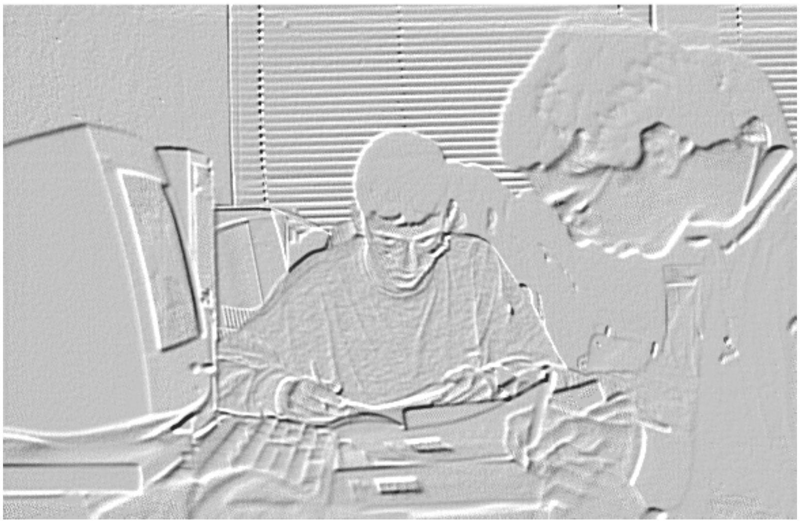
Modellus: Learning Physics with Mathematical Modelling

Vitor Duarte Teodoro

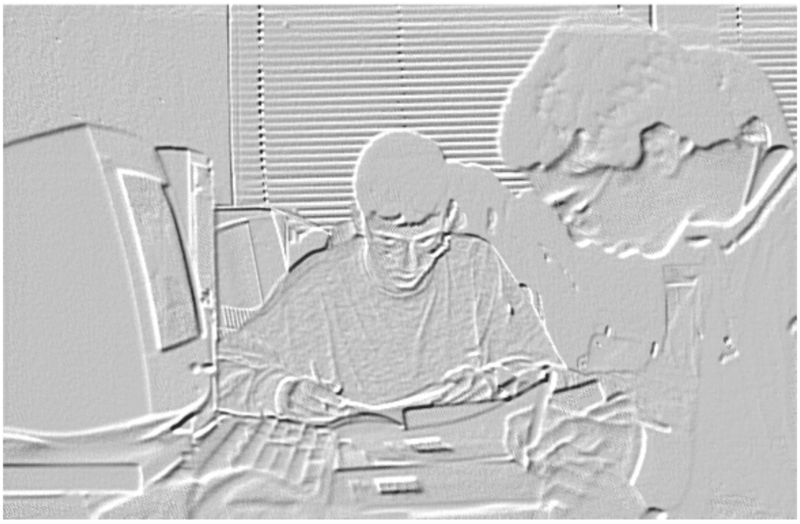


Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
2002

Modellus: Learning Physics with Mathematical Modelling

Vitor Duarte Teodoro

Dissertação apresentada para obtenção do grau de Doutor em Ciências
da Educação – especialidade de Teoria Curricular e Ensino das Ciências,
sob orientação conjunta da Professora Doutora Maria Odete Valente e
do Professor Doutor Cândido Marciano da Silva



Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
2002

Table of Contents

Chapter 3 Modellus: A Tool for Doing Experiments with Mathematical Objects [87]

Chapter 4 Modelling and Modellus in the Physics Curriculum: Some Examples and a Framework [127]

List of Figures

List of Tables

Abstract

Computers are now a major tool in research and development in almost all scientific and technological fields. Despite recent developments, this is far from true for learning environments in schools and most undergraduate studies.

This thesis proposes a *framework* for designing curricula where computers, and computer modelling in particular, are a major tool for learning. The framework, based on research on learning science and mathematics and on computer user interface, assumes that: 1) learning is an active process of creating meaning from representations; 2) learning takes place in a community of practice where students learn both from their own effort and from external guidance; 3) learning is a process of becoming familiar with concepts, with links between concepts, and with representations; 4) direct manipulation user interfaces allow students to explore concrete-abstract objects such as those of physics and can be used by students with minimal computer knowledge.

Physics is the science of constructing models and explanations about the physical world. And mathematical models are an important type of models that are difficult for many students. These difficulties can be rooted in the fact that most students do not have an environment where they can explore functions, differential equations and iterations as primary objects that model physical phenomena—as objects-to-think-with, reifying the formal objects of physics.

The framework proposes that students should be introduced to modelling in a very early stage of learning physics and mathematics, two scientific areas that must be taught in very closely related way, as they were developed since Galileo and Newton until the beginning of our century, before the rise of overspecialisation in science. At an early stage, functions are the main type of objects used to model real phenomena, such as motions. At a later stage, rates of change and equations with rates of change play an important role. This type of equations—differential equations—are the most important mathematical objects used for modelling Natural phenomena. In traditional approaches, they are introduced only at advanced level, because it takes a long time for students to be introduced to the fundamental principles of Calculus. With the new proposed approach, rates of change can be introduced also at early stages on learning if teachers stress semi-quantitative reasoning and use adequate computer tools.

In this thesis, there is also presented Modellus, a computer tool for modelling and experimentation. This computer tool has a user interface that allows students to start doing meaningful conceptual and empirical experiments without the need to learn new syntax, as is usual with established tools. The different steps in the process of constructing and exploring models can be done with Modellus, both from physical points of view and from mathematical points of view. Modellus activities show how mathematics and physics have a unity that is very difficult to see with traditional approaches. Mathematical models are treated as concrete-abstract objects: concrete in

the sense that they can be manipulated directly with a computer and abstract in the sense that they are representations of relations between variables.

Data gathered from two case studies, one with secondary school students and another with first year undergraduate students support the main ideas of the thesis. Also data gathered from teachers (from college and secondary schools), mainly through an email structured questionnaire, shows that teachers agree on the potential of modelling in the learning of physics (and mathematics) and of the most important aspects of the proposed framework to integrate modelling as an essential component of the curriculum.

Schools, as all institutions, change at a very slow rate. There are a multitude of reasons for this. And traditional curricula, where the emphasis is on rote learning of facts, can only be changed if schools have access to new and powerful views of learning *and* to new tools, that support meaningful conceptual learning and are as common and easy to use as pencil and paper.

Resumo

Os computadores são actualmente uma ferramenta essencial na investigação e no desenvolvimento em quase todos os campos científicos e tecnológicos. Apesar dos desenvolvimentos recentes, este facto está longe de ser verdadeiro para os ambientes de aprendizagem nas escolas secundárias e para a maioria dos estudantes universitários, nomeadamente nos anos iniciais dos respectivos cursos.

Esta tese propõe uma nova perspectiva na aprendizagem e na definição dos currículos de física, em que o computador e, em particular, o uso do computador como ferramenta de modelação, é considerado como uma ferramenta chave no processo de aprendizagem. Esta nova perspectiva, fundamentada na investigação sobre a aprendizagem das ciências e da matemática, e na investigação em interfaces entre o computador e o utilizador, assume que: (1) a aprendizagem é um processo activo de criação de significados a partir de representações; (2) a aprendizagem decorre numa comunidade de prática em que os estudantes aprendem a partir do seu próprio esforço e a partir de orientação externa; (3) a aprendizagem é um processo de familiarização com conceitos, com ligações entre conceitos e com representações; (4) os interfaces baseados na manipulação directa permitem aos estudantes explorar conceitos concreto-abstractos, como é o caso dos conceitos físicos, mesmo quando possuem uma competência reduzida na utilização de computadores.

A física é a ciência que constrói modelos e explicações acerca do universo físico. Os modelos matemáticos são um tipo importante de modelos, de difícil aprendizagem para muitos estudantes. A origem destas dificuldades pode ser identificada no facto de muitos estudantes não disporem de ambientes computacionais em que possam explorar funções, equações diferenciais e iterações como objectos que podem ser utilizados na construção de modelos de fenómenos físicos—como objectos-para-pensar-com, tornando concretos os objectos formais utilizados pela física.

Nesta tese, argumenta-se que os alunos devem começar a utilizar modelação matemática no início da aprendizagem da física e da matemática, duas áreas científicas que, argumenta-se também, devem ser ensinadas de um modo integrado, tal como foram inicialmente desenvolvidas desde Galileu e Netwon—antes do aparecimento da elevada especialização na ciência, iniciada no princípio do século XX. Numa fase inicial, a modelação dos fenómenos físicos, como por exemplo, o movimento, é feita com funções. Numa fase posterior, os modelos são construídos recorrendo a taxas de variação e a equações com taxas instantâneas de variação. Este tipo de equações—equações diferenciais—são o tipo de objectos matemáticos mais importantes para modelar os fenómenos físicos. Nos currículos tradicionais, os modelos com equações diferenciais são abordados apenas em níveis avançados porque os alunos necessitam de ser previamente introduzidos nos princípios fundamentais do Cálculo. Nesta nova perspectiva, podem ser abordados em níveis mais elementares, se se reforçarem os aspectos semi-quantitativos e se se utilizarem ferramentas computacionais adequadas.

Nesta tese, descreve-se igualmente o software Modellus, uma nova ferramenta computacional para modelação e experimentação. Este software tem um interface com o utilizador suficientemente simples que permite aos estudantes a realização de experiências conceptuais e empíricas, sem a aprendizagem de sintaxes específicas, tal como acontece com outros programas, comummente utilizados. Os diferentes passos de construção e exploração de modelos com o Modellus podem ser feitos quer a partir de dados e registos físicos, de experiências reais, quer apenas a partir de um ponto de vista exclusivamente matemático. As actividades com o Modellus evidenciam a unidade da matemática e da física, algo que é muito difícil de evidenciar nas abordagens tradicionais. Os modelos matemáticos são tratados como objectos concreto-abstractos: concretos no sentido que podem ser manipulados directamente com um computador e abstractos no sentido em que são representações de relações entre variáveis.

A evidência obtida com dois estudos de caso, um com estudantes do ensino secundário e outro com estudantes do primeiro ano do ensino superior, suportam as ideias principais defendidas nesta tese. Dados obtidos de um conjunto de professores (de escolas secundárias e do ensino superior), essencialmente através de um questionário estruturado, administrado por correio electrónico, mostram que os professores estão de acordo sobre a importância da modelação na aprendizagem da física (e da matemática) e com os aspectos mais importantes das propostas sobre integração da modelação como uma componente essencial do currículo.

As escolas, como todas as instituições, mudam muito lentamente. Há muitas e variadas razões para tal facto. Os currículos tradicionais, em que a ênfase está na aprendizagem mecânica e na aprendizagem de factos, apenas podem ser modificados se as escolas tiverem acesso a novas e poderosas visões sobre a aprendizagem *e* a novas ferramentas, que suportem a aprendizagem conceptual significativa e que sejam tão comuns e fáceis de utilizar como o papel e o lápis.

Résumé

Les ordinateurs son actuellement un outil essentiel pour la recherche et le développement en presque toutes les champs scientifiques et technologiques. Malgré les développement récents, ce fait est loin d'être vrai pour les environnements d'apprentissage des écoles secondaires et pour la plupart des étudiantes universitaires, nommément pour les étudiantes des premières années.

Cette thèse propose une nouvelle perspective pour l'apprentissage et pour la définition des curricula de physique, où l'ordinateur et les outils de modélisation sont envisagés comme des outils essentiels pour le procès d'apprentissage. Cette nouvelle perspective, fondé sur la recherche en didactique des sciences et de la mathématique, et sur la recherche en interaction utilisateur-ordinateur, admette que : (1) l'apprentissage est un procès actif de création de signifies à partir de représentation ; (2) l'apprentissage s' écoule dans une communauté de practice où les étudiantes apprennent à partir de sont propre effort et de orientation externe ; (3) l'apprentissage est un procès de familiarisation avec des concepts, des liaison entre concepts et des représentations ; (4) les interfaces fondés sur la manipulation directe rendre possible aux étudiantes explorer des concepts concret-abstraits, comme les concepts physiques, même quand lés étudiantes ne sont pas des experts avec des ordinateurs.

La physique est la science qui crie des modèles et des explications sur l'univers physique. Les modèles mathématiques sont un genre important de modèles, d'apprentissage difficile. L'origine de cettes difficultés se peut identifier dans le fait de beaucoup d'étudiantes n'utilisent pas des ordinateurs pour explorer des fonctions, des équations différentielles et des itérations, comme des objectes qui peut être utilisés pour la construction de modèles de phénomènes physiques—comme des objecte-pour-penser-avec, rendent concrètes les objectes formelles utilisés en physique.

En cette thèse, on argument que les élèves doit commencer à utiliser la modélisation mathématique au début de sont apprentissage de physique et de mathématique, deux champs scientifiques qui, on argument aussi, doit être enseignés de manière intégré. Au début, la modélisation des phénomènes physiques comme, par exemple, le mouvement, est faite avec des fonctions. Postérieurement, les modèles son construit avec des taxes instantanés de variation—équations différentielles, le type d'objectes mathématiques plus importants pour la modélisation des phénomènes physiques. Dans les curricula traditionnelles, les modèles avec des équations différentielles sont étudiés seulement en niveaux avancés, parce que les étudiantes doivent préalablement étudier les principes fondamentaux du Calcul. Dans cette nouvelle perspective, cette type de modèles peut être étudier en niveaux plus élémentaires, si on renforce les aspectes semi quantitatifs y si ont utilisent des outils informatiques appropriés.

Dans cette thèse, ont décrit aussi le logiciel Modellus, un nouveau outil informatique pour la modélisation et la expérimentation. Ce logiciel a un interface assez simple pour permettre aux étudiantes la réalisations des expériences conceptuelles et empiriques, sans l'apprentissage de syntaxes spécifiques, comme es le cas d'autres logiciels. Les

modèles son construit soit a partir de donnés réelles soit a partir d'un point de vue exclusivement mathématique. Les activités avec Modellus rendre évident l'unité de la physique et de la mathématique, ce qui est difficile de faire avec des approches traditionnelles.

L'évidence obtenue avec deux études de cas, un avec des étudiantes de secondaires et l'autre avec des étudiantes de premier année de l'université, soutienne les idées principales soutenues dans cette thèse. Les réponses a un questionnaire faite a professeurs de secondaire et d'enseignement supérieure, administré par courrier électronique, indique que les enseignants sont d'accord avec l'importance de la modélisation dans l'apprentissage de la physique (et de la mathématique) et avec les propositions sur l'intégration de la modélisation dans les curricula. Les écoles change très lentement. L'accent sur l'apprentissage mécanique dans les curricula traditionnelles seulement peut être modifié si les écoles ont l'accès a nouvelles visions et nouvelles outils.

Chapter 0 Summary, in Portuguese (*Modellus: aprendizagem da Física e modelação matemática*)

0.1 Aprendizagem, software educativo e a natureza da Ciência

O discurso dominante na educação neste início de um novo milénio reforça frequentemente a necessidade da utilização das tecnologias da informação, nomeadamente a Internet e software educacional de qualidade. A maioria dos países desenvolvidos e em desenvolvimento tem programas específicos de intervenção que visam promover essa utilização, quer em contextos extra sala de aula quer em contextos de ensino na sala de aula.

O debate existente desde o início da utilização de computadores na educação, na década de 1960, sobre que papel deve o computador desempenhar no ensino tem revestido diversas formas. No essencial, confrontam-se duas perspectivas: uma que reforça o computador como *máquina de fornecer informação* e outra como *ferramenta para auxiliar a construção de conhecimento* (De Corte, Verschaffel, & Lowyck, 1998; Taylor, 1980). A perspectiva da "máquina de informação" tem sido concretizada através de diversas formas, desde os programas de ensino tutorial até à maioria dos actuais produtos multimédia ou páginas de Internet. A segunda perspectiva, de que Papert (Papert, 1980) foi e é um expoente, concebe o computador como

> (...) object-to-think-with, object in which there is an intersection of cultural presence, embedded knowledge, and possibility for personal identification (p. 11).

É esta perspectiva que é dominante em grande parte dos discursos dos educadores mais informados sobre a investigação na utilização de novas tecnologias na educação (veja-se, por exemplo, o programa de Matemática do Ensino Secundário em Portugal, os *Principles and Standards for School Mathematics* nos EUA, as *Benchmarks for Science Literacy*, da *American Association for the Advancement of Science*), em contraste com os discursos de políticos, responsáveis pela administração da educação e gestores dos seus programas (por exemplo, os programas portugueses de introdução das tecnologias na educação têm nos últimos anos destinado muito pouca atenção à utilização de software educativo como ferramenta de aprendizagem—o que é patente no tipo de acções de formação promovidas na maioria dos Centros de Competência do Programa Nónio), que tendem, actualmente, a privilegiar a Internet como suporte de ensino, tal como privilegiaram em anos recentes o uso de tutoriais multimédia e ensino assistido e gerido por computador. O mesmo sucede, aliás, com os editores e distribuidores de software educativo: uma observação dos títulos disponíveis mostra a quase inexistência de oferta de software de tipo ferramenta. Apenas publicam ou distribuem obras de referência e títulos multimédia, que na sua maioria combinam apresentações electrónicas de conteúdos com jogos sobre esses conteúdos.

Usar o computador como ferramenta intelectual, como artefacto cognitivo (Norman, 1991), é coerente com o que sabe sobre o modo como se aprende. Note-se que, de acordo com Olson (1974), tecnologia, cognição e inteligência estão profundamente associados:

> Almost any form of human cognition requires one to deal productively and imaginatively with some technology. To attempt to characterize intelligence independently of those technologies seems to be a fundamental error (p. 356).

Se bem que seja difícil existir acordo total sobre a natureza do processo de aprendizagem, alguns princípios emergem como consensuais (Resnick & Collins, 1998). Entre estes princípios está a natureza construtivista da aprendizagem: aprende-se construindo relações e significados. Mas, de acordo com Resnick e Collins, *didactic teaching*, por contraposição à *descoberta* e *invenção*, é essencial para a aprendizagem:

> For many years, particularly under the influence of Piagetian interpretations of cognitive development, constructivism was taken to mean that there should be no ‘didactic’ teaching. Instead it was proposed that educators should arrange rich exploratory environments for children. In such environments, students would discover or invent knowledge for themselves. It is now known that arranging for students to construct their own knowledge is a far more complex matter, filled with challenges that derive from the nature of expertise and learning (Resnick & Collins, 1998).

Ainda de acordo com estes autores, são quatro as características básicas do processo de aprendizagem:

1. *O pensamento e a aprendizagem dependem do conhecimento prévio*. Quanto melhor organizado estiver o conhecimento prévio mais provável se torna a aprendizagem de novos conhecimentos e capacidades.
2. *Os bons aprendizes são bons construtores de conhecimento estratégico*. O conhecimento em contextos específicos tem influência determinante, mais relevante que a simples aprendizagem descontextualizada de “general learning skills”.
3. *“Os ricos ficam mais ricos”*. Os aprendizes mais ricos em conhecimento e estratégias de aprendizagem valorizadas pela escola tendem a beneficiar de novas oportunidades de conhecimento.

4 *A construção de conhecimento demora muito tempo*, muito mais tempo do que é normalmente suposto.

Aprender não é, no entanto, apenas, um processo de *construção pessoal*. Na tradição de Vygotsky (1978), Resnick e Collins reforçam ainda a importância das *interacções sociais* e das *ferramentas* no processo de construção de conhecimento:

> (…) cognition is assumed to be shared both with other individuals and with tools and artifacts. This means that thinking is situated in a particular context of intentions, social partners, and tools.

É esta visão sobre o processo de aprendizagem que fundamenta a utilização de software exploratório, como o *Modellus* (Knowledge Revolution, 1997; Teodoro, Vieira, & Clérigo, 2000). *Modellus* é uma ferramenta cognitiva para auxiliar a internalização de conhecimento simbólico, preferencialmente em contexto de actividades de grupo e de classe, em que a discussão, a conjectura e o teste de ideias são actividades dominantes, por oposição ao ensino directo por parte do professor. Isto não significa, no entanto, que os alunos reinventam o conhecimento quando constroem ou exploram modelos com o *Modellus*. De facto, *ninguém pode aprender explorando sem conhecimento relevante sobre o campo de exploração*. A aquisição de conhecimentos e capacidades não é um processo completamente claro e definido no tempo e no espaço. É demorado, contextual, dependente de estruturas cognitivas e conhecimento prévio. E, essencialmente, é um processo de *familiarização* com novas ideias e representações (como afirmaram muitos dos mais notáveis criadores científicos, como Newton, Planck, Feynman).

O conhecimento científico é um conhecimento limitado sobre a *natureza das coisas*. É, fundamentalmente, uma *representação* da realidade. A Ciência discute as representações das coisas, não as coisas em si (Giere, 1989). Por exemplo, Richard Feynman, um dos mais famosos físicos do século XX, considera irrelevante a discussão sobre a *natureza* das forças (para Feynman, conhecemos os modelos das interacções ou forças, não a *natureza* das interacções). Num dos seus mais famosos livros, escreve:

> While Kepler was discovering these laws [as leis da gravitação de Kepler], Galileo was studying the laws of motion. The problem was, what makes the planets go around? (In those days, one of the theories proposed was that the planets went around because behind them were invisible angels, heating their wings and driving the planets forward. You will see that this theory is now modified! It turns out that in order to keep the planets going around, the invisible angels must fly in a different direction and they have no wings. Otherwise, it is a somewhat similar theory!) (Feynman, Leighton, & Sands, 1963, p. 7-2)

Modellus, como outras ferramentas computacionais, permite ao utilizador fazer e refazer *representações*, *explorando-as sobre as mais diversas perspectivas*. Deste modo, facilita a *familiarização* com essas representações, criando de certo modo uma *intimidade entre aprendiz e representação*, intimidade essa que muito dificilmente resulta da simples *observação ocasional* de equações e representações feitas pelo professor ou apresentadas nos livros. Essa *intimidade*, por outro lado, é fundamental para a *reificação* dos objectos formais, algo que, de acordo com Roitman (1998), é imprescindível no desenvolvimento do pensamento científico.

As ferramentas computacionais revolucionaram e estão ainda a revolucionar muitas actividades humanas. Foi na Ciência e na Tecnologia que, provavelmente, essa revolução foi mais significativa. Desde a Física à Biologia Molecular, a utilização de computadores tornou-se fundamental na produção de conhecimento científico. Essa importância é tão relevante, que um relatório do *National Research Council* (EUA) afirma que se pode considerar que há mais uma nova metodologia científica:

> Scientific computation has become so much a part of everyday experience of scientific and engineering practice that it can be considered a third fundamental methodology of science parallel to the more established paradigms of experimental and theoretical science.
> (National Research Council, 1989, p. 36)

A influência das novas ferramentas computacionais na educação é menos evidente mas é muito provável que essa situação seja alterada nas próximas décadas. Mas a presença de tecnologia não garante por si só a mudança dos processos de ensino e de aprendizagem: só o uso *adequado* da tecnologia pode auxiliar essa mudança (Bransford, Brown, & Cocking, 2000). Claro que essa mudança não depende apenas da qualidade da tecnologia, é uma questão global, sistémica, complexa. Começa, necessariamente, com uma visão do processo de ensino e aprendizagem coerente com a acima sintetizada. E uma prática coerente com uma visão desse tipo só é possível com "ferramentas poderosas", no sentido dado por De Corte (1989). Note-se, no entanto, que é útil distinguir entre características *inerentes* às ferramentas e as características *impostas* (Pryluck, 1968). As características *inerentes* têm a ver com a natureza da tecnologia. As características *impostas* referem-se às situações didácticas em que as ferramentas são utilizadas. As características inerentes são de importância evidente (é por isso que o uso de computadores se tornou tão ubíquo nos últimos anos). Mas apenas as características *impostas* são susceptíveis de investigação didáctica. Reflectir sobre usar computadores no ensino é, pois, uma discussão sobre *como usar*, sobre quais as melhores características *impostas*.

De acordo com Collins (1991), a utilização de computadores na educação pode auxiliar as seguintes mudanças:

1. Do ensino dirigido para a turma para o ensino dirigido para pequenos grupos.
2. Do ensino magistral, expositivo, para um ensino assistido.
3. Do ensino dirigido aos melhores alunos para um ensino dirigido aos alunos com mais dificuldades de aprendizagem.
4. Do ensino dirigido para alunos pouco empenhados para um ensino dirigido a alunos empenhados na sua própria aprendizagem.
5. De uma avaliação baseada em testes para uma avaliação baseada em produtos, progresso e esforço.
6. De estruturas sociais competitivas para estruturas sociais co-operativas.
7. De um ensino em que todos os alunos aprendem o mesmo para um ensino em que diferentes alunos podem aprender diferentes coisas.
8. Da primazia da aprendizagem verbal para uma integração entre o pensamento visual e verbal.

Esta lista refere-se, claramente, às características *impostas*. Daí, pois, a enorme importância que tem a natureza das situações didácticas criadas com as ferramentas computacionais. Não é possível esperar que estas características *impostas* sejam apenas função das ferramentas, por mais “poderosas” que possam ser. São, também, função da cultura, dos contextos didácticos, e da prática da aprendizagem.

0.2 *Modellus*: uma ferramenta cognitiva

Uma boa parte do trabalho científico pode ser considerado como fazendo parte de “ciclos de modelação” (American Association for the Advancement of Science, 1993). Um “ciclo de modelação” envolve (não necessariamente em sequência ordenada):

1. O uso de abstracções para representar objectos ou ideias.
2. A manipulação das abstracções de acordo como certas regras lógicas.
3. A verificação do grau de acordo dos resultados obtidos nessa manipulação com as ideias e abstracções iniciais.

Antes do uso de computadores, construir e explorar modelos era um processo complexo e muito exigente em capacidade de abstracção. Por exemplo, a simples representação de um movimento acelerado por uma função do segundo grau, exige do aprendiz a utilização de uma função do tipo $x = \frac{1}{2}a_x t^2$. Um sistema de modelação permite ao aprendiz utilizar esta função, num contexto específico, e explorar facilmente o significado dos parâmetros e da função em si.

Papert foi um dos primeiros autores a conjecturar sobre a importância de concretizar o “formal” num computador:

> Stated most simply, my conjecture is that the computer can concretize (and personalize) the formal. Seen in this light, it is not just another powerful educational tool. It is unique in providing us with the means for addressing what Piaget and many others see as the obstacle which is overcome in the passage from child to adult thinking. I believe that it can allow us to shift the boundary separating concrete and formal. Knowledge that is accessible only through formal processes can now be approached concretely. And the real magic comes from the fact that this knowledge includes those elements one needs to become a formal thinker (Papert, 1980, p. 21).

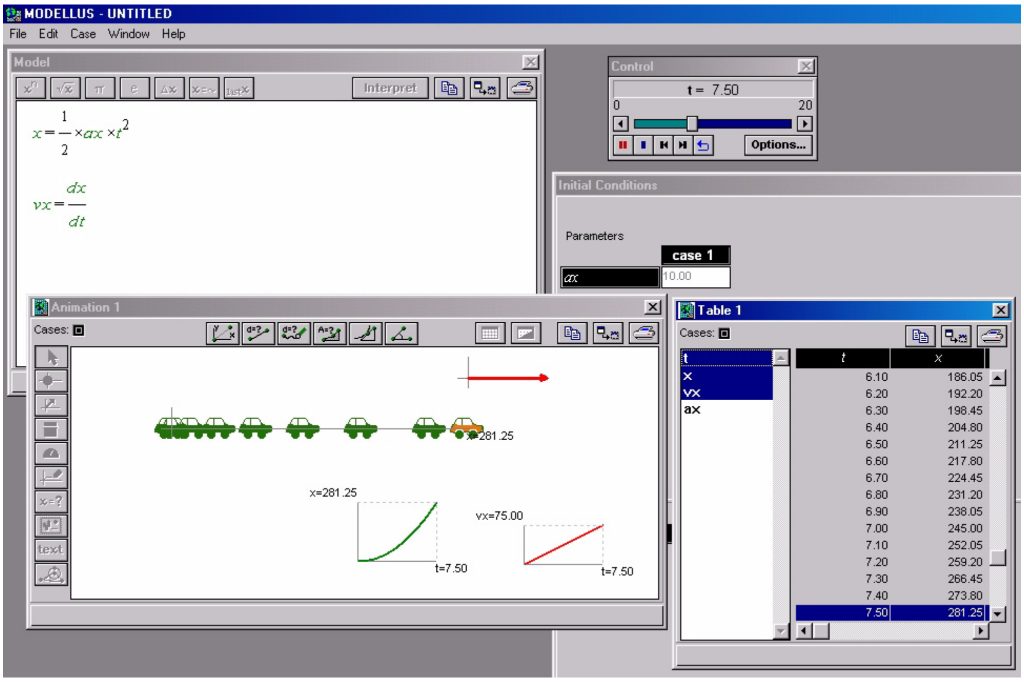
A importância da concretização do “formal”, sem perder a ideia da relevância do “abstracto” na construção do conhecimento científico, tem sido retomada por vários outros autores. Por exemplo, Hebenstreit (1987) considera que uma das principais potencialidades do computador é permitir a construção de objectos “concreto-abstractos”: *concretos* porque podem ser directamente manipulados no computador, *abstractos* porque são representações de ideias ou relações.

A Figura 0.1 mostra um exemplo da exploração de um movimento acelerado no *Modellus*. Na janela “Model” estão indicadas as equações do modelo do movimento. A posição x é definida explicitamente como função de t, e v_x (a componente da velocidade segundo o eixo Ox) é definida como a derivada da posição em ordem ao tempo. Ao parâmetro a_x, a componente da aceleração no eixo Ox, é atribuído o valor 10 unidades (m/s^2, no sistema internacional de unidades), na janela “Initial Conditions”. Uma vez construído o modelo, construiu-se uma “Animação” e uma “Tabela”. Na tabela, representam-se os valores das diversas variáveis. Na animação, representam-se um objecto a acelerar de acordo com os valores de *x* (representação estroboscópica, isto é, representação da posição em diversos instantes separados por iguais intervalos de tempo), o vector velocidade, acompanhando o movimento do objecto, e dois gráficos, um da posição e outro da velocidade.

Este exemplo mostra uma das mais importantes características de um programa de modelação: a possibilidade de construir *múltiplas representações* da mesma situação. A importância da observação e “navegação” entre múltiplas representações tem sido assinalada pelo menos desde 1988 (Harvard Educational Technology Center, 1988). De certo modo, compreender um modelo e o respectivo fenómeno é ser capaz de construir múltiplas representações desse fenómeno e “navegar” de uma para a outra representação.

Há alguns anos, Nickerson (1995) assinalou que os investigadores que estudam a utilização de computadores na educação, não davam suficiente atenção à criação de programas onde os alunos pudessem construir simulações. Nickerson refere ainda que criar programas que permitam aos alunos desenvolver simulações *é difícil, mas não impossível*. Antes da criação de ambientes de modelação, alguns autores, como Papert, por exemplo, tinham proposto o uso de linguagens de programação, nomeadamente linguagens especialmente adequadas para uso por crianças e jovens (como a linguagem *Logo*). Mas, para Nickerson:

> For student-developed simulations to be practical for educational purposes, it will probably be necessary to develop tools that are designated to facilitate the building of simulations by people without such language facility and programming experience (idem, p. 16).



$$x = \frac{1}{2} \times ax \times t^2$$

$$vx = \frac{dx}{dt}$$

t = 7.50

Parameters

	case 1
ax	10.00

t	x
6.10	186.05
6.20	192.20
6.30	198.45
6.40	204.80
6.50	211.25
6.60	217.80
6.70	224.45
6.80	231.20
6.90	238.05
7.00	245.00
7.10	252.05
7.20	259.20
7.30	266.45
7.40	273.80
7.50	281.25

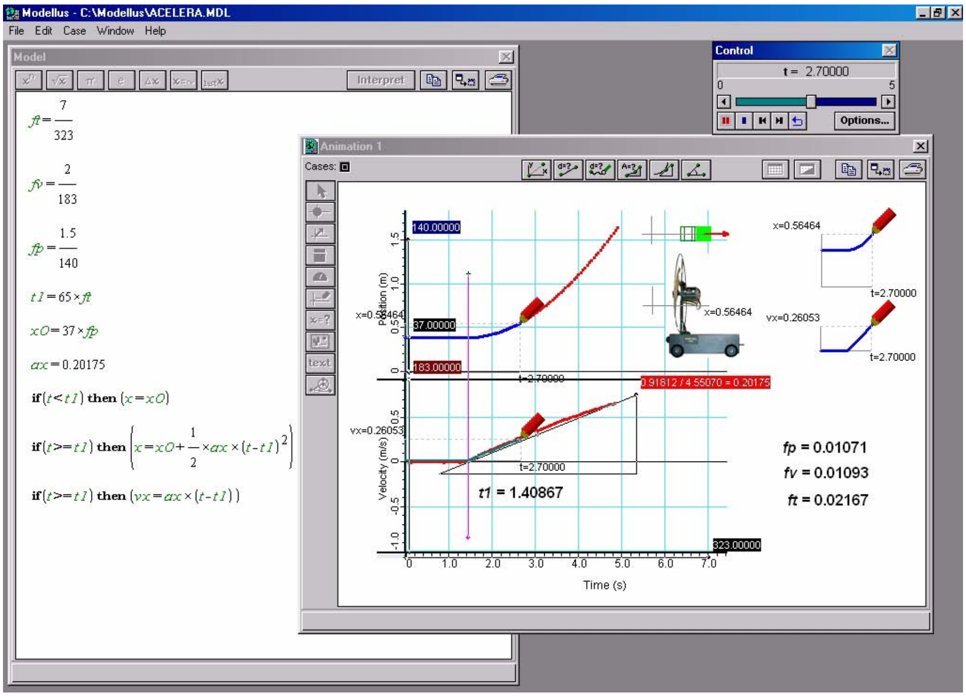
Figura 0.1 Explorando múltiplas representações (equações, tabelas, gráficos, trajectórias, estroboscopia, vectores) de um movimento acelerado no *Modellus*.

Esta foi a opção adoptada no design do *Modellus*: a construção de modelos é feita tão próxima quanto possível do modo como se constrói e se utiliza um modelo sem computador. Deste modo, procura-se que o utilizador, aluno e professor, *pense com o computador como pensaria se estivesse usando papel e lápis*. Claro que esta opção só é possível porque, hoje em dia, os ambientes gráficos e de manipulação directa (Shneiderman, 1983) são praticamente de conhecimento intuitivo para os utilizadores de computador. Apesar da evolução dos interfaces computacionais e do aparecimento de novas ferramentas computacionais, alguns autores ainda continuam a propor que os ambientes de programação são os únicos suficientemente poderosos e com as mais profundas implicações no processo de construção de conhecimento (e.g., diSessa, 2000). Esta perspectiva não tem, no entanto, qualquer influência nos currículos actuais, ao contrário do que sucedeu no início da década de 80 (Hoyles, 1995).

Um modelo no *Modellus* pode ser construído a partir de equações gerais, como no caso da Figura 0.1, ou a partir da análise de dados experimentais, da análise de fotografias, ou da análise de vídeos. Nas Figuras 0.2 e 0.3 mostram-se dois modelos (para uma descrição mais completa destes exemplos, ver o Manual do *Modellus*):

1. O modelo da Figura 0.2 foi construído a partir da análise de dados experimentais referentes a um movimento acelerado, obtidos com sensores de posição.
2. O modelo da Figura 0.3 foi construído a partir da análise de um vídeo de um pêndulo.

Em ambos os casos, uma vez construído o modelo, comparou-se este com a situação experimental. Deste modo, avalia-se a *razoabilidade* do modelo.



