

Motivating Teachers and Pupils to Engage with Modeling.

Laurence Rogers

School of Education, University of Leicester, England

Abstract

It has been found that science teachers in secondary schools have embraced the use of simulation software more enthusiastically than modeling software. Some simulations are visual aids, chosen for their ability to help pupils visualize complex or abstract phenomena. Others feature virtual experiments which allow pupils to perform pseudo-laboratory activities and obtain quasi-experimental data. In both cases it is common for the software to facilitate activities which support the development of valuable skills for scientific investigation. It is argued that modeling software has even greater potential for developing these skills towards a deeper level of scientific understanding. However, many modeling software systems possess a conceptual and presentational format which appears to be less accessible than graphically-rich simulations. The paper describes a new type of software of hybrid design which attempts to build bridges between apparently successful simulations and potentially more demanding modeling activities. The development involved careful consideration of the language used for expressing scientific concepts and relationships, contextual factors which influence motivation and the design of tasks to promote effective use of the software tools.

The adoption of ICT in science lessons

At the present time in the UK young teachers entering the profession have sometimes been described as ‘digital natives’, having been surrounded from birth by the products of information and communication technology. Their environment of computers, Internet, mobile phones and wireless technology has equipped them with technical skills and confidence which many ‘digital migrants’ of the previous generation of teachers have struggled to achieve. The effect of this has helped transform the adoption in schools of ICT tools for teaching and learning. However, some tools have succeeded more than others. In a survey of science teachers in the UK, one of the most popular types of application was simulation software (Rogers & Finlayson, 2003). The possibilities for simulating physical phenomena with the use of animated graphics were widely appreciated as valuable teaching tools. In a wider survey across several European countries prepared by the IT for US project (www.itforus.oeiizk.waw.pl), of the different types of software to be found in secondary schools, simulation software comes out with a consistently higher score than do other types.

In particular, there is a contrasting response for modeling software, whose function is very closely associated with simulations, but appears to be less widely used.

Country Software type	Cyprus (%)	Netherlands (%)	Poland (%)	Portugal (%)	UK (%)
11.1 Presenting and reporting software (word processor, electronic presentations, (web pages)	63	79	67	72	90
11.2 Tutorial software	7	32	52	17	29
11.3 Information storage systems (databases, encyclopedias, multimedia CD's, web-sites, etc)	43	75	57	63	90
11.4 Simulation and visualization software	47	83	36	65	86
11.5 Calculating software (spreadsheets, statistical or mathematical programs)	10	62	32	56	52
11.6 Modeling software (Coach, Modellus, Insight, PowerSim, Stella, etc)	3	81	10	59	24
11.7 Measuring tools (data logging, video measurement)	50	84	6	35	76
11.8 Sound analysis software	0	50	11	19	20
11.9 Drill and practice programs (revision etc)	3	42	24	15	62
11.10 Software for monitoring student's progress	7	24	12	13	29

Table 1. Types of software used in science lessons (IT for US Project, 2006)

Simulations and Modeling software compared

For the purpose of developing the present discussion, it is necessary to distinguish between simulation software and modeling software. Both

types of software share similar pedagogical aims and employ similar mathematical techniques. Both attempt to help the learner explore and understand physical phenomena in a virtual environment. Both may be used to facilitate investigative inquiry, the exploration of relationships between variables, the testing of hypotheses and so on (page 84, Newton and Rogers, 2001). However, on inspection, an obvious distinction is in the design of the user interface; modeling software is usually of a generic, symbolic design (e.g. as in Stella, Excel, Modellus programs) whereas simulations usually offer a graphical interface customized to the needs of the topic under consideration. A more significant difference is that, although every simulation employs a mathematical model as an ‘engine’ to perform calculations, the model is usually not explicit or accessible to the user. In contrast, modeling software allows all the mathematical definitions and assumptions in the model to be scrutinized and, if desired, edited and modified; there is greater freedom for controlling and choosing the mathematical expression of relationships between variables. Thus, activities with modeling software can probe more deeply into the assumptions inherent in the design of the model, potentially pose more questions about possible alternatives and ultimately facilitate a deeper scientific understanding. Table 2 suggests a comparison of the distinctive differences between modeling software and simulations.

It should be remarked that a considerable number of simulations are mainly offered as graphical animation aids for visualizing phenomena. Many so-called ‘applets’ fall into this ‘visual aid’ class of simulation. The other class of simulation may be termed as ‘virtual experiments’ in which data is generated for analysis and relationships may be explored.

