad \text{equation 1.1 [9]}$$

where g= acceleration due to gravity of an object, G= the universal gravitational constant, me = mass of a body at the origin, R =distance of the object from the body at the origin.

```
rocklaccn is (G * sunmass )/((
rockldistr ) * ( rockldistr ));
```

Figure 1.1 - Excerpt of code from sun-rock.v1 EM model. [10]

Figure 1.1 shows the *tkEden* application of equation 1.1. What this shows is several variables and importantly the dependencies of the variables. It can be seen that *rocklaccn* represents the acceleration due to gravity g acting on a rock. The *rockldistr* is the distance of the rock from the sun, which is the body at the origin. It can be seen from both figure 1.1 and equation 1.1 that the acceleration is dependent on the mass of the body at the origin, in this case the *sunmass*.

The knowledge used to build the model was obtained from the author's prior knowledge of how massive objects attract one another, represented in literature such as books or observed on video, for example when a comet or meteorite collides with a planet. The knowledge to apply a mathematical equation model the motion of the rock was from experience in physics and engineering.

The model can be represented diagrammatically in figure 1.2

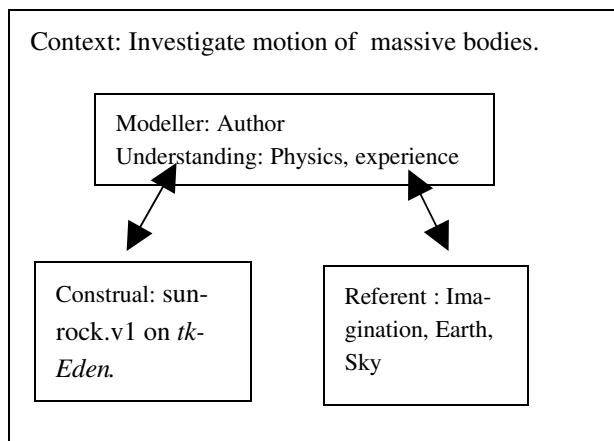


Figure 1.2 - Diagram representing the relationship between components in the empirical sun-rock.v1 (E.M.) model.

It is important to note concerning the model that the code was constructed according to figure 1.2 and not in the manner of specification through abstract-

tion. This is according to the principles of building a empirical model shown in section 1.2.

The observables were deemed to include anything that the external observer believed significant and interesting. Including the sun, the orbiting rock, the gravity and/or mass of the sun. Also the potential for a tangential component acting on the rock which effects the trajectory of the rock, and potentially prevents the rock falling into the sun.

A dependencies not explicitly shown in equation 1.1, is shown only graphically and that is the link between the apparent size of the sun and the mass.

The agents can be identified at the author and the sun which has a significant impact on the rock.

The model features a simple user interface in which a user can change the mass of the sun and due to the identified dependencies, the acceleration of gravity acting on the rock. A tangential component is acting on the rock which can be varied also, this is meant to represent the original velocity that acted on the rock at its genesis, in reality this could be when rock broke from a larger piece.

An interesting aspect of the model is that *tkEden* conforms to several of the necessities of an interactive graphics system stated by work by M.Beynon (1989) [1]. Namely flexibility in not requiring to meet the perceived constraints of the model. It was thought by the author that whilst the model is not a simulation it could represent the motion of two bodies closely. However due to screen size in tkEden, in order to make a better visual representation the universal gravitation constant was modified to allow the whole orbit to fit the screen. The existence of dependencies crucially makes restoring the original universal gravitation constant value when this screen size hurdle is overcome, a simple matter of entering the correct value into the *tkEden* command line. All the dependencies of the universal gravitation constant will be updated.

The overall graphical interaction of the model is produced via the visual representation of the solar system and the GUI controls for the sun mass and tangential acceleration component. A user can use the *tkEden* user interface to manipulate and change variables and instantly see the results on the display

screen. Users can learn not only the syntax by merely interacting with *tkEden* but also learn how varying variables that form the equations and dependencies effect the motion of the rock. This model is not intended to act as a simulation but demonstrate the suitability of tkeden in providing a graphical demonstration of the motion of bodies using mathematical formula.

1.6 Conclusion

To conclude this report, the work on the empirical model will be reviewed and a reflection on the topic of graphical modelling and design.

The model itself is superficially simple but underneath there is extensive code which allows the model to function. The initial ambition of the model was to explore how the motion of bodies can be represented graphically using empirical modelling tools. Overall the model was a success. The model shows how a massive rock orbits a sun and the effect of gravity of mass of the sun has on the rock. It can be seen visually how as the distance between the rock and the sun decreases its acceleration increases. This can be seen by analysing the equation, but is shown graphically in a obvious manner. Increasing the mass of the sun increases the acceleration due to gravity acting on the object.

The model has much room for development. Due to the nature of empirical modelling, as stated before, a another user can learn how the model operates and how it was built by interacting with the model. From there several developments can be made. The scale of the screen can be modified such that a simulation of the earth and any nearby massive objects can be simulated effectively. This could be done for educational purposes, and there is no reason why the model can not return accurate data to use in calculations.

The graphical interactivity of the model can also be increased. The actual rock can be repositioned anywhere within 'space' by capturing the mouse clicks and placing the rock in the new location by building a dependencies. The dependency of the rock's current location can match that of the mouse position.

The next step would be to build the model using 3D tools which can be done using the empirical modelling tool Sasami to increase the immersion and learning experience in desinging as shown in work by W.Sung and S.Ou [8].

Acknowledgements

I would like to thank Meurig Beynon and Steven Russ for providing the lectures concerning empirical modelling and the extensive online resources.

References

- [1] M.Beynon, Evaluating definitive principles for interactive graphics, *New advances in computer graphics*. 1989
- [2] P.K.. Machamer. Feyerabend And Galileo: The Interaction Of Theories, And The Reinterpretation Of Experience. *Studies In History and Philosophy of Science Part A*. Volume 4, Issue 1, pgs 1-46.
- [3] C. Matthews, *Case studies in Engineëring Design*, John Wiley, 1st Ed., New York, 1998, pgs 1-5.
- [4] J Meriam and L.G. Kraige, *Engineering Mechanics: Statics*, John Wiley, 4th Ed, Canada, 1998, pgs 7-8.
- [5] B.C. Smith, The Foundations of Computing, *Unpublished*, URL:
<http://www.ageofsignificance.org/people/bcsmith/print/smith-foundtns.pdf>
- [6] M.Beynon, Definitive Principles for Interactive Graphics, *Research Report 93*. 1987.
- [7] J.Pettersson, Visualising interactive graphics design for testing with users, *Digital Cretivity*, Volume 13, Issue 3, pgs 133-156, 2002
- [8] W.Sung and S.Ou, Learning Computer Graphics Using Virtual Reality Technologies Based on Constructivism - Case Study of the WebDe-
Grator System, *Interactive Learning Environments*, Volume 10, Issue 3, pgs 177-197, 2002
- [9] J Meriam and L.G. Kraige, *Engineering Mechanics: Dynamics*, John Wiley, 5th Ed, Hoboken NJ, 1998, pgs 8-9.
- [10] V.Kabukoba, sun-rockv1.e, tkEden 1.73, unpublished.