$ft = \frac{7}{323}$

$fv = \frac{2}{183}$

$fp = \frac{1.5}{140}$

$t1 = 65 \times ft$

$x0 = 37 \times fp$

$ax = 0.20175$

$\mathbf{if}(t < t1)\ \mathbf{then}\ (x = x0)$

$\mathbf{if}(t >= t1)\ \mathbf{then}\ \left(x = x0 + \frac{1}{2} \times ax \times (t - t1)^2\right)$

$\mathbf{if}(t >= t1)\ \mathbf{then}\ (vx = ax \times (t - t1))$

t = 2.70000

$t1$ = 1.40867

fp = 0.01071

fv = 0.01093

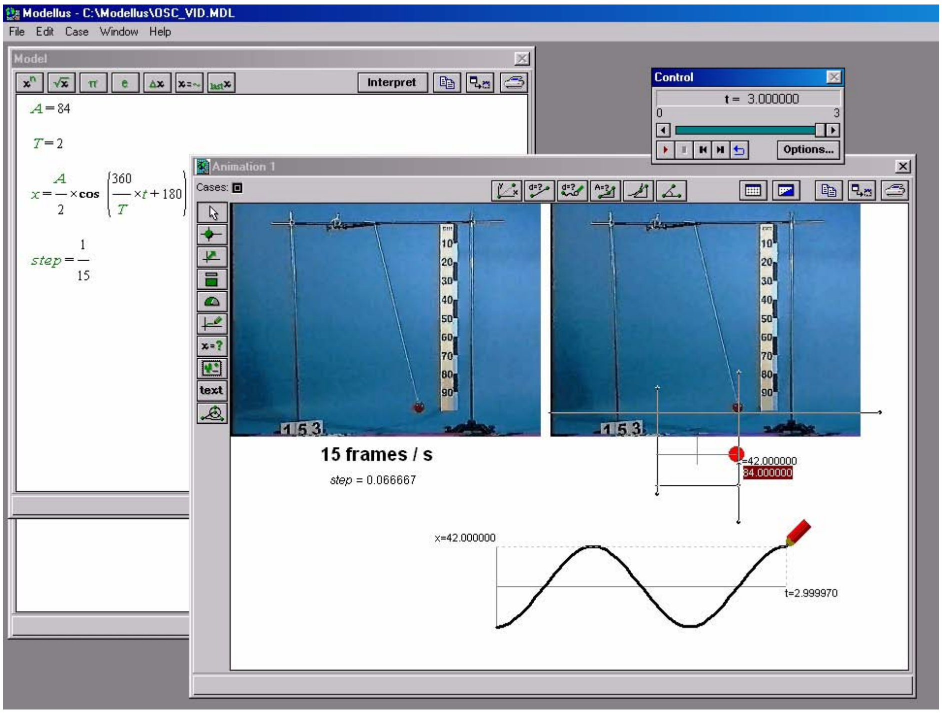
ft = 0.02167

Figura 0.2 Construção de um modelo a partir da análise de dados experimentais. O gráfico correspondente aos dados experimentais foi utilizado para obter informação para a construção do modelo. Uma vez construído o modelo, este foi comparado com os dados experimentais.

Os exemplos acima apresentados são relativamente simples e correspondem a temas dos currículos de Matemática e de Física do ensino secundário. Mas, como é conhecido dos professores, e verificado pela investigação educativa, uma grande maioria dos alunos tem muitas dificuldades na aprendizagem significativa destes temas (Pfundt & Duit, 1991). A hipótese chave que está na base do desenvolvimento do *Modellus* é, precisamente, que estas *dificuldades de aprendizagem* estão relacionadas com o facto do tratamento destes temas ser quase *exclusivamente formal*, envolvendo a *mecanização* da resolução de problemas rotineiros, *sem* contextos *experimentais*, *sem* exploração de representações *visuais*, *sem* exploração dos *modelos* matemáticos, *sem* ênfase em raciocínios do tipo "e o que se sucede se...".

Com o *Modellus* é, pois, possível:

1. Construir e explorar múltiplas representações de modelos matemáticos (baseados em funções, em equações diferenciais, em iterações, em objectos geométricos, etc.), a partir de especulação puramente teórica ou a partir de dados experimentais ou registos em imagem fixa ou em vídeo.
2. Analisar a razoabilidade dos modelos, quer em termos de coerência teórica quer em termos de coerência com dados experimentais ou registos de imagem.
3. Reforçar o pensamento visual, sem menorizar os aspectos de representação formal através de equações e outros processos formais.
4. Abordar de uma forma integrada os fenómenos naturais, ou simplesmente representações formais.



$A = 84$

$T = 2$

$$x = \frac{A}{2} \times \cos\left(\frac{360}{T} \times t + 180\right)$$

$$step = \frac{1}{15}$$

t = 3.000000

15 frames / s

step = 0.066667

=42.000000

84.000000

x=42.000000

t=2.999970

Figura 0.3 Construção de um modelo a partir de um vídeo. O modelo foi, em seguida, comparado com o movimento do pêndulo. O vídeo foi colocado no fundo da janela de "Animação". A imagem da direita é uma cópia da imagem da esquerda. Apenas se pode fazer medidas e colocar objectos sobrepostos na imagem da direita.

0.3 Currículo: uma visão interdisciplinar

Nos últimos quinze anos reacendeu-se em diversos países a discussão sobre a renovação do currículo das Ciências e da Matemática. Esta discussão, ainda em curso, está relacionada com o enorme corpo de conhecimentos obtido pela investigação educacional, em particular o resultante dos estudos relacionados com a identificação de “misconceptions” e as dificuldades de mudança conceptual. Enquanto as reformas nas décadas de 60 e de 70 privilegiaram o desenvolvimento curricular, envolvendo a elaboração de materiais para alunos e professores por cientistas da respectiva área científica em colaboração com professores criativos e experientes (Raizen, 1991), as reformas nas décadas de 80 e 90 enfatizaram a definição de padrões de qualidade na aprendizagem, procurando influenciar o desenvolvimento curricular e as práticas pedagógicas a todos os níveis, desde a formação inicial de professores ao uso de tecnologia e aos estilos de ensino (American Association for the Advancement of Science, 1989, 1993, 1998; National Council of Teachers of Mathematics, 1989, 2000; National Research Council, 1996; National Science Teachers Association, 1996). Ao contrário das reformas das décadas de 60 e 70, as reformas recentes deram mais importância às dificuldades de aprendizagem e de ensino (Eylon, 1998), naturalmente porque aumentou o corpo de conhecimentos sobre os processos de aprendizagem.

Uma outra característica das reformas recentes é a importância dada a uma visão mais integrada do currículo, desde a aprendizagem da comunicação escrita e oral, até à necessidade de aprendizagem em contextos interdisciplinares e às conexões entres as abordagens das diversas ciências. Por exemplo, os *Program and Standards for School Mathematics* estabelecem que:

> The opportunity for students to experience mathematics in a context is important. Mathematics is used in science, the social sciences, medicine and commerce. The link between mathematics and science is not only through content but also through process. The processes and content of science can inspire an approach to solving problems that applies to the study of mathematics (National Council of Teachers of Mathematics, 2000, p. 77).

Por sua vez, nos *National Science Education Standards* afirma-se que:

> If teachers of mathematics use scientific examples and methods, understanding in both disciplines will be enhanced. For mathematics, coordination reinforces the perspective of investigation and experimentation that is emphasized in the National Council of Teachers of Mathematics (NCTM) standards (National Research Council, 1996, p. 218).

Em Portugal, os programas de Matemática tendem também a reforçar a importância do uso da Matemática em contexto, inclusivamente em contextos experimentais, de modo semelhante às ciências físicas:

> (...) as aprendizagens significativas em Matemática não podem excluir características típicas do ensino experimental, sendo que as competências adquiridas por via da Matemática devem contribuir para alicerçar conhecimentos e formas de pensar sobre a ciência experimental (Departamento de Ensino Secundário, 2002, p. 2).

A necessidade desta visão interdisciplinar, em que cada disciplina reforça explicitamente conexões com outras disciplinas, já tinha sido identificada há muito tempo, em muitos países. Por exemplo, Rómulo de Carvalho escreveu, há mais de cinquenta anos:

> A lição dos factos é, pois, esta: o programa de Matemática não pode ser gizado num compartimento e o da Física em outro à parte. Nem aqui nem em qualquer grau de ensino, evidentemente (Carvalho, 1947, p. 12).

Uma perspectiva semelhante é apresentada por Sebastião e Silva no seu inovador *Compêndio de Matemática*, um dos poucos esforços de renovação curricular, nunca continuado, realizado em Portugal na década de 60:

> Entre os exercícios que podem ter mais interesse, figuram aqueles que se referem a *situações reais, concretas*. O nosso ensino (...) peca também por ausência de contacto com o húmus da intuição e com a realidade concreta. Ora, um dos pontos assentes em reuniões internacionais de professores (...) é que o professor de Matemática deve ser, primeiro que tudo, um professor de *matematização*, isto é, deve habituar o aluno a reduzir situações concretas a modelos matemáticos e, vice-versa, aplicar os esquemas lógicos da matemática a problemas concretos (Silva, 1975, p. 12-13).

A importância da conexão entre as ciências físicas e a matemática recebeu recentemente um novo impulso com a introdução do currículo *Advancing Physics*, desenvolvido pelo *Institute of Physics* no Reino Unido, em que o *Modellus* é considerado como uma ferramenta “integral to the course” (Lawrence & Whitehouse, 2000). Esse curso é, provavelmente, o primeiro curso de ensino não superior em que a utilização de ferramentas computacionais—nomeadamente o *Modellus* e a folha de cálculo—desempenha um papel essencial, quer para a exemplificação de situações quer para a aprendizagem da construção de modelos. Por exemplo, no início do segundo ano do curso (o último ano do ensino secundário) o aluno constrói modelos utilizando funções e equações diferenciais. A importância de uma abordagem integrada com a Matemática é bem evidente em vários dos documentos do curso. No CD-ROM do professor, escreve-se:

> When planning the Advancing Physics course, we decided that we must be very positive about mathematics in physics. Students should not see mathematics as a necessary evil to complain about, but as fundamental to the pleasure and power that physics has to offer. An example is vector quantities, which need to be presented as an exciting first step on a long road to constructing new quantities that can do more than represent single numerical values. Another, more important, example is simple differential equations. Students need to understand them as recipes that can predict the value of a quantity at a tiny step in the future. And that those tiny steps can be repeated again and again so that, for example, eclipses can be predicted with precision a long time before they happen. Here again, computing can play a crucial role (Lawrence & Whitehouse, 2000).

A utilização de computadores no ensino das ciências físicas e das ciências matemáticas será, certamente, um facto no futuro próximo, no ensino secundário e no ensino superior, tal como é um facto a sua utilização na produção de conhecimento científico nessas disciplinas. A natureza do conhecimento nestas disciplinas, com profundas conexões, bem como a natureza das diversas ferramentas computacionais, exige uma abordagem mais integrada do seu ensino. Essa abordagem integrada não significa necessariamente a redução a uma única disciplina escolar, se bem que tal já esteja a ser experimentado a nível dos primeiros anos do ensino superior (ver, por exemplo, Rex & Jackson, 1998). É necessário uma reflexão sistemática sobre o melhor processo de concretizar esta visão integrada. E, evidentemente, uma reflexão semelhante sobre qual o papel das ferramentas computacionais na concretização dessa visão. No que diz respeito à utilização do *Modellus*, esta reflexão está em curso em diversos países, nomeadamente a Inglaterra, Portugal e o Brasil. Os próximos anos deverão ser esclarecedores sobre estas questões essenciais para a renovação do currículo das ciências físicas.

0.4 *Modellus*: algumas opiniões de alunos e professores

Seis anos após a primeira edição, o *Modellus* está amplamente divulgado em Portugal e em muitos outros países. Algumas das primeiras opiniões de professores assinalaram o facto do programa ser de “difícil utilização”. Esta opinião tem vindo a ser alterada, à medida que os professores conhecem melhor qual é “a ideia do programa”. Os alunos, pelo contrário, quando devidamente iniciados na sua utilização, consideram-no fácil de usar, mesmo nas primeiras vezes que usam o programa.

Foram realizados dois estudos com alunos. Num, em que participaram doze alunos dos 10.º e 11.º anos, os alunos (em grupos de dois) efectuaram diversas actividades experimentais com sistemas de aquisição de dados e, em seguida, actividades de modelação com o computador, durante uma semana. Nas respostas ao questionário sobre as actividades, foi unânime a opinião de que o *Modellus* é fácil de usar, mesmo por quem não sabe utilizar um computador. Um dos alunos escreveu:

> Na minha opinião, o programa *Modellus* é bastante acessível, mesmo para quem mal saiba mexer em computadores e não perceba nada de Matemática e Física (como eu!). É claro que é preciso professor e manual de instruções, mas depois de alguma experiência torna-se bastante fácil [CD].

Um dos aspectos chave no design do *Modellus* era, precisamente, a criação de um interface suficientemente intuitivo, em que o utilizador *pensa com o computador* praticamente como pensaria se estivesse apenas usando papel e lápis.

Todos os alunos que participaram no estudo foram capazes de construir os modelos adequados às situações experimentais que analisaram (movimentos uniformes, acelerados e retardados), com maior ou menor apoio. Para alguns alunos, o conhecimento prévio era insuficiente (por exemplo, desconhecimento do significado dos parâmetros da função quadrática) mas tal foi ultrapassado com apoio individual.

No outro estudo, em que participaram 10 estudantes do 2.º ano da licenciatura em Ensino das Ciências da Natureza da FCTUNL, os alunos utilizaram um texto, disponível na página Web do Modellus, sobre *Funções e Movimentos com o Modellus*. As actividades decorreram durante apenas três dias, o que se revelou insuficiente para fazer todas as actividades propostas no texto.

No início do segundo dia os alunos resolveram um problema de cálculo de distância de travagem, apenas com papel e lápis. No final do terceiro dia resolveram o mesmo problema, utilizando o Modellus. Em seguida, comentaram, por escrito, a resolução do problema sem e com computador. No final do terceiro dia, com o computador, todos os alunos resolveram correctamente o problema (no início do segundo dia, quatro dos alunos não tinham sido capazes de o resolver). Nos seus comentários sobre a resolução do problema, a maioria dos alunos considera que a "visualização" e o "controlo dos resultados" são as diferenças mais importantes entre a resolução sem e com computador.

Uma das ideias chave no design do *Modellus* é a importância dada à concretização dos conceitos abstractos. Foi muito frequente a opinião dos alunos que o programa os *auxiliava a pensar*, porque lhes permitia *concretizar e testar as ideias*. Uma opinião semelhante é manifestada por muitos dos inquiridos num questionário respondido por 75 professores de 11 países, de diversos níveis de ensino. Esses professores (registados como utilizadores do Modellus) responderam, por correio electrónico, a onze questões sobre o Modellus e sobre uma visão integrada do ensino da Física e da Matemática. Por exemplo, um dos professores escreve:

> (…) writing models with *Modellus* is teaching me a lot about phasors that a degree in physics never did! [IL].

Este facto tem, necessariamente, a ver com a *reificação* dos objectos formais que os utilizadores manipulam na janela de "Animação" do *Modellus*. Esta manipulação concreta de objectos formais no *Modellus* é evidenciada por vários dos professores inquiridos:

> (…) For me, 'formal reasoning' (Piaget sense) is precisely concrete reasoning (Piaget sense) with symbolic objects (*Modellus* sense) [JO].

> It helps develop reasoning and abstract skills, and to create an observational attitude and analysis skills before experimentation [EM].

Uma visão integrada das ciências físicas e da matemática é outro dos aspectos chave no design do *Modellus*. Todos os inquiridos consideram essa visão integrada como essencial, porque, como afirma um dos professores, não há, muitas vezes, distinção entre o que é "física" e o que é "matemática":

> Much of science, especially physics, is at the same moment mathematical. In doing theoretical physics, or in analysing an experiment, there simply is no real distinction. Physicists do not so much 'use' mathematics, as 'do mathematical style physics' at these times [JO].

A utilização integrada do *Modellus* no currículo é corroborada pela quase totalidade dos inquiridos. Mas, note-se, que é também comum a opinião que sua utilização é, apenas, *potencialmente* útil. De facto, para ocorrerem significativas mudanças na prática curricular é necessário, como vimos atrás, muito mais do que uma ferramenta computacional.

0.5 Coda

Num livro recente, diSessa escreveu:

> "Computers can be the technical foundation of a new and dramatically enhanced literacy, which will act in many ways like current literacy and which will have penetration and depth of influence comparable to what we have already experienced in coming to achieve a mass, text-based literacy" (2000, p. 4).

O conceito de "literacia" tem vindo, como argumenta diSessa, a ser significativamente ampliado nos últimos anos. A emergência das "tecnologias da inteligência" (Lévy, 1994), de que as ferramentas computacionais são o exemplo mais marcante, ainda não teve influência significativa nos processos de ensino e aprendizagem. A tendência para a simplificação do uso de computadores, a diminuição de preços, a melhoria das condições organizacionais das escolas, o aparecimento de uma nova geração de professores e alunos familiarizada desde cedo com o computador, a crescente difusão da Internet (responsável, em grande medida, por exemplo, pela difusão do *Modellus*, desde a China ao Chile), darão um contributo essencial para a alteração das práticas curriculares e para uma literacia mais ampla, em que o uso de "ferramentas cognitivas" é determinante. Mas há um factor essencial que não pode ser menosprezado: é fundamental um esforço continuado de desenvolvimento curricular, em que as ferramentas computacionais sejam integradas no processo de ensino e aprendizagem, um esforço baseado na investigação educacional e envolvendo o contributo de cientistas, engenheiros e professores, e acompanhado de um processo continuado de desenvolvimento profissional de professores.

Esta tese argumenta que a utilização de ferramentas computacionais, para apoiar os processos de pensamento e de construção de conhecimento científico por estudantes do final do ensino secundário e dos primeiros anos do ensino superior, é uma perspectiva incortornável para o futuro do ensino da Física. O domínio destas ferramentas computacionais é, sem dúvida, uma componente das novas literacias.

Os resultados obtidos nos dois estudos com alunos e as respostas ao questionário a professores sugerem que é possível a alunos e professores utilizar software de modo semelhante ao modo como ele é utilizado na investigação: como uma *ferramenta para pensar*. O objectivo de criar um tal tipo de ferramenta, facilmente acessível a alunos e professores, era um objectivo pessoal de longa data, originado numa larga experiência de ensino e de confrontação com as dificuldades de aprendizagem de muitos alunos. Husén (1998), considera que a finalidade da investigação educacional é

> "provide a basis for action, be it policy or methods of teaching in the classroom".

Fiz o melhor dos meus esforços para dar um contributo para atingir tal objectivo.[1]

---

[1] Uma tese é, essencialmente, um esforço individual. Mas, um esforço individual só é possível com apoio e suporte nos momentos adequados. Foram muitas as pessoas (jovens programadores, amigos e colegas, alunos, familiares) que me apoiaram, alguns sem, provavelmente, terem consciência de quão importante foi o seu contributo. Esse apoio e suporte não foi esquecido: estará sempre presente na minha memória. A orientação, crítica e suporte dos meus orientadores M. O. Valente e C. M. Silva, bem como de alguns colegas estrangeiros, em particular J. Ogborn (University of London) e J. L. Schwarz (Harvard University) é também motivo de profundo reconhecimento, muito mais vasto que uma nota de rodapé pode supôr...

Chapter 1 Introduction

A model for educational research: innovate, disseminate, create a community, do research, and disseminate the research (Schwartz, 1990).

1.1 Why this Thesis?

This thesis is the result of about two decades of work, reflection, research and development in the pursue of a solution to problems I came across in my teaching in the late seventies. As a secondary school teacher, starting a career in teaching physics, I found that most, if not all, senior high school students didn't learn even the basics of what they were supposed to learn, even when exposed to *good and careful teaching*. This was particularly true for topics involving *graphs*, *motion* and other phenomena that are *modelled mathematically*. Soon I found that it was not only my students who had "difficulties": many other teachers had the same problems and a few researchers were then publishing the first studies showing the "misconceptions" most students have before and after formal instruction in the laws of motion (see e.g., Caramazza, McClossky, & Green, 1980; Viennot, 1979).

At the same time—about 1980—, I became interested in computers, mainly because I needed to do statistical computation for evaluation and assessment studies I was then involved. Computers at that time were not of the same "species" we are now familiar with. A typical user was only introduced to a terminal, connected through a telephone line to a central computer. The terminal was just a Teletype machine and all the interaction with the computer was with text, printed in a continuous sheet of paper. When my students knew that I was able to operate the "computer" they urged me to show how a computer could be used in science. I programmed a few lines in FORTRAN to show a "motion" of a letter on a page, according to with a certain equation of motion…

After this my first use of computers in education it was clear to me that we needed time to have hardware sufficiently reliable and user-friendly. But it took only one year for a very small computer, ZX81, to appear everywhere, available for a relatively small amount of money. I remember giving my first course on using computers in physics education somewhere around December 1982, at the Portuguese Physical Society. I borrowed some computers and a few TV sets from a TV reseller company. Teachers in the course spent two days making motion simulations in a dialect of Basic, with lots of enthusiasm. The activities in the course were based on worksheets I was using with 12th graders at the secondary school I was then working at.

After ZX81, NewBrain, another British computer with a very interesting Basic, with lots of graphical facilities appeared. And then ZX Spectrum, the computer that influenced many youngsters in the mid-eighties to pursue careers in computer science and engineering. Some of these students worked with me in the late eighties as programmers and co-authors of software. ZX Spectrum was, for the time, a fascinating computer: small, with good design, good graphics, not very difficult to program, not very expensive and extremely popular in many homes. There was also an increasingly supporting literature on how to use the ZX Spectrum in Science Teaching (e.g., Sparkes, 1984). In a visit to the UK, I met Brian Kahn, the author of one of the first books about the use of computers in science education (Kahn, 1985) and Jon Ogborn, one of the most influential authors on my work. I showed the software I had developed by that time, mainly for studying motion, and we discussed many issues about using computers in education. Jon was then (late 1980s) maturing his ideas about a new concept of using computers in education—modelling—an idea that was starting to be used in the physics curriculum developed by the Nuffield Advanced Physics project.

It took me three more years to start thinking that modelling was probably the most interesting use of computers in science education. And I must say that this is due mainly to the fact that, in Jon's view students should model physical phenomena not with functions or differential equations but with difference equations, something that was then strange to me. Jon visited Portugal in 1987 and gave a conference and a workshop in Braga. He brought his own computer, a BBC machine, popular in the UK but unavailable in Portugal, and showed to a small audience the modelling program DMS (*Dynamic Modelling System*). This program, one of the first modelling systems for secondary school physics, was then popular in A-level Physics in the UK, as part of the *Nuffield Physics* course, but seemed to me yet too much primitive. I wondered if teachers would really think DMS would be an added value to their teaching. In 1989 I visited Jon Ogborn in London where he gave me the successor of DMS, a new package called CMS (*Cellular Modelling System*) that was then ready to be published.

At that time I designed a few titles to explore graphs, motions, some ideas in chemistry (periodicity, atomic models, etc.). We were then shifting from the ZX Spectrum to the PC world, still under MS-DOS. But the user interface that was offered by most of the MS-DOS programming environments was very primitive. It was necessary to use libraries of graphical routines (some made by the programmers themselves, others bought mainly from the USA). One of the titles—*Newton*—needed such a sophisticated user interface that it was necessary to develop it only on the Macintosh, the true direct manipulation graphical interface really available late in the 1980s.

The software that I developed in the early eighties was mainly simulation software. I was then becoming more and more aware that simulation software was not what students really needed. In one paper written in 1990 (Teodoro, 1992) I wrote that it was important to design software tools that could link *simulation*, *modelling* and *data logging*. I participated in the design of a modelling environment, *Dinamix*, which implemented some of these ideas. But it was still very early to have a powerful tool, due to the relatively poor design of the software, the difficulty implementing direct manipulation and also because the ideas needed further development.

During the academic years of 1990-91 and 1991-92 I tested some of the software developed with students in secondary schools. This testing showed that computers were still a novelty to students and teachers alike but they were capable of using the software. But this doesn't mean that students were able to learn easily with the software: I was intrigued why students had difficulties in making sense of what they saw on the screens. It was evident that the software was not enough, as I always suspected.

Between 1992 and 1995 many interesting titles were developed, this time running under Windows 3.11. They became available for teachers, including in other countries. Direct manipulation, full graphical environments, multiple representations, etc., become standard on these titles. But, again, it was not this I *really wanted*: I wanted a *general tool*, which could be used in almost everything, either in physics or in the part of mathematics that was really essential to physics. It was only in late 1995 that I started designing Modellus, the modelling system now available worldwide and used by hundreds of teachers and thousands of students.

Modellus answered most of the questions I had been working with in the previous 15 years. This thesis is, partially, a *narrative* of a search for a solution of teaching problems, and partially the proposing of scientific arguments to support both the *search* and the *tentative solution*. But, with Bruner, we all recognise "that there are powerful constraints on what schools can do" (Bruner, 1996, p. xv). So, besides thinking of new ideas and new tools, it is also necessary to think about how these new ideas and new tools can become part of the curriculum and of the classroom culture.

This thesis is an exercise on the links between *theory* and *practice* in education. The educational literature is full of ideas, convictions, and arguments, about what is "good teaching" according to the prominent theories of teaching and learning. As a practitioner, I have always been interested on the "how to". As a researcher, I always felt that the "how to" was a limited way of changing things in education. How can we make compelling arguments about the usefulness and validity of new perspectives of teaching and learning? Surely, one way is developing tools "embedded with theory" that can be used to test the soundness of the theory. This is one of the goals of Modellus: to serve as a tool that can be used to test fundamental ideas about the role of computers in education, about the construction of meaning from experience, in a word, about learning and teaching.

1.2 Cognitive Artefacts and Physics Education

Physics is a relatively new subject in the secondary curriculum (ages 13-18). Only in the second half of the 19th century did science education become part of the curriculum and only in our century did physics, or physics and chemistry, become an autonomous subject in the developed countries (Carvalho, 1985; Jenkins, 1991).

Teaching and learning physics has always been considered a difficult task by most teachers and students and common people (McDermott, 1993; Peters, 1982). Some experienced policy analysts even say that physics/scientific literacy is a *myth* (Shamos, 1995). Feynman used to tell a story about a conversation with the Queen of Sweden during which she said, after asking what his field of work was: “Oh, well, we can’t talk about *that*. *Nobody* knows anything about physics.” Feynman, very politely, answered, “On the contrary, Madam, we can’t talk about physics precisely because somebody *does* know something about physics. What we *can* talk about is philosophy or psychology, because nobody knows anything about those subjects.” And “he would go on to say that subjects like philosophy and psychology are hard, but physics is easy and that’s precisely why we know so much about it” (Goodstein, 1992). But, as Goodstein says, “If physics is easy, the question is, why do we do so badly at teaching it?”

Certainly, there are multitudes of reasons for that. One, surely not the least important, is that teachers soon face the “discovery of poverty” in their classrooms and that they somewhat naively tend to assume that kids are just as enthusiastic about the curriculum as they are (Bruner, 1996).

Besides the many social-cultural problems teachers face in their teaching, it can be argued that learning science, and physics in particular, is like learning a second language—a *new* language where words are not what they seem to be:

> (...) When dealing with the definition of terms we do well to remember how abstract are some of the concepts we use in physics. If we recall the difficulty that Galileo, a superb physicist, had in dealing with acceleration we may have more patience with our students (Ebison, 1993, p. 361).

The language of physics was created in the last three hundred years. The history of physics is also an evolution of this language, the “invention of new vocabularies and new ways of talking about the world” (Gregory, 1988, p. 3). Physicists are aware that the language of physics is not the “proper” language to express themselves in many contexts. We all know, for example, that most of the language of physics is unsuitable to maintain an understandable social conversation about cold and hot things. Learning a second language is not a problem of knowing the structure of the grammar of that language. It is, essentially, a matter of *familiarization* with the language and its proper use in specific contexts. *Familiarization* is, I shall argue recurrently in this thesis, an important issue when learning science (and mathematics). And, for some eminent scientists, *becoming familiar with* is so important for the success of scientific ideas that new ideas only become triumphant because supporters of old ideas die, as Planck wrote in his autobiography:

> (…) A new scientific truth does not triumph by converting its opponents and making them see the light, but rather because its opponents eventually die, and a new generation grows up that is familiar with it (Planck, 1950, pp. 33-34).

There are some relevant arguments to support the importance of familiarization in learning, as contrasted with *understanding*. One is that scientists frequently say that they do not understand some of the most fundamental concepts or theories in their own field. For example, Feynman wrote that he didn't know what force or energy *really* was and that no one *really* ever understood the theory of quantum mechanics… In an elegant manner, Feynman wrote (Feynman, 1982):

> *We have always had a great deal of difficulty*
> *understanding the world view*
> *that quantum mechanics represents*
>
> *At least I do;*
> *because I'm an old enough man*
> *that I haven't got the point*
> *that this stuff is obvious to me.*
>
> *Okay, I still get nervous with it...*
>
> *You know how it always is,*
> *every new idea,*
> *it takes a generation or two*
> *until it becomes obvious*
> *that there's no real problem.*
>
> *It cannot define the real problem,*
> *therefore I suspect there's no real problem,*
> *but I'm not sure*
> *there's no real problem.*

Experienced physics teachers also alert us about *our* ignorance of the most fundamental issues in physics. In a very curious paper, Brian Davies, an English physics educator, wrote:

> At the heart of the problem lies what amounts to a connivance among successive generations of teachers regarding the nature of their science approaches to understanding abstract concepts, leading to a self-defeating reluctance to open and share their partial understandings with their equally perplexed students.
>
> If I asked you, here and now, to write me a one- or two-paragraph explanation of your understanding of the nature of the Holy Ghost, you'd probably think it a tough challenge. If we then collected up your answers, we could be pretty sure that (i) they'd all be different, (ii) not one would satisfy all of us, and (iii), importantly, we'd be hard put to say which interpretation or explanation would be *wrong*, because there is no absolute in knowledge and understanding regarding The Holy Ghost.
>
> Most of the useful concepts of physics are, for teenagers, as mysterious and as difficult to grasp as the concept of the Holy Ghost: the *nature* of an electrostatic charge, of a magnetic field, of electromagnetic wave propagation in a vacuum, or of charm and colour are examples. There is, as the Holy Ghost, no absolute understanding or knowledge of the nature of these intangibles, yet any half-curious young adult will ponder on their nature. In physics education we need much more of the kind of humility shown by Feynman, who said openly that he felt no one really understood quantum mechanics.

> If teachers continue to give the impression that they do have a better basic understanding of such fundamentals than their students, the students will see their own perplexity and uncertainty as a negative reflection on their own capabilities. Even in this group today there will be some of you who will remember the relief you felt when you could use some equation, and your mathematics, to answer a problem, rather than stay with your uncertainties regarding the concepts involved. We learn and teach others to use mathematics to manipulate the symbols associated with mysteries. This does not mean that we or they have a grasp of the mysteries themselves (Davies, 1997, pp. 420-421).

Physics deals with a special type of "objects": "objects" such as *force*, *velocity*, *energy*, *radiation*, etc. At a first glance, some, or even most, of these words seem *familiar* to a student. Nevertheless, they are not and they can't be. *Force*, in the language of physics, is the "instantaneous rate of change of linear momentum". The same sort of specifications are established for the other physics concepts. However, in the students' first language (Portuguese, English or any other), *force* means many things and in many contexts, surely not "an instantaneous rate of change"!

An important issue in learning such abstract concepts, one that is intimately related to familiarization is the issue of *reification*, i.e., of *concretisation of abstract objects*. According to Wright & Wright:

> Reification is a central goal [… of learning science and mathematics]; it essentially defines scientific literacy. It is the foundation for common sense about how the world works […] (1998, p. 128).

Reification is an essential issue both for the learning of physics and mathematics as well. It is Roitman, a mathematician, which tells us that students can only learn at an abstract level when they consider mathematical objects as real as everyday objects:

> The objects of mathematics are real objects, in a psychological, not necessarily ontological sense—they feel real; we act as though they are real. For example, 'number sense' is based on reification: we can compare numbers, operate on them, and look at their properties because they are real. Or another example, many young children have not reified the notion of fraction—for them, 1/2 implicitly carries with it the question "1/2 of what?" When the concept of '1/2' takes its place in the number system as just one of many rational numbers, to be thought about and used as we think about and use all rational numbers, it has been reified. (….) To take a third example, algebra cannot really be understood unless variables are reified—'x' is not a placeholder standing in for some unknown number, but an object in its own right. Reification cannot be forced, but its encouragement is a major part of the art of teaching mathematics (Roitman, 1998, p. 26).

Assuming *reification* and *familiarization* as essential aspects of learning physics (and mathematics), we must ask how can this be improved with technology and, specially, with computers? Hebenstreit, writing about the role of computers in education, coined a term that seems essential to understand how computers can help in the reification of knowledge. For Hebenstreit, computers allow us to manipulate a new type of object, objects that he calls *concrete-abstract* objects. *Concrete* in the sense that they can be manipulated on the screen and react as "real objects" and *abstract* because they can be only physical or mathematical constructs such as vectors, equations, fields, etc. (Hebenstreit, 1987).

We are now coming to an end of this section with the proper subsumers to understand how computers can be used in physics education: they can be powerful *cognitive* *artefacts* (Norman, 1991). A cognitive artefact is a *tool* to enhance *cognition*, a tool to create and explore “concrete abstract-objects”, a tool to create “worlds from ideas” and check how well these “worlds” can fit “real worlds”, or make sense of “imagined worlds”. That’s one of the goals of this thesis: to show how Modellus is a cognitive artefact and how students and teachers can use cognitive artefacts to *reify* the abstract and *become familiar with* and *reify* some of the most powerful ideas in physics (and mathematics).

Teachers tend to *teach what they can teach*, not necessarily what they think it would be useful to teach. This is what Osborne calls *technological determinism* (Osborne, 1990): “that which we do teach is limited by that which we can teach” (p. 193). He gives a few classic examples and shows how most of the practical and theoretical teaching is really dependent on the available technological devices and on the *limited* mathematics that students (and also teachers) can use: *simple analytical tools* but that need *complex* algebraic manipulation. He follows his line of reasoning to propose that:

> The advent of powerful computational tools in the past decade has resulted in more emphasis being placed on numerical methods of solution in physics. The study of chaos and the generation of Mandelbrot plots would have been severely limited without this technology. Yet school physics has yet to deploy such tools to enhance the education provided. Introductory kinematics courses place pre-eminence on the analytical solutions of objects moving with constant acceleration. My argument is that one implicit reason why this is done is because it is one physical situation that is accessible to an analytical solution with limited mathematics, another instance of that which can be taught, being taught. Yet the solution is lost in a confusion of algebraic manipulations whilst the numerical approach is, in a fundamental way, easier than the analytic approach.
>
> The alternative approach through numerical methods forces attention on the basic physics. It asks the child to consider what are the dynamics of the situation? How can the acceleration be predicted? How can the velocity be calculated if the acceleration is known and then how can the new position be calculated? The solution is then generated by iterative calculation and the pupil is forced into judging whether the answer suggested is appropriate. The rule used for calculating acceleration can easily be changed to incorporate friction or to model a harmonic oscillator. Thus the emphasis is on the physics, not the mathematics. The issue that must he resolved is whether we should present pupils with problems of real-world complexity or adopt the reductionist approach, stripping the problem of anything but the simplest detail? The inevitable idealization of the latter approach, enhancing the separation of the world of physics from the real world of the child, again weakens our argument that physics can explain ‘how we know’ since the phenomena described are patently not commensurate with the child' s perception of reality (Osborne, 1990, p. 194).

Osborne proposals only now seem feasible because hardware and specially software evolved to give us the possibility to emphasize *meanings*, even if complex calculations are necessary, in spite of simple analytical solutions that describe only idealized phenomena. For example, with Modellus a student can easily explore complex damping situations, based on *physical reasoning*, instead of using meaningless functions.

A characteristic feature of using a computer as a cognitive artefact is that the emphasis is on *meaning* and *semi-quantitative reasoning* instead of algorithms and routine thinking. A good example of what is semi-quantitative reasoning can be done with the computation of, let’s say, sin (35º). Using a computer, or a calculator, we can easily get the result: 0.574. From a *semi-quantitative* point of view, let’s look if this result *makes sense*: it is smaller than 1 (*OK!*), is bigger than 0 (*OK!*); if I do a sketch of a trigonometric circle, I can easily *estimate how big* is the ratio between the segment that represents the sine value and the radius of the circle for an angle of 35º (*a little more than 1/3 of the 90º angle…*); with some sense of estimation, is possible to check that that ratio *can* be *a little more than 0.5*.

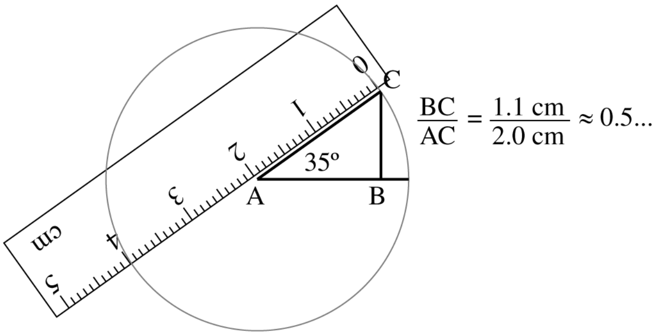


Figure 1.1 How to estimate sin 35º?

An *expert*—either student or teacher—can easily do this semi-quantitative reasoning even when either the teacher or the student *never* knows how that value is really obtained! Moreover, they really do not think that is essential: it will not be of any help if you have a computer or calculator… The *experts* are *experts* when they can evaluate how *reasonable* that value is, not when they can *compute* it. Sure there is also place for a certain kind of expert that can think, let's say, of better and faster algorithms to compute trigonometric functions, but that is not for the "rest of us", who can, at best, be interested as a curiosity in such algorithms but understand that it is not the knowledge of the algorithms that help make sense of "sin (35º) = 0.574"…

The same kind of semi-quantitative reasoning can be done with a mathematical object such as $dx/dt = 4 \times t$. What does this tell us? First, we see that the rate of change of x is proportional to t. So, for a bigger t, we will have a bigger rate of change. More precisely, when t is 5 units, for example, the rate of change will be, at that instant, $4 \times 5 = 20$ units. If t is 10 units, then the rate of change will be $4 \times 10 = 40$ units. That is, if t doubles, the rate of change doubles. And x is always increasing, for positive values of t. Let us see another example: $dx/dt = 4 \times x$. Now we have a rate of change of x that is proportional to x at any instant of time. For example, is x is nil, then the rate is also nil. For a positive value of x, at any instant of time, the rate of change is positive and so x increases.