To explain the apparent preference for simulations over modeling, it is possible that the simplified subset of tools and implicit guidance inherent in simulations makes them more accessible than modeling programs, which, although they allow greater degrees of freedom, their successful use demands more insight and skill on the part of the user. Another explanation might be found in the teaching styles adopted by teachers. In a recent UK teacher training program in ICT methods in science teaching, it was found that the most common mode of use of simulations was as a teacher demonstration. (Table 3). Teacher exposition is a dominant pedagogy for many teachers, and it appears that teachers’ skills in explanation and demonstration are readily adapted to the use of simulations. It may be argued that teachers readily identify simulations as tools with which they can engage in discourse with a whole class of pupils. The thesis to be developed in this paper is that suitably designed software can exploit teachers’ positive disposition towards simulations to lead them towards more demanding modeling techniques.

Simulations	Models
<p>The designer has a major role in defining the user interface and scope of possible investigations.</p> <p>Appropriate tools for presentation and analysis are pre-selected (displays, graphs, axes, cursors, controls and calculating aids)</p>	<p>The user has freedom to examine, adjust and modify the model.</p> <p>The user may choose from a broad range of tools for analysis and presentation.</p> <p>A model makes the physical principles more explicit.</p> <p>A model can show how a solution to a complex problem may be synthesized from simple elements based on first principles.</p>

Table 2. Differences between simulations and models

Type of activity	Individual (%)	Group (%)	Demonstration (%)
Internet	46	44	10
Data-logging	2	33	65
Simulation	19	13	68
Spreadsheet	39	39	22
Using models	30	38	32

Table 3. Comparison of teaching formats for different ICT activities. (Rogers and Finlayson, 2004)

Pedagogical objectives of modeling activities

Gilbert and Boulter (2000, page 13) describe three main contributions of modeling to science education. First, models have a central role in the development of the scientific understanding of any phenomenon. Second, the testing of models is an important process of science in action. Third, scientific models are major creative outcomes of science. Translating these into specific objectives for modeling activities, the most commonly stated objectives in practice are:

- To prompt thinking and exploration of scientific ideas.
- To assist the interpretation and understanding of real data (through a process of comparing data from a model with data from an experiment)

- To test the accuracy of a model and to evaluate its implicit assumptions.
- To make possible the virtual exploration of otherwise expensive, remote or dangerous experiments.
- To simplify the solution of complex problems in terms of simple elements.
- To facilitate the extension or reinforcement of previous learning.
- To provide opportunities for revision exercises.
- To rehearse experimental procedures employed in laboratory work.
- To build confidence in analyzing data. (Models can generate ‘noise-free’ data which allow clear conclusions to be drawn about relationships.)

As previously indicated, activities with modeling software share many common objectives to those of activities with simulations; the main distinction arises from the level of access and the depth of investigation. Of course all models and simulations are built upon assumptions about the variables involved and the scientific principles thought to govern them. The very process of challenging such assumptions is an integral aspect of science in action, so modeling activities which allow those assumptions to be challenged and tested provide an extremely valuable means of developing an understanding of science (Webb, 1993). Lawrence has argued that modeling tools fulfill the need of individual learners to find ways of expressing their own thinking about scientific problems (Lawrence, 2005)

The discussion will be developed by considering two case studies to illustrate a gradient of activities, starting with simulations and progressing towards tasks with models. In each case, a simulation provides a welcoming context which relates to a pupil’s previous experience and attempts to stimulate their interest by posing questions for investigation. These require them to experiment with the variables involved and evaluate the results through applying their previous knowledge. This can involve graphical analysis of data generated by the simulation. Progression to modeling tasks usually begins with examining the model to review all the variables and the relationships between them. Crucially, new tasks with the model involve modifying the model to test alternative relationships. If desired, completely new models may be built. The modeling program used is *Simulation Insight* (Logotron, 2005) which contains in one window an authoring system for creating simulations and in another window a modeling system employing graphical objects to represent variables. The simulation window contains animated graphics whose movement and changes are driven by variables defined in the modeling window.

Case Study 1: Simulations of house insulation

The context for these simulations is very familiar to people who endure winters in central and northern Europe. House designers incorporate various ways of improving the thermal insulation of the building; cavity walls, double glazing and loft insulation. The first simulation allows the insulating properties of a cavity wall to be investigated. The user may control the outside temperature, the inside house temperature, the cavity thickness and the insertion of insulating wool.

As each of these variables are adjusted in turn, the effect on the rate of heat transfer may be observed and students can gain a qualitative perception of the relationship between the variables. Through more systematic control of the variables, data may also be recorded, graphs analyzed and a quantitative description of a relationship achieved.