In addition to supporting semi-quantitative reasoning, cognitive artefacts also can play an essential role in externalisation and negotiation of learning, as mentioned by (Orhum, 1995):

> The main function of cognitive tools is to enable learners to make explicit and negotiate meaning. Making meaning explicit requires the representation of thought processes in external models for examination and reflection, and it may help learners improve these cognitive processes. Negotiation of meaning involves exchanging views and interpretations in communicative acts among learners (Orhum, 1995, p. 314).

This view is coherent with the now dominant view of learning, the constructivist view.

Since physics is a science where visualisation plays an important role, even when visualisation is only used to show mathematical objects, such as vectors or field lines, it seems very reasonable to suppose that computer visualisation can help learners create meaning from manipulations of abstract objects. This capability of computers has been extensively used in many contexts (e.g., in the beautiful TV-based course *The Mechanical Universe*, Goodstein & Olenick, 1988) and is stressed by many authors, such as Kozma, who points out the capability of making dynamic representations of non-concrete formal objects:

> Computers […] have the capability of creating dynamic, symbolic representations of non concrete, formal constructs that are frequently missing in the mental models of novices. More importantly, they are able to proceduralize the relationships between these objects. Learners can manipulate these representations within computer microworlds to work out differences between their incomplete, inaccurate mental models and the formal principles represented in the system (Kozma, 1991, p. 179).

This trend accompanies the increasing importance computer visualisation and simulation is playing in science and in physics in particular. Galison, for example, wrote about the new “epistemic position” of computers and simulations in the production of physics knowledge:

> Computers and simulations ceased to be merely substitutes for mechanical parts, they come to stand in a novel epistemic position within the gathering of knowledge—not quite a piece of empirical machinery, and not quite one with theoretical apparatus (Galison, 1997, p. xix).

Nickerson pointed out that researchers have not yet focused on students as authors of simulations:

> What has not yet received much attention from researchers is the possibility of having students develop simulations themselves as a way of fostering a greater understanding of the processes they attempt to simulate (Nickerson, 1995, p. 16).

He follows arguing that “it is only difficult, not impossible, and the work that goes into the successful building of a microworld is likely to deepen one’s understanding of whatever the microworld is intended to simulate” (Nickerson, 1995, p. 16). To build simulations, one can use programming languages, but these require technical knowledge and skill outside of the domain being simulated. This is the reason why Nickerson propose the development of specific tools that can be used by people without that knowledge:

> For student-developed simulations to be practical for educational purposes, it will probably be necessary to develop tools that are designated to facilitate the building of simulations by people without such language facility and programming experience (Nickerson, 1995, p. 16).

Modellus is, certainly, a tentative approach to create such a tool, not only for students but also for teachers and curriculum developers.

There are still two other important aspects that must be considered about cognitive artefacts such as computer tools, mentioned by Bruner in some of his recent writings. Bruner says that the computer reintroduced the capability to make routine work without human “servants”. Bruner also points out that computers can be a kind of “intellectual mirrors”, in the sense of Schwartz (1989):

> One last word and I am done. I have said nothing about computers, which seems strange in this day and age. I really have nothing to say about them, aside from the fact that I love them and my life would be much more tedious without them. They can be a boon to scientific consciousness and, besides, they have reintroduced the servant in an era when the sages all said we would forever more be servantless. Best of all, we can construct programs that can ‘simulate’ what we might with great cost and effort do in our heads or on paper, and, in so doing, making us aware of what it is that we must still do ourselves in our own heads (Bruner, 1992, p. 12).

1.3 Technological Change and Science Education

When I started using computers with students, in the early 80s, it was not clear how important and ubiquitous computers would be in our society. Now we know that “computers have pervaded all aspects of life in the developed world, changing working practices and leisure activities” (Ross, 1993, p. 69). Everybody now agrees that young people and adults must “become aware and unafraid of computers, just as they need to become literate and numerate” (Ross, 1993, p. 69). Most of this familiarization with computers is done, especially with young people, without any formal teaching—just learning with peers in informal settings, like resource centres and homes.

Ross also points out that it would be a waste of resources if we use computers just to develop computer literacy when we know that computers can help extend, improve and change the traditional curriculum significantly. And, more important, computers are now recognised as fundamental tools in the production of scientific knowledge:

> Scientific computation has become so much a part of everyday experience of scientific and engineering practice that it can be considered a third fundamental methodology of science—parallel to the more established paradigms of experimental and theoretical science (National Research Council, 1989).

Then, why should we not use extensively computers in teaching?

Some authors, such as Cuban (Cuban, 1989), pointed out that computers, like all technological innovations in schools, tend to follow a cycle of four phases: high expectations; rhetoric about the need to innovate; oriented policy and finally limited use. This cycle is certainly true for innovations such as educational television but it is not true for other innovations such as radio or the teaching machines since for these there is a fifth phase: *no use at all*. It is also not true for computers, as Cuban himself seem to admit recently—see, e.g., the debate between Roy Pea, a strong advocate of the use of computers in education, and Larry Cuban (Pea & Cuban, 1998). Contrary to the other innovations, which declined very early after the first three initial phases mentioned by Cuban, computers are increasingly present in schools, as they are everywhere. For example, nowadays a school or university laboratory without data logging systems (computers, interfaces, sensors and software) is unthinkable. The same is true for school libraries: all have access to digital books, either off-line or on-line, at least in Portugal and in other European countries. Moreover, in the near future we will see an increase in the use of computers in education, at least in certain subjects such as mathematics and the physical sciences. For example, the *Principles and standards for school mathematics* (NCTM, 2000) now explicitly states the importance of technology in learning, considering the use of technology one of the six fundamental principles of teaching and learning:

> **Technology principle**: Technology is essential in teaching and learning mathematics; it influences the mathematics that is taught and enhances students' learning (NCTM, 2000, p. 24).

The same is true for the mathematics curriculum in Portugal and in many other countries that are strongly influenced by NCTM *Principles and standards*.

But it is not only mathematics educators and curriculum developers who reinforce the importance of technology in learning. For example, the new Institute of Physics project Advancing Physics uses computer tools (one of the tools is Modellus) and the Internet as an integral part of the curriculum (Ogborn, 1997).

To understand the role of computers in education it is useful to use Pryluck's distinction between "inherent" and "imposed" characteristics of a medium (Pryluck, 1968, p. 372). The first, *inherent characteristics*, are "symbols and combinations thereof selected from the symbol system that was developed in connection with the specific technology of transmission." The second, *imposed characteristics*, are "situations of exposure (...), teachers' instructions, or even the didactic structure of the presentation. They are imposed simply because one could easily remove them, apply them differently, or apply them to another medium." Moreover, Pryluck reminds us that the imposed characteristics are "at best correlates of the medium". Since computers are now normal tools in the production and communication of scientific knowledge, their *inherent* characteristics seem unquestionable. This leaves us only with the discussion about the *imposed* characteristics, that is, the discussion about using computers in physics education (and, probably, in most school subjects) should *not* be a *discussion about using computers* but a *discussion about how to use* them.

According to Collins (Collins, 1991), we can expect computers to help in the following eight shifts:

1. A shift from whole-class to small-group instruction.
2. A shift from lecture and recitation to coaching.
3. A shift from working with better students to working with weaker students.
4. A shift toward more engaged students.
5. A shift from assessment based on test performance to assessment based on products, progress, and effort.
6. A shift from a competitive to a co-operative social structure.
7. A shift from all students learning the same thing to different students learning different things.
8. A shift from the primacy of verbal thinking to the integration of visual and verbal thinking.

This list clearly relates to the "imposed" characteristics of computers in schools, not with the "inherent" ones. However, if we want technology to make a real difference, students must be in a state of 'mindfulness' for technology to work. Mindfulness, in this context, is the employment of non-automatic, effortful, and thus metacognitively guided processes (Salomon, Perkins, & Globerson, 1991).

According to Salomon, computers are more than add-on devices since "computer tools carry with them implicit assumptions about self-guided exploration and design, even playful activity, team collaboration, integrated curricula, mutual consultation, and teachers' orchestration of activities rather than teacher domination" (Salomon, 1992, p. 251). He follows alerting us that "to make computer use affect education it cannot just be introduced as an addition to otherwise unchanging classroom practices the way, for example, television could; with its proper introduction everything in the classroom, possibly in the school as a whole, changes."

This helps us understand why it seems so difficult to make computers become part of regular practice in schools, particularly in classrooms. Computers *are not* add-on tools like television. Their use implies profound cultural changes—changes in the teaching culture, in the learning culture, in the school culture— as well as changes in the school organisation. If the dominant cultures of teaching, despite their diversity and evolution, are still essentially individualist—"teachers' desire to be left to themselves" (Feiman-Nemser & Floden, 1986, p. 522)—and defensive, the constraints for the generalisation of computer use in classrooms become evident, particularly when another characteristic of the culture of teaching is the use of "little research-based technical knowledge" (Feiman-Nemser & Floden, 1986, p. 522).

But everything can change with time, social pressure and teacher involvement. It is also Feiman-Nemser and Floden who wrote that one particularly important change in progress in the culture of teaching is that the passive teacher moulded by the bureaucracy is being substituted by an "active agent, constructing perspectives and choosing actions" (Feiman-Nemser & Floden, 1986, p. 523). And the renovation of teaching and learning can only be done with teachers able to work in groups, open to learn continuously, open to learn and share difficulties with their students, open to criticism and improvement (Ponte, 1994).

Some critics, such as Apple (1991), introduce a different perspective when analysing the role of the computers in education:

> At root, my claim will be that the debate about the role of the new technology in society and in schools is not and must not be just about the technical correctness of what computers can and cannot do. These may be the least important kinds of questions in fact. At the very core of the debate instead are the ideological and ethical issues concerning what schools should be about and whose interests they should serve (Apple, 1991, p. 61).

These critics fear, for example, that computers will support "the creation of enhanced jobs for a relative few and deskilled and boring work for the majority" (Apple, 1991, p. 65). This is, undoubtedly, an important issue, but more general than the ones I am analysing in this thesis. All members of our societies must be aware of this danger, not only educational researchers and teachers. Apple also alerts us that computers can be extensively used to "rationalise and control the act of teaching" (Apple, 1991, p. 66).

This is certainly true for computer uses such as computer-managed instruction, but not for exploratory environments such as Modellus and other computer tools. On the contrary, these tools give more control to teachers and can help them be more creative in the management of the curriculum. This can be exactly the *opposite* of what Apple fears, the "deskilling of teachers":

> Of the major effects of the current (over) emphasis on computers in the classroom one may be the deskilling and depowering of a considerable number of teachers (Apple, 1991, p. 67).

Apple is right when he says that new technologies embody a *form of thinking*, primarily *technical thinking*:

> The new technology is not just an assemblage of machines and their accompanying software. It embodies a *form of thinking* that orients a person to approach the world in a particular way. Computers involve ways of thinking that under current educational conditions are primarily *technical*. The more the new technology transforms the classroom into its own image, the more a technical logic will replace critical political and ethical understanding. The discourse of the classroom will center on technique, and less on substance. Once again 'how to' will replace 'why,' but this time at the level of the student. This situation requires what I shall call social, not technical, literacy for all students (Apple, 1991, p. 75).

But it is not correct to assume that *technical* and *not technical* thinking are opposite ways of thinking. Most of the time, intelligent thinking is done with tools and it is not possible to separate intelligence from tools:

> Almost any form of human cognition requires one to deal productively and imaginatively with some technology. To attempt to characterise intelligence independently of those technologies seems to be a fundamental error (Olson, 1986, p. 356).

In a certain way, Apple recognises that sooner or later computers will be normal tools in schools. In this case, students should not only be technically proficient but also "have a serious understanding of the issues surrounding their larger social effects" (Apple, 1991, p. 75). *Social literacy* must have a considerable importance in the curriculum:

> Where are computers used? What are they used to do? What do people *actually* need to know in order to use them? Does the computer enhance anyone's life? Whose? Does it hurt anyone's life? Whose? Who decides when and where computers will be used? (Apple, 1991, p. 76).

Another point raised by Apple, teacher education and new technologies, must always have a clear goal: teacher education is about "*skilling*", not *deskilling*; about giving power to control technology, not giving technology to control teaching.

There have been many promises of radical change in education for educational technologists, researchers and computer enthusiasts. For example, one of the early advocates of computers in education, Patrick Suppes, wrote in 1966 that "in a few more years millions of school children will have access to what Philip of Macedon enjoyed as a royal prerogative: the personal service of a tutor well-informed and responsive as Aristotle" (quoted by De Corte, 1994, p. 206). We know now that is not feasible, at least in the foreseeable future. The enthusiasm for intelligent tutoring systems is something of the past. In the 1990s, a "clear transition has been initiated in educational computing in general (…) toward supportive systems that are less structured and less directive, that are more focussing on coaching and scaffolding" (De Corte, 1994, p. 116). Groups such as the group that worked with the Education Technology Center in Harvard between 1985 and 1995 have initiated this perspective. Their goals were, for the time, counter-current, but are now dominant. The Harvard perspective was based on four principles:

> **Goals**: Focus on key concepts and on the overall nature of knowledge, evidence, and inquiry in a discipline.
>
> **Teaching Approaches**: Help students develop a deep understanding of the subjects they study by taking into account their prior theories and by integrating teacher-directed instruction with opportunities and challenges for critical inquiry.
>
> **Technology**: Use technologies selectively to make a distinct contribution to teaching and learning, for example, to present dynamic models of key ideas or to enable students to participate in disciplined inquiry.
>
> **Implementation**: Design technology-enhanced teaching modules and approaches that can be gradually and gracefully integrated into existing curriculum and practice (Educational Technology Center, 1988).

As we can see in these statements, *technology is not a goal in itself* but a selective contribution "to make a distinct contribution to teaching and learning". This contribution can, in many circumstances, be a "Trojan horse" to change education (Schwartz, 1993b). But, as many authors point out, is the *teacher* that really can make the difference in creating powerful educational environments *with* technology. Or, as Hooper, one of the British pioneers of research in computers in education, stated in an interesting paper (*Computers and sacred cows*):

> (…) the teacher as human being is both the form and content of education, both means and end (Hooper, 1990, p. 4).

Compared with other institutions and areas of work, schools are less influenced by technologies. Cuban (1993) presents two reasons for this: one is what he calls "cultural beliefs" about what teaching is and how learning and teacher-student relationship occurs. He argues that popular views of proper schooling emphasise the role of the teacher, not the role of a machine. A second reason is the organisation of the age-graded school, with sequences of 50 minutes classes, that "has profoundly shaped what teachers do and do not do in classrooms". To Cuban, using computers in traditionally organized schools is an almost impossible task. Only a minority of "enthusiastic teachers" can do it. And, in present circumstances, he is probably right, even with innovations such as using computers as data logging tools, as Rogers pointed out:

> Despite the fact that the software and hardware tools for this type of activity are now refined and very easy to use, school science departments have been rather slow to adopt data logging technology. The reasons for this reticence are often cited as a mixture of limited funds, limited time and limited training opportunities for science teachers. It is also possible that limited awareness of the learning benefits has caused a failure to gain the professional commitment of teachers (1996, p. 130).

It can be difficult to "gain the professional commitment of teachers" when research shows that the effective use of Information Technologies (IT) needs *substantial demands on teachers and schools* as pointed out in a large scale evaluation in UK:

> Overall indications were that in particular circumstances the use of IT had a highly positive impact on children's achievement, but this was not without substantial demands on teachers and schools (Johnson, Cox, & Watson, 1994).

This high level of demand, this "high threshold of effort" (Wilson, 1997, p.24) for teachers and schools is recurrently considered as major obstacle for the regular use of computers.

But technological innovations in schools have always been very slow and not only due to teachers but also to cost. Computers are expensive commodities as paper was some centuries ago:

> The high cost of paper stimulated the use of substitutes—the wax tablet, the slate, the smooth wooden board, as well as the board painted black. Even these developments were slow. Brinsley mentions the blackboard in his *Ludus Literarium* of 1612, Comenius had pictured one in 1658, but we have no record of its use in schools until about 1800, and no mention is made of slates for individual pupils until about 1815. Again and again the records of the early schools disclose the complaints of the parents over the cost of each new innovation, and the introduction of student slates was the cause of public disturbances (Brooker, 1949, p. 12).

Other difficulties with the introduction of a proper use of computers can be related to the fact that empowering environments—such as Logo—have been replaced by more appealing multimedia presentations. According to Robertson (1998), this is due to the fact that investment in support for teachers and curriculum development based on educational research on computers has almost disappeared. Schools manage their own budgets and buy directly from publishers—the trend is buying what is more "attractive", not what can help explore the potentialities of the technology to help create powerful learning environments.

It seems reasonable to admit, with Joyce (1974, p. 411) that the "structure of the school is in many senses the medium of instruction—it facilitates certain kinds of learning modes and inhibits others". Joyce gives the example of programs such as "Sesame Street" that "would not have anywhere near the effect they are having if there were not television sets in most homes and if the parents were not delighted to have the children occupied before them". Would it be possible to change the structure of the school? A change in the direction of more active engagement of learners in their own learning, a change in the direction of the transformation of schools and classrooms in communities of situated practice, in the sense given by Brown, Collins & Duguid (1989)?

Computers are commonly associated with fun and enjoyment, including in learning environments. Learning can certainly be fun but, in most cases, is slow and difficult. If we want students and teachers to use computers as learning tools they must be aware that popular myths can be true for games and browsing through most of multimedia titles, but are certainly not true when reflection and hard work is needed (Stoll, 1995). Using computers as scientific tools is a demanding experience, as is all scientific work, both for students and teachers.

Recent research in innovation and knowledge dissemination tend to insist on "a social constructivist approach to dissemination and use of knowledge" (Hutchinson & Huberman, 1994, p. 43). Users are not passive recipients of novelties and it is not possible to transfer expertise and information as we transfer bits and data. As a matter of fact,

> (…) work on organizational life has shown clearly that, within any given social setting, there are a sufficient number of tensions, differences in perception, differences in influence or authority, etc., to preclude any straightforward communication of information or innovation. A constructivist view of knowledge use also shows us that users must transform inputs simply to apprehend them, even if they are as unaware of the process (…). When we look at outcomes, then, we must assume that users have reconfigured their understanding and use of a given practice simply to integrate it into their repertoire (Hutchinson & Huberman, 1994, p. 43).

No longer do we need short-term programs that assume that innovation is granted because is has proven with the enthusiastic. We need programs that “encourage cumulative improvement over the long haul” (Holton, 1994), committed to ongoing slow but clear change. “Cumulative improvement” is, certainly, a more reasonable view to envision how computers tools will be assimilated and change learning and teaching.

Our modern institutions are profoundly dependent on abstract systems, what Giddens calls “expert systems” (1991). Computers and computer networks are good examples of these expert abstract systems. Their potential and social impact is enormous and will increase as technology advances. But, as Papert pointed out twenty years ago, “there is a world of difference between what computers can do and what society will choose to do with them” (Papert, 1980, p. 5). In the near future, we all face the challenge to use technology to empower learning (and all other human activities), not to create any kind of Aldous Huxley *Brave New World*, where machines control everything, dehumanising schools and learning.

1.4 The Importance of this Thesis: Research and a Guide to the Thesis

1.4.1 Can we do it?

This work received a considerable influence from the work done at the Educational Development Center, in Harvard. A recurrent theme of their work was expressed by the idea that research must be meaningful both for the advance of the theory of education and for the classroom practice, for teachers and students in real schools. This work shares a vision, an old vision that many educators have had at least since Socrates: *learning can be an active process for students*—and for teachers alike!—and that the result of this process is *meaningful learning*, not *rote learning*, in Ausubel’s sense (Ausubel, Novak, & Hanesian, 1978).

I argue that tools, and computer tools in particular, can have a determining role in the realization of this vision, *at least from the upper secondary school onwards* where science and mathematics are more and more formal and the power of many ideas comes precisely from this formalisation. But, it is evident that, as in all fields, *tools are not enough*: innovation is a systemic process with many variables influencing one another.

The experience of the “Supposers” (Schwartz, Yerushalmy, & Wilson, 1993) show how new visions and tools can help teachers transform their classrooms in more active settings, in the learning of geometry. Now, active learning in geometry through exploration using computer tools is widely recognised as a normal way of learning, helping students’ develop insight about properties of geometrical objects and their relations (NCTM, 2000). Schwartz asks if we can “do this sort of thing in other areas of mathematics education” (Schwartz, 1993a, p. viii) and I add *in other areas such as physics education.* The work presented in this thesis is an attempt to show that “we can do it” and that learning physics will never be the same after the introduction of modelling tools.

In a certain way, something similar happened one decade ago with the introduction of microcomputer-based laboratories (MBL). Physics teachers and physics education researchers now take for granted that sensors, interfaces and data logging software are normal tools in a classroom laboratory, as they are in a research laboratory. But modelling software of all kinds (such as *Matlab*, *Mathematica*, *LabView*, *Extend*, etc.), also usual in research laboratories, are not considered normal tools in school laboratories—and most probably they will never be because of their complexities and broad uses, very far from the “zone of proximal development” (Vygotsky, 1978) of students and teachers. But modelling software, such as Modellus, has many relevant features of the most powerful professional software (e.g., the ability to solve systems of differential equations) but doesn’t have the complexities of the professional software (e.g., a student can start doing complex visual representations of a model without any programming in Modellus, differently from professional software, where it takes a lot of effort to make them). And more: since Modellus is based on research on learning, many common difficulties can easily be addressed, helping students construct their own knowledge, as I try to show in Chapter 4.

Will schools use modelling software, as they use MBL? There is no straight answer to this question but it is reasonable to expect that that it will depend largely on:

1. The curriculum (the official curriculum and that presented in school textbooks).
2. The evaluation procedures, with a special importance of final examinations.
3. The professional development of teachers and their familiarisation with computer tools.
4. The organisational conditions of schools, such as the availability and functionality of computers in laboratories, resource centres and homes.

This thesis gives a framework to think about all these issues but I’m aware that it is *not the solution to all problems*, including *lack of resources and ideas on how to change practices in physics teaching in schools*. Or, as Buchmann wrote:

> Supposed implications for practice—as recommendations for action—are neither deducible from nor logically contained in research results. Action and decision depend instead on moral frameworks and networks of power and authority that affect the work of practitioners, as well as on legal and political knowledge, the resources at hand, and (importantly) know-how (Buchmann, 1992, p. 325).

1.4.2 Research questions

From the previous sections, it is now clear that this thesis is both a *narrative of a search* and a *research report* on the development and evaluation of a framework and a tool to think about modelling in the physics curriculum.

The *narrative* intends to show how many years of software development ended in a general tool, Modellus. This development received inputs from many fields of knowledge: studies in human-computer interaction, studies in learning and concept development, and from research in science and mathematics education. In the part of the thesis more related with the *narrative*, the research questions come out from work in specific contexts and were not formulated as *a priori* questions. The most relevant research questions on the *narrative* can be formulated as:

1 What are the appropriate features of direct-manipulation software that allow students to express and explore mathematical and physical ideas?

2 What are the "primitive objects" of that software?

3 What methodology is most suitable to design this software?

4 How can exploratory software be included as a normal tool in the curriculum?

A second part of the thesis reflects on issues related to curriculum design in physics education using computers, and modelling software in particular. In this part of the thesis I discuss some relevant aspects of what can be a *framework for modelling in the physics curriculum*, from high school to university, with special emphasis in the transition between secondary schools and higher education. In particular, I'm interested in describing and justifying answers for the following:

5 When are modelling approaches fundamental to knowledge building in physics?

6 What features must have a physics curriculum based on the intensive use of computers for modelling (and for data logging)?

7 How does knowledge building in physics relate to knowledge building in mathematics?

8 What features must the classroom culture have when modelling and data logging are as ubiquitous and pervasive as books or other traditional tools in the classroom?

A third part of the thesis shows some *empirical evidence*, both from students and teachers, about *how students learn with Modellus and how effective that learning is*. In particular, I'm interested in getting evidence for the following:

9 Can students use Modellus without being disturbed by the specific features of the software?

10 Can students have exploratory learning approaches when using Modellus?

11 What do physics and mathematics teachers, and researchers in education, familiar with Modellus, think of (a) Modellus features and use? (b) The place of modelling in the physics curriculum? (c) The interactions in the physics and mathematics curriculum when adopting a strong emphasis on modelling?

The first and the second set of questions are explored mainly in chapters 2 to 5, and 8. The third set of questions are the guiding questions for chapters 6 and 7.

1.4.3 The nature of research in education and research in computers in education

Until one decade ago, most of the research on the use of computers in education was based on an experimental agricultural research metaphor: research questions asked whether "using computers" (the good "seed") can led to "better results" ("better crops") than "traditional methods" (not so good "seeds"). Synthesis of research, using complex methodologies such as meta-analysis, demonstrated "a typical learning advantage for 'newer' media of about 0.5 standard deviations on final examination performance, compared with 'conventional' treatments" (Clark & Salomon, 1986, p. 466). This learning advantage becomes smaller as the time spent with computers increased (Clark & Salomon, 1986).

The validity of these experimental studies has been questioned by many authors (e.g., Berger et al., 1994; Clark, 1990; Eisenberg, 1995; Hooper, 1990; Salomon, 1974, 1990). E.g., Hooper argues that

> (…) control group experiments in education leave much to be desired. First and most obvious of all, the holding of variables constant whilst one is varied is all but impossible. Interpersonal communication—between teacher and taught—is full of unknown idiosyncrasies. It is highly subjective with each partner to the communication making his own meanings. Secondly, the Hawthorne effect in education remains substantial—particularly with computers around (Hooper, 1990, p. 6).

As Hooper says, the Hawthorne effect is particularly foreseeable when computers are new and fascinating tools for students (and that is still much the case now, almost twenty years after the start of the personal computer revolution).

Gross media comparison studies can be considered as part of a teaching effectiveness program of research (in the sense given by Shulman, 1986). This kind of program of research, in particular if the dominant methodology is experimental, comparing groups with and without computers, is now questioned by almost all contemporary authors, and has led to a decline in comparison studies (Berger et al., 1994). It is not possible to compare learning "with computers" with "learning without computers" (whatever this means) since there are many other variables that change from one setting to another, besides the presence/absence of the computer:

> Computer tools carry with them implicit assumptions about self-guided exploration and design, even playful activity, team collaboration, integrated curricula, mutual consultation, and teachers' orchestration of activities rather than teacher domination (Salomon, 1992, p. 251).

It is interesting to note, as Hooper does, that it is a paradox that, for so many decades, educational research has been influenced by a misinterpretation of the systems engineering model of research:

> The simplistic notions of causality, certainty, proof, linearity, are no longer believed in post-Einsteinian science—if they ever were. Ambiguity is at the centre of modern

> science—light being both waves and particles is the classic example, or Werner Heisenberg's principle of uncertainty. Heisenberg's contention that the act of observation alters the objects being observed, has its obvious analogy in the arts—we call it 'subjectivity' (Hooper, 1990, p. 6).

If research has an "arts perspective" (Hooper, 1990), i.e., a perspective that "focuses attention on values" instead "on proof", where "science's attention is", we can create a much more useful perspective of what it means to use computer tools for learning, since "education is fundamentally about values, not about skills and facts. And values, we all know (…) can't be proved". That means, we can't prove—and it doesn't make sense to prove!—that using computers is better then not using them. And since the computer is becoming so common, "to compare its use to current non technological practice would be equivalent to comparing instruction with books to the same amount of instruction without books 20 years ago" (Berger et al., 1994, p. 486).

Comparison studies can have only a residual importance in very specific contexts. And, as a matter of fact,

> (…) technology provides an instructional approach that cannot be matched by traditional non technological instruction. The ability to see graphs develop in real time, to see multiple representations of phenomena from macro to micro, cannot be duplicated even with the best lecture or reading. Thus, it is even less appropriate to try to design a comparison study (Berger et al., 1994, p. 486).

The decline in comparison studies has been accompanied with an increase in the study of what is going on when students learn with computers. More recent studies focus on research questions about qualitative understanding of learning, about concept development, and about personal and cooperative/group learning strategies. This new program of research—or *paradigm of research* (in the sense of Husén, 1988)—emphasizes holistic, qualitative and interpretative approaches. The gathering of evidence from teachers and students to support the ideas presented in this thesis must be seen in the light of this paradigm.

All educational research efforts have powerful limitations. As Shulman pointed out, "there is no 'real world' of the classroom, of learning and of teaching. There are many such worlds, perhaps nested within one another, perhaps occupying parallel universes which frequently, albeit unpredictably, intrude on one another" (Shulman, 1986, p. 7). We may study students, classrooms, schools. We may "become involved in these different worlds as elements of our puzzle because we most often must make a particular level or strand the subject of empirical study, but then we attempt to infer properties of other strands from the one we have investigated" (Shulman, 1986, p. 7). Can our inferences be valuable? Can our inferences be sustainable in other contexts? Shulman also gives a reasonable answer to these questions:

> The essence of the puzzle lies in recognizing that no benevolent deity has ordained that these parallel lives be consistent with one another, nor that the principles found to work at one level must operate similarly at others (Shulman, 1986, p. 7).

1.4.4 A guide to the thesis

After chapter 1—the chapter that explains the research problem and provides an overview of the work undertaken and of the global constraints that shaped the thesis and its line of argument—follows chapter 2, *Computers in Physics Education (with a Note on Mathematics Education)*. This chapter briefly describes the evolution of the use of computers in physics teaching (and in mathematics, when relevant to understand their use in physics) and analyses the potential and the pitfalls of the different modalities of use of computers in the physics curriculum. It also synthesizes the research literature about the effects of using computers in physics education and how the nature of physics provides a framework for modelling.

Chapter 3—*Modellus: A Tool for Doing Experiments with Mathematical Objects*—describes in detail the modelling software that is fundamental in this thesis. It also shows how it can be used, and how teachers and curriculum developers are using it all over the world, in different areas of the science and the mathematics curriculum, with emphasis on the physics curriculum at secondary and at the undergraduate level. This chapter also discuss design issues that guided most of the software I developed between 1985 and 1995, and presents a very brief description of each title. In the last part of this chapter, I argue about the need for a coherent model for the design of exploratory software, based either on my own arguments or on the work undertaken by others. A model for designing this kind of learning tool is proposed.

Chapter 4—*Modelling and Modellus in the Physics Curriculum: Some Examples and a Framework*—presents a framework for modelling as a fundamental activity, integrated with more traditional laboratory work, in the secondary and undergraduate physics curriculum. After the analysis of concepts and type of tools, and its uses, I suggest issues for horizontal and vertical approaches to modelling.

Chapter 5—*Teachers, Modelling and Modellus*—show what teachers, both from secondary schools and university, and also curriculum developers and science education researchers (from different continents and cultures), say about the role of modelling in physics education and how Modellus can help fulfil that role.

Chapter 6—*Students Using Modellus*—reports a study with first-year undergraduates with insufficient knowledge of physics. In this study, I had first-hand experience with students confronting themselves with Modellus and modelling activities, in active learning exploratory environments. The study gave evidence to support claims about the potentialities both of Modellus and of a physics curriculum where modelling and exploratory learning with computers are as natural as doing observation and experimentation of physical phenomena.

This thesis *is not* about social and leadership issues on how to improve the quality of teaching and learning both in schools and universities. But it has become clear to me that proposing somewhat *radical* changes in the culture of teaching and learning can only deserve any attention if it is followed by reflection on strategic problems on innovation in education. Based on the work of Cuban and on the literature on professional development of teachers, I argue in the last chapter (*From Theory to Practice: Computers, Modelling and Modellus in Education*) about how the proposed changes can be achieved. The reflection in this last chapter also tries to take into consideration the history of “educational technology” in the last one hundred years with its unfulfilled promises and expectations.

1.4.5 Who can benefit from this thesis?

As Landsheere (1988) wrote, “(…) education is an art. That is why advances in research do not produce a science of education, in the positivist meaning of the term, but yield increasingly powerful foundations for practice and decision making”.

Writing a thesis is both a personal academic obligation and an obligation to the communities that made the work possible and, at the end, must have a proper benefit, directly or indirectly. I’m thinking of society as a whole but also of students who many times struggle to have some knowledge of physics (and mathematics), without the opportunities to carry out most of their own explorations and investigations, because they don’t have the necessary tools to help them. I’m also thinking of teachers, who also struggle with difficulties to make sense for themselves of difficult concepts and to devise ways to help students learn and explore.

This is research done with a clearly identified target population: teachers and students. But it is also a small contribution to creating new knowledge on learning and teaching with computers, knowledge that is much more than just “useful” for teachers and students. My hope is that it can also be useful for a better understanding of how people learn when using “cognitive artefacts”, and about how to create them.

Chapter 2 Computers in Physics Education (with a Note on Mathematics Education)

> The point is that exploratory software is *not* simple and its use is not simple. It is easily misrepresented and its purpose easily abused by inadequate conceptualisation leading to crude means of evaluation. Understanding and facing up to this complexity is the only way out of the cycle of inertia; the only chance of realizing the potential of exploratory software. This is the challenge we need to face (Hoyles, 1995).

2.1 Introduction

This chapter presents a description of the different modalities of use of computers in physics education and a synthesis of the research done in the last decades. It also discusses related research in mathematics education, wherever it is important to understand the use of computers in physics teaching.

There are four major trends in the use of computers in physics teaching:

1. presentations/texts/tests;
2. data logging (or computer-based laboratory systems);
3. simulation;
4. modelling.

The first trend, *presentations/texts/tests*, is exemplified by titles such as *Interactive journey through physics* (Schwarz & Beichner, 1997). The suggested use for this type of titles is “learn/reinforce physics concepts, practice problem solving, study for exams or as a visual/interactive study guide” (Schwarz & Beichner, 1997, p. 10). The authors suggest that *Interactive journey through physics* can also be used in classroom to present animations, videos and simulations. This trend had some commercial impact and it is now common to find these titles in bookstores and other retailers. Most titles are available only in CD format but there is an increasing interest in creating similar materials on the Internet (as can be seen at the website http://www.riverdeep.net [retrieved August 25, 2002], for interactive presentations, texts and tests, and also simulations, or at the website http://webassign.ncsu.edu [retrieved August 25, 2002] for assessment).

The second trend, *data logging*, i.e., the use of computers and sensors as laboratory tools—*computer-based laboratories* also known as *MBL*, *microcomputer-based laboratories*. Most, if not all physics laboratories suppliers have now their own interfaces and sensors and more and more schools are regularly integrating them as routine equipment in experiments, at least in some experiments such as those related to motion. For motion experiments, motion sensors, based on ultrasound reflection, are really a significant advance over traditional ways of studying motion (ticker-timers; Atwood machine, etc.).

A third trend is *simulation* software, best represented by titles such as the well-known *Interactive Physics* (Knowledge Revolution, 1998)—mainly for mechanics simulations—or *Albert* (Wullenweber, 1996), a collection of simulations in many fields of physics. Some college books, at least in the USA, have companion books with activities based on simulation software and some courses include these activities as complements to laboratory work. Simulation software is also becoming increasingly more available directly on the Internet, using Java applets (e.g., the collection of simulations know as Physlets available on http://webphysics.davidson.edu [retrieved August 25, 2002]) or special purpose browsers (such as the one available at http://www.riverdeep.net [retrieved August 25, 2002]).

The fourth trend is *modelling*[1]*, the use of computers to create and explore physical* phenomena *and* the underlying mathematical relationships behind the phenomena.

Schecker pointed out (1993, p. 102) that "the use of computers in physics teaching is dominated by data loggers and simulation programs". This is still probably true now as it will be in the near future, but *modelling* has the potentiality to change this state of affairs as we can see with the new curriculum Advancing Physics, where modelling is considered an essential aspect of the course:

> A focus on modelling enables the course to look at and compare a considerable variety of phenomena. This not only broadens the perspective on physics, but also offers a base of concrete experience with physical phenomena which guides thinking and checks out ideas. Thus work on modelling should—oddly it may seem—have a large component of very physical activity (Advancing Physics Project, 1998).

---

[1] Also *modeling* in US English.

2.2 From Programming Languages to Educational Applications

2.2.1 Changes in computer technology and in views of education

The first advocates of the use of computers in physics education started in the early 70s (e.g., Bork, 1975). During the last 30 years, significant changes occurred in computer technology in general and in the use in physics and physics teaching in particular. These changes are not only due to technological development but also to new conceptions of teaching and learning.

Major technological changes include:

1. an enormous increase of computational power, simultaneous with an equally enormous increase of access to computers due to size and price reduction;
2. a shift from text and command based interaction to graphical user interfaces, with multimedia capabilities;
3. a shift from stand-alone computers to computers that can interact locally and globally, either with other computers or with laboratory equipment.

Simultaneous with these technological changes, educational and other social researchers contributed to establish a now dominant view of teaching and learning (Bruner, 1996), where the word "constructivism" seems to express the dominance of *learning over teaching*. This dominant view of teaching and learning combines a view about the genesis of knowledge in individuals—*knowledge is the result of a process of personal construction and involves epistemic conflict, self-reflection and self-regulation* (Forman, 1988)—with a view about the dialectic between empiricism and rationalism—*knowledge stands on previous conventions, that support and guide observation and theoretical formulations*.

Other authors, such as Papert (1994), a pioneer in the field of computers in education, use the word *constructionism* instead of *constructivism* to emphasise the importance of the child's construction of meaning when involved in creating things that are *personally* meaningful.

The different views of using computers in education were confronted in a well-known pioneer book with a title that is a synthesis of the different views: *The computer in the school: tutor, tool and tutee* (Taylor, 1980). In his introduction, Taylor describes the *tutor* function as presentation of subject material, questioning and control of the path the student must follow in order to master the subject, and tailoring the presentation to accommodate student differences. The *tool* function is exemplified by "a calculator in math and various science assignments, as a map-making tool in geography, as a facile, tireless performer in music, or as a text editor and copyist in English" (Taylor, 1980, p. 3). Finally, "To use the computer as a *tutee* is to tutor the computer; for that, the student or teacher doing the tutoring must learn to program, to talk to the computer in a language it understands" (Taylor, 1980, p.4). When using a computer as a *tool* or as a *tutee*, students and teachers are using open-ended software (*open environments*) where the control is in the hands of the user, not of the software.

In an update to his paper (Taylor, 2000), Taylor uses four new categories: *access*, *collaborate*, *communicate*, and *experience*. The following table summarizes the meaning of each of these categories:

Access	Students and teachers can access directly a broad and extensive body of information around the world, far beyond the limits of their own books and libraries.
Collaborate	Students and teachers can collaborate not only with each other, but also, as appropriate, with those outside their school, even across the world, to develop and refine ideas, as they construct and refine projects realizable only through collaboration.
Communicate	Students and teachers can communicate with peers and with experts of all sorts anywhere in the world, through a global language, English, or one of a number of broadly used regional languages, thus potentially broadening their own understanding of life through reference to those with different relevant experiences.
Experience	Students and teachers can experience things about the world not just through these first three possibilities but also through radically improved and network resident simulations of all kinds.

2.2.2 Papert' s Logo and its successors

Programming languages were the first kind of computer open environments used in education, with young children and high school students, mainly in mathematics education. Behind Basic, a language specially created to make programming available to the general user, the most disseminated was Logo, created in early 70s at the MIT by a group led by Seymour Papert, the author of *Mindstorms* in 1980, now a landmark in the use of computers in education. The enthusiasm over the ideas presented by Papert spread over the world for at least a decade. This enthusiasm led to many projects, from Costa Rica to France, from the US to Australia, from Portugal to Chile. Twenty years after, there are many versions of Logo, in many languages. The "official" version, from Logo Computer Systems Incorporated, Microworlds, is a kind of multimedia development environment for children, using a mixture of graphical programming features and traditional Logo syntax, allowing multitasking, a feature considered by Michael Tempel, President of the Logo Foundation, as "the most significant change in the Logo programming language since it was first implemented in 1967" (Tempel, 1998). Files created with Microworlds can be published on the Internet and read using a browser with a special plug-in.

The basic "Logo philosophy"—the computer a "tutee" for learners—is well represented by the following quotation from *Mindstorms*:

> In my vision, the child programs the computer and, in doing so, both acquires a sense of mastery over a piece of the most modern and powerful technology and establishes an intimate contact with some of the deepest ideas from science, from mathematics, and from the art of intellectual model building (Papert, 1980, p. 5).

Papert argued that “Learning to use computers can change the way they [children] learn everything else” (1980, p. 8) since it is the most powerful tool one can have to help students become formal thinkers, in a Piagetian sense:

> Stated most simply, my conjecture is that the computer can concretise (and personalize) the formal. Seen in this light, it is not just another powerful educational tool. It is unique in providing us with the means for addressing what Piaget and many others see as the obstacle which is overcome in the passage from child to adult thinking. I believe that it can allow us to shift the boundary separating concrete and formal. Knowledge that is accessible only through formal processes can now be approached concretely. And the real magic comes from the fact that this knowledge includes those elements one needs to become a formal thinker (Papert, 1980, p. 21).

The impact of Logo, thirty years after, is much diminished. For example, Tempel wrote recently:

> Logo has been around for 30 years. It has not taken the world by storm. It has not gone away. On the one hand, simply surviving for three decades in the fad-frenzied world of educational technology should be considered success. But those of us who had hopes that Logo would spearhead a major change in education are naturally disappointed (Tempel, 1998).

Why that happened? Sure there are a multitude of reasons, some institutional (Hoyles, 1995) and some related to Logo itself. For example, the basic Logo metaphor is programming and the many advantages attributed to programming by Papert and others are strongly controversial (see, e.g., Clark, 1992).

Classic Logo environments have been used for teaching physics. Papert himself mentions the use of Logo to teach physics in *Mindstorms* and other members of the MIT group that helped create special Logo environments for teaching physics (e.g., diSessa, 1982). In 1985 was published a book of Logo projects for physics college or bright senior high school students (Hurley, 1985) and as recently as 1997 a group of Argentine researchers created an improved version of Logo, Graphic Logo (Grant & Antueno, 1997) using a physics paradigm instead of a geometry paradigm for turtle motion. This version of Logo emphasizes simulations and simultaneous animations and includes primitives (Logo basic instructions) to define acceleration, friction, collisions, gravitation, etc. A typical set of instructions (three Logo procedures) in Graphic Logo looks like the following:

```
TO ROCKET1
     TELL 1
     SETPOS [-100 0]
     SETSHAPE 1
     VISIBLE
     SETHEADING 90
     SETSPEED 10
     END
TO ROCKET2
     TELL 2
     SETPOS [100 7]
     SETSHAPE 2
     VISIBLE
     SETHEADING 270
```

```
          SETSPEED 15
          END
     TO STARTUP
          ROCKET1
          ROCKET2
          ONBUMP 1 [REBOUND]
          ACTION
          END
```

This program creates two sprites (i.e., moving objects on the screen) and “when they collide they rebound according to the rules of elastic collision of two bodies of equal mass. This was established by means of primitive ONBUMP” (this example is presented in the Graphic Logo web page at http://nalejandria.com/fundaustral [retrieved August 25, 2002]).

Another effort to create a more powerful version of Logo, more suitable for physics and other scientific disciplines, was made by diSessa and the Boxer project at Berkeley. diSessa (1997, p. 47) describes Boxer as a “new genre of educational software”, an “open toolset”, i.e., “an open collection of tool-styled software units that are aimed toward learning in some particular subdomain, like constructions in geometry, system dynamics, particular pieces of ecology or evolutionary theory”. Both Boxer and Logo assume “that programming, in some form, is essential in truly liberating the computer’s power as a learning tool” (diSessa, 1997, p. 8). A Boxer file, like a Logo file, looks uncommon for a user not familiar with programming (a program in Boxer is a set of screen boxes, each with their own set of instructions).

As recently as 1996, still another effort has been made to create a general purpose programming language, suitable for education, by Travers (1996). Travers created a visual programming environment, LiveWorld, based on programming with *agents*, “as a means to help people create worlds involving responsive, interacting objects”. In this context, “an agent is a simple mechanism intended to be understood through anthropomorphic metaphors and endowed with certain lifelike properties such as autonomy, purposefulness, and emotional state” (Travers, 1996).

As mentioned above, the *programming metaphor* (a metaphor that stresses the importance of *writing commands* to create animations and simulations) as the basic metaphor for computer educational tools has been challenged and criticized by many researchers, teachers and curriculum developers. Even researchers who initially have been involved with Logo, changed views. For example, Cockburn and Greenberg (1998, p. 780) describe how they “moved away from Logo and the command-line interface towards a fully graphical system” in order to “provide a seamless interface that allowed all the student’s cognitive effort to be directed at the contents of the microworld”, instead of the particularities of the Logo syntax.

Logo has never been popular in the physics education research community, with physics teachers, and curriculum developers, despite efforts of Logo advocates to popularise Logo in other subject areas than mathematics. Curriculum guidelines for Physics, in Portugal or England, for example, never mentioned Logo when mentioning computers.

2.2.3 Pascal and programming utilities

A different approach was taken Joe Redish and his group at the University of Maryland. They started from the assumption that "The computer has changed the way the physics professional carries out his or her job" (Redish & Wilson, 1993, p. 222). Comparing student's activities with those of a professional physicist, they conclude that:

Students:	*Professionals:*
Solve narrow, pre-defined problems of no personal interest.	Solve broad, open-ended and often self-discovered problems.
Work with laws presented by experts. Do not "discover" them on their own or learn why we believe them. Do not see them as hypotheses for testing.	Work with models to be tested and modified. Know that "laws" are constructs.
Use analytic tools to get "exact" answers to inexact models.	Use analytic and numerical tools to get approximate answers to inexact models.
Rarely use a computer.	Use computers often.

In order to reduce these significant differences between students and professionals, and arguing that "Physics is not an exact science, rather, it is a science where we believe we understand the accuracy of our approximation" (Redish & Wilson, 1993, p. 223) they propose that students should solve physics problems using programming languages, like Pascal. In order to reduce the programming load, they created modular programs—the "programming utilities"—that can be linked to create more sophisticated programs, and examples of self-explaining programs:

> We set up utilities for interactive input and for graphics output, and we provide self-documenting sample programs that allow even non-programmers to learn by example and to begin to build programs themselves without extensive training (Redish & Wilson, 1993, p. 226).

Using this programming environment, "many of the professional skills traditionally short-changed at the introductory level can be introduced", "more realistic problems may be treated than in the traditional approach", "contemporary topics may be introduced at an early stage", and "students may begin designing and carrying out their own research, even in the introductory course" (Redish & Wilson, 1993, p. 228).

A typical program can look like the one presented in Redish & Wilson (Redish & Wilson, 1993). The physical computations, the core of the program, are done with a Pascal procedure that receives, as inputs, the initial conditions and the time step to integrate the equations of motion, using the force law written in a Pascal function:

```
FUNCTION Force(x,v,t:Real) : Real;
    BEGIN
        Force := -m*g - b*v*abs(v)
    END;
PROCEDURE StepRK2(xIn, vIn, tIn, aIn,tStep:Real;
    VAR xOut,vOut,tOut,aOut:Real);
    VAR
```

```
          xHalf,vHalf : Real;
          tHalf,aHalf : Real;
     BEGIN
          tHalf := tIn + 0.5*tStep;
          xHalf := xIn + 0.5*vIn*tStep;
          vHalf := vIn + 0.5*aIn*tStep;
          aHalf := Force(xHalf,vHalf,tHalf)/m;
          tOut := tIn + tStep;
          xOut := xIn + vHalf*tStep;
          vOut := vIn + aHalf*tStep;
          aOut := Force(xOut,vOut,tOut)/m;
     END;
```

They conclude that their approach "opens many possibilities for changing the curriculum. Elements may be rearranged in a more natural order; professional skills may be introduced at an earlier stage than is traditional; contemporary topics such as chaos and quantum theory may be introduced; and students may begin research immediately" (Redish & Wilson, 1993, p.232).

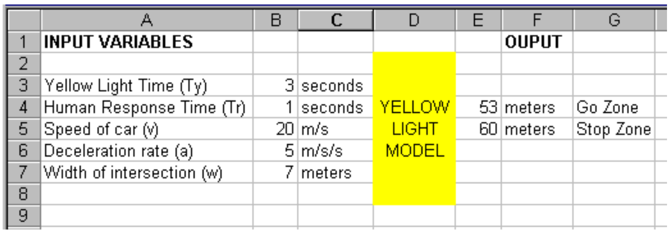
The approach used at Maryland has been used as the basis for a large effort do develop similar utilities and simulations for all fields of physics, for upper-college and graduate students, by the CUPS consortium (Consortium for Upper-Level Physics Software), a group of 27 physicists "with extensive backgrounds in research, teaching, and development of instructional software" (Hiller, Johnston, & Styer, 1995, p. 5).

Using programming languages such as Pascal, with or without "programming utilities" has never been common in the general physics curricula, even with college students. It seems that this kind of approach can only be used by those who developed the ideas, with their own students. Generalizing the use means less time dedicated to physics itself and time is a very scarce resource for teachers. There are only a few books that make use of this approach, and even this use is only a limited use (e.g., Deus et al., 2000). On the web and in an electronic format, there is one general physics book that makes extensively use of programming (in True Basic, an even simpler language) as a basic approach to many physics concepts (Huggins, 2000).

2.2.4 Spreadsheets and mathematical tools

Spreadsheets are one of the most used computer tools in all fields, from science and engineering to economics and other social sciences. The use of spreadsheets is now considered by many curriculum authors as a normal part of the curriculum, even for junior high school physics (see, e.g., the *Active Physics* curriculum, developed in association with the American Association of Physics Teachers). Spreadsheets are particular useful for repetitive computations, such as computing values of functions for different values of independent variables, for numerical integration, where Riemann sums can easily be computed, and for plotting graphs of the computed values. For more advanced students, typical examples of the use of spreadsheets in physics can be seen in the Jay Koop book (Koop, 1991) or in *Workshop Physics* (Laws, 1997). This curriculum has a set of add-on tools for *Excel* and an introductory set of activities (Gastineau et al., 1998).

A typical example of a spreadsheet, from one activity of *Active Physics* (p. 30-31, *Transportation*) look like the one presented in Figure 2.1.

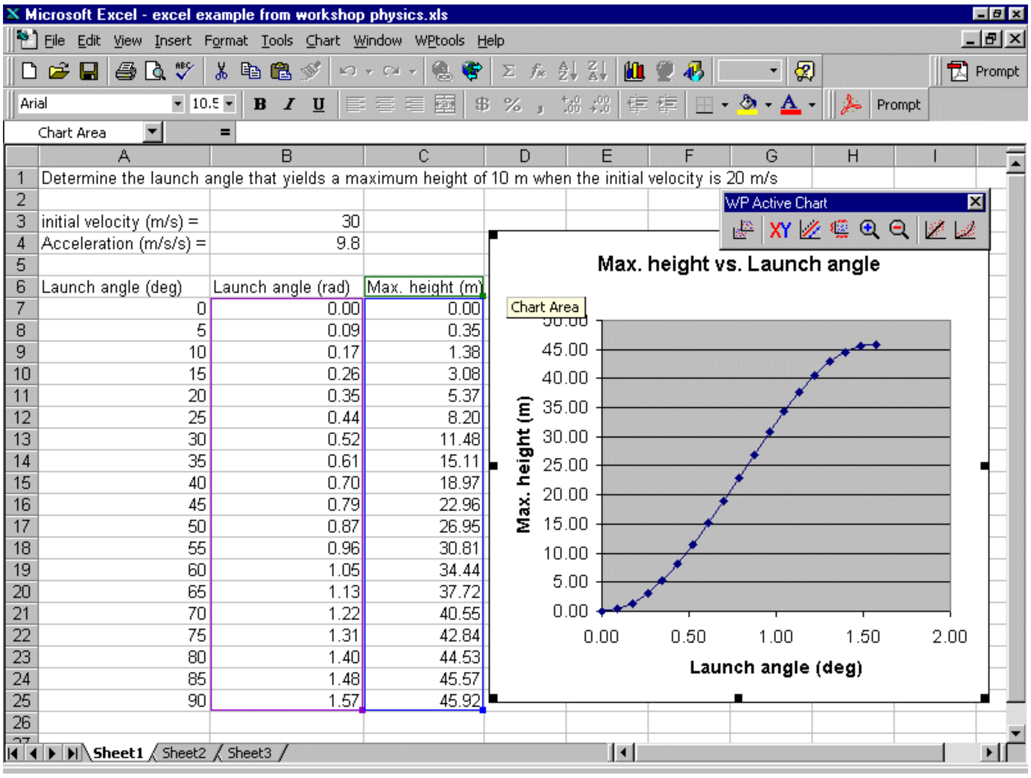


	A	B	C	D	E	F	G
1	**INPUT VARIABLES**					**OUPUT**	
2							
3	Yellow Light Time (Ty)	3	seconds				
4	Human Response Time (Tr)	1	seconds	YELLOW	53	meters	Go Zone
5	Speed of car (v)	20	m/s	LIGHT	60	meters	Stop Zone
6	Deceleration rate (a)	5	m/s/s	MODEL			
7	Width of intersection (w)	7	meters				
8							
9							

Figure 2.1 Spreadsheet used in an investigation in the *Active Physics* curriculum. In the cell E4 and E5 the formulas are (B5*B3)-B7 and (B5*B4)+(B5^2)/(2*B6), respectively.

With this spreadsheet file, students can easily compute safety braking distances when a car approaches a traffic light, and investigate the effect of the different factors that affect this distance.

An example at a more advanced level is also presented in the Figure 2.2: in this example, taken from *Workshop Physics*, students can investigate how the maximum height of a projectile depends on the launch angle, neglecting air resistance.

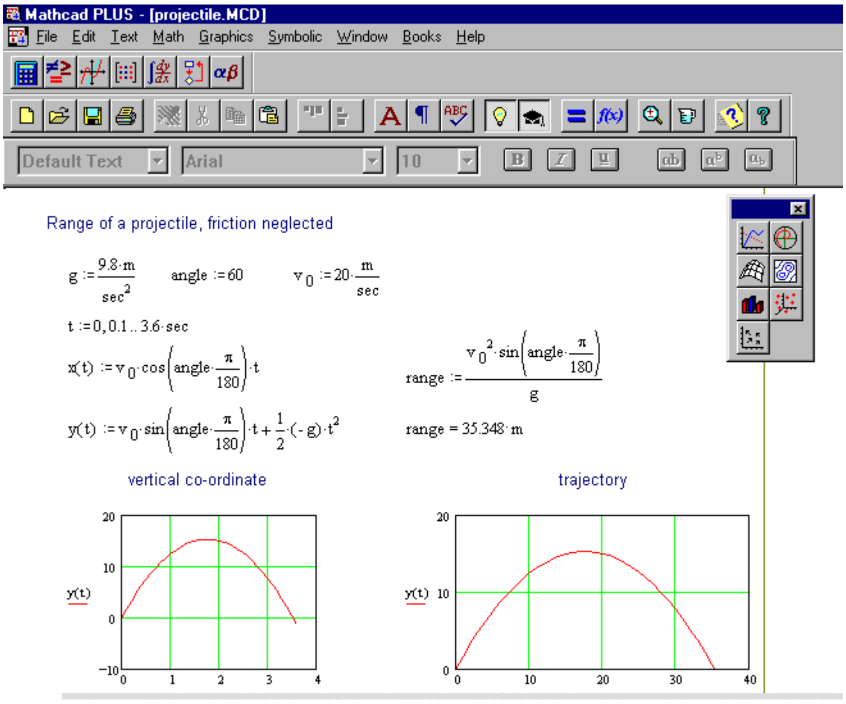


	A	B	C
1	Determine the launch angle that yields a maximum height of 10 m when the initial velocity is 20 m/s		
2			
3	initial velocity (m/s) =	30	
4	Acceleration (m/s/s) =	9.8	
5			
6	Launch angle (deg)	Launch angle (rad)	Max. height (m)
7	0	0.00	0.00
8	5	0.09	0.35
9	10	0.17	1.38
10	15	0.26	3.08
11	20	0.35	5.37
12	25	0.44	8.20
13	30	0.52	11.48
14	35	0.61	15.11
15	40	0.70	18.97
16	45	0.79	22.96
17	50	0.87	26.95
18	55	0.96	30.81
19	60	1.05	34.44
20	65	1.13	37.72
21	70	1.22	40.55
22	75	1.31	42.84
23	80	1.40	44.53
24	85	1.48	45.57
25	90	1.57	45.92
26			

Figure 2.2 An *Excel* worksheet from *Workshop Physics* to compute the maximum height of a projectile as a function of the launch angle. Cells in column C have the formula $h = v_0^2 \sin(\theta) / (2g)$.

Other general mathematical tools are also becoming extensively used in physics education, mainly at college level. During the 90s, these tools became more powerful, having both numerical and symbolic capabilities. The most used are *Matlab*, *Mathcad* and *Mathematica*. All have programming capabilities but, with the possible exception of *Mathcad*, many physics problems need some sort of programming to be solved.

Using *Mathcad* is somewhat similar to using a word processor. Most of *Mathcad* instructions are close to the usual mathematical expressions. The example below (Figure 2.3) computes the range of a projectile, using parametric equations, and shows the trajectory.



Range of a projectile, friction neglected

$g := \frac{9.8 \cdot m}{sec^2}$ $\quad angle := 60$ $\quad v_0 := 20 \cdot \frac{m}{sec}$

$t := 0, 0.1 .. 3.6 \cdot sec$

$x(t) := v_0 \cdot \cos\left(angle \cdot \frac{\pi}{180}\right) \cdot t$

$range := \frac{v_0^{\,2} \cdot \sin\left(angle \cdot \frac{\pi}{180}\right)}{g}$

$y(t) := v_0 \cdot \sin\left(angle \cdot \frac{\pi}{180}\right) \cdot t + \frac{1}{2} \cdot (-g) \cdot t^2$

$range = 35.348 \cdot m$

Figure 2.3 A *Mathcad* file showing the computation of the range of a projectile, the representation of the trajectory and the vertical co-ordinate as a function of time.

To create a *Mathcad* object, such as a graph, or to do a mathematical operation, such as the symbolic computation of a derivative, the user can use buttons and, or, pull-down menu options. Recent versions of *Mathcad* have some animation capabilities.

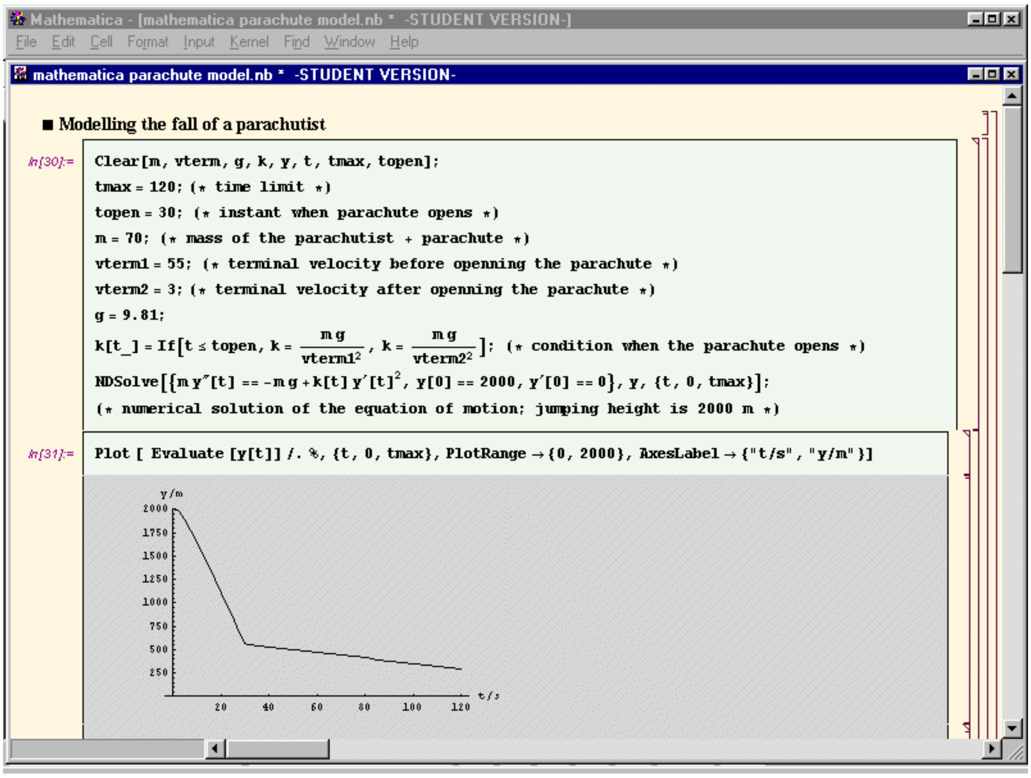
Mathsoft, the company that developed *Mathcad*, also created a limited version but with hundreds of examples targetted at senior high school students studying science. The product, called *StudyWorks*, is a mixture of a mathematical tool with a hypertext and multimedia environment, with links to the Internet.

Mathematica is widely used in physics research—as can be seen in the pictures and calculations in the leading physics journals. Besides research, *Mathematica* is also extensively used in teaching, mainly at college level but also in secondary schools, particularly in mathematics. According to the author of *Mathematica*, Stephen Wolfram,

> "The results are impressive. Over and over again what was once a calculation too lengthy to be reproduced as part of a course now becomes a few lines of *Mathematica* input that can be executed in a matter of seconds. And instead of having to explain in a painful detail the mechanics of the calculation, one can concentrate on the conceptual issues that underlie it" (from the *Preface* written by Stephen Wolfram to the Zimmerman and Olness (1995) book).

Wolfram, besides pointing out how important a mathematical tool can be to foster conceptual thinking, also points out that computational tools "narrowed the gap between physics learning and physics research".

To use *Mathematica* the user must write commands or click on buttons to create them. A typical file can look like the one in Figure 2.4 that shows a numerical solution of a fall of a parachutist. Terminal velocities are used to compute the drag coefficient.



■ Modelling the fall of a parachutist

```
Clear[m, vterm, g, k, y, t, tmax, topen];
tmax = 120; (* time limit *)
topen = 30; (* instant when parachute opens *)
m = 70; (* mass of the parachutist + parachute *)
vterm1 = 55; (* terminal velocity before openning the parachute *)
vterm2 = 3; (* terminal velocity after openning the parachute *)
g = 9.81;
k[t_] = If[t ≤ topen, k = m g/vterm1^2, k = m g/vterm2^2]; (* condition when the parachute opens *)
NDSolve[{m y''[t] == -m g + k[t] y'[t]^2, y[0] == 2000, y'[0] == 0}, y, {t, 0, tmax}];
(* numerical solution of the equation of motion; jumping height is 2000 m *)
```

```
Plot [ Evaluate [y[t]] /. %, {t, 0, tmax}, PlotRange → {0, 2000}, AxesLabel → {"t/s", "y/m" }]
```

Figure 2.4 A *Mathematica* file showing the numerical solution of the height of the fall of a parachutist, before and after opening the parachute.

As can be seen in the above example, to use the software one must be familiar with the syntax of many commands. This syntax is recognised as not easy to learn and some authors even developed a "point and click" interface shell to facilitate the use of *Mathematica* (e.g., *The joy of Mathematica*, Shuchat & Shultz, 1994).

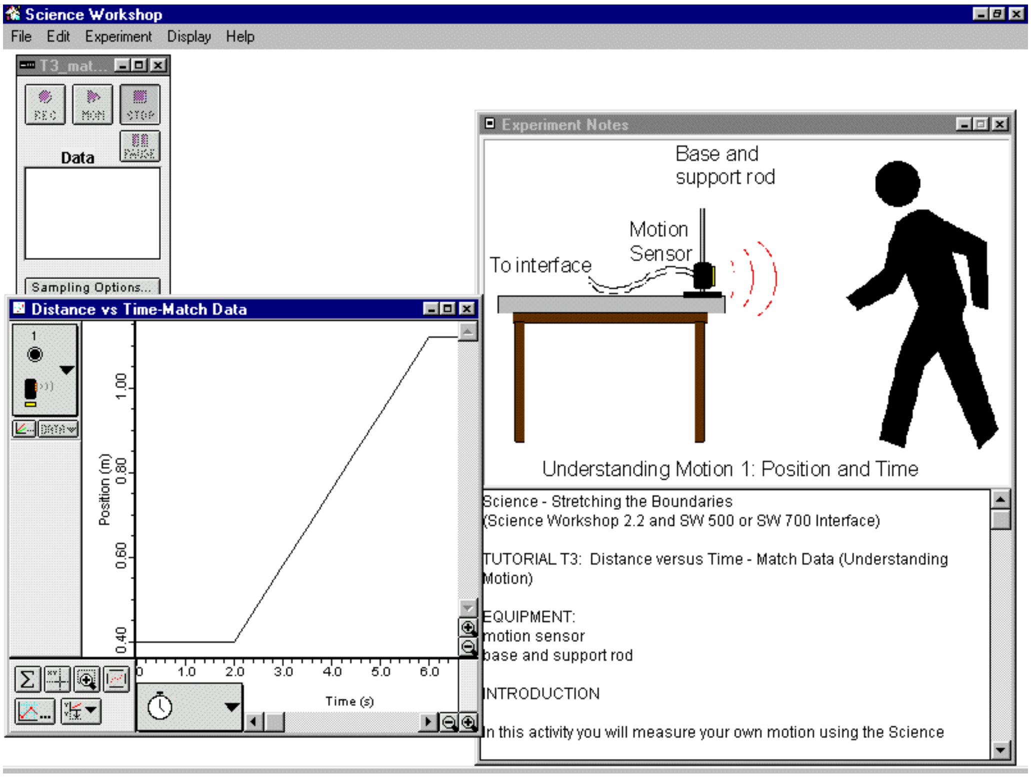
There are a few *Mathematica* books for physicists and physics students (e.g.: Tam, 1996; De Jong, 1999; Zimmerman & Olness, 1995), but none can be considered a complete course on physics such as the one developed by mathematicians to introduce calculus at college level (Davis, Porta, & Uhl, 1994). This course, *Calculus&Mathematica*, has a "hands-on" approach to calculus, stressing calculus "as the introduction to the science of measurements—both exact and approximate". Quoting the 1989 National Research Council report *Everybody Counts* (National Research Council, 1989) they start from the assumption that "Scientific computation has become so much a part of everyday experience of scientific and engineering practice that it can be considered a third fundamental methodology of science—parallel to the more established paradigms of experimental and theoretical science" (p. 36). *Calculus&Mathematica*, according to their authors, is "close to empirical mathematics. Every effort is made to replace rote learning with learning by experimentation through plotting and calculating", ie., doing "active mathematics instead of formalism". Formal definitions of the fundamental concepts of calculus are missing in the course but all are "present in their active forms". All the text is available both in electronic format and paper. The electronic files are interactive: students execute the computations, change parameters or the *Mathematica* commands, analyse different solutions, etc. The full text is also available on the Internet (http://www-cm.math.uiuc.edu [retrieved August 25, 2002]). In *Calculus&Mathematica* students do not study *Mathematica* as a programming language before they can begin their study of *Calculus&Mathematica*: they learn *Mathematica* on a "just in time basis", "gradually and always in context", since the lessons themselves contain all the *Mathematica* code. The same can also be said of the physics books with *Mathematica*, probably to a lesser extent.

2.2.5 Data logging (*computer-based laboratory systems*)

Data logging systems—or, more appropriately, *computer-based laboratory systems*—, are systems that can acquire and, or, analyse experimental data. These systems had been, for decades, a normal tool in experimental research and in the 90s finally came to school laboratories. A typical experimental set-up uses one or more sensors (either digital or analogue), an interface, and software to display data in tables or graphs and to do curve fitting or other mathematical treatment such as FFT.

Many courses have been created with a very significant use of data logging—and other computer tools—, such as *Workshop Physics* (Laws, 1997). This course emphasises the process of scientific investigation, replacing the lecture method by active involvement of students in laboratory experiments that "include prediction, qualitative observation, explanation, equation derivation, mathematical model building, quantitative experiment, and problem solving" (Laws, 1997).

The most distinctive feature of data logging systems is the ability to get a "real time" picture of the data. One of the most convincing examples of the potentialities of data logging is given by the real time display of position-time graphs. The experimental set-up includes the computer, the interface, an ultrasonic motion sensor, and the software. Once running the software, if there is an object in front of the sensor, the student can see a position-time graph of the object (or any other quantity that depends on the position and time, such as velocity and acceleration). The object in motion can be the student itself, as represented in the Figure 2.5.



Science - Stretching the Boundaries
(Science Workshop 2.2 and SW 500 or SW 700 Interface)

TUTORIAL T3: Distance versus Time - Match Data (Understanding Motion)

EQUIPMENT:
motion sensor
base and support rod

INTRODUCTION

In this activity you will measure your own motion using the Science

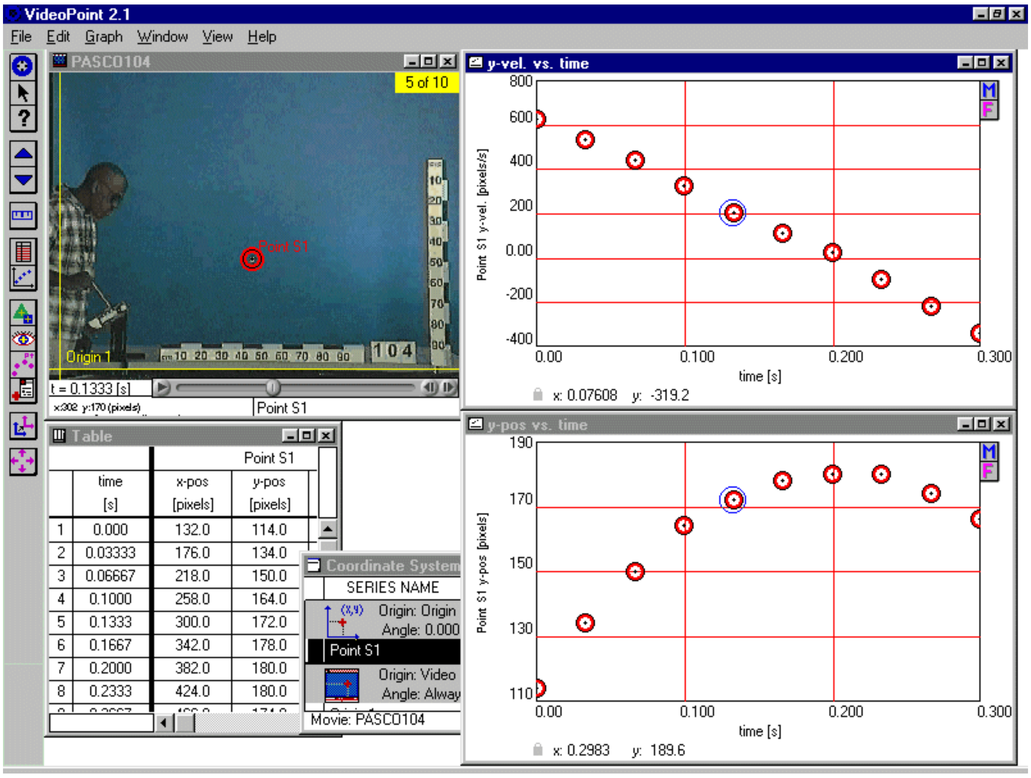
Figure 2.5 Using a motion sensor to understand position-time graphs. In this experiment, the student tries to match the graph on the left moving in front of the sensor (from *Pasco Science Workshop*).

Motion sensors are particularly useful to make thinking from representations more consciously, as can be seen in this personal experience reported by an experienced physics educator and researcher (Redish, 1994, p. 797):

> Ron Thornton visited the University of Maryland a few years ago to give a seminar on his now famous work on using the Sonic Ranger to teach the concept of velocity. The Ranger detects the position of an object using sonar and can display the position or the velocity of the detected object on a computer screen in live time. Thornton set up the Ranger to display velocity and had the computer show a pre-set pattern (a square wave). He then called me up to the front of the room to serve as a guinea pig and try to walk so my velocity matched the pre-set pattern.
>
> I had no hesitation in doing this. I had been teaching physics for nearly twenty years and felt perfectly comfortable with the concept of velocity. I did my first trial without thinking; I walked backward until my velocity reached the height of the pre-set square wave. Then I stopped and my velocity dropped to zero immediately! I asked for another chance, and this time, putting my brain in “velocity mode”, I was able to reproduce the curve without difficulty.
>
> What this experience said to me was that, for normal walking, I still maintained a naïve (but appropriate!) position-dominated proposition in my mental model of motion. I also had a correct proposition for the concept of velocity, but I had to consciously apply a rule telling me to use it.

It is not only Redish who maintains naïve views about motion and graphs. Research has shown in the last twenty years that most students and many teachers have learning difficulties with graphs (McDermott, Rosenquist, & Zee, 1987).

Another type of data logging system is video analysis software (Beichner et al., 1990). Video analysis software allows the user to collect data from video sequences: on a frame by frame mode, the user clicks on special locations on the image and a table of data is generated—see Figure 2.6.



		Point S1	
	time [s]	x-pos [pixels]	y-pos [pixels]
1	0.000	132.0	114.0
2	0.03333	176.0	134.0
3	0.06667	218.0	150.0
4	0.1000	258.0	164.0
5	0.1333	300.0	172.0
6	0.1667	342.0	178.0
7	0.2000	382.0	180.0
8	0.2333	424.0	180.0

Figure 2.6 Video analysis of a projectile motion using *VideoPoint* (Luetzelschwab, Laws, Gile, & Cooney, 1997). Clicking on the projectile, at the different images of the video sequence, generates a table of data. This table was used to create two graphs, $x(t)$ and $y(t)$.