The next step of the activity is to examine the model (Figure 2a). This is viewable in an alternative window. The model representation of connected blocks is designed to indicate the relationship between variables; each block represents one variable or constant. The geometrical layout may be freely organized to maximize the lucidity of the model. The arrows show how values are used in the calculation of secondary variables. In this case, the rate of heat flow H is calculated from three other variables, T_i , T_o and th . K is a constant.

Whilst the model is running, values are displayed in each block and input variables may be adjusted with slider and spinner controls (Figure 2b). The model may be investigated in a very similar manner to the way in which the simulation window was controlled and can be seen to generate similar results. However, the analytical appearance of the model, although less stimulating visually, gives a clear focus on the numerical changes occurring as input variables are adjusted, but more importantly, it makes the mathematical basis of the calculations quite explicit. In this case, the model assumes that the rate of heat transfer varies in proportion to the temperature difference across the wall section and inversely with the thickness of the cavity. The language of mathematical formulae allows this relationship to be specified precisely, and the program contains a formula builder to facilitate this. However the program also offers an alternative exploratory method of defining relationships which is textually based and less dependent on algebraic skills.

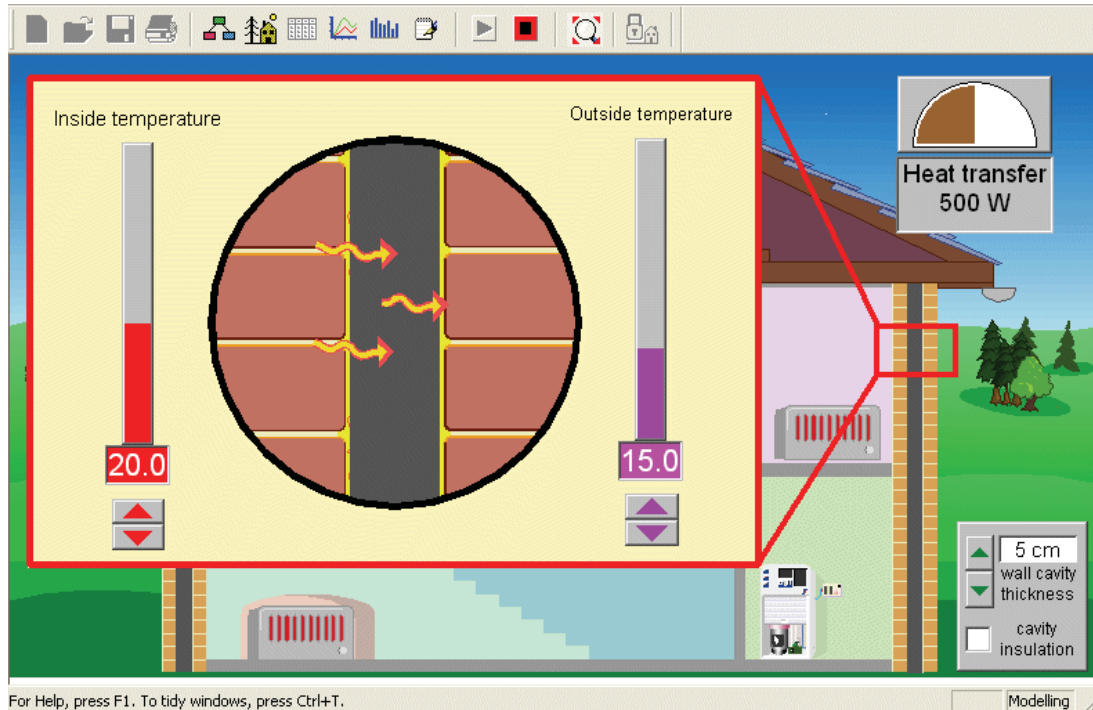


Figure 1. Simulation of the insulating properties of a cavity wall.