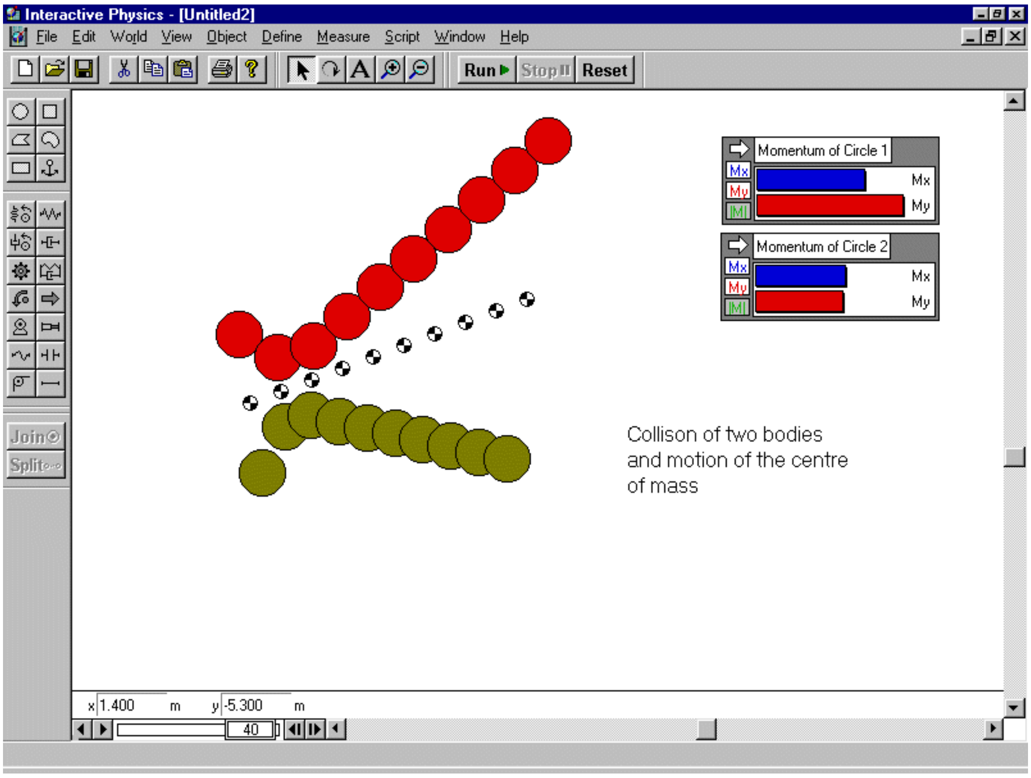
Video analysis software is more useful for motion analysis but can also be used for other types of experiments. Users can use their own movies or select from the movies that come with the programs. According to the creators of the first video analysis software, “this software not only makes it easier and faster to collect motion data, but also helps students make the cognitive link between the physical event and the mathematical graph representing it” (Beichner et al., 1990, p. 244).

2.2.6 Simulation tools

A simulation is “a computer program where elements of a specific system are represented in a model” (de Jong & Joolingen, 1998, p. 180). Users can interact with the model through one or more interfaces of the simulation. Typical uses of a simulation consist of "what-if" actions, changing parameters and, or, initial conditions and observing how these changes affect the system. As all computer tools mentioned in this section, simulations are conceived to be part of discovery learning environments. "Discovery learning is a highly self-directed, and constructive form of learning" (de Jong & Joolingen, 1998, p. 179).

It can be useful to distinguish between *simulation by itself* and *authoring environments for creating simulations* but in this thesis I will not distinguish, since in most cases I'm only referring to authoring environments, where the user can create many different examples of simulations.

There are many simulation programs for physics teaching, some commercially available but some freely available on the Internet. The most renowned program is *Interactive Physics*, described in its manual as "a complete motion lab on the computer that combines a simple user interface with a powerful engine that simulates the fundamentals of Newtonian Mechanics" (Knowledge Revolution, 1998, p. xvii). A simulation in *Interactive Physics* can be created using a direct manipulation interface, selecting tools, drawing and dragging objects, editing its properties or the properties of the environment, etc. The outputs are the motions of the objects, graphs, stroboscopic representations, vectors, etc. Figure 2.7 shows a simulation of a collision, with the stroboscopic representation of the motion of the centre of mass and bars representing linear momentum components for each body.



Collison of two bodies
and motion of the centre
of mass

Figure 2.7 A simulation of a collision of two disks made with *Interactive Physics*. Both the disks and the centre of mass of the system are shown in stroboscopic representation.

The instructional use of simulations has four features (de Jong, 1992):

1. Presence of a formalised and manipulable model.
2. Presence of learning goals, such as conceptual knowledge or procedural knowledge.
3. Elicitation of specific learning processes, characteristic of exploratory learning, such as hypotheses generation, predicting, and model exploration.
4. Presence of learner activity, ie, the learner must manipulate input variables and parameters, collect data, make choices in procedures, set data presentation, etc.

Levin & Waugh (1988) argue that simulations must be analysed from various dimensions.

The first dimension is *fidelity* which refers to "how faithfully does the simulation represent the part of the world being simulated" (p. 72). It is possible to distinguish between *perceptual fidelity*, ie, "the extent to which the computer program is seen (and sometimes heard) in a way similar to the situation being modelled" (p. 72), *manipulative fidelity*, "the extent to which the learner's action correspond to the actions to be taken in the domain being modelled" (p. 72), and *functional fidelity* which relates to the "correspondence between the internal structure of the model (the parts of the simulation that mediate between the learner's actions and the perceptual scene presented by the learner) and the internal structure of the domain being modelled" (p. 73).

The second dimension is *dynamic support*, the "sequence of systematically decreasing the amount of assistance provided to learners as they progress from novices to experts" (p. 73). This notion of dynamic support is derived from the learning principle "the zone of proximal development" (Vygotsky, 1978).

The third dimension is *breadth of scope*—the extent to which it is possible to range the mental activity, observation, or outlook of the system being modelled.

The fourth dimension is *learner control*, a dimension that can have very different values from one simulation environment to another.

Simulations can also be analysed on whether they have "multiple coordinated levels of presentation" (Levin & Waugh, 1988), or "multiple representations" (Perkins, Schwartz, West, & Wiske, 1995). Multiple representations are alternative coordinated views of a phenomenon, model or process in a certain cognitive domain, such as a graph and an equation, referring to the same phenomena.

If we look at the titles published by the *Physics Academic Software* (a project of the American Institute of Physics, with the most extensive catalogue of software for teaching physics) or the catalogues of commercial educational software publishers, we can conclude immediately that simulations are the most published type of software in physics teaching. Surveys had shown that simulations are also the most used type of software in physics teaching (de Jong, 1992; Carvalho, 1994).

Recently there have been some interest in virtual reality based simulations, such as those developed by the project *ScienceSpace* at NASA's Johnson Space Center and George Mason University (http://www.vetl.uh.edu/ScienceSpace, [retrieved August 25, 2002]). This project aims to design a series of virtual reality microworlds for teaching abstract science concepts and skills that students typically have difficulty mastering. Most important characteristics of virtual reality based simulations are immersion in 3D environments (with help from additional hardware such as head-mounted displays) and haptic devices that provide feedback through touch and pressure. Besides some simulations developed for research and testing, there is not known yet any virtual reality simulation for general use, with the above mentioned characteristics.

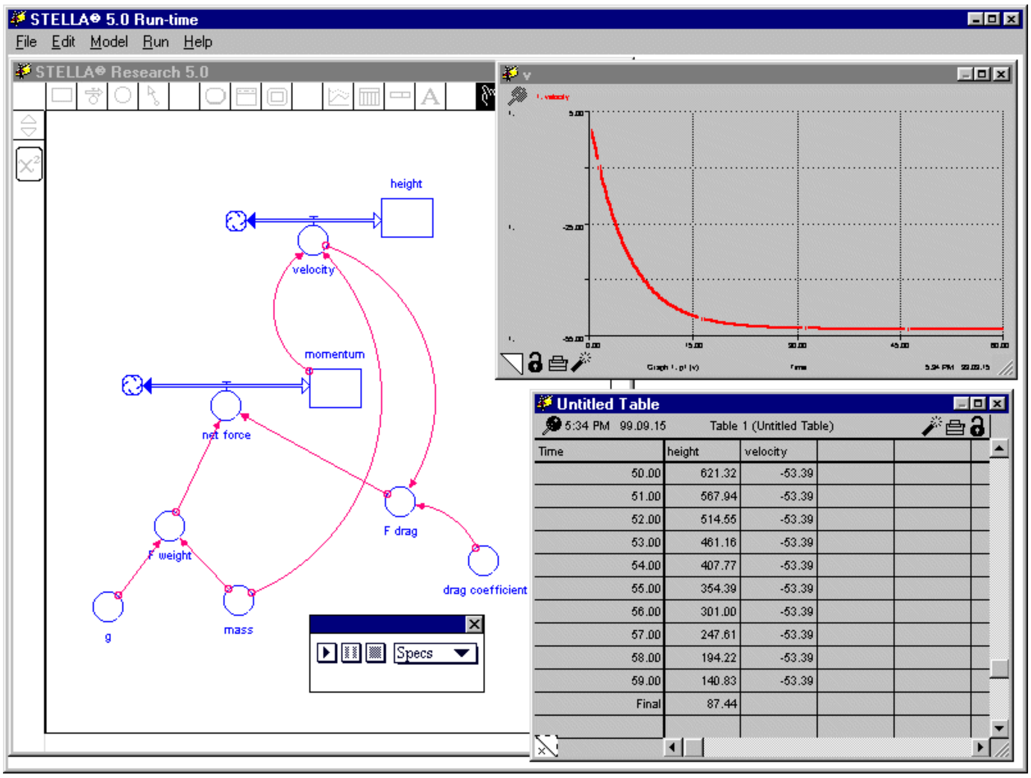
2.2.7 Modelling tools

It is not easy to define clearly what a computer modelling tool is. Spreadsheets and other mathematical tools can be considered modelling tools and they are used as such in certain curricula, as in *Workshop Physics*. In this thesis, when referring to modelling tools, I will refer only to software such as *Stella* (High Performance Systems, 1988; High Performance Systems, 1997), *PowerSim* (Baugsto, Byrknes, Krakenes, & Myrtveit, 1993), *Cellular Modelling System* (Ogborn & Holland, 1989), or Modellus, where students can create and explore dynamic mathematical models by entering variables and relations between variables in a graphical format and, or, using equations.

Modelling tools are powerful tools to make computations but the fundamental purpose of modelling is *insight* and *qualitative analysis* of phenomena, not computation:

> (…) the purpose of modelling is insight into and understanding of the process under consideration. Hence the more simple the model, the better, provided of course that the model behaves, at least qualitatively, as does the actual process itself (Dorn, 1977, p. 141).

A typical example of a model made with *Stella* is shown in Figure 2.8 that represents a model of a falling parachutist.

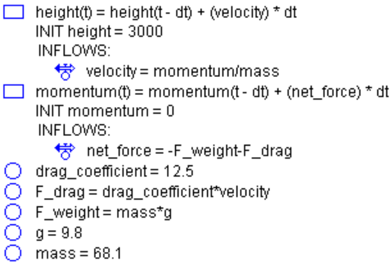


5:34 PM 99.09.15 Table 1 (Untitled Table)

Time	height	velocity
50.00	621.32	-53.39
51.00	567.94	-53.39
52.00	514.55	-53.39
53.00	461.16	-53.39
54.00	407.77	-53.39
55.00	354.39	-53.39
56.00	301.00	-53.39
57.00	247.61	-53.39
58.00	194.22	-53.39
59.00	140.83	-53.39
Final	87.44	

Figure 2.8 Model of a parachutist made with *Stella*. The left side of the screen shows the graphical representation of the model. On the right side, the graph of the velocity shows that the parachutist reaches a terminal velocity about 30 s after start falling.

Each rectangle represents an integrated variable—or, in *Stella* language, a *stock*. Stocks have *flows*—in the above example, “velocity” is the flow of “height” and “net force” is the flow of “momentum”. Flows are the instantaneous rates of change of stocks. Flows can depend of *converters*, the isolated circles. Converters are the parameters of the model. Arrows represent the dependence of flows from converters, other flows or stocks. Once having entered these dependencies, it is necessary to give initial values to stocks, values to independent converters (e.g., “g” and “mass” in the above model) and mathematical expressions to dependent converters (e.g., “F_weight = mass * g”). The full set of mathematical equations can be seen using an appropriate button and looks like the one in Figure 2.9.

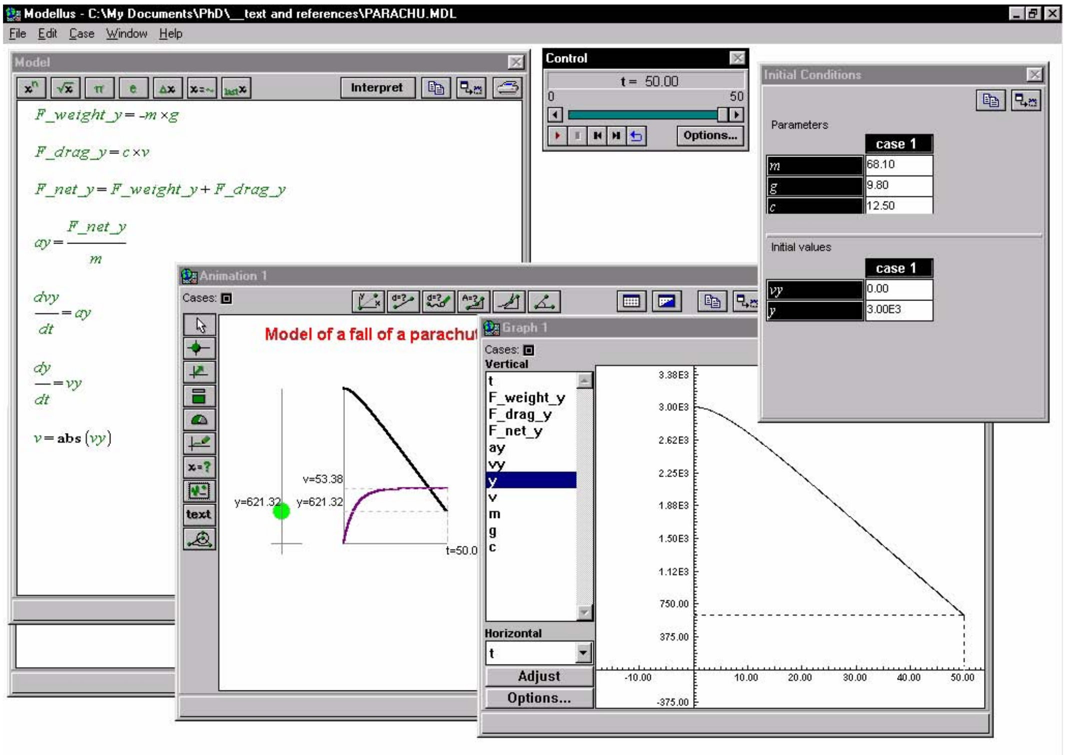


```
□ height(t) = height(t - dt) + (velocity) * dt
  INIT height = 3000
  INFLOWS:
    velocity = momentum/mass
□ momentum(t) = momentum(t - dt) + (net_force) * dt
  INIT momentum = 0
  INFLOWS:
    net_force = -F_weight-F_drag
○ drag_coefficient = 12.5
○ F_drag = drag_coefficient*velocity
○ F_weight = mass*g
○ g = 9.8
○ mass = 68.1
```

Figure 2.9 Mathematical equations used in the *Stella* model of the parachutist.

Stocks correspond to difference equations in the form $f(t) = f(t - dt) + flow \times dt$. Converters are shown as constant parameters or as equations. The user can choose, selecting the appropriate option in one of the menus, the integration method and the time range. He can also create graphs and tables for any of the variables. A strange feature of the more recent versions of *Stella* is that a flow can be "uniflow" or "biflow". A "uniflow" is a flow that it can have only positive values (things entering the stock) and a "biflow" is one that can have both positive and negative values (things entering or leaving the stock). By default, flows are "uniflows"—this means that in almost all physics models one must be very careful and not forget to change from "uniflow" to "biflow".

Modellus uses a completely different approach to modelling. A model of a parachutist can look like the following in Modellus:



Modellus - C:\My Documents\PhD__text and references\PARACHU.MDL

$$F\_weight\_y = -m \times g$$

$$F\_drag\_y = c \times v$$

$$F\_net\_y = F\_weight\_y + F\_drag\_y$$

$$ay = \frac{F\_net\_y}{m}$$

$$\frac{dvy}{dt} = ay$$

$$\frac{dy}{dt} = vy$$

$$v = \mathrm{abs}(vy)$$

t = 50.00

Parameters

	case 1
m	68.10
g	9.80
c	12.50

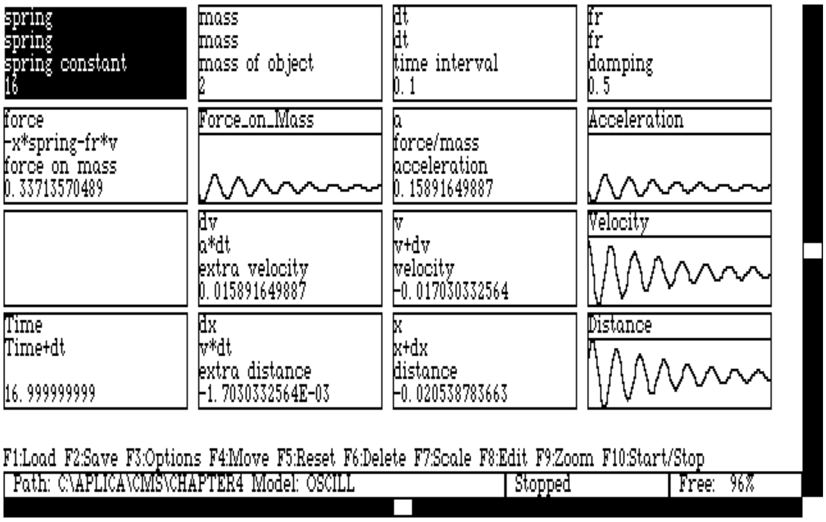
Initial values

	case 1
vy	0.00
y	3.00E3

Figure 2.10 Model of a parachutist in Modellus, using differential equations.

While *Stella* and *Powersim* use visual representations of equations and dependence between variables, Modellus uses equations written almost in the same way as in normal mathematical writing. Integrated variables are written as *instantaneous rates of change equal to some expression* and dependence between variables is written as *some variable equals some expression*. Integrated variables can be used as inputs to other variables, which allow the user to write nth order differential equations. Dependent variables can depend on the independent variable (which is used for integration) or on any other variable or parameter. The user must give the initial values of the integrated variables and of the parameters in the Initial Conditions Window. The output can be in tabular form, graphic or as an animation. Since Modellus is described in Chapter 4 in more detail, I will not give more information about it here. I shall only notice that while other modelling tools create a *new visual language* to represent models, Modellus uses a straightforward approach to modelling, using equations, either as functions or as differential equations (*instantaneous rates of change*).

Another approach to modelling was adopted by the software *Cellular Modelling System* (Ogborn & Holland, 1989). This package uses a "cell" metaphor for creating models. A cell can be a variable or a constant, an explicit equation, an iteration or a graph. A typical model made with this system can look like the one in Figure 2.11.



spring spring spring constant 16	mass mass mass of object 2	dt dt time interval 0. 1	fr fr damping 0. 5
force -x*spring-fr*v force on mass 0. 33713570489	Force_on_Mass	a force/mass acceleration 0. 15891649887	Acceleration
	dv a*dt extra velocity 0. 015891649887	v v+dv velocity -0. 017030332564	Velocity
Time Time+dt 16. 999999999	dx v*dt extra distance -1. 7030332564E-03	x x+dx distance -0. 020538783663	Distance

F1:Load F2:Save F3:Options F4:Move F5:Reset F6:Delete F7:Scale F8:Edit F9:Zoom F10:Start/Stop

Path: C:\APLICA\CMS\CHAPTER4 Model: OSCILL | Stopped | Free: 96%

Figure 2.11 A model of a mechanical oscillator made with Cellular Modelling System.

As with Modellus, creating a model with *Cellular Modelling System* means manipulating equations. But while in Modellus differential equations can be written in the form $dx/dt = rate$ in the *Cellular Modelling System* all integrations must be written as difference equations, $x_{new} = x_{old} + rate \times dt$ (Modellus can also use this type of equation—see chapter 4).

2.2.8 Modelling vs. simulation: classifying the use, not the tool

Most simulation and modelling tools present themselves as modelling *and* simulation tools, not as modelling *or* simulation tools. Indeed, it is not possible to distinguish completely these two types of computer tools. What we can distinguish, in most cases, is the *use* of the tool, not the tool itself.

Some authors consider the criteria of the explicit presence of the mathematical model to distinguish modelling from simulation (e.g., de Jong, 1998). Dorn (1977) argues that these two types of computer use are "quite different", linking modelling to the control of the mathematical details of the program:

> We contrast and catalogue these two quite different types of computer use as modelling and simulation. Modelling is being used if the student knows enough about the program to be able to write the program if needed. (…) Simulation, on the other hand, is the case in which the details of the computer program are deliberately and intentionally withheld from the student (p. 161).

A computer simulation can be defined as a software program that shows one or more representations of phenomena, hiding the "model" from the user, i.e. the details of the mathematics or logical relations between the elements or variables that represent some features of the phenomena. For example, when a learner uses *Interactive Physics* to explore projectile motion, he or she does not need to have present the equations of motion. These equations are automatically assumed by the software when an object is created in the "drawing area". On the contrary, in a modelling environment, the learner must implement the equations of motion to represent the projectile. The output can then be seen in a table, a graph or as an animation (a *simulation*), such as in Modellus. Animations can also be done in other modelling tools (for example, one can represent a graph of y vs. x in *Stella*, obtaining a simulation of the trajectory), in a spreadsheet, and even in a more standard mathematical tools, such as *Mathcad* and *Mathematica* that now have animation capabilities that allow the user to create animated objects from equations. But with simulation software, such as *Interactive Physics*, it is also possible to have some control of the equations governing the simulation (this can be done in the "properties menu" for each object, in the case of *Interactive Physics*).

It seems then clear in more recent software *modelling and simulation is not necessarily a characteristic of the software itself: it depends on the way one uses the software*. In this regard, Modellus is *not* a *modelling* environment or a *simulation* environment: it *can be used* as a modelling tool or as a simulation tool. The criterion of the "presence of the mathematical model" is a criterion that one can use for a specific example, for a specific context, not for the software as a whole.

2.2.9 The rise of educational applications

In the 70s and early 80s "the most visible and spirited debates in the educational computing community focused on a few central questions—e.g., whether drill-and-practice programs were pedagogically valuable; or how best to incorporate cognitive models and artificial intelligence techniques into educational software; or whether (for instance) Basic or Logo was the best programming language for children to learn" (Eisenberg, 1995, p. 177). As Eisenberg points out, "While none of these issues has disappeared, the traditions of software design that they reflect—computer-assisted instruction, intelligent tutoring systems, and child-friendly programming environments—have collectively been challenged, if not eclipsed, by the advent of what might be called 'educational applications'" (Eisenberg, 1995, p. 177). These "educational applications" (programs such as *SimCity, SimLife, Interactive Physics, Kid Pix*, the *Geometer's Sketchpad*, and the *Explorer* series) have become dominant in those schools where computers are seen as learning tools, not devices for multimedia presentations. Frequently, these "educational applications" are recommended by curriculum developers and authorities. For example, dynamic geometry software (such as *Geometer's Sketchpad*) is mentioned fifteen times in the NCTM *Principles and Standards for School Mathematics* (NCTM, 2000) while "programming" and "Logo" is only mentioned once, in a very specific context for young children. Effective educational applications are as "transparent" as possible and allow the user "to focus on the subject matter rather than on the computer" (Ronen, 1995, p. 150).

Educational applications must be relatively simple to use, have a clear benefit for the curriculum (e.g., because they extend the range of curriculum experiences students can have) and have the possibility of an extended use in multiple curricular levels. Typically, a physics curriculum can require three or four educational applications (data logging, modelling, simulation, image analysis). For example the new IOP Advancing Physics curriculum makes intensive use of Modellus for modelling, *Scion Image* for image analysis and data logging tools (not included as part of the curriculum materials).

Eisenberg (1995, p. 180) argues that "designers might start by thinking about what sorts of (e.g.) physics applications would be useful for the professional community; the educational version of such an application would then be a system based on the very same framework." This is probably true for upper level teaching, particularly in the University, but it is much more problematic for lower level teaching such as teaching in grades 7 to 12. For example, it will be very difficult to use *Matlab* or *Mathematica* with nine graders since the necessary basic knowledge to use these systems is considerably more than the one students have.

2.3 Research on Learning Physics with Computers

2.3.1 Which methodology?

The dissemination of the use of computers in physics education has been accompanied by an effort of educational research. Initially, research focused on the effectiveness of learning, comparing experimental groups using computers with control groups using traditional ways of teaching. These types of studies have been criticized by many authors (e.g., Salomon, 1990) with the argument that using computers is not merely the use of a tool, *it transforms the character of the educational learning environment*. Salomon (1992, p. 262) argues that "in an environment in which almost everything changes, it would be impossible and conceptually unsatisfactory to try to attribute such cognitive changes to one single agent—the computer". More recent studies focused on the *quality of learning* instead of experimental studies comparing groups *with* and *without* computers (e.g. Roth, 1995; Roth, 1996).

2.3.2 Research on physics learning using data logging

Rogers & Wild (1996) studied the effects of data logging systems in practical investigations. After recognizing how difficult is to do research on the quality of learning on a short time scale, they concluded that their results "demonstrated quantitative changes in the time spent by pupils on different categories of activity when Information Technology was used: the traditional emphasis on the mechanical aspects of measuring, recording and reporting in conventional practical work was diminished, with a commensurate enhancement of time spent on observation and discussion" (p. 140). Recognizing that students must have a minimum "threshold of IT capability" to make regular use of software and hardware, they found that many students rapidly gained confidence and familiarity, having no major technical problems.

Some studies have reported that data logging systems facilitate graph interpretation. As earlier as 1987 (Mokros and Tinker) reported that students are encouraged to understand graphical representations because changes over time are dynamically displayed in real time. Recent studies, such as Thornton & Sokoloff (1998), comparing traditional introductory courses with courses supported by MBL activities, have shown that performance of students using MBL tools are significantly better than students following traditional courses.

Brasell (1987) and Thornton & Sokoloff (1990) found that students using real-time graphs with MBL significantly improved their kinematics graphing skills and their understanding of the qualitative aspects of motion as compared to students using graphs produced after the motion of an object.

Beichner (1990) analysed the effect of MBL on student learning in both high school and college physics classrooms. He compared the understanding of kinematics students who were taught using teacher oriented demonstrations and computer simulations of videotaped images with other students who were taught using MBL-based lessons. His results showed that students who were taught using MBL techniques got better results. Beichner concluded that direct personal control of the computer and/or the experience of producing the graphs were responsible by the better results of MBL-based learning.

Ronen (1995) investigated the use of a 3-D motion tracing system in High School Physics teaching. The system, V-scope, is a computer-based three-dimensional motion tracking system designed for physics teaching, combining infrared and ultrasonic pulses. Teachers and students considered that it facilitates the understanding of concepts and stimulates interest in physics, helping the understanding of graphical representations of motion, vectors, relation between location, velocity and acceleration, harmonic motion and circular motion. Difficulties mentioned by teachers and students are mainly of a practical nature—limited availability of equipment and shortage of time.

Other researchers used student analysis of videodisc motion images and MBL real time graphical analysis (Brungardt & Zollman, 1995). They found no significant learning difference between using real-time and delay-time (recorded video) analysis for understanding of kinematics graphs. They also found that real-time analysis resulted in increased student motivation, more discussion, and less confusion between velocity-time graphs and acceleration-time graphs than delay-time analysis.

A curriculum that uses MBL extensively is *Workshop Physics*, an activity-based introductory college physics course with no formal lectures that integrates computer technology into laboratory experiences. The curriculum is intensively based on inquiry activities (Laws, 1991), and integrates various computer applications in active environments in which students observe, take carefully records and measurements, analyse data, and develop verbal and mathematical models. Computer applications used in the course include: MBL tools to collect data and to display the data in real-time graphs; spreadsheets (with special purpose macros) to analyse data and solve numerical problems; computer simulation programs to model phenomena that cannot be directly observable; and video analysis tools to analyse two-dimensional motion of objects (Laws, 1995). Laws analysed the effect of *Workshop Physics* on student learning by comparing students' performance before and after the course. She found dramatic improvements in student conceptual learning in kinematics, dynamics, and other topics. However, these improvements in student learning resulted only after discussion, observation, and prediction were integrated into the program.

2.3.3 Research on physics learning using computer simulations

As with other modalities of use of computers in physics teaching, it is not an easy task to infer clear conclusions from research on the use of computer simulations in physics teaching. Indeed, most "proponents of simulations and games felt that the priority was their invention rather than their evaluation" (Robinson, p. 309, quoted by de Jong & Joolingen, 1998). Another difficulty comes from "the involvement and the excitement which is common amongst participants in a simulation can blind the critical eye of the observer" (Tonks & Armitage, 1997, p. 51).

Hennesy et al. (1995b) investigated how effective were simulations and practical activities on promoting meaningful conceptual change on the topics of force and motion with 29 students aged 12-13 years. The experimental group used interactive simulations and relevant practical activities, while the comparison groups received more traditional instruction, without any use of computers. Previous to the intervention, both groups used activities designed in such a way to make students aware of their conceptions. Posttest results reveal that the experimental group displayed more sophisticated reasoning and less alternative conceptions of force and motion.

After previous studies on learning difficulties in understanding Newtonian mechanics White (1983) designed a simulation-based learning environment, ThinkerTools (White, 1993). Comparing results of junior high school students that had followed a ThinkerTools based curriculum to more advanced students, who followed a traditional curriculum, on a test on qualitative understanding in real-world situations, White found a large effect of the ThinkerTools based curriculum on results on the test.

Brna (1987) reported on research using DYNLAB, a dynamics and kinematics simulation software. To use DYNLAB students write short instructions (such as KICK, GRAVITY ON, TABLE FORCE ONE 4N, etc.) that guide a screen object. Brna designed a set of physical situations which would present conceptual difficulties for the students, assuming that meaningful learning is promoted through confronting students with inconsistencies in their beliefs and knowledge. The author reported that when using the software, students were confronted with their misconceptions, and some of these were resolved.

Finegold and Gorsky (1992) investigated the use of computer simulations to restructure students' conceptions of force. There results showed that simulations were effective in eliciting students' beliefs about forces acting on objects at rest and in motion and that students who directly experienced the outcomes of their own misconceptions apparently rejected their incorrect views and accepted the scientific ones, at least in the context of the simulation.

Ronen, Langley & Dorothy (1992) reported on a curriculum project to integrate physics simulations into 18 Israeli schools. They used data from teacher-generated reports about each lesson, from student final assessments, and from class observations and personal interviews. They found large logistic difficulties and no agreement on teachers opinions about the best use of simulations: "initial subject presentation," "exploration during teaching," "drill and practice," "assisting laboratory work," and "summary and review" were the most common. The authors concluded that teachers can only take advantage of simulation tools if they have personal experience of the new tools and this is not an easy task for many teachers, overwhelmed with work.

Carlsen and Andre (1992) studied the effectiveness of simulations along with conceptual change texts in overcoming typical preconceptions about electric circuits. The study involved college students divided in three groups: one using computer simulation of electric circuit design and testing, another using only the electricity textbook and a third one using both text and simulations. The main conclusion of the study was that the use of simulations did not improved learning over the use of the textbook. But, when used in conjunction, the students did acquire a more developmentally advanced model of a series circuit.

In a study about *Interactive Physics,* Roth (1996) argues that "educators and software developers often seem to assume that students appropriate ideas represented in these microworlds and develop from these a scientific understanding of the physical world. But it is not clear if and how such ideas are learned." They analysed high school student conversations about computer displays to understand how a "computer microworld became a tool for social actors to coordinate their activities through talk" (p. 173). They clearly found that "students did not instantly change from inappropriate to appropriate science talk. Rather, the Newtonian science talk emerged slowly and tentatively (..)" (p. 183). Computer displays worked as "cultural tools", as "objects and sites for conversation", as "tools for collaborative sense-making".

Tao & Gunstone (1999) investigated the potential of a suite of 10 computer simulations to foster conceptual change in mechanics within high school students. Students worked in pairs and their conversations were recorded and analysed to identify processes of construction of shared understanding or conflict. The authors concluded that students engaged in the tasks experienced "co-construction of shared understanding" but personal construction and personal sense making of the new understanding was important to achieve "long-term and stable conceptual change".

2.3.4 Research on physics learning using modelling

Empirical research on modelling in physics education started in late 1980s but even now it is hard to find literature on the subject. Before empirical studies, there is some literature reporting software development (e.g. Ogborn, 1984). Early software was very similar to programming languages (Hartley & Lewis, 1982), but with time modelling software had become a more specific concept and different from programming languages.

Modelling researchers (e.g., Mellar et al., 1994; Schecker, 1993) claim that modelling software makes model creation accessible to students, when compared with previously available software designed for experts and scientists.

Schecker (1993) investigated the impact of curricular innovation involving systems thinking and the *Stella* modelling software on student learning and transfer (senior high school physics). From evaluating long-term case studies in several schools, he found that modelling in physics *works*—in the sense that students actively engage in discussions in small groups that promoted meaningful learning. In the beginning, students had some difficulties in using the modelling tool but they surpassed these difficulties after becoming more familiar with the tool. The difficulties were related to the *Stella*'s flow and reservoir metaphor used to think of the relationship between acceleration and velocity. Model building forced students to make vague, imprecise ideas into more explicit ideas and relations. Schecker argues that his studies give evidence that the modelling system served the students as a tool to help thinking and problem solving activities and that model building processes allow students to engage in a more qualitative, principle-based analysis of problems, prior to working on equation-based formal representations. When approaching phenomena and problems with computational models, physics concepts are introduced more qualitatively, more independently from analytic solutions. Thus, modelling provides students the opportunity to express their own views. Schecker is positive, but prudent, about his results for positive effects of model building software on the development of physical understanding.

Mandinach and Cline (1994) have engaged in a series of studies of implementing a technology-based learning environment, for high school students, centred on modelling with systems thinking, also using *Stella*. In early studies, they concluded (Mandinach, 1988) that the systems thinking approach affected learning and teaching activities, in different ways for different disciplines. The initial results reported were inconclusive about learning and transfer, but later studies (Mandinach, 1989) report that students were able to apply system-thinking concepts to complex scientific problems. They also conclude that there was necessary a strong effort on curriculum development.

In a review of research about modelling with *Stella*, Doerr (1996) concludes:

> There is some evidence that the use of STELLA for system dynamics modeling may lead to improvements in students' abilities to qualitatively reason about problem situations, particularly in the domain of introductory physics. Despite the difficulties with the plumbing metaphor of flows and accumulators, STELLA provides a means for discerning the structural similarities among problems whose symbolic algebraic representations appear to have little in common (p. 219).

These studies are positive but cautions should be taken about the impact of using modelling on the physics curriculum. They also show how important is curriculum development to facilitate the dissemination of modelling views. Or, as Doerr concludes, "The development of re-formulated curricula in light of such [modelling] software tools is a critical next step" (p. 221).

2.3.5 Other studies of physics learning with computers

The educational literature contains other studies about the use of computers in physics education. Of particular interest are studies of multimedia courses or multimedia based courses.

Watkins, Augousti & Calverley (1997) report a study on the evaluation of SToMP (Software Teaching of Modular Physics), a project under the Teaching and Learning Technology Programme (TLTP) in the UK. SToMP was used for primary delivery of first-year physics courses in six universities as a replacement for traditional lectures—the instructor only support students in attendance at timetabled computer-laboratories. The goal was to "promote a combination of self-paced learning and one-to-few tuition which should lead to an improvement in the effectiveness of undergraduate teaching" (p. 165). The main conclusion of the evaluation was that the computer-based method performs only marginally better than traditional teaching methods, being particularly useful for weaker students. The evaluation had made also clear how important and problematic is timetabling and managing resources in computer based courses for large groups of students.

In another paper, members of the same research group (Calverley, Fincham, Bacon, 1998) discuss the introduction of such courses in university, supporting their proposals from current thinking in educational theory. A common problem also found by the authors was the attitude of the students: "Initially, as the course developed, there was some resentment to feeling used as guinea pigs" (p. 164). They also found problematic study skills with electronic media. For example, "the students showed a strong tendency towards wanting to draw the course style back to being strongly lecture-based, despite being given all the necessary information in other forms and having ample opportunities for a shared learning experience" (p. 165). Between resource-based learning and explicit teaching is a large gap, and many students have difficulty in overcoming it. The authors then concluded that "it is not unreasonable to expect a certain amount of insecurity and uncertainty when introduced to an unanticipated and very different style of learning" (p. 165).