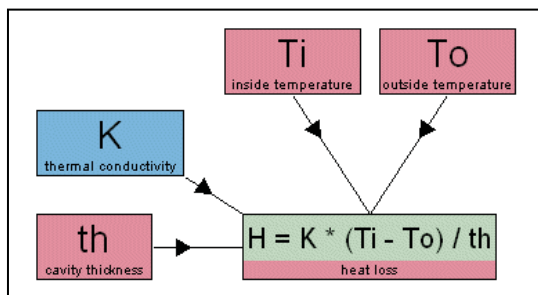


Figure 2a. Thermal insulation model

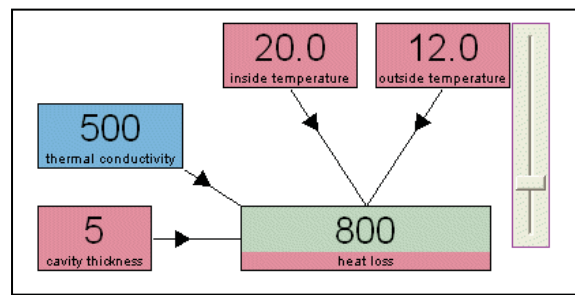


Figure 2b. Model in run mode

The introductory tasks investigating the effect of each variable on the heat transfer had previously demonstrated the success of the pre-defined model in making credible calculations and predictions which seem to match experience of the phenomenon in real life. A design challenge to consider for less mathematically able pupils is how a successful alternative model might be built using only descriptive phrases to define relationships between the variables. The program allows this to be done using a text-based dialogue in place of the formula builder (Figure 3).

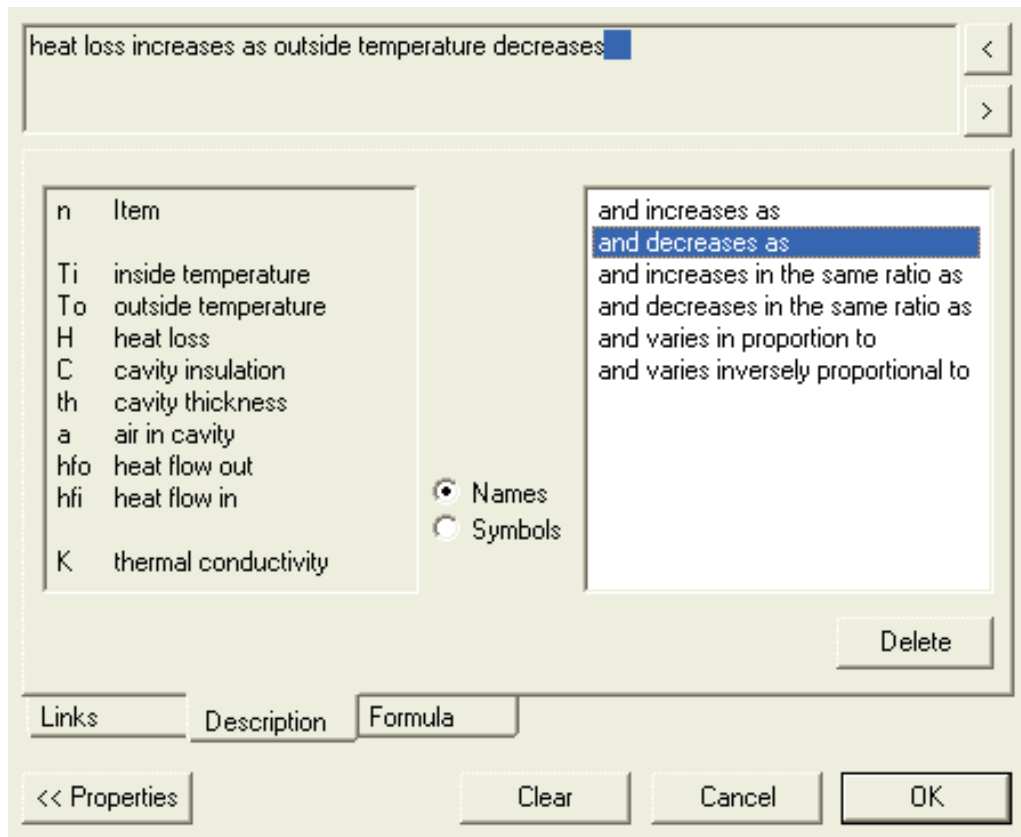


Figure 3. Dialogue for defining a relationship between variables

Having cleared the previous formula, pupils can point and click on appropriate phrases and variable names to build a description in words which expresses their idea of how the variables might affect each other. For example, each of these alternatives might be tried:

heat flow increases as inside temperature increases
heat flow increases as outside temperature decreases
heat flow decreases as cavity thickness increases

For each description, the model may set to run and the resulting behavior compared. In this example, each description successfully produces the correct trend which serves to confirm pupils' thinking about a relationship. This textual method works well for simple relationships, but unfortunately struggles to adequately describe complex multi-variable relationships, so is unable to offer a model embracing all the variables. To answer this problem, one must return to the original pre-defined model to see how these relationships are combined and expressed in a single formula. Ultimately the formula definition method gives greater precision and clarity, but modeling activity with the descriptions can be used in a formative way to promote thinking about relationships.

This fairly simple simulation and model is useful as a preparatory exercise to a second more sophisticated simulation which calculates the

internal temperature of a house, taking into account the insulating properties of doors, windows and the loft under the roof. The user may adjust the external air temperature and the rate of heat production by the boiler and observe how a new equilibrium becomes established according to the various insulation options. This is a dynamic model in the sense that, by including a time variable, it handles rates of change and shows how long it takes for changes to occur.

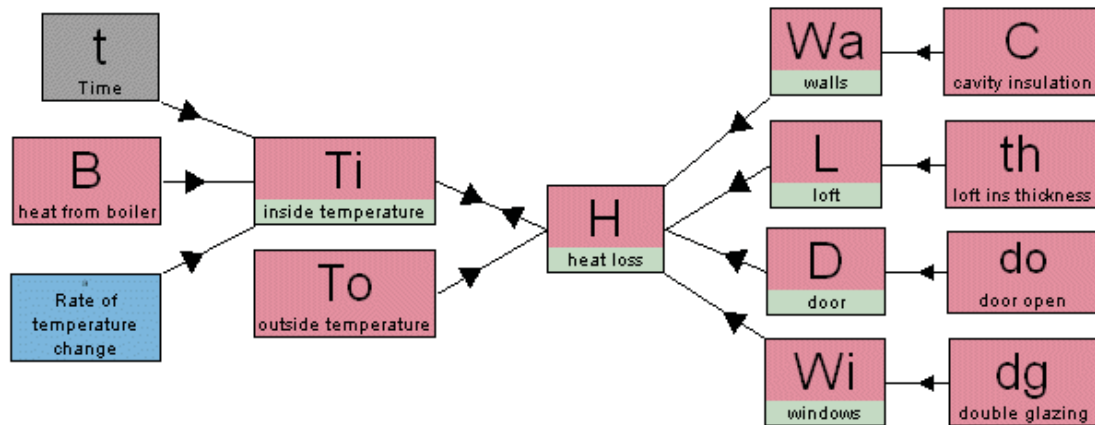


Figure 4. Model for house insulation

Case Study 2: Simulations of accelerated motion

These simulations consider various forces acting on a body and calculate the velocity and displacement for the resulting motion. The context for the first is the problem of pushing a car whose engine will not start. A car is the mass to be accelerated and the external force is supplied by a choice of one, two or three men. When 'Push' is activated, the car moves and a simultaneous graph shows the acceleration, velocity and displacement (Figure 5).

It is immediately evident that the constant force produces a constant acceleration, whereas the velocity increases linearly. Altering the number of men pushing changes the acceleration and the gradient of the velocity graph. If the men stop pushing, acceleration drops to zero, velocity becomes constant and the displacement graph is linear. These explorations give pupils an opportunity to distinguish between uniform acceleration and uniform velocity, and the role of external forces in determining the motion. Quantitative comparisons may be made by analyzing the graphs; for example, measurements of gradient on the velocity graph can be related to the acceleration.

As previously, the next step is to view the model in the alternative window. This reveals the mathematical basis of the calculations (Figure 6).

The model defines incremental calculations for velocity and displacement rather than absolute values which, in this program, are automatically

computed. Newton's Second Law is assumed for calculating acceleration. Being based on basic principles and definitions, this model will be recognized as a basic building block of models for many other examples of motion such as that for a parachute which follows here. There are many opportunities for elaborating this basic model, the most obvious of which is to introduce a variable which takes account of friction. Herein lies the potential of modeling activities; complete access to the model, adapting it to accommodate additional conditions, modifying it to express alternative assumptions.