2.4 Computers in Mathematics Education

2.4.1 Mathematics education under change

Since Galileo, the language of mathematics is considered the language of Nature. Mathematics is extremely valuable for expressing scientific ideas unambiguously, providing a grammar for science—ie, rules for creating and analysing ideas and data with rigour (American Association for the Advancement of Science, 1989).

As Wright & Wright (1998, p. 128) say, "Mathematics enters the curriculum to support scientific and technological experiences for two reasons: (1) it is the only way for students to understand the relationships that define many physical phenomena, and (2) it is important for students to develop the ability to master abstract ideas".

The nature of mathematics is under a process of change: mathematical proof, mathematical fact, experimental mathematics, geometric intuition, etc., are old and new concepts that are under discussion mainly because of the introduction of computers in the production of knowledge (Mandelbrot, 1992). The ability to make powerful computations and graphics with computers is changing mathematics. Doing mathematics with computers is now part of the day to day routine for many mathematicians, not only applied mathematicians (Mandelbrot, 1992). But *proof is not being replaced by mere pictures:*

> All that is happening now is that new methods of searching for new facts provide mathematics with a powerful 'front end' of unexpected character, one that involves more than just the proverbial pencil and paper. Thus, pictures have already demonstrated their astonishing power to *help* in early stages of both mathematical proof and physical theory; as this help expands, it may well lead to a new equilibrium and to changes in the prevailing styles of completed mathematical proof and of completed physical theory (Mandelbrot, 1992, p. 2).

Mandelbrot argues that "the use of computer graphics is changing the role of the eye", "bringing it back as an integral part of the very process of thinking, search and discovery" (p. 2) in science and mathematics.

2.4.2 The calculus reform movement

Calculus has always been a traditional subject for physics students. Even at junior high school, physics students are introduced to calculus in indirect ways (e.g., computing distances from speed-time graphs using an integral approach without any mention of calculus). At higher levels of senior high school, many calculus tools are explicitly used, such as derivatives and limits.

During the 1980s, at the same time as personal computers got more and more common in society and schools, the mathematics education community discussed the role and the ways of teaching calculus. The calculus reform movement started. The Tulane 1986 “Conference/Workshop To Develop Curriculum and Teaching Methods for Calculus at the College Level” is generally identified with the birthplace of calculus reform. The Conference discussed discrete vs. continuous mathematics and ways of improving the teaching of calculus (Douglas, 1986). A typical claim at the Tulane Conference was that computer algebra systems could be used to improve conceptual understanding, overcoming limitations imposed by poor algebraic skills. Teaching should focus on the big ideas of calculus, rather than manipulative skills, and students should use calculus to solve problems that go beyond “typical” problems, even these problems involve advanced calculus ideas.

The situation and the new need were clearly stated by Peter Lax, a former president of the American Mathematical Society:

> Calculus as currently taught is, alas, full of inert material… The real crisis is that at present [calculus] is badly taught; the syllabus has remained stationary, and modern points of view, especially those having to do with the roles of applications and computing are poorly represented… There is too much preoccupation with what might be called the magic in calculus. For instance, too much time is spent in pulling exact integrals out of a hat, and, what is worse, in drilling students how to perform this parlor trick. Summing infinite series is another topic that has the aura of a magic trick, and is overemphasized at the expense of the concept of approximation… I feel that rigor at this level is misplaced; it appears as an arid game to those who understand it, and mumbo jumbo to those who don’t…. Many students have difficulty in grasping the idea that the integral of a function over an interval is a number. The reason is that this number is difficult to produce by traditional methods, i.e. by antidifferentiation, and so the central idea is lost. Numerical methods have the great virtue that they apply universally. When special methods are introduced to deal with each one of the pitifully small class of [differential] equations that can be handled analytically, students are apt to lose sight of the general idea that every differential equation has a solution and that this solution is uniquely determined by initial data… That today we can use computers to explore the solutions of [differential] equations is truly revolutionary; we are only beginning to glimpse the consequences (quoted by Davis, Porta & Uhl, 1991, pp. 74-75).

The reformers initiated a movement of curricular reform both for college courses and high school. This movement has not yet established a new consensus but it induced changes in courses all over the world, including Portugal and the curriculum reformulation made in 1997. New topics, fundamental to physics teaching, have been introduced (e.g. *rate of change*). More intuitive approaches have been suggested to limits, derivatives, etc. And modelling and data analysis took an important place.

The New Calculus Conference promoted by the Mathematical Association of America (Gordon et al.1994, p. 56) summarized the trends. New calculus courses should:

1. cover fewer topics and give more emphasis on fundamental concepts;
2. place less emphasis on complex manipulative skills and emphasize modelling the real world;
3. promote experimentation and conjecturing;
4. teach students to think and reason mathematically, develop problem-solving skills;
5. make use of calculators and computers.

2.4.3 NCTM principles in mathematics education and its importance to science education

Computers are now considered so relevant in mathematics education that one of the six principles for school mathematics stated by the NCTM Standards (NCTM, 2000) establishes that “technology is essential in teaching and learning mathematics; it influences the mathematics that is taught and enhances students’ learning” (p. 24). Accordingly to the Standards (p. 24), computers:

1. “furnish visual images of mathematical ideas”;
2. “facilitate organizing and analyzing data”;
3. “compute efficiently and accurately”;
4. “support investigation by students in every area of mathematics, including geometry, statistics, algebra, measurement, and number”;
5. allow students to “focus on decision making, reflection, reasoning, and problem solving.”

The Standards makes also a note on caution: “Technology should not be used as a replacement for basic understandings and intuitions; rather, it can and should be used to foster those understandings and intuitions” (p. 25).

Computers are one of the many ways of doing mathematics. The same is happening more and more in mathematics education, particularly in senior high school. Dynamic geometry, graphing and quantitative modelling are the three most important modalities of use mentioned in the Standards and in most literature of mathematics education. Dynamic geometry software “can allow experimentation with families of geometric objects, with an explicit focus on geometric transformations” (NCTM, 2000, p. 27). Graphing software facilitates the exploration of characteristics of families of functions. Quantitative modelling software allows students to explore and create quantitative mathematical relations, usually in relation with real phenomena and contexts. These types of software are also available in handheld graphical calculators, that students must use compulsorily in classrooms and examinations in some countries (e.g., Portugal).

Making experiments and modelling data gathered in experiments is becoming a common practice in mathematics classrooms. Accordingly to the Standards, high school students should “create and interpret models of phenomena drawn from a wider range of contexts —including physical and social environments —by identifying essential elements of the context and by devising representations that capture mathematical relationships among those elements” (p. 71). Usual topics in physics classrooms tend now to be also taught in mathematics classrooms (e.g., measurement errors and uncertainties, data fitting, etc.). Laboratory equipment companies that sell equipment for school physics laboratories are now also developping and selling similar equipment to mathematics laboratories (the Portuguese curriculum establishes that all schools must have such a laboratory). As I will show in the the next chapter, many mathematics teachers are using modelling software, such as Modellus (and also spreedsheets and data logging software), in very similar ways to physics teachers.

Connecting mathematics and science has been a recurrent theme in mathematics education in the last decade (see, e.g., NCTM, 2000). Accordingly to Wright & Wright (1998), "one of the central beauties of the [mathematics] national standards is that they include mathematics, science, and technology" (p. 129). But there has been a "tragic failure" in the lack of integration between science and mathematics in the structure of the equivalent National Science Education Standards (National Research Council, 1996). Although the science standards do explicitly include coordination of the mathematics and science curricula as one of the standards, "the examples and descriptions in the content standard and professional development standards contain only superficial connections" (p. 128). This lack of integration of science, and specially physics, with mathematics is common to many other countries, including Portugal, nowadays and decades before (Carvalho, 1947).

One of the main goals of Modellus is changing this lack of connection between science and mathematics. This has been recognized by a review on the NCTM journal:

> Modellus is an excellent tool for integrating mathematical models with applications in other fields. It will be as useful to science teachers as to those in mathematics, and I recommend it especially for teachers collaborating with colleagues in science (Dickey, 1998, p. 529).

2.5 Coda

In this chapter I tried to show how the use of computers in physics education evolved from the use of programming languages to specific "educational applications" and general computer tools such as spreadsheets and other mathematical tools. Teaching physics (and mathematics) has been significantly changed by the use of computers tools in the last decade, in some cases simply as accessory tools but in other cases as completed integrated tools in the curriculum (case of *Workshop Physics* in the USA and Advancing Physics in the UK).

Simulations and data logging are still probably the most common modalities of use of computers in physics classrooms but modelling is becoming more and more common. In the last fifteen years, a few studies have been made of the influence of using computers in learning physics. Results of these studies show that careful use can improve learning, helping students focus on more qualitative aspects of reasoning and on graphical analysis. A few studies were done about modelling in physics education using *Stella*, an icon oriented software tool. The results of these studies showed some evidence that modelling may lead to improvements in students' abilities also to qualitatively think about problem situations. But there is a huge work to be done in curriculum development in order to integrate modelling tools. This is considered by Doerr (1996), a "critical next step". The goal is clear:

> (…) technological products, from pencils to computers, provide tools that promote the understanding of natural phenomena (National Research Council, 1996, p. 24).

Chapter 3 Modellus: A Tool for Doing Experiments with Mathematical Objects

What I can not create I can not understand.
(Text found on Richard Feynman's blackboard at the time of his death).

3.1 Introduction

This chapter presents the most relevant aspects of Modellus and how it can be used to explore scientific ideas. Modellus culminates a decade of development of computer software for science and mathematics and its design received many inputs from previous software created by me or other colleagues—working in Portugal or elsewhere. Its design was also influenced by other software, either professional software such as *Mathcad* or educational software such as *Interactive Physics*, *Cabri Geometre* and *Geometer's Sketchpad*.

"Most software isn't designed. Rather, it emerges from the development team like a zombie emerging from a bubbling vat of Research and Development juice" (Cooper, 1995, p. 11). This is probably true for software designed by programmers, that centre the user interface on technology. That was not the case of Modellus and of all the software developed before it and briefly described on section 3.3.

3.2 Issues in Software Design

Software design is a relatively new field in computing, not yet completely recognized (Winograd, 1996). Part of the work done for this thesis are exercises of software design. It is, then, important to discuss some basic issues of the field.

As Winograd says “There is no direct path between the designer’s intention and the outcome. As you work with a problem, you are continually in the process of developing a path into it, forming new appreciations and understanding as you make new moves” (Winograd, 1996, p. 171). Or, in other words, product and process are intimately related—it is not possible to completely specify the product before the development process.

Software design is an *iterative process* (Microsoft, 1995): each instance influences the next. The starting point of the design is a *concept*, “what the program will do, what it will look like, and how it will communicate with the user” (Cooper, 1995, p. 24).

From where comes the *concept*? In education, software concepts come from experience in learning environments, from literature on learning, from adaptation of previous ideas. An interesting personal reflection about the origin of a software concept is described by (Schwartz, 1993), the original creator of the now common dynamic geometry environments. Schwartz, a physicist by training, later involved in educational reform (with a large experience on the design of learning experiences and on research on learning science and mathematics), was interested in images, geometry, and the power of computer graphics. His *concept*—interactive geometry environments—was a consequence of these multiple interests.

It is useful to consider three types of models of a tool—digital tools, like software, or mechanical devices (Cooper, 1995):

1. the **Implementation Model**; refers to the actual method: how the tool works;
2. the **Mental** or **Conceptual Model**; refers to the simplified view of how users think about how the tool works;
3. the **Manifest Model**; refers to the way the tool represents its functioning to the user. A Manifest Model can be closer to the Implementation Model or to the Mental Model.

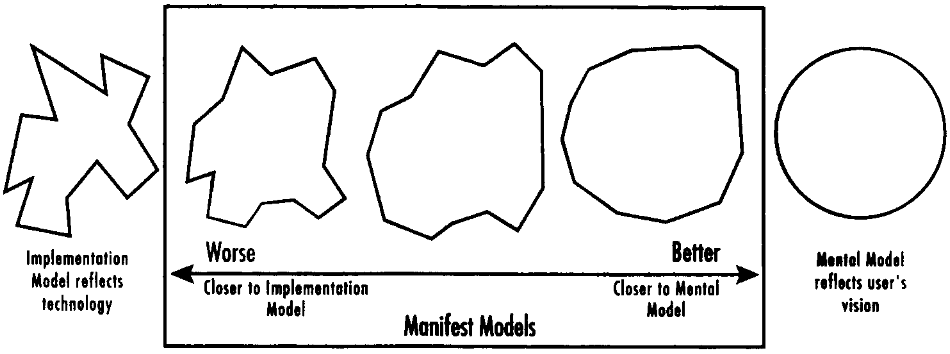


Figure 3.1 Manifest Models of a tool can be closer to the Implementation Model or to the Mental Model. Good software is closer to the Mental Model (Cooper, 1995, p. 29).

The “concept of the Manifest Model has no widespread counterpart in the mechanical world” (Cooper, 1995, p. 29) but is particular relevant in the digital world. According to

Cooper, bad user interfaces design conform to Implementation Models (they are close to the mode of operation) and good ones conform to Mental Models (they tend to reflect the way users think with the tool). But, still, according to this author, "good user interface design" is strongly dependent on the context of the use of the tool.

Digital tools can import ideas and representations from the pre-digital world (such as static geometric objects) or can create new ideas, only possible in the new digital world. The most powerful software concepts are related with ideas and actions only possible in the digital world. A good example of this is, again, dynamic geometry software: the user can draw geometrical objects as one does on paper, but he also can manipulate directly the shape of the object, and of his associated objects, using the mouse. This action doesn't have any equivalent action on the non-digital world, but is a fundamental action on the new digital world. It is the *concept* under dynamic geometry software. Similar ideas can be found in Modellus. For example, a Modellus user can create a particle, in a certain reference frame, and use a graph to move the particle—there is not any equivalent action to this in the material world, where graphs are only obtained from moving objects.

Ease of use is considered by most naïve users as a very important aspect of a software tool. This idea is not shared within the user interface design community (see, e.g., Cooper, 1995). Certainly, ease of use is an important consideration but good interface design combines power and ease of use. Ease of use is an attribute that develops with use, not necessarily a first impression. And good design always make the user more effective (Cooper, 1995).

The now common GUIs (graphical user interfaces) were not available when I started designing software in middle 1980s when user interfaces were command and menu based. One of the software titles I designed, Newton, was implemented on the Macintosh because its concept was only possible to implement on a direct manipulation interface. Other titles, such as *Funções* (Functions) were an interesting exercise of how to nearly have a graphical interface without having a computer environment to support it.

Briefly, a graphical user interface "is a collection of techniques and mechanisms to interact with something. In a graphical interface, the primary interaction mechanism is a pointing device of some kind. (...) What the user interacts with is a collection of elements referred to as objects. (...) People perform operations, called actions, on objects. The operations include accessing and modifying by pointing, selecting, and manipulating" (Galitz, 1997, p. 13).

Graphics has indeed revolutionized software design (Galitz, 1997). In early 1980s, it was expected that user interface will tend to intelligent conversations between user and software, using written commands or voice. The emergence of artificial intelligence (the 5th generation computer...) created expectations that were not fulfilled. Instead, graphical and direct manipulation user interfaces took the scene—and the personal computer revolution started.

There are three dominant paradigms in user interface design (Cooper, 1995):

1 the **technology paradigm**, based on *understanding* how things work;

2. the **metaphor paradigm**, based on *intuiting* how things work;
3. the **idiomatic paradigm**, based on *learning* how to accomplish things.

Initial GUIs, invented in late 1970s at XEROX Palo Alto Research Center and successful commercially implemented only in middle 1980s on the Apple Macintosh, were explicitly based on the metaphor paradigm. “The metaphor paradigm relies on intuitive connections in which there is no need to understand the mechanics of the software, so it is a step forward from the technology paradigm, but its power and usefulness has been inflated to unrealistic proportions” (Cooper, 1995, p. 57). But, according to Cooper, the majority of the characteristics of a GUI interface are idioms, not metaphors (e.g., windows, caption bars close boxes, screen-splitters and drop-downs menus, pointers, mouse): we learn them “idiomatically rather than intuiting them metaphorically”. And learning—not necessarily understanding—is a human natural process:

> We are inclined to think that learning is hard because of our conditioning from the technology paradigm. Those old interfaces were very hard to learn because you also had to understand how they worked. Most of what we know we learn *without* understanding (Cooper, 1995, p. 58).

Learning is an intuitive, albeit intentional, process. Learning to perform actions with a good graphical user interface is very easy. With one or two examples, a child can master it. A basic principle of software design assumes that “all idioms must be learned. Good idioms only need to be learned once” (Cooper, 1995, p. 59).

Direct manipulation (Shneiderman 1982) is strongly associated with graphical user interfaces. Direct manipulation systems possess the following characteristics (Galitz, 1997):

1. the system appears as an extension of the real world, assuming that the user is already familiar with real objects and actions;
2. the system replicates objects and actions on the screen;
3. the user works on the screen as in a familiar environment and in a familiar way, focusing on the documents and data, not on the application;
4. objects have continuous visibility and actions have immediate feedback;
5. cursor action and motion is physically obvious.

Direct manipulation actions are, in many software tools, *idioms*—in the sense that their meaning must be learnt, they are not intuitive. But most environments combine direct with indirect manipulation, mainly because certain actions may be difficult or even impossible to conceptualize in a graphical system. The advantages of graphical systems, combining direct and indirect manipulation, are so overwhelming that it is now a standard on the industry, particularly on the personal computer.

Graphical systems provide faster learning and recognition, easier remembering, natural and spatial visual clues, concrete thinking, fewer errors, feeling of control, visible feedback, less typing requirements, etc. The disadvantages, like unfamiliarity for non expert users, window manipulation requirements and slowness of use, are surpassed with learning and familiarity (Galitz, 1997).

Combining direct and indirect manipulation is also fundamental in educational software design— they have complementary strengths (Eisenberg, 1995). Direct manipulation is particular useful for "experiential" cognition and indirect manipulation (command/written manipulation, in particular) for "reflective" cognition (Norman, 1993).

The discussion about direct and indirect manipulation in educational software has been particularly strong between Logo advocates and dynamic geometry supporters (see, e.g., diSessa, Hoyles & Noss, 1995). But, as Hoyles (1995) noted, the difference between direct manipulation geometric environments and programming/indirect manipulation environments "is becoming increasingly blurred". New versions of Logo Microworlds incorporates many aspects of direct manipulation and more recent versions of dynamic geometry software allows macro programming—a construction can be repeated or adapted using the macro.

After many years of software development, I could not be more in agreement with Ogborn (1994, p.126):

> The design of all learning environments remains a black art. In reality, people usually start with some given computational resources, and see some problems for which those resources may be able to be exploited. Finding a way to exploit those resources, in a way which engages the attention of students in some desirable fashion on a given problem, is not a general problem with a general solution, but a specific problem in a specific context with a few specific solutions. Some people find solutions which are more imaginative and suggestive of further exploitation than do others, just as some write better books than others. As to writing, it was Bernard Shaw who gave an excellent set of rules for writing well, the most important of which was the last, which read, "Break all or any of the above rules when it is in good taste to do so". The same rule applies here.

3.3 Some Contributions to the Creation of New Computer Tools for Learning Physics (and Mathematics)

3.3.1 The motivation: students' learning difficulties and misconceptions

Students encounter many difficulties in learning science and mathematics. In the 1980s, a large body of research was done on student learning difficulties and on misconceptions they develop before and during learning. A bibliography of such studies published in 1991 (Pfundt & Duit) lists about 2000 references. This line of research was then influencing many other fields and educational software development was a clear example of this influence (Harvard Educational Technology Center, 1988; Teodoro, 1990). The majority, if not all, of the titles I designed, or co-designed, between 1984 and 1994 can not only be considered tentative answers to help students overcome difficulties but also tentative tools to help teachers explain visually difficult topics. As I discuss in section 3.5, the literature on learning difficulties was always a starting point for choosing topics and ways of dealing with it. The concepts behind almost all of the titles described on this section can now be implemented in Modellus, without major difficulties.

At that time, it was difficult to find schools with enough computers to allow student direct use— the software was conceived for student use but also to allow teacher projection and class discussions based on screen images.

The first titles I designed (and programmed) were done for the ZX Spectrum between 1984 and 1987. They will not be presented here, since later on they were adapted to the MS-DOS environment. The following sub-section gives a brief description of all titles published in Portugal (some have also been published in other european countries, in the scope of EPES, European Pool of Educational Software, a project of educational software interchange). *Functions*, *Change*, *Mouse in motion*, and *Function games* have been used in four master theses presented in different portuguese universities. Besides manuals (some with classroom work sheets), collections of activities to guide teachers to use the software have been published by different poles of the MINERVA project or by the Ministry of Education—e.g., Ribeiro & Rocha (1993).

Research has shown that learning with computer exploratory environments is a difficult task (de Jong & Joolingen, 1998), but the same is true for more concrete environments, such as school laboratories. As Driver (1983) pointed out:

> If we wish children to develop and understanding of the conventional concepts and principles of science, more is required than simply providing practical experiences. The theoretical models and scientific conventions will not be 'discovered' by children through their practical work. They need to be presented. Guidance is then needed to help children assimilate their practical experiences into what is possibly a new way of thinking about them (p. 9).

Children need guidance to explore scientific ideas, in all settings. They will not "re-discover" science just by clicking with a mouse or manipulating a keyboard. Providing guidance is a essential part of planning computer based exploratory learning environments. Guidance can be given on the screen, on paper, and directly by the teacher. In the titles describe below, guidance was supposed to be given on paper (usually work sheets, such as those available in Ribeiro & Rocha, 1993) and by the teacher, specifically in the format of all-class discussion.

3.3.2 *Early titles: Kinematics* ("Cinemática"), *Projectiles* ("Projécteis"), *Space* ("Espaço") and *Functions* ("Funções")

The following paragraphs give a brief description of the first titles, all done for the MS-DOS operating system. It was difficult to implement them due to the almost total absence of graphic tools for programming in the then available programming languages. For some of the titles, it was necessary to use a (primitive) graphical author environment—*Mosaik*—made in Norway (and based on *MetaWindows*, then one of most common collections of graphical routines for the MS-DOS operating system).

Three of the titles are about motion and one about functions. Motion is a relatively simple topic to implement in computers—moving objects, trajectories, and graphs can easily appear simultaneously, helping linking multiple representations of motion.

Kinematics—("Cinemática") (Teodoro & Seabra, 1990)—is a multiple representation tool for linear motion with constant acceleration or with constant velocity. The user can:

1. observe the motion of one or two particles;
2. control initial position, initial velocity, acceleration;
3. see the equations of motion;
4. see graphs and tables of position, velocity and acceleration.

Projectiles ("Projécteis") (Teodoro, Gonçalves & Melo, 1990) allows the user to explore two-dimensional motion in a vertical plane, with or without air resistance. The user can also analyse multiple trajectories and graphs.

Space ("Espaço") (Teodoro & Santos, 1990) is a direct manipulation tool to discuss one of the most persistent conceptual difficulties with students—the distinction between coordinate and distance travelled. It shows simultaneous multiple representations of movement, trajectory, and graphs of position and distance travelled.

Functions ("Funções")—Figure 3.3—(Teodoro & Batalha, 1990) is graphical analysis utility with a very simple but powerful interface. This software is still recommended by the portuguese curriculum for senior high school mathematics, despite its evident limitations when compared with current software. With *Functions*, the user can graph families of functions, specify intervals for domains and counter-domains, specify scales, see tables of values, etc. A user can also "hear" how a function sounds. Loading and saving is done with a special interface, where meaningful names (with spaces and more than eight characters) can be used. This software was used in a qualitative study by Domingos (1994), with 10th graders. Domingos was particularly interested in investigating how students deal with multiple representations in studying functions and solving equations. He concluded that the computer was a valuable aid to learn about functions and equations.

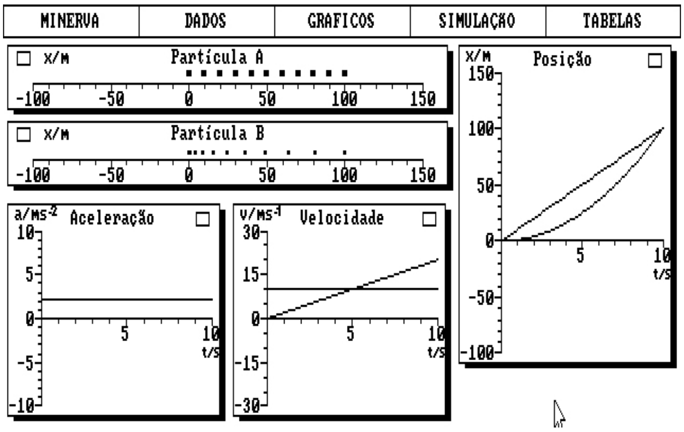
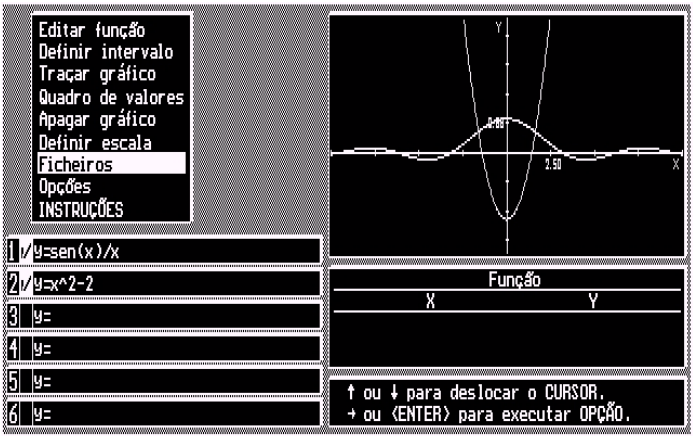


Figure 3.2 Kinematics ("Cinemática") (Teodoro & Seabra, 1990), a multiple representation tool for linear motion.



1 ✓ y=sen(x)/x
2 ✓ y=x^2-2
3 y=
4 y=
5 y=
6 y=

↑ ou ↓ para deslocar o CURSOR.
→ ou <ENTER> para executar OPÇÃO.

Figure 3.3 "Funções", *Functions*, a graphical analysis tool.

3.3.3 Dinamix and Change ("Variações")

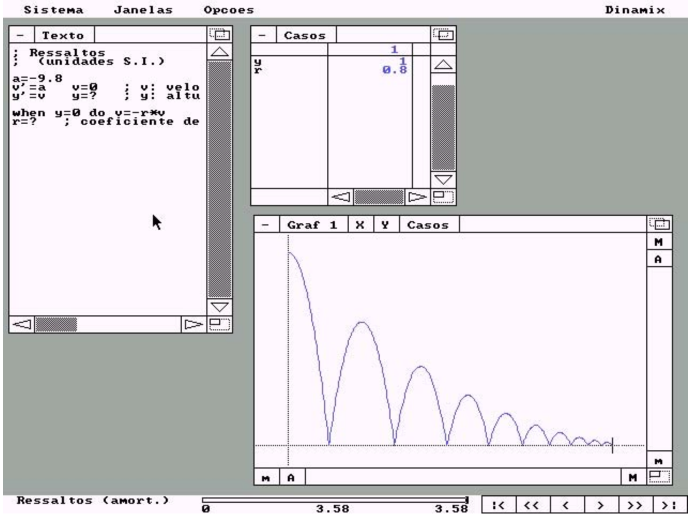
Dinamix (Lobo, Clérigo, Martins, Lobo & Teodoro, 1993[1])—Figure 3.4—is a modelling system that tested some of the ideas that were used in Modellus a few years later. It is an equation based modelling system. In the *Text* ("Texto") window the user can write functions, differential equations, and conditions. It is possible to have different sets of parameters and initial conditions, defined in the *Cases* ("Casos") window. After running the model (it is possible to define a domain for the independent variable *t*), the output can be seen as a graph. The use of differential equations in *Dinamix*, instead of difference equations (then the common way of making computer modelling), "can be seen as a bridge between using the computer to give numerical solutions to problems and using the calculus to give analytical solutions" (Boohan, 1994, p.55).

Dinamix has never been used extensively in Portugal, compared with other software such as *Kinematics*: its *concept* was too difficult and unfamiliar to be understood by teachers—and the interface was too primitive to allow easy learning and use. But it was an important step in order to test the idea of making models directly with functions and differential equations. During three years I gave courses for first year engineering students with it and they had no major difficulties in making models with this software. I've also taught a unit on forces and motion for high school students (10th graders) using *Dinamix*, and all students were also able to make meaningful models with it, and solving traditional physics problems.

Change ("Variações")—Figure 3.5—(Teodoro & Batalha, 1993) is a computer environment to explore, in a semi-quantitative way, the basic ideas of quantitative modelling: rates of change and functions. The user can move a person (top window) slower or faster, change the direction of the motion, stop the motion, etc. This can be done with direct control of the person, controlling the graph of the position (a function of time, left graph), or controlling the rate of change of position (a rate of change of a function, right graph). With *Change*, it is possible to discuss *derivation* (the mathematical process how we describe how a function changes—it is the connection

---

[1] The initial design was done in 1989.



```
; Ressaltos
;  (unidades S.I.)

a=-9.8
v'=a    v=0    ; v: velo
y'=v    y=?    ; y: altu

when y=0 do v=-r*v
r=?    ; coeficiente de
```

	1
y	1
r	0.8

Figure 3.4 *Dinamix* (Lobo et al.), a primitibe modelling system with functions, differential equations, and conditions.

from the graph on the left to the graph on the right) and *integration* (the mathematical process how we describe how the rate of change can inform us about the function—it is the connection from the graph on the right to the graph on the left).

Change was used by Ribeiro (1995) in a quasi-experimental study about group work in computer learning environments. She concluded that the learning environment, based on a guided discovery approach, using the software, written worksheets, group discussions and teacher support was a *powerful learning environment*, in the sense of De Corte (1989).

3.3.4 *Galileo* (Macintosh), *Galileo* (Windows), *Mouse in Motion* ("Ratos em movimento")

Galileo for the Macintosh (Teodoro & Silva, 1990)—Figure 3.6—was the only title developed specifically for Macintosh computers, since its design was then (1988) not possible to implement in MS-DOS environments.

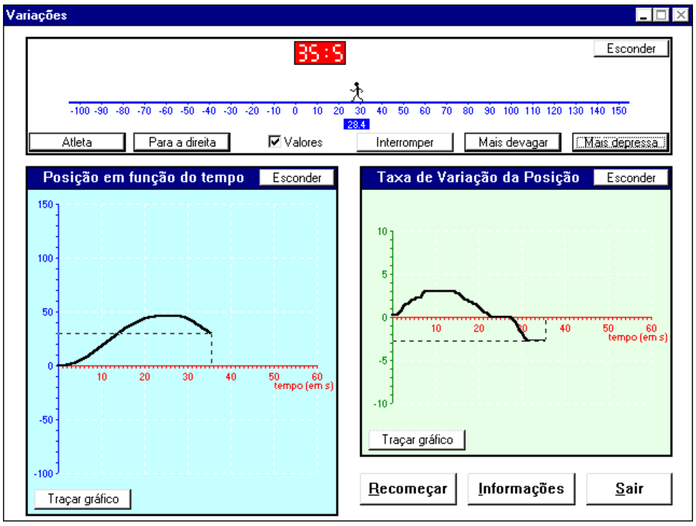


Figure 3.5 *Change* ("Variações")

The basic concept of *Galileo* is to allow the student to explore abstract physical concepts by direct manipulation, functioning as a *conceptual laboratory* (Teodoro, 1992). With the software, the user can:

1. "act" on a particle, applying forces in certain directions;
2. define velocity and other physical concepts acting on vectors;
3. control the environment where the particle moves, "connecting" or "disconnecting" friction and gravity;
4. see the motion and, or, representations of the motion, such as trajectories, vectors, strobe images, graphs, etc., simultaneously and in real time;
5. explore representations in different reference frames;
6. zoom in or zoom out the trajectory and other representations.

Galileo's interface presented a few innovative aspects, namely multiple (five) levels of complexity of the interface, related to the cognitive development levels stages of learning in dynamics (Teodoro, 1992). Each level is accessed through a password. The first has a very high level of *perceptual fidelity* but, as the level increases, more complex representations, with less perceptual fidelity, are available (such as vectors, graphs, etc.)—see Figure 3.6.

Galileo was later completely redesigned for the *Windows* environment. The new version (Teodoro & Clérigo, 1993)—Figure 3.7—is much less flexible:

1. the user can't change windows size and position;
2. most icons were substituted by text buttons;
3. new functionalities such as an hide/show button have been introduced;
4. graphs have less options;
5. etc.

These changes were made after observations of students using the software. In early 1990s, students (and teachers) were not familiar with graphical interfaces and they easily

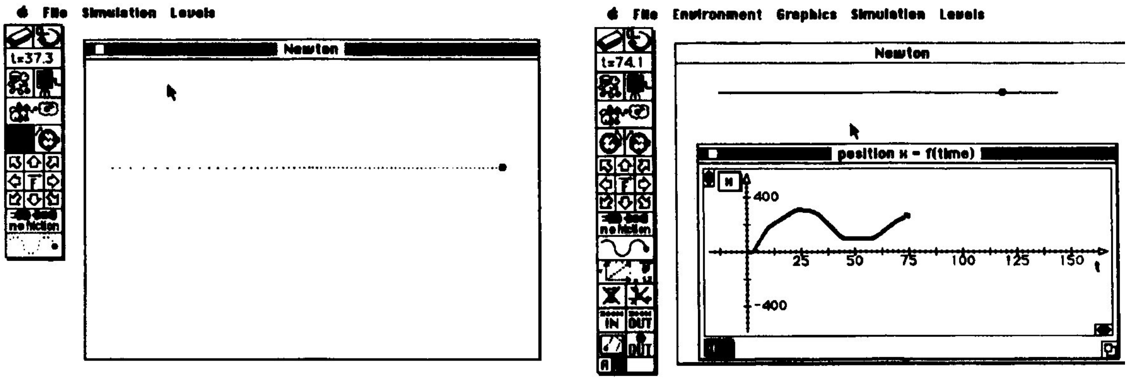


Figure 3.6 Galileo for the Macintosh (Teodoro & Silva, 1990). The first level (left image) has less options available and a higher degree of perceptual fidelity. Higher levels have more options, demanding more knowledge and skills from the user.

got confused with all the flexibility of this type of interface (this is not true now, when *all* software has graphical interfaces and most users are familiar with it). *Galileo for Windows* also introduced some useful new functionalities, such as the possibility of moving a particle using the mouse. In this case, the mouse behaves as a motion sensor and the software can be called a "mouse-based laboratory" (original idea of Judah Schwartz, from the Harvard Graduate School of Education, discussed in a seminar he gave at FCTUNL in 1990).

Mouse in Motion ("Rato em Movimento") (Teodoro & Clérigo, 1994)—Figure 3.8—is a development of the concept of *mouse-based laboratory* with some more powerful functionalities, such as the ability to draw graphs before the motion and then see the motion that correspond to the graphs. The motion can be obtained not only from position-time graphs but also from velocity- and acceleration-time graphs.

Mouse in motion was used in a master thesis by Féteira (1996) to study how high school students learn graphical representations of motion in a normal classroom environment. She concluded that students significantly improved their knowledge of graphical representations.

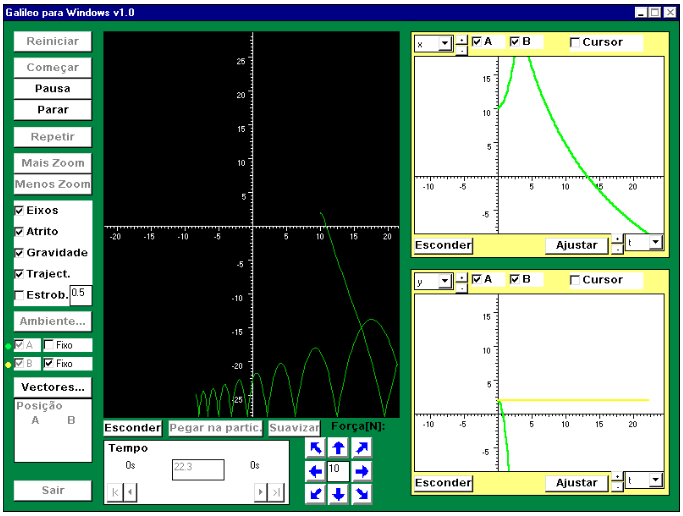


Figure 3.7 Galileo for Windows (Teodoro & Clérigo, 1993). The same concept, but much less flexibility, being closer to the skills of a typical user in early 1990s.