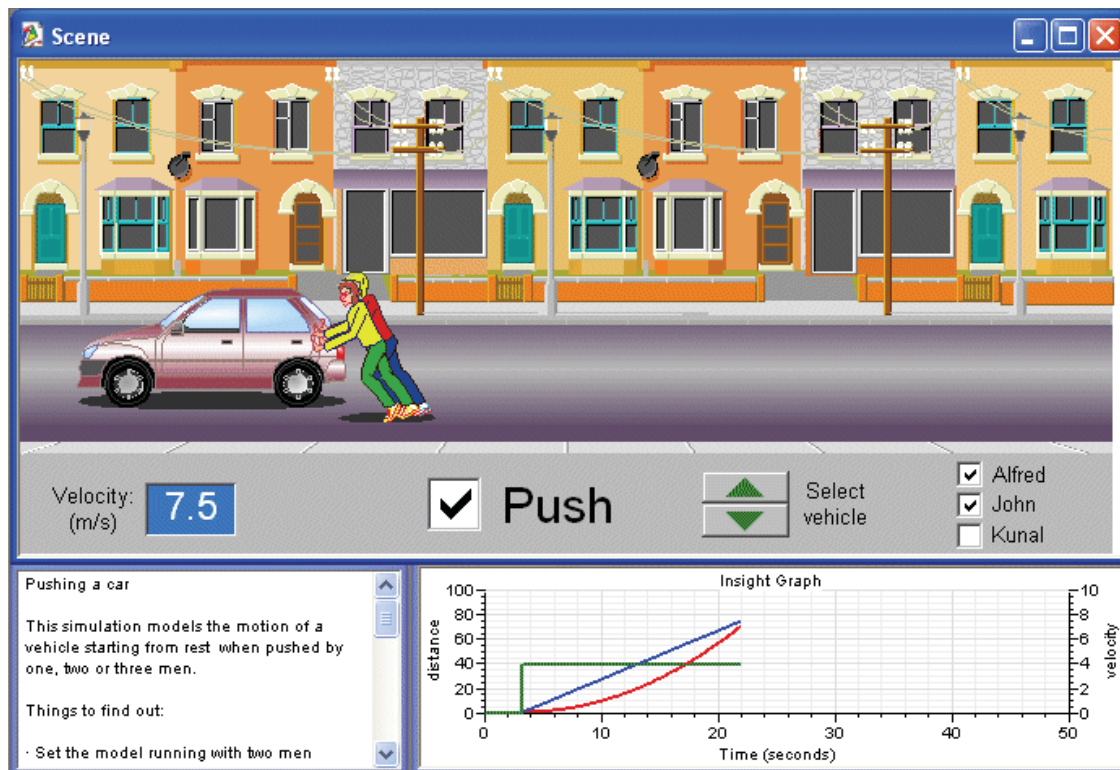


Figure 5. Simulation of moving a car by pushing.

The second simulation described here illustrates this sort of elaboration. This example features the motion of a parachutist jumping from an aeroplane. The simulation window (Figure 7) sets the context and invites the user to experiment with controlling a limited number of variables: the mass of the parachutist, the diameter of the parachute, the time of the jump and the time for opening the parachute. Initial activities can establish how the variables affect the time taken to reach the ground and how they may be controlled to ensure a terminal velocity which is safe for landing. Turning to the model, the core variables of time, acceleration, velocity and height fallen are recognized, with the additions of weight and air resistance as the forces involved. The total of nine variables suggests a moderately complex problem, but the geometrical layout with directional links shows a simplified pattern of dependencies. The summary list of definitions indicates that individually their origins are very simple: height, velocity and weight express basic principles, acceleration comes from Newton's Second Law. Air resistance is

assumed to increase with diameter and velocity. Since air resistance affects acceleration, which in turn affects velocity, a ‘feedback’ loop of dependencies exists between these three variables.

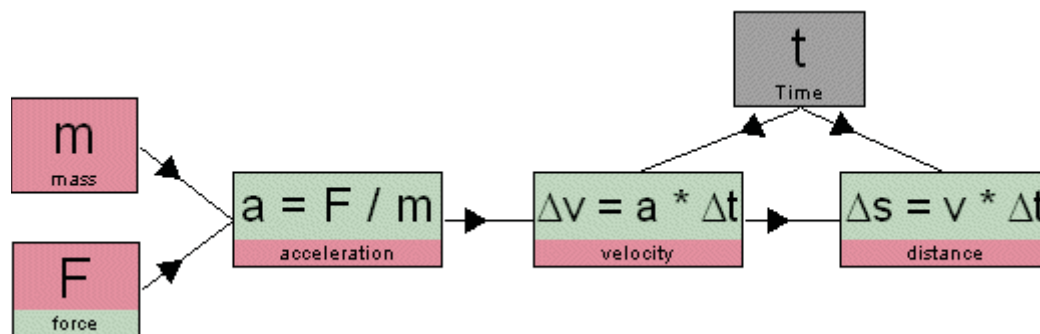


Figure 6. Model for motion of car

To optimize the learning value of the simulation and model, the activities need to be carefully planned with clear objectives. The following sequence of outline questions and tasks is a possible framework for achieving this:

The general need for establishing an agenda of activities with simulation and modeling tools will be considered next.