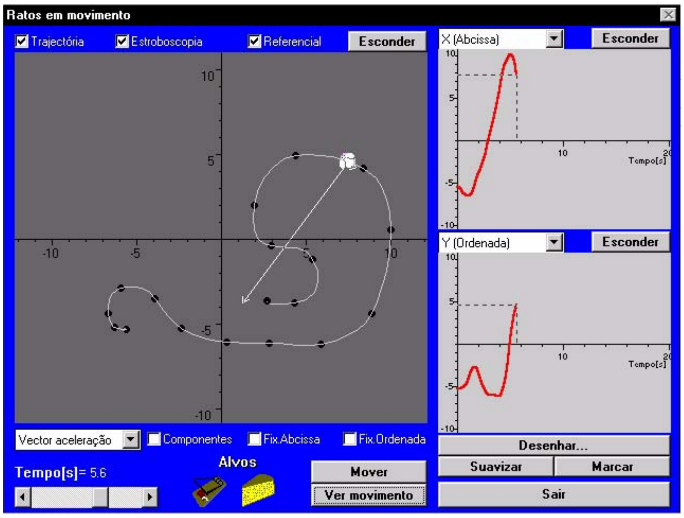


Figure 3.8 Mouse in Motion ("Rato em movimento") (Teodoro & Clérigo, 1994) is an exploratory tool to make motion experiments with the mouse of the computer. The user can also draw graphs and then explore what is the motion that correspond to the graphs.

3.3.5 *Descriptive geometry* ("Geometria descritiva") and *Functions and derivatives* ("Funções e derivadas")

These titles are exercises in software design for exploring multiple mathematical representations. *Descriptive geometry* ("Geometria descritiva") (Clérigo & Teodoro, 1994)—Figure 3.9—is a direct manipulation exploratory environment for 3D geometry. The user can create a 3D representation of an object, using coordinates, and then see a 3D representation of it or of its projections on the horizontal and vertical plane. The most interesting feature, from a software designer point of view, is the possibility of acting with the mouse on any of the visual representations and then see what happens in the other representations.

As with other software tools described above, the windows of *Descriptive geometry* are fixed, and have a hide/show button. This hide/show button can be particular useful to allow the teacher to promote discussions of the type "what happens if...".

Functions and derivatives ("Funções e derivadas") (Teodoro & Clérigo, 1994) is a graphical utility that has a distinguishable feature: the user can get a graph of a function, ask the software to compute its derivative (it has a symbolic motor, later used in Modellus) and then manipulate the derivative, with the mouse, to see how the function changes. It is also possible to do other usual explorations with graphs and tables, such as observing tangent lines, investigating the effect of parameters, etc.

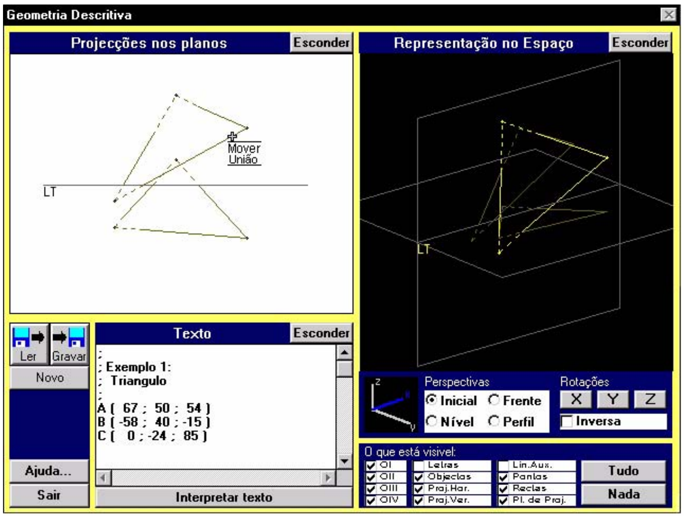
3.3.6 *Function games I & II* ("Jogos de funções I & II"), *Field force games* ("Jogos de campos de forças") and *Vrum... vrum: motion games* ("Vrum... vrum... Jogos de movimento")

These four titles can also be considered as design exercises in educational computer games. As games, they have a goal, based in some intuitive or explicit understanding of scientific concepts and representations, one or more tasks to accomplish, and feedback. The interfaces of these games were very simple—they were planned to be used without any written or oral instructions.

In *Function games* ("Jogos de funções")—Figure 3.11—(Teodoro & Seabra, 1993; 1994) the goal, and the task, are related with one or more movements (e.g., "how big must the speed of the fox be in order to catch the rabbit?"), given some constraints, and the scientific concepts are related with speed, rates, computing rates from graphs, etc.

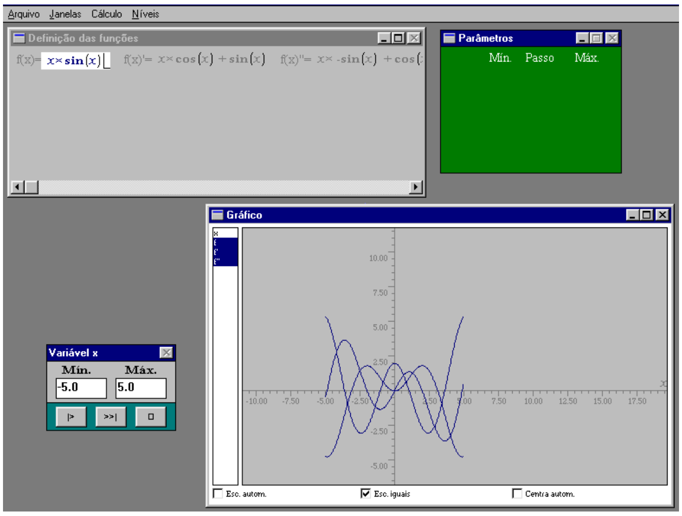
Field force games ("Jogos de campos de forças") (Teodoro & Vieira, 1994)—Figure 3.12—allow the user to test his or her understanding of the effect of field in moving charges. "Charges" move in real time and leave a stroboscopic trail, that can be used to find the position of fixed charges. The goal can only be attained when the user correctly interprets stroboscopic representations of the motion of the charges.

Vrum... vrum motion games ("Vrum... vrum jogos de movimentos") (Teodoro & Clérigo, 1994)—Figure 3.13—is a unique game where graphical interpretation is used to



```
;
; Exemplo 1:
;  Triangulo
;
A (  67 ;  50 ;  54 )
B ( -58 ;  40 ; -15 )
C (   0 ; -24 ;  85 )
```

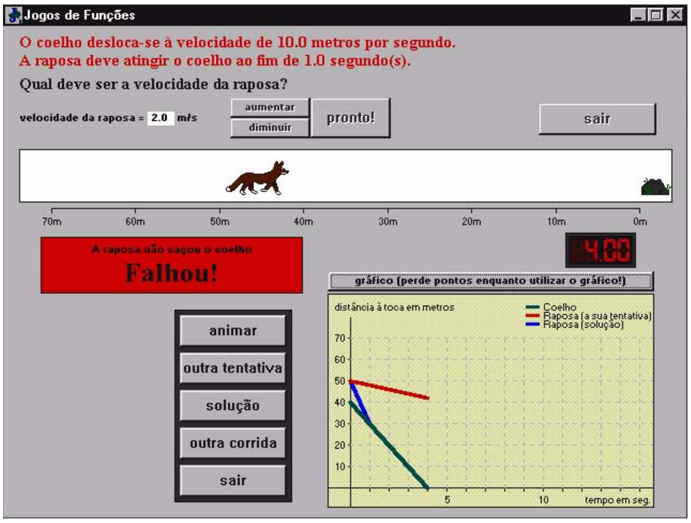
Figure 3.9 Descriptive geometry ("Geometria descritiva") (Clérigo & Teodoro, 1994), a direct manipulation environment to explore 3D representations from multiple perspectives. It is possible to move the projections and see what happens in the 3D representation.



f(x)= x× sin(x)   f(x)'= x× cos(x) + sin(x)   f(x)''= x× -sin(x) + cos(

Variável x — Mín. -5.0 Máx. 5.0

Figure 3.10 Functions and derivatives ("Funções e derivadas") (Teodoro & Clérigo, 1994), a graphical utility where the user can obtain derivatives (it has a symbolic motor) and change these derivatives—with the mouse—seeing what happens to the function.



O coelho desloca-se à velocidade de 10.0 metros por segundo.
A raposa deve atingir o coelho ao fim de 1.0 segundo(s).
Qual deve ser a velocidade da raposa?

velocidade da raposa = 2.0 m/s

A raposa não caçou o coelho
Falhou!

Figure 3.11 Function games I ("Jogos de funções") (Teodoro & Seabra, 1993).

create motions, in the more complex levels of the game, or motion is used to create graphs, in less complex levels.

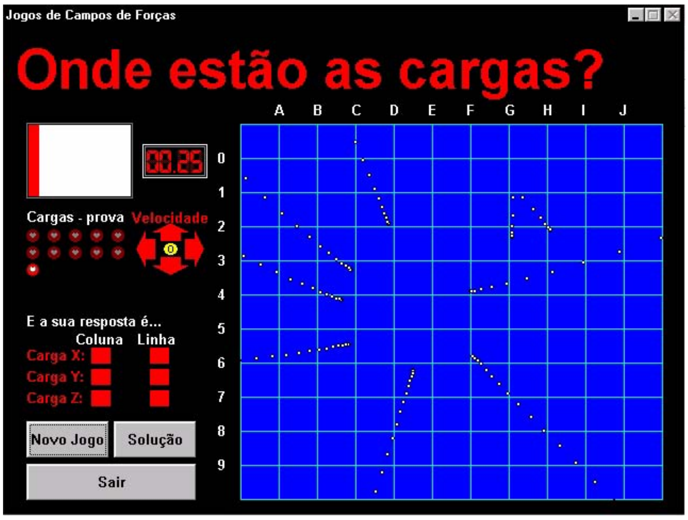
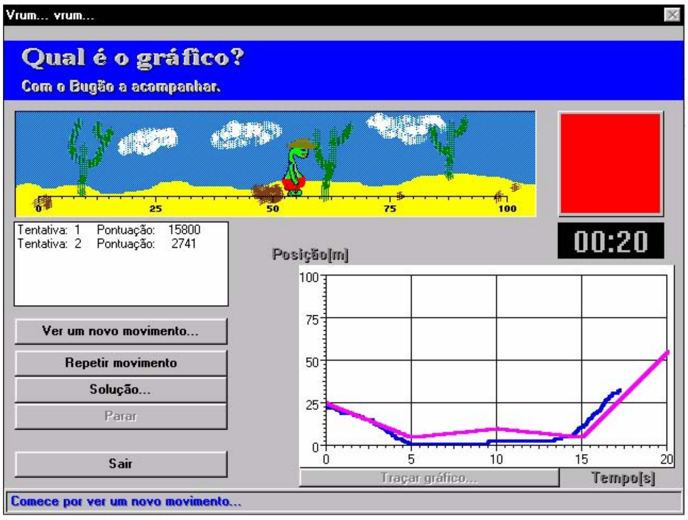


Figure 3.12 Field force games ("Jogos de campos de forças") (Teodoro & Vieira, 1994) is a game where the student can become familiar with the interaction between moving charges and fields created by fixed charges. The goal is to find the unknown position of fixed charges.



Tentativa: 1 Pontuação: 15800
Tentativa: 2 Pontuação: 2741

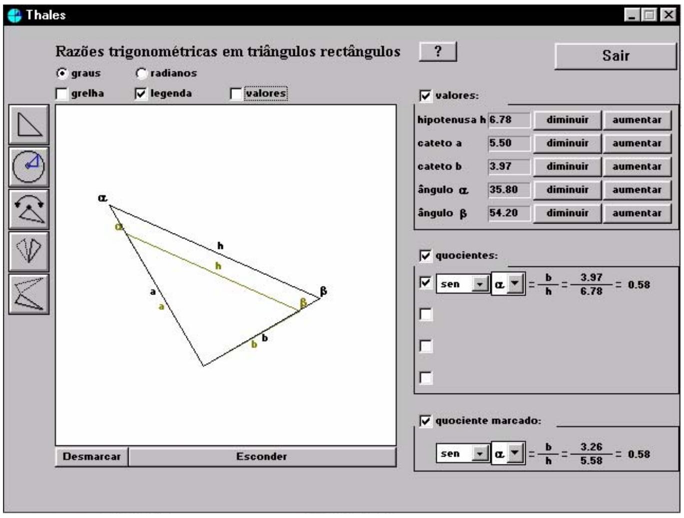
Comece por ver um novo movimento...

Figure 3.13 Vrum... vrum motion games ("Vrum... vrum jogos de movimentos") (Teodoro & Clérigo, 1994) is a game to test the understanding of graphical representations of motion.

3.3.7 *Thales* and *Statistics: exploring basic concepts* ("Estatística: explorar conceitos básicos")

Thales (Junqueira, Valente, Teodoro & Seabra, 1993)—and *Statistics: exploring basic concepts* ("Estatística: explorar conceitos básicos") (Teodoro & Seabra, 1994) are two software design exercises in simple exploratory environments for mathematical topics fundamental to physics learning. Simple in the sense they are not of the same genre as general and open tools (e.g., *The Geometer's Sketchpad*, Key Curriculum Press, 1995). They address specific topics and specific learning difficulties in a close manner.

Thales—Figure 3.14 and Figure 3.15—has four screens and is directed to trigonometric relations, in triangles and in the trigonometric circle. *Statistics: exploring basic concepts*—Figure 3.16—also has four screens. In two of them, it is possible to make probability experiments and, in the other two, experiments with one or two variables. The interface combines a direct manipulation interface (e.g., the user can act directly on the images) with an indirect interface (e.g., the user can use text buttons to act



● graus ○ radianos

☐ grelha ☑ legenda ☐ valores

☑ valores:

hipotenusa h 6.78

cateto a 5.50

cateto b 3.97

ângulo α 35.80

ângulo β 54.20

☑ quocientes:

☑ sen α = $\frac{b}{h} = \frac{3.97}{6.78} = 0.58$

☐

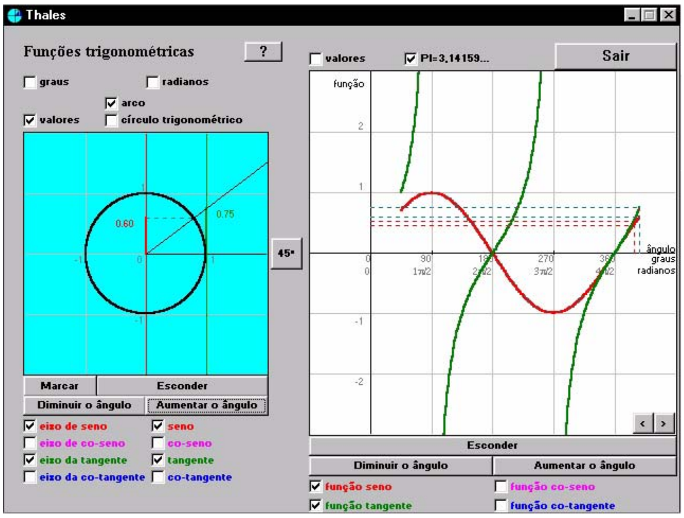
☐

☐

☑ quociente marcado:

sen α = $\frac{b}{h} = \frac{3.26}{5.58} = 0.58$

Figure 3.14 One of the *Thales* (Junqueira et al., 1993) screens is directed to the exploration of trigonometric relations in triangles. The user can easily compare trigonometric ratios in different triangles.



☐ graus ☐ radianos

☑ arco

☑ valores ☐ círculo trigonométrico

☐ valores ☑ PI=3,14159...

☑ eixo de seno ☑ seno

☐ eixo de co-seno ☐ co-seno

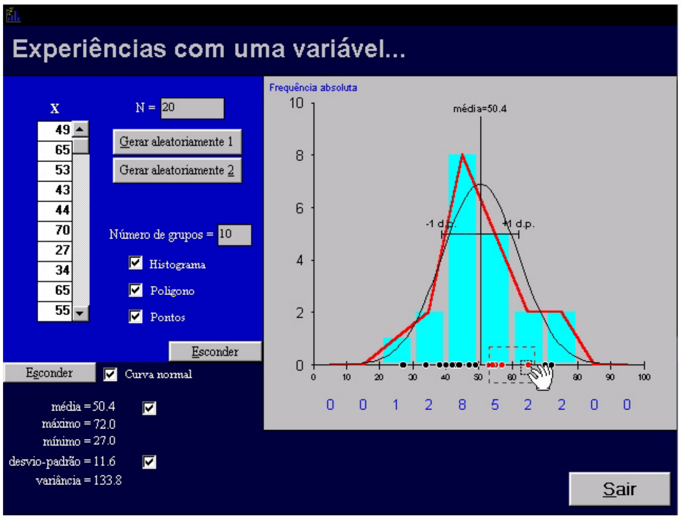
☑ eixo da tangente ☑ tangente

☐ eixo da co-tangente ☐ co-tangente

☑ função seno ☐ função co-seno

☑ função tangente ☐ função co-tangente

Figure 3.15 Another of the *Thales* (Junqueira et al., 1993) screens. This one allows the exploration of trigonometric relations in the trigonometric circle and on the corresponding graphs of the trigonometric functions.



X: 49, 65, 53, 43, 44, 70, 27, 34, 65, 55

N = 20

Número de grupos = 10

☑ Histograma

☑ Polígono

☑ Pontos

☑ Curva normal

média = 50.4 ☑

máximo = 72.0

mínimo = 27.0

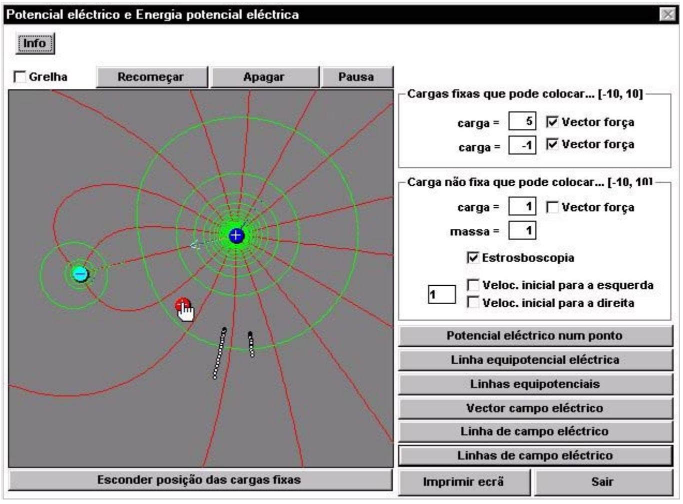
desvio-padrão = 11.6 ☑

variância = 133.8

0 0 1 2 8 5 2 2 0 0

Figure 3.16 One of the four screens of Statistics: exploring basic concepts (Teodoro & Seabra, 1994). In this screen, the user can create bar and polygon graphs of frequency tables, computer measures of dispersion, etc. It also possible to use the mouse to change raw data and then see the change in graphs and computed parameters

on the images that represent mathematical objects). The use of these programs is very straightforward due to the combination of direct and indirect manipulation, and absence of pull-down menus.



☐ Grelha

Cargas fixas que pode colocar... [-10, 10]

carga = 5 ☑ Vector força
carga = -1 ☑ Vector força

Carga não fixa que pode colocar... [-10, 10]

carga = 1 ☐ Vector força
massa = 1
☑ Estrosboscopia
1 ☐ Veloc. inicial para a esquerda
☐ Veloc. inicial para a direita

Figure 3.17 Faraday (Teodoro et al., 1993) is an exploratory tool to make experiments with electric fields. "Charges" can be directly manipulated with the mouse, even when they are moving.

With *Statistics*... is possible to make some interesting mathematical explorations, such as changing single or multiple points in scattergrams with regression lines. This feature is now one of the more interesting features of *Fathom*, an exploratory environment for statistics education recently published (Key Curriculum Press, 2000).

3.3.8 *Faraday, Another world* ("Outro mundo") and *Signals and oscilloscope* ("Sinais e osciloscópio")

These three titles addressed different aspects of fields and periodic signals, in the case of *Signals and oscilloscope*.

Faraday (Teodoro et al. 1993)—Figure 3.17—allows the user to make experiments with electric fields, holding "charges" on the field, then leaving the charges and observing its motion, draw field lines and equipotential lines, draw field vectors, etc.

Another world ("Outro mundo") (Teodoro & Vieira, 1994) is an exploratory environment for gravitational fields. It is possible to create configurations of one to three fixed stars and then place planets or starships in space. As in the case of *Faraday*, the user can interact with any object (or with its properties, such as velocity) with the mouse, change gravity constant (even when the simulation is running), etc.

Signals and oscilloscope ("Sinais e osciloscópio") (Teodoro, Romano, & Cardoso, 1994) has several screens dedicated to the exploration of what happens to an electron beam when it crosses an electrical field and screens dedicated to the simulation of an oscilloscope. It also has screens for exploring the combination of different signals in vertical and horizontal plates of the oscilloscope.

3.3.9 *Mendeleieff*

The last title in this series, *Mendeleieff* (Teodoro & Batalha, 1992)—Figure 3.18—is a visual database on the periodic properties of the elements, designed as an exercise of multiple representations. With it, the user can get the usual information about each element and obtain graphs of different properties, either as scattergrams or as bar charts or other types of graphs. But the most useful characteristic of this software is the possibility of linking place of an element in the periodic table with place on the graph (clicking on the graph or on the periodic table), which allows the exploration of trends in properties and on the way the periodic table is organized.

3.3.10 A note and a synthesis

I described in this section 21 titles designed or co-designed between 1986 and 1994. Almost all were done as "design exercises" to test how computer software can help students overcome typical learning difficulties. They were not necessarily thought of as *final* products intended for general use in schools. Seven to fifteen years after it is possible to understand how important they were for the design of Modellus. They were "conjectures" about a *software concept*. And they were essential to bring out mistakes and dubious conjectures and, as Popper wrote, "make us understand the difficulties of the problem which we are trying to solve. This is how we become better acquainted with our problem, and able to purpose more mature solutions" (Popper, 1989, p. vii).

Do the 21 titles have something in common? In spite of the fact that they addressed different topics of physics, mathematics, and chemistry, they all share the following characteristics:

1. a graphical user interface;
2. an exploratory approach to one or more topics;
3. a multiple representations view of the topics.

Some titles also use extensively a direct manipulation interface, where the mouse directly acts on *concrete-abstract objects* (Hebenstreit, 1987). These concrete-abstract objects can be closer to real objects, such as particles, or physical and mathematical constructs such as vectors, geometrical objects, graphs, and fields.

All titles address one or more learning difficulties, most of them identified in the research literature. These learning difficulties have one thing in common: visualization

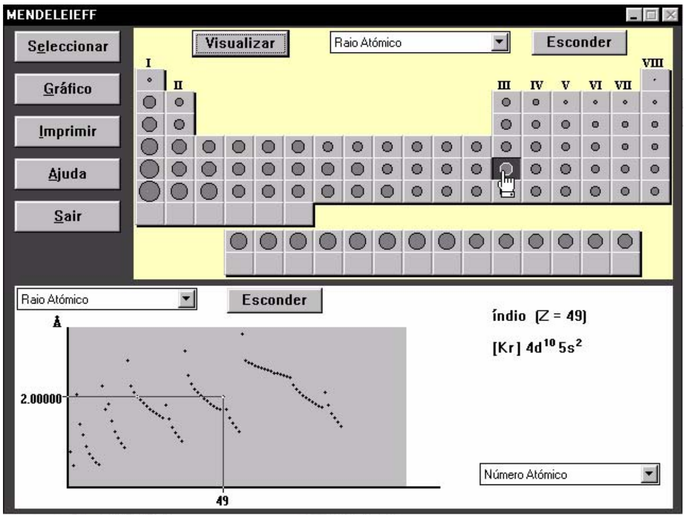


Figure 3.18 Mendeleieff (Teodoro & Batalha, 1992). This visual database allows the exploration of the relations between the properties of atomic elements and its place in the periodic table.

can help to overcome it. Table 3.1 gives a synthesis of the goal of each title and identify the learning difficulty associated with it.

Table 3.1 Synthesis of titles published between 1986 and 1994

Title	Topic	Learning difficulties addressed
Kinematics (“Cinemática”)	Linear motion	Correspondence between graphs and motion
Projectiles (“Projécteis”)	Two-dimensional motion (projectiles)	Correspondence between graphs and motion
Space (“Espaço”)	Coordinate and distance	Cumulative variable *vs.* coordinate at a certain instant
Functions (“Funções”)	Graphical and tabular representation of functions	Correspondence between graphs and equations
Dinamix	Modelling with differential equations	Rate of change and numerical integration
Change (“Variações”)	Rate of change and functions	Correspondence between rates and functions
Galileo (Macintosh)	Motion and forces	Relation between force and velocity. Correspondence between graphs and motion
Galileo (Windows)	Motion and forces	Relation between force and velocity. Correspondence between graphs and motion
Mouse in Motion (“Rato em movimento”)	Two-dimensional motion	Correspondence between graphs and motion. Vectors
Descriptive geometry (“Geometria descritiva”)	Projection of 3D objects	Visualization of projections in vertical and horizontal planes
Functions and derivatives (“Funções e derivadas”)	Graphical representation of functions and derivatives	Correspondence between derivatives and functions
Function games I & II (“Jogos de funções I & II”)	Graphical representation of functions that describe motion	Correspondence between graphs and motion
Field force games (“Jogos de campos de forças”)	Electrical fields	Interaction between fields and moving charges
Vrum... vrum: motion games (“Vrum... vrum... Jogos de movimento”)	Graphical representation of functions that describe motion	Correspondence between graphs and motion
Thales	Angles and trigonometry	Correspondence between trigonometric ratios and triangles. Correspondence between trigonometric functions and angles
Statistics: exploring basic concepts (“Estatística: explorar conceitos básicos”)	Probability, histograms and correlation	Probability concept. Statistical description of variables and relations between variables.
Faraday	Electrical fields	Interaction between charges and fields. Vector and potential description of fields
Another world (“Outro mundo”)	Gravity fields	Interaction between fields and moving masses

Table 3.1 Synthesis of titles published between 1986 and 1994 (Continued)

Title	Topic	Learning difficulties addressed
Signals and oscilloscope ("Sinais e osciloscópio")	Electrical fields and periodic signals	Action of an electrical fields on an electron beam
Mendeleieff	Periodic properties	Correspondence between physical and chemical properties of the elements and place of the element on the periodic table

3.4 Modellus: the *Concept* and a Technical and Pedagogical Description

3.4.1 The *concept*

As discussed on section 3.2, a software *concept* is "what the program will do, what it will look like, and how it will communicate with the user" (Cooper, 1995, p. 24). The Modellus concept can be briefly described as:

1. It is a software tool to *create and explore multiple representations* of mathematical models using functions, differential equations and iterative equations.
2. It has a multiple window environment. In one of the windows, the user can write a model, writing equations as they are written on paper; in other windows, the user can create and interact with animations of the models, using abstract objects, such as vectors and graphs, or more concrete objects, such as video and photos.
3. The communication with the user is based on the concept of "intellectual mirror" (Schwartz, 1989)—the software acts as a mirror of what the user thinks.

From where comes the Modellus *concept*? It comes from multiple influences: teaching experience, experience on design of computer learning environments, literature on learning difficulties.

First, from experience in learning environments. As a teacher, I had, for many years, direct contact with students puzzled by the meaning, forms and implications of mathematical models. As a supervisor and teacher educator, also for many years, I had a similar experience: teachers (physics and mathematics teachers) have many difficulties with mathematical models. John Dewey wrote that "We know an object when we know how it is made, and we know how it is made in the degree in which we ourselves make it" (John Dewey, *Experience and nature*, 1925, quoted by Garrison, 1997). And students and teachers have little opportunities to make objects in learning environments, particularly *abstract objects* such as mathematical models. Modellus is a tool to help change this situation.

The personal computer revolution and the Internet changed the way people live and work. This revolution started with graphical and direct manipulation computer user interfaces. Computer tools, such as Modellus, could not be done in previous interface styles, such as written command-based or menu selection interfaces. The software described in the previous section tested most of the design ideas used in Modellus. Of particular interest was the experience in the use of multiple representations. Before Modellus, this concept of multiple representations had been largely experienced and tested in many contexts and by many software developers, including myself. Graphical and direct manipulation user interfaces also allowed the design of software where its "Manifest Model" can be very close to the user's "Mental Model" (Cooper, 1995).

The enormous literature on alternative conceptions and learning difficulties of specific science and mathematical concepts also had a significant role in the specification of the Modellus concept. From the early studies (e.g., Viennot, 1979; Caramazza, McClossky & Green, 1980), which identified persistent misconceptions and learning difficulties associated with force and motion, to other studies such as Ponte (1984) that showed how difficult it was for students to make meaning of graphs, to later studies (a good synthesis for physics studies can be found on Wandersee, Mintzes & Novak, 1994; for mathematics, see Grouws, 1992) that categorized and theorized about misconceptions and learning difficulties.

Wandersee, Mintzes & Novak (1994) synthesize the research claims made about alternative conceptions (and learning difficulties):

1. Learners come to formal science instruction with a diverse set of alternative conceptions concerning natural objects and events.
2. The alternative conceptions that learners bring to formal science instruction cut across age, ability, gender, and cultural boundaries.
3. Alternative conceptions are tenacious and resistant to extinction by conventional teaching strategies.
4. Alternative conceptions often parallel explanations of natural phenomena offered by previous generations of scientists and philosophers.
5. Alternative conceptions have their origins in a diverse set of personal experiences including direct observation and perception, peer culture and language, as well as in teachers' explanations and instructional materials.
6. Teachers often subscribe to the same alternative conceptions as their students.
7. Learners' prior knowledge interacts with knowledge presented in formal instruction, resulting in a diverse set of unintended learning outcomes.
8. Instructional approaches that facilitate conceptual change can be effective classroom tools.

Research on computers in education targeted, since early studies, some of these claims. Many authors argued that computer tools can be powerful cognitive tools to help teach for understanding and facilitate conceptual change (e.g., Nickerson, 1995; Horwitz, & Barowy, 1994; De Corte, 1994). According to Nickerson (1995), to do these, software must:

1. deal with misconceptions;
2. promote active learning and discovery;
3. use dynamic and interactive representations;

4. allow student-developed simulations;
5. and give supportive environments.

As I noted in the previous section, most of the software I designed before Modellus target one or more misconceptions or learning difficulties. Similar goals have been also identified by other researchers (e.g., diSessa, 1982; White, 1984). To deal with misconceptions, a major point is allowing students test the consequences of their naïve ideas (Teodoro, 1990).

Active learning and discovery is a major issue with exploratory software, but as mentioned before, guidance is essential to make it useful since learners encounter many problems with discovery learning environments. Most common problems are (de Jong, 1998):

1. hypothesis generation;
2. design of experiments;
3. interpretation of data;
4. and regulation of learning.

Of particular importance is *regulation of learning*. A common situation is that "learners don't know what they have done, and have no clear idea on what to do next" (de Jong, 1998, p. 185). This problem shows how important is guidance, given by teacher, by peers, and by the written documentation: the goal is to give a *supportive environment*, essential to make learning a *personal* but also a *social* process.

Student-developed simulations have been a goal for software designers for many years. Computer microworlds, based on programming languages (e.g., Papert's Logo) were the first attempt. Similar approaches, based on different computer metaphors, have also been tried recently (e.g., Repenning, 1993; Resnick, 1994). But none of these attempts seem to have been widely adopted in schools, either because they are difficult to use or because they cannot be easily related with the school curriculum.

Computer exploratory learning environments are usually seen as constructivist tools (de Corte, 1989). Constructivist tools in the sense they are designed to be used by learners to help them make sense of concepts, theories and representations. This is the case of Modellus. The following subsections describe it, first from a technical point of view and after from a pedagogical point of view.

3.4.2 Modellus: a direct manipulation interface

Almost all the software now available has a direct manipulation interface, first proposed by Shneiderman (Shneiderman, 1983). The idea behind direct manipulation interfaces is amazingly simply: the actions must be done directly on the objects, without being mediated by written language. The best example Shneiderman considers of direct manipulation is driving a car. *Driving a car* is, certainly, a good metaphor for the design of direct manipulation software:

> The scene is directly visible through the front window, and performance of actions such as braking or steering has become common knowledge in our culture. To turn left, the driver simply rotates the steering wheel to the left. The response is immediate and the scene changes, providing feedback to refine the turn. Imagine trying to turn by issuing a command LEFT 30 DEGREES and then another command to see the new scene (…) (Shneiderman, 1992, p. 183).

A fundamental design choice made in Modellus was to identify what could be done using graphical metaphors and what should be done using other representations. Some designers choose graphical metaphors, even for equations (e.g., *Stella*, at least partially; Feurzeig, 1993). In Modellus, the choice was different: the user should write equations as he writes on paper (as far as possible). The environment for manipulating equations and all other features are implemented in standard graphical Windows elements: windows, pull-down menus, dialogue boxes, buttons, etc.

Another important design choice was that the user should be able to interact with animations of the models *when they are running*. This interaction can be done in all variables that are *independent* (such as parameters) or *integrated*. The user cannot interact with *dependent* variables defined as functions. This feature, as well as others, will be explained below in more detail.

3.4.3 The structure of the program

After running *Modellus*, the screen looks like the following:

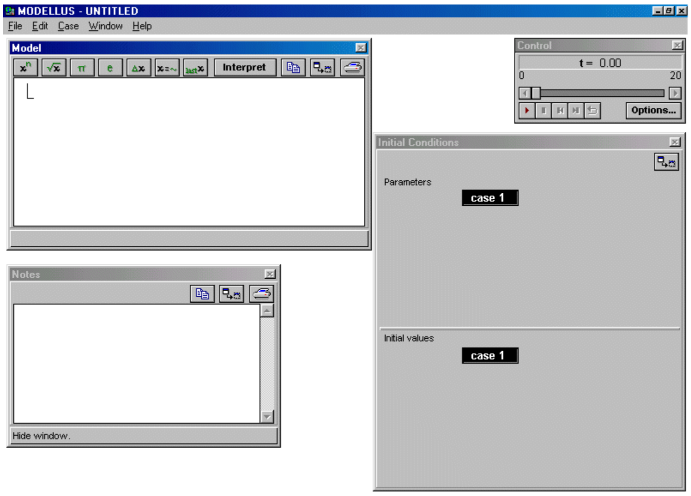


Figure 3.19 Modellus appearance after run.

Modellus has a main window and the following subordinate windows:

1. a *Model Window*, to write and edit mathematical models or comments;
2. a *Control Window*, to start, pause, and stop the model and also to control general specifications of the model, such as the letter and the domain of the independent variable;
3. a *Notes Window*, to write notes and comments;
4. one to three *Graph Windows*, available through the Window menu;

5. one to three *Table Windows*, also available through the Window menu;
6. and one to three *Animation Windows*, available through the same Window menu.

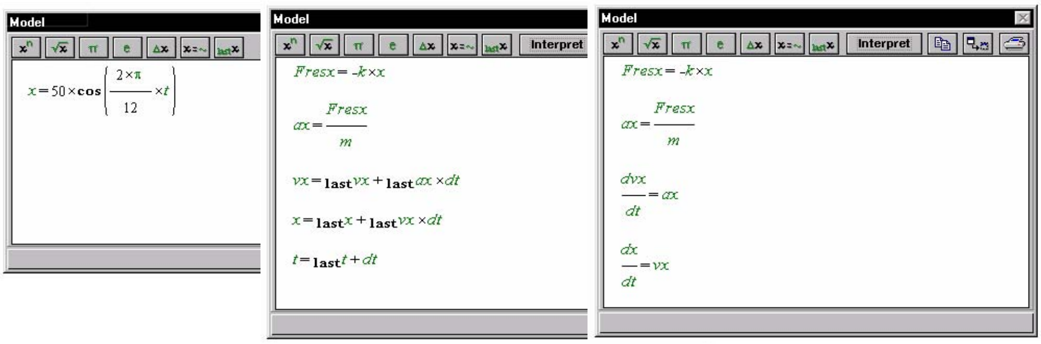
Modellus interface follows a standard Windows interface, according to *Windows Interface Guidelines* (Microsoft, 1995), with minor exceptions. These exceptions are:

1. it is not possible to close or minimize windows, except the main window, since each Modellus file can have multiple windows;
2. it is possible to *hide windows* (button:), without closing them; closing a window destroys its content; hiding a window preserves the content of the window;
3. it is not possible to print a Modellus file: the user can only print the content of each window; the best way to print Modellus data is to copy the contents of one or more windows to a word processor file—this allow the user to easily add comments and notes to Modellus data.

3.4.4 Creating and running a file

To create and execute a Modellus file the user must write one or more functions (or differential equations, or difference equations) in the *Model Window*, create an output window (a graph, a table or an animation) and run the model using the start button in the control window.