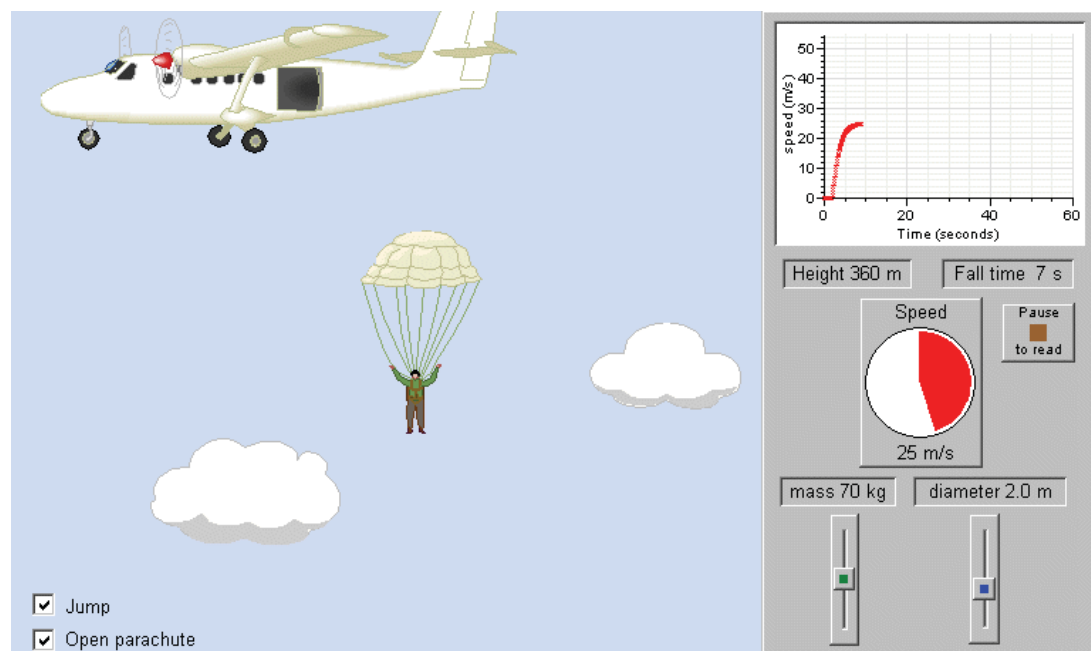


Figure 7. Simulation of a parachute drop.

Planning curricular activity

Modeling and simulation tools are most effective when there is personal engagement of the learner. All the exploration and thinking described in this paper demands not simply response but commitment of the learner. The broader experience of teachers makes them familiar with this requirement generally in science education and most have developed strategies for stimulating and motivating pupils to achieve this in a whole

range of science skills. For modeling to succeed in the classroom, the challenge is to provide a framework within which the software tools can flourish. In recent years the concept of ‘scaffolding’ pupils’ thinking to support an essentially personal learning process has emerged in pedagogy. Scaffolding implies a careful balance between instruction and learner autonomy. It demands clear learning objectives, an understanding of pupils’ needs, dialogue, questioning, discussion and so on. The teacher is the best judge of how to manage this in detail, but let us consider some principles for guiding the planning of modeling activities.

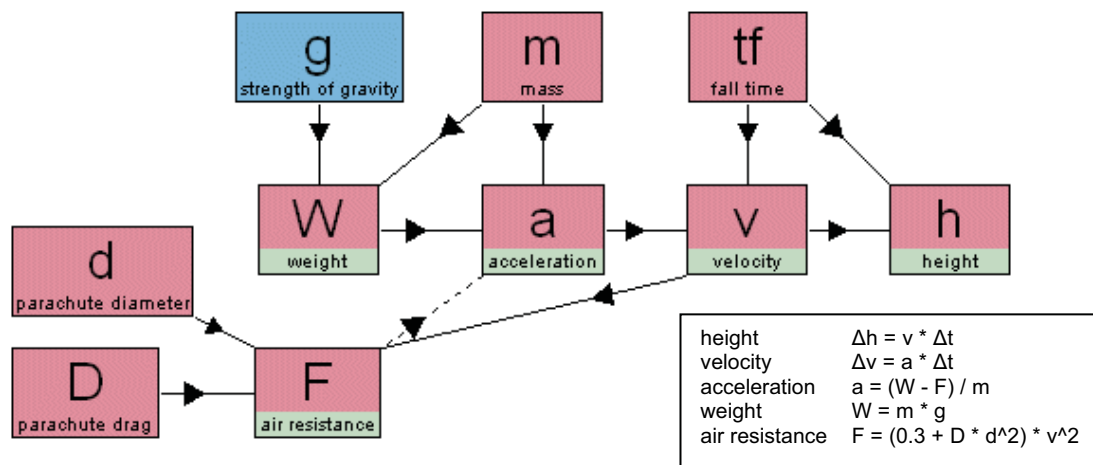


Figure 8. Model for parachute motion

The freedom offered to pupils when presented with modeling software in a content-free state can be extremely daunting. Designing and building a model from scratch probably poses the most sophisticated demands on pupils and is a skill to be aspired to rather than to be exercised as a first activity. Newton and Rogers (2001, pages 89-93) have described a hierarchy of task levels in the use of spreadsheet software, which may be readily adapted to modeling software thus:

Task Level 1: Exploring an existing model.

Pupils are presented with a previously prepared model and activities focus on studying patterns and relationships in the data generated by the model. Familiarization with a range of data analyzing tools is required. Activities should build confidence in obtaining useful data from the model and interpreting their significance.

Task Level 2: Modifying a model

The additional demand at this level is to understand more of how the model works and to edit some of its components or add new components. The purpose of activities is to make the model behave differently to yield new or different data which more accurately reflect observations of the phenomenon in real life.

Task Level 3: Designing and building a model

This is the most sophisticated level of use, requiring pupils to identify variables and define relationships to replicate the behavior of a physical system. It demands all the skills exercised in the previous levels.

Simulation	Is the parachute big enough? – vary mass of parachutist How long does it take to reach the ground? – vary diameter of parachute When is it safe to open the parachute? – vary delay before opening
Graph	Describe changes of speed during the fall. Make suitable measurements. Compare the graph of velocity with the graph of air resistance.
Model	Examine the model and justify each relationship between variables. Alter the model to take account of thermal currents, changes of air density, wind, changes to the parachute shape etc.

Reflecting on the example tasks featured in the case studies, we can identify examples of task levels 1 and 2. If we also consider how activities with simulations fit into the hierarchy of tasks, it can be argued that simulations offer an even greater degree of scaffolding than Task Level 1. Not only is the model pre-defined, a simulation adds more support by limiting the scope of investigation, focusing attention on significant variables, selecting appropriate tools for display and analysis of the data produced. Taken with the graphical user interface, the simulation presents more user-friendly access than Task Level 1 such that it may be assigned Task Level 0. Thus simulations can provide an entry level for what ultimately can develop into an autonomous command of modeling tools at Task Level 3.

Conclusion

The case has been made for considering simulation software as a suitable entry point for engagement in the world of scientific modeling. The use of graphics and images in simulations reinforces context and enhances motivation. The guidance implicit in the pre-selection of variables for control and display and analysis options helps to minimize ‘inauthentic’ labor associated with the tasks (Wellington, 2005) and to maintain an emphasis on appropriate scientific thinking. Simulations vary in the amount of freedom ceded to pupils to make choices and fashion their own investigation (Osborne and Hennessy, 2003) but access to the model

which drives the simulation is usually not allowed. It was to facilitate this transfer of thinking activity to the model itself that *Simulation Insight* was conceived. It has blurred the closed/open distinction between simulations and modeling. The program has also attempted to break new ground in providing a non-algebraic approach to developing thinking about relationships.

As is often the case with ICT applications, there is still much for the teacher to determine, in particular the schedule of activities involving the scripting, sequencing and grading of tasks.

References

- Gilbert, J.K. and Boulter, C.J. (2000) *Developing Models in Science Education*. Dordrecht: Kluwer.
- Lawrence, I. (2005) 'Modeling – a core activity', in Sang, D. and Frost, R. (eds.) *Teaching Secondary Science Using ICT*. London: Hodder Murray.
- Logotron (2005) *Simulation Insight* (software and curriculum materials). Cambridge: Logotron (www.logo.com)
- Newton, L. and Rogers, L.T. (2001) *Teaching Science with ICT*, London: Continuum.
- Osborne, J. and Hennessy, S. (2003) *Literature Review in Science Education and the Role of ICT: Promise, Problems and Future Directions*. Bristol: NESTA Futurelab. Page 30.
- Rogers, L.T. and Finlayson, H. (2003) 'Does ICT in science work in the classroom? Part 1, The individual teacher experience' *School Science Review* (2003) 84 (309) 105-111
- Rogers, L.T. and Finlayson, H. (2004) 'Developing successful pedagogy with ICT – How are science teachers meeting the challenge?' *Technology, Pedagogy and Education* 13, 3: 287-305.
- Webb, M. (1993) 'Computer-based modeling in school science', *School Science Review*, 74(269): 33-47.
- Wellington, J.J. (2004) 'Multimedia in Science Teaching' in Barton, R. (ed.) *Teaching Secondary Science with ICT*. Maidenhead: Open University Press.