For example, to create a model of an oscillator, using a function, a system of differential equations, or a system if iterative equations, the user can write one of the following models in the *Model Window*:



$$x = 50 \times \mathbf{cos}\left(\frac{2\times\pi}{12}\times t\right)$$

$$Fresx = -k \times x$$
$$ax = \frac{Fresx}{m}$$
$$vx = \mathbf{last}\,vx + \mathbf{last}\,ax \times dt$$
$$x = \mathbf{last}\,x + \mathbf{last}\,vx \times dt$$
$$t = \mathbf{last}\,t + dt$$

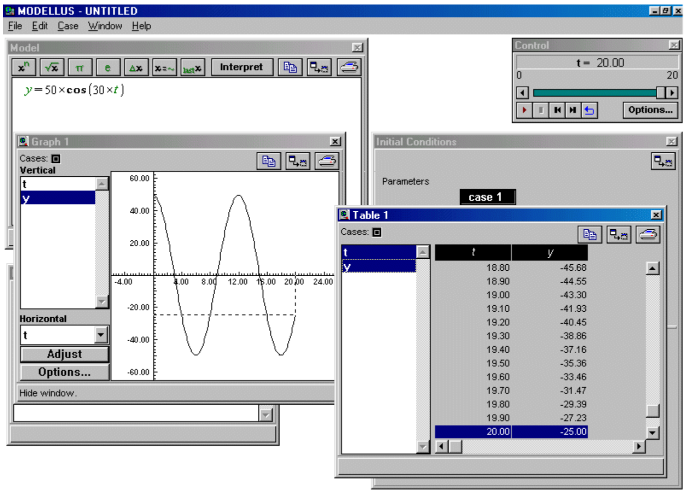
$$Fresx = -k \times x$$
$$ax = \frac{Fresx}{m}$$
$$\frac{dvx}{dt} = ax$$
$$\frac{dx}{dt} = vx$$

Figure 3.20 Three ways of writing a model of an oscillator on the *Model Window*.

Equations look like a normal written equation. The parser formats equations using a few conventions: variables are written in *green italic*, primitive functions in **black bold**, numbers and operation symbols in plain black. To get the symbols of the algebraic operations the user must press the usual keys (e.g., the * for multiplication—this operation can also be obtained with the space bar). Units are not represented in the model but the user must always be aware of them if they exist. Angles can be expressed in degrees and radians. By default, Modellus uses degrees, but this can be changed pressing the *Options*... button on the *Control Window*.

Once the model is written, it is necessary to press the *Interpret* button to activate the parser and check if the syntax is correct. This action is also automatically performed when the user presses the start button on the *Control Window*.

To see an output of the model the user must activate another window, such as a *Graph*, a *Table* or an *Animation*. *Graph* and *Table Windows* are very easy to create through the *Window* menu. It is possible to see one or more variables on the *Graph* or *Table* windows, clicking on the names of the variables with the Ctrl key down (a standard procedure for multiple selection on Windows):



MODELLUS - UNTITLED

$y = 50 \times \cos(30 \times t)$

t = 20.00

Table 1

t	y
18.80	-45.68
18.90	-44.55
19.00	-43.30
19.10	-41.93
19.20	-40.45
19.30	-38.86
19.40	-37.16
19.50	-35.36
19.60	-33.46
19.70	-31.47
19.80	-29.39
19.90	-27.23
20.00	-25.00

Figure 3.21 Two output windows: *Graph* and *Table*.

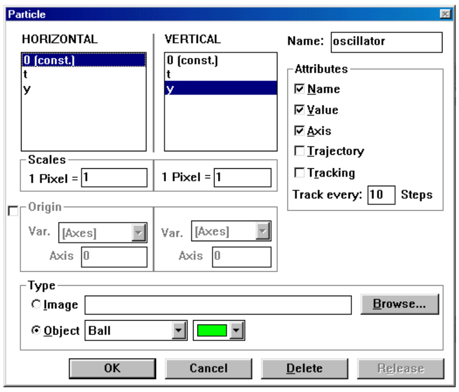
It is possible to resize any window, which is particularly useful for the *Graph Window*. It is also possible to *Adjust* automatically the scale on the *Graph Window* (button *Adjust*) and choose different viewing options. For example, the user can choose *Equal scales*, *Projection lines*, *Points* instead of line, etc. pressing the button *Options*…on the *Graph Window*. Other actions on this window include zoom in (click and drag) and change the origin (double click).

After making visible an *Animation Window*, using the *Window* menu, it is possible to create a visual representation of the model—in this example, an oscillating object. Any *Animation Window* has a set of objects that can have properties, such as position or size, accordingly with the available variables. For example, to create an oscillator, the user can choose a particle, clicking on the button ✛, and attribute the values of variable *y* to its vertical coordinate (Figure 3.22).

Once created the particle, its motion is animated when the model is running (Figure 3.23).

The *Animation Window* can have many types of objects. For example, in Figure 3.24, the *Animation Window* has a particle, a vector, a graph, and a block of text and digital displays of position and time.

Objects (position of the particles, length of the vectors, limits of the analog and protractors displays, graphs, etc.) on the *Animation Window* behave in accordance with a scale. By default, the scale is 1 pixel to 1 unit. To make a useful animation it is necessary,



Particle

HORIZONTAL: 0 (const.) / t / y — selected: 0 (const.)

VERTICAL: 0 (const.) / t / y — selected: y

Name: oscillator

Attributes

☑ Name

☑ Value

☑ Axis

☐ Trajectory

☐ Tracking

Track every: 10 Steps

Scales: 1 Pixel = 1 | 1 Pixel = 1

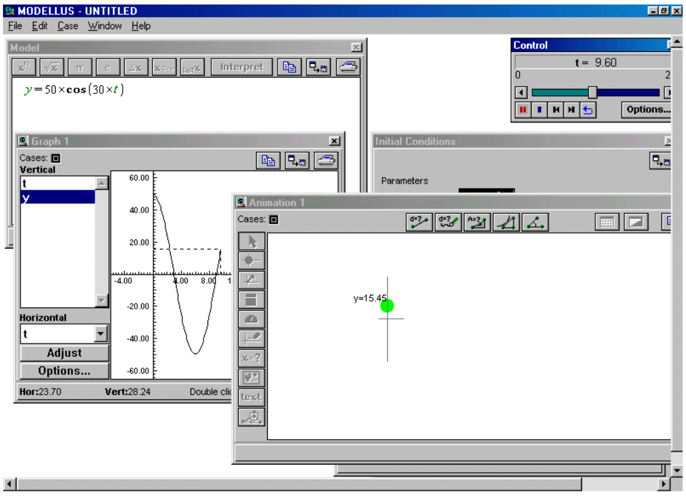
☐ Origin: Var. [Axes], Axis 0 | Var. [Axes], Axis 0

Type

○ Image

● Object Ball

Figure 3.22 Properties of the oscillating particle: the vertical coordinate is *y* and the horizontal coordinate is 0.



$y = 50 \times \cos(30 \times t)$

Figure 3.23 With the model running, the motion of the particle is animated. The graph is generated simultaneously with the motion.

frequently, to change the scale of one or more objects. For example, on the above animation, the scale of time *t*, on the graph, is 1 pixel to 0.1 units. If the default scale were unchanged, the graph would look like the one in Figure 3.25.

It is possible to write symbols to represent parameters in the model. For example, a model of an oscillator can look like the one on Figure 3.26, where the amplitude and the angular frequency are represented by *A* and *w*, respectively.

A and *w* are parameters of the model. The values for the parameters must be introduced in the *Initial Conditions Window*, after pressing the *Interpret* button on the *Model Window*. A set of values of the parameters is a *Case*. Using the *Case Menu*, the

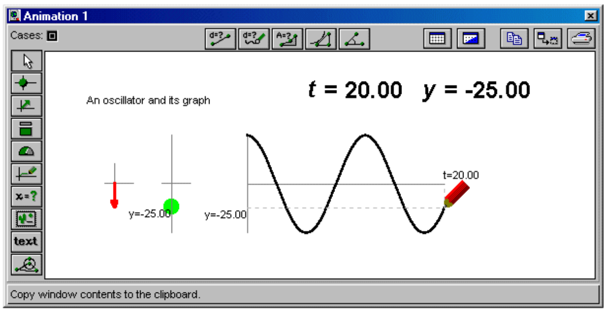


Figure 3.24 An *Animation Window* with different types of objects: text, vector, particle, graph, digital displays.

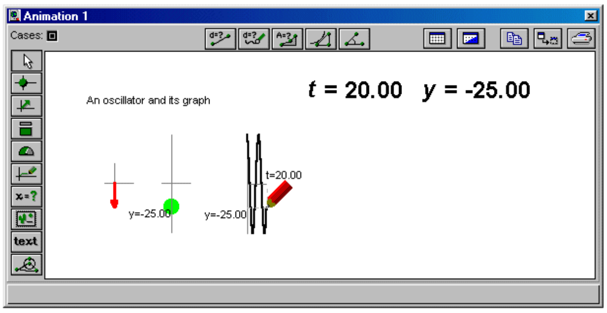
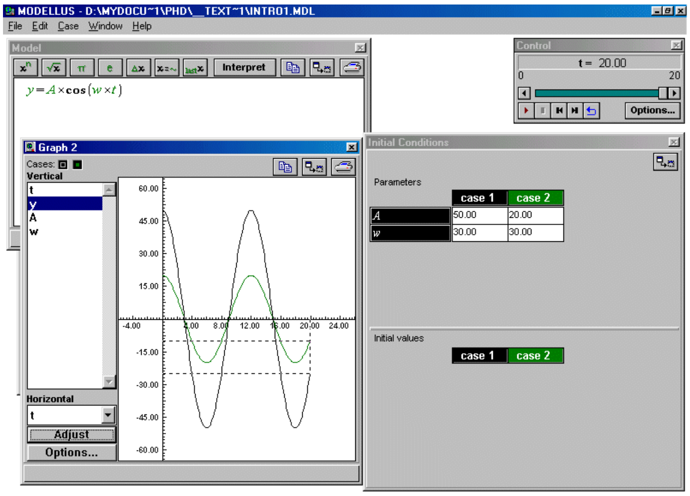


Figure 3.25 An inadequate scale on the horizontal axis of the graph.

user can create a new *Case*. This is particularly useful to analyse the effect of changing the values of the parameters, as shown on Figure 3.26 above.

Modellus has a “motor” to compute symbolic derivatives of functions that explicitly depend on the independent variable or are derivatives themselves. Figure 3.27 shows how to compute the first and second derivative of a function.

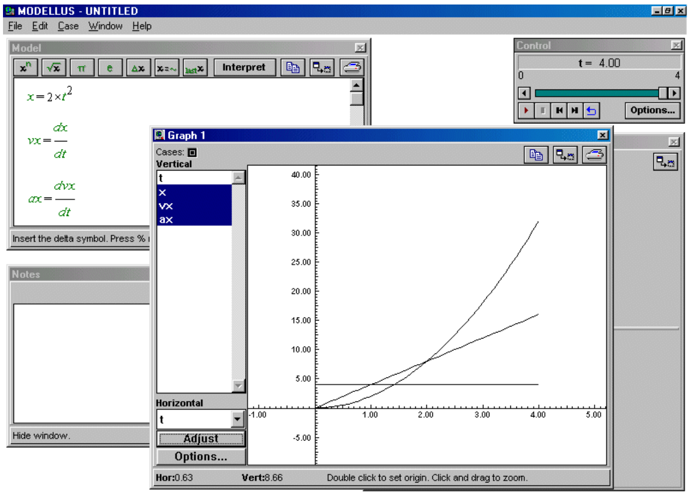
As usual, if the function is a trigonometric function, its argument must be in radians (to use radians, as mentioned above, the user must select this option using the button *Options*... on the *Control Window*).



$y = A \times \cos(w \times t)$

Parameters	case 1	case 2
A	50.00	20.00
w	30.00	30.00

Figure 3.26 A model of an oscillator with two parameters, *A* and *w*. The values for the parameters are attributed on the *Initial Conditions Window*. In this example, a second case was created, using the *Case Menu*, to analyse the effect of changing *A* from 50 units to 20 units. Both case are seen on the *Graph*, after pressing the small coloured button on the top-left of the *Graph Window*.



$x = 2 \times t^2$

$vx = \frac{dx}{dt}$

$ax = \frac{dvx}{dt}$

Insert the delta symbol. Press %

Figure 3.27 Using Modellus to compute derivatives symbolically.

Modellus can use any step and any single letter for the independent variable. To change the default letter *t* and the default step 0.1 (and, if necessary, the lower and upper limits of the independent variable) it is necessary to press the *Options*... button on the *Control Window*. For example, it is possible to analyse the succession with limit $e = 2.718\ldots$ with a model like the one of Figure 3.28.

Modellus can also solve ordinary differential equations (ODEs), or systems of ordinary differential equations. A system of *n* differential equations,

$$\frac{\mathrm{d}y_1}{\mathrm{d}t} = f_1(t, y_1, y_2, \ldots, y_n)$$

$$\frac{\mathrm{d}y_2}{\mathrm{d}t} = f_2(t, y_1, y_2, \ldots, y_n)$$

$$\ldots$$

$$\frac{\mathrm{d}y_n}{\mathrm{d}t} = f_n(t, y_1, y_2, \ldots, y_n)$$

is solved iteratively, using Runge-Kutta fourth-order method with fixed step *dt*,

$$y_{n,\,t+\mathrm{d}t} = y_{n,\,t} + \frac{1}{6}\mathrm{d}t(k_{1,y_n} + 2k_{2,y_n} + 2k_{3,y_n} + k_{4,y_n})$$

where

$$k_{1,y_n} = f_1(t_t, y_{1,\,t}, y_{2,\,t}, \ldots, y_{n,\,t})$$

$$k_{2,y_n} = f_1\left(t_t + \frac{1}{2}\mathrm{d}t, y_{1,\,t} + \frac{1}{2}\mathrm{d}t \cdot k_{1,y_1}, y_{2_t} + \frac{1}{2}\mathrm{d}t \cdot k_{1,y_2}, \ldots, y_{n,\,t} + \frac{1}{2}\mathrm{d}t \cdot k_{1,y_n}\right)$$

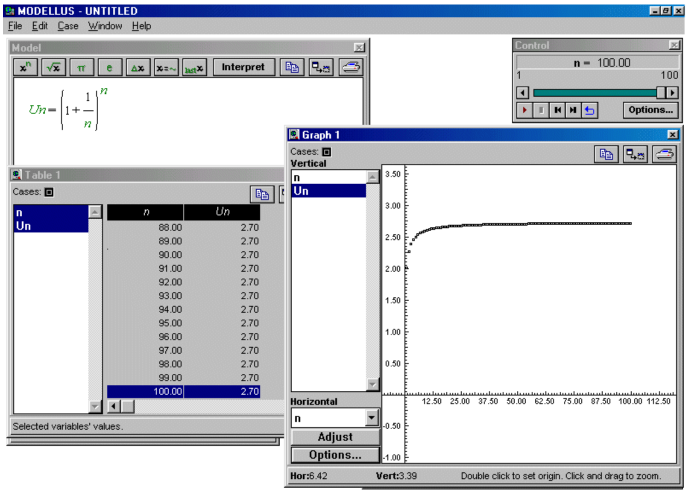
$$k_{3,y_n} = f_1\left(t_t + \frac{1}{2}\mathrm{d}t, y_{1,\,t} + \frac{1}{2}\mathrm{d}t \cdot k_{2,y_1}, y_{2,\,t} + \frac{1}{2}\mathrm{d}t \cdot k_{2,y_2}, \ldots, y_{n,\,t} + \frac{1}{2}\mathrm{d}t \cdot k_{2,y_n}\right)$$

$$k_{4,y_n} = f_1(t_t + \mathrm{d}t, y_{1,\,t} + \mathrm{d}t \cdot k_{3,y_1}, y_{2,\,t} + \mathrm{d}t \cdot k_{3,y_2}, \ldots, y_{n,\,t} + \mathrm{d}t \cdot k_{3,y_n})$$

With this method it is possible to solve the most common ODEs or systems of ODEs. For models with rapid changes in dependent variables, it can be necessary to choose smaller steps. For example, the Figure 3.29 shows a model of an oscillator where the step is too big. Using a smaller step (0.01) the integration is done using more points and the graph becomes more smooth and correct (Figure 3.30).

An important feature of Modellus is that all integrated variables can be changed when the model is running. Consider, for example, a particle with mass *m* where a constant horizontal force is applied (component F_x). A Modellus model for this phenomenon can be the one on Figure 3.31.

The model starts with a simple computation to compute the horizontal component of the acceleration, a_x, from the horizontal component of the applied force, F_x, and from the mass of the particle, *m*. These two quantities are independent variables—it is necessary



$$Un=\left(1+\frac{1}{n}\right)^n$$

n = 100.00 (1 to 100)

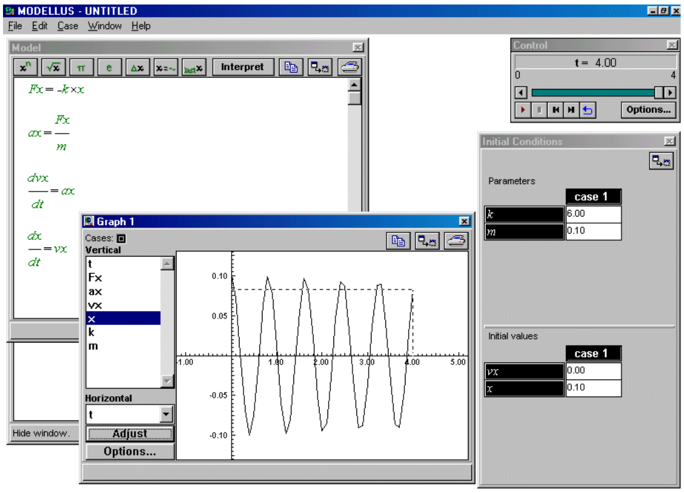
n	Un
88.00	2.70
89.00	2.70
90.00	2.70
91.00	2.70
92.00	2.70
93.00	2.70
94.00	2.70
95.00	2.70
96.00	2.70
97.00	2.70
98.00	2.70
99.00	2.70
100.00	2.70

Selected variables' values.

Vertical: Un; Horizontal: n

Hor:6.42 Vert:3.39 Double click to set origin. Click and drag to zoom.

Figure 3.28 Analysing a succession. On the graph, points are used instead of lines, since *n* is an integer variable (step 1). The *Control Window* shows that the lower limit for *n* is 1 and the upper limit is 100.



$$Fx=-k\times x$$

$$ax=\frac{Fx}{m}$$

$$\frac{dvx}{dt}=ax$$

$$\frac{dx}{dt}=vx$$

t = 4.00 (0 to 4)

Parameters

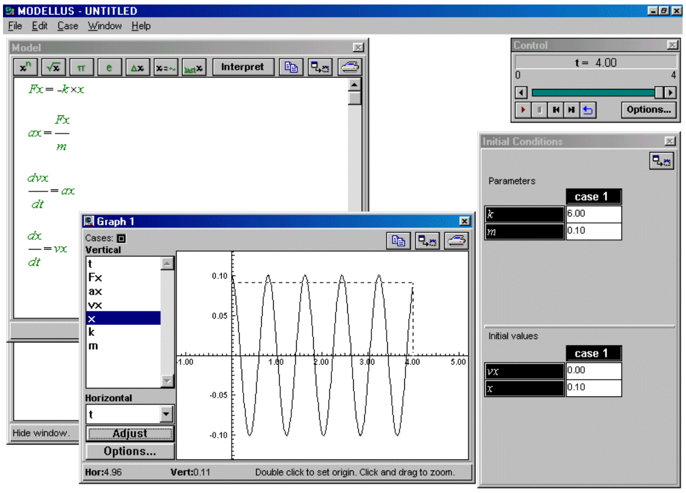
	case 1
k	6.00
m	0.10

Initial values

	case 1
vx	0.00
x	0.10

Vertical: x; Horizontal: t

Figure 3.29 A model of an oscillator with differential equations. The step, 0.1, is too big for these parameters: the graph is not smooth, since there are not enough points computed for the integrated variables (x and v_x). The step can be changed to a smaller value using the *Options*... button on the *Control Window*.



$$Fx=-k\times x$$

$$ax=\frac{Fx}{m}$$

$$\frac{dvx}{dt}=ax$$

$$\frac{dx}{dt}=vx$$

t = 4.00 (0 to 4)

Parameters

	case 1
k	6.00
m	0.10

Initial values

	case 1
vx	0.00
x	0.10

Vertical: x; Horizontal: t

Hor:4.96 Vert:0.11 Double click to set origin. Click and drag to zoom.

Figure 3.30 The model of the oscillator with differential equations with a convenient step, 0.01. The graph is smoother than the one obtained with step 0.1.

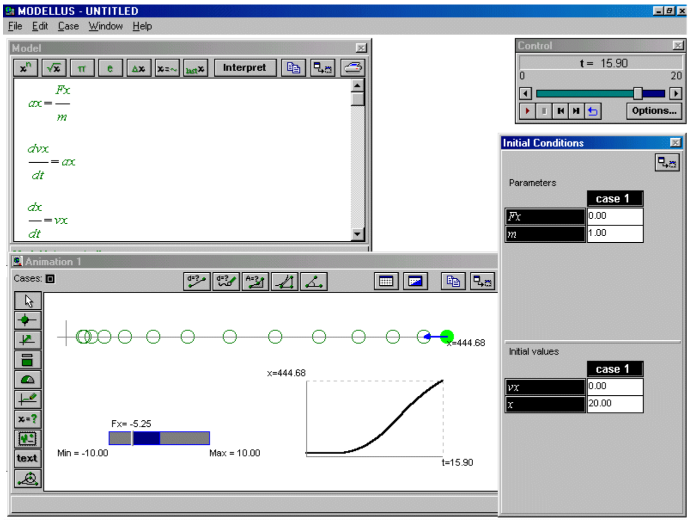
to give initial values for them in the *Initial Conditions Window*. Knowing a_x, *Modellus* can compute v_x by integration; again, knowing v_x, Modellus computes x, also by integration. These integrations are computed systematically at each time step. If F_x

changes, all variables depending on F_x, either explicitly, like a_x, or implicitly, like the integrated variables v_x and x, also change at each time step.

Besides functions and differential equations, it is also possible to write another kind of model: models with difference equations or iterations. For example, Figure 3.32 shows a simple model of decay of variable N using a difference equation.

Iterative models are very useful particularly to study basic or advanced numerical methods, as I shall exemplify in the next chapter.

Despite more than a decade of criticism of exploratory environments based on programming metaphors, some authors (e.g. Eisenberg, 1995) still argue about the importance of programming in these kind of environments. Programming is an evolving concept and it is now difficult to say if constructing a model with Modellus is comparable with creating a program. Building a model with Modellus involves certain characteristics of programming (e.g., there are syntax rules for building models) but, on the other side, many aspects of programming are "hidden" from the user (e.g., it is not necessary to create loops to assign different values for expressions). Modellus has also



$ax = \frac{Fx}{m}$

$\frac{dvx}{dt} = ax$

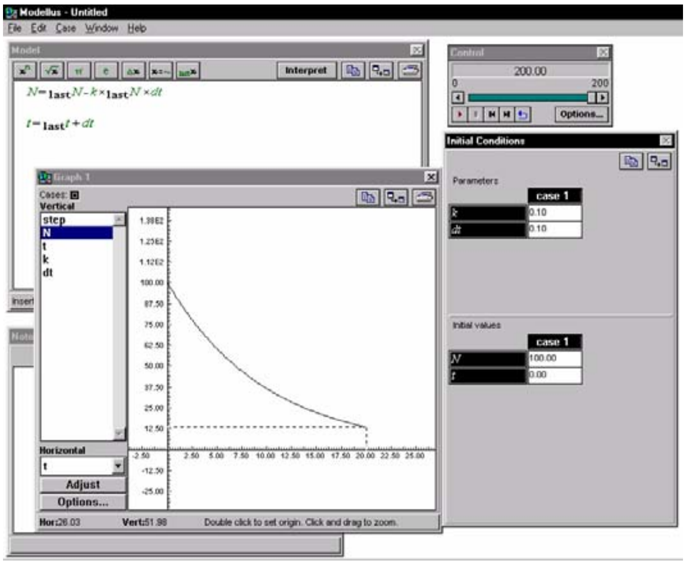
$\frac{dx}{dt} = vx$

t = 15.90

Parameters, case 1: Fx 0.00; m 1.00

Initial values, case 1: vx 0.00; x 20.00

Figure 3.31 A Modellus model of a particle moving on a straight line. The force on the particle, with an initial value of 0 N, can be controlled using the horizontal bar. In this example, after an initial interval of zero force, a force was applied to the right for a while (the particle increased its velocity), and then to the left (the particle decreased its velocity). Changing the value on the bar changes F_x. Since v_x and x are computed by integration from F_x and a_x, its values are adjusted interactively.



$N = {}_{\text{last}}N - k \times {}_{\text{last}}N \times dt$

$t = {}_{\text{last}}t + dt$

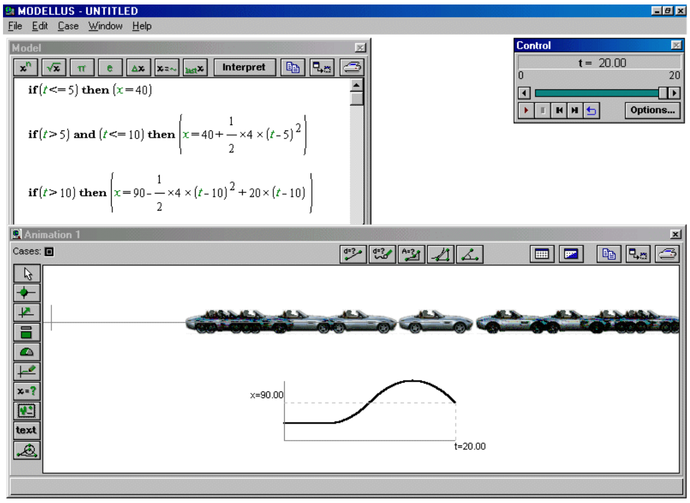
200.00

Parameters, case 1: k 0.10; dt 0.10

Initial values, case 1: N 100.00; t 0.00

Hor:26.03 Vert:51.98 Double click to set origin. Click and drag to zoom.

Figure 3.32 An iterative model of decay of N (no independent variable on the *Control Window* and check button *Iterative Model* activated on the *Options…* dialogue box).



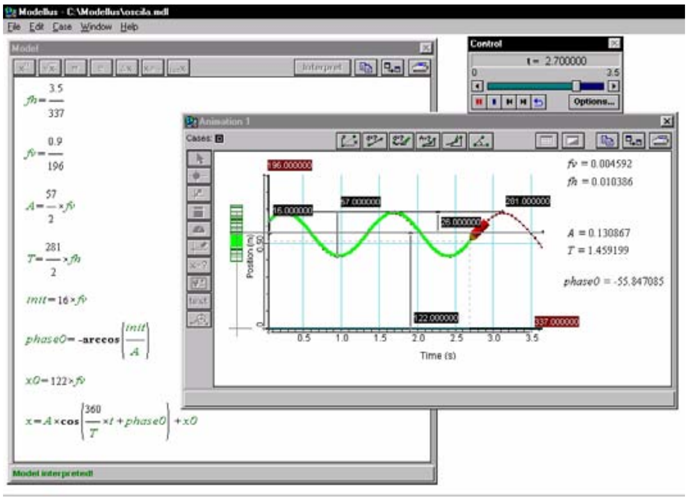
MODELLUS - UNTITLED

$$\text{if}(t<=5)\ \text{then}\ (x=40)$$

$$\text{if}(t>5)\ \text{and}\ (t<=10)\ \text{then}\ \left(x=40+\frac{1}{2}\times4\times(t-5)^2\right)$$

$$\text{if}(t>10)\ \text{then}\ \left(x=90-\frac{1}{2}\times4\times(t-10)^2+20\times(t-10)\right)$$

Figure 3.33 Using conditions to define different functions for different domains of *t*.



Modellus - C:\Modellus\oscila.mdl

$$fh=\frac{3.5}{337}$$

$$fv=\frac{0.9}{196}$$

$$A=\frac{57}{2}\times fv$$

$$T=\frac{281}{2}\times fh$$

$$init=16\times fv$$

$$phase0=-\text{arccos}\left(\frac{init}{A}\right)$$

$$x0=122\times fv$$

$$x=A\times\cos\left(\frac{360}{T}\times t+phase0\right)+x0$$

Model interpreted!

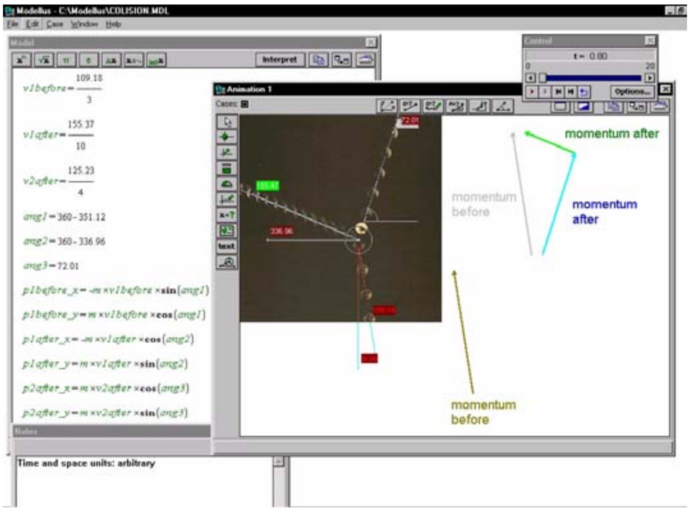
fv = 0.004592
fh = 0.010386
A = 0.130867
T = 1.459199
$phase0$ = -55.847085

Figure 3.34 A Modellus model made from experimental data—in this case, a graph of the coordinate of an oscillator obtained with data logging software. After obtained the model, this is compared with the data. The model was also used to make an animation of an oscillator—the small green square of the left of the graph.

some typical features of programming, such as *conditions*, particularly useful to define functions with different analytic expression for different parts of its domain. For example, the model in Figure 3.33 uses conditions to change the dynamic state of the car.

With Modellus it is also possible analyse experimental data, either from photos or videos, or from any type of image such as screens from data logging software. For example, the user can take measures from a graph of experimental data, using Modellus measuring tools (located at the top of the *Animation Window*). To make the model, the experimental data is placed as an image on the background of the *Animation Window.* The first step is, usually, to establish scales for the image (in the form of factor scales). Then, it is necessary to make convenient measures of specific parameters (e.g., inflexion points, period, amplitude, etc.). Finally, use the parameters to write the model and compare the model with the experimental data (taking factor scales into consideration)—see an example on Figure 3.34. A similar process can be used with photos (Figure 3.35) and videos (Figure 3.36).

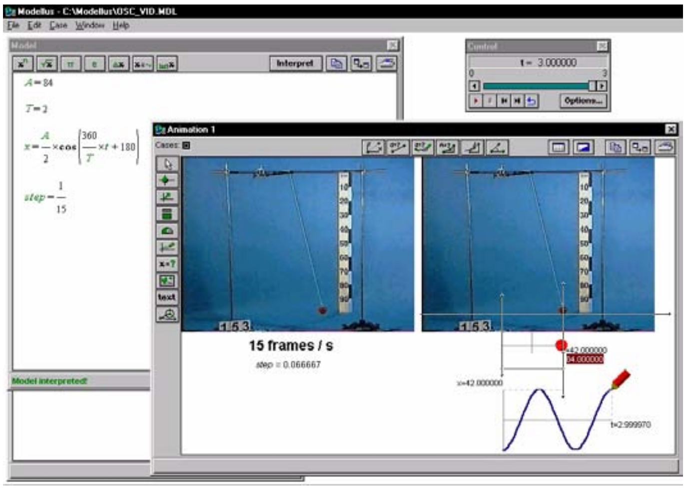
There are many other features of Modellus not described here but documented in the Modellus User's Manual (Knowledge Revolution, 1997; Teodoro et al., 2000).



$$v1before = \frac{109.18}{3}$$
$$v1after = \frac{155.37}{10}$$
$$v2after = \frac{125.23}{4}$$
$$ang1 = 360 - 351.12$$
$$ang2 = 360 - 336.96$$
$$ang3 = 72.01$$
$$p1before\_x = -m \times v1before \times \sin(ang1)$$
$$p1before\_y = m \times v1before \times \cos(ang1)$$
$$p1after\_x = -m \times v1after \times \cos(ang2)$$
$$p1after\_y = m \times v1after \times \sin(ang2)$$
$$p2after\_x = m \times v2after \times \cos(ang3)$$
$$p2after\_y = m \times v2after \times \sin(ang3)$$

Time and space units: arbitrary

Figure 3.35 Using Modellus to analyse a strobe photo of a collision. The measuring tools (for angles and distances) helped to find momentum before and after the collision. Momentum of each object is then represented as vector to check if there is conservation of momentum.



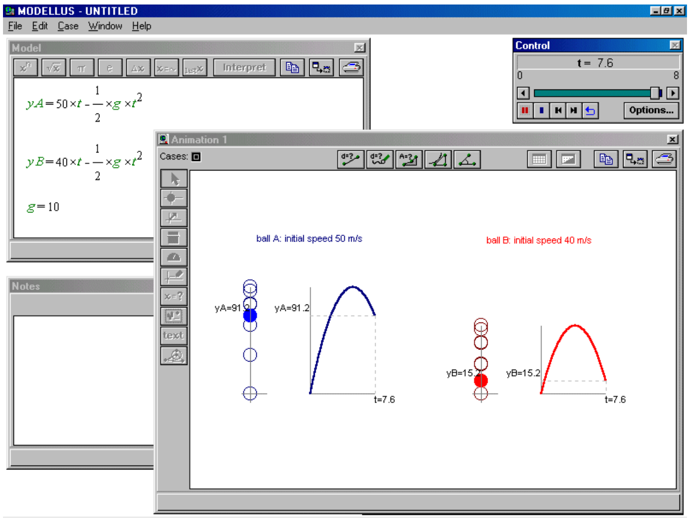
$$A = 84$$
$$T = 2$$
$$x = \frac{A}{2} \times \cos\left(\frac{360}{T} \times t + 180\right)$$
$$step = \frac{1}{15}$$

Figure 3.36 A model made from a video. After finding a time scale (15 frames per second), using a small step and counting images, it was necessary to measure the period and the amplitude of the pendulum. The model can easily be compared with the pendulum. The window shows two videos simultaneously. The left is the "original" video and the one on the right is a "copy" where the user can take measurements and overlap objects.

3.4.5 Modellus: a pedagogical description

One of the most important features of Modellus is the possibility of exploring multiple representations of abstract objects. The concept of multiple representations has been a recurrent concept in exploratory software design for science and mathematics, at least since the publication of *Making sense of the future* (Harvard Educational Technology Center, 1988). In this position paper, the authors argue about how computers can make a difference in learning environments. Besides other points, they stress the fact that computers can easily present simultaneously representations of the same object, such as a function (the analytical expression, a table of values, and a graph). Multiple representations, emphasizing different aspects of the same idea and affording different sort of analyses, are now a "taken for granted" issue in most educational software for science and mathematics.

Modellus broadens the concept of multiple representations introducing the capability of creating visual representations of phenomena with a *lesser degree of formalism* than equations, tables, or graphs. For example, a traditional program for analysing functions,

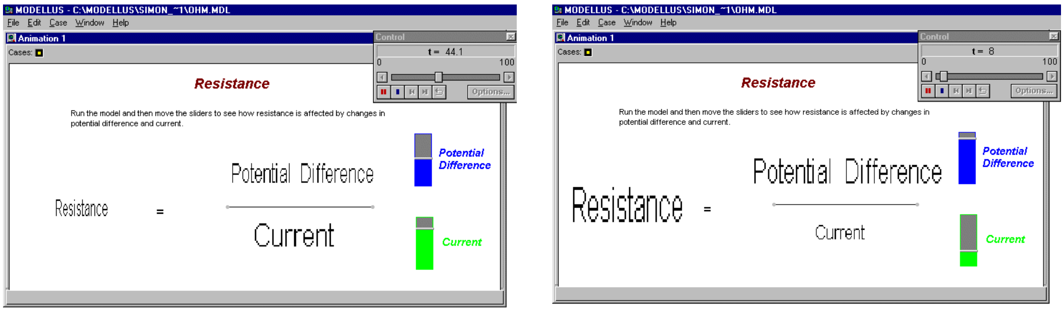


$yA = 50\times t - \frac{1}{2}\times g\times t^2$

$yB = 40\times t - \frac{1}{2}\times g\times t^2$

$g = 10$

Figure 3.37 Modellus broadens the concept of multiple representations since it allows users to create visual animations of phenomena described by functions. In this example, it is possible to see animations of balls launched with different initial speeds.



Run the model and then move the sliders to see how resistance is affected by changes in potential difference and current.

Resistance = Potential Difference / Current

Figure 3.38 A visual explanation how change in the denominator and, or, on the numerator, affects the quotient (example made by Simon Carson, UK). Bars can be moved to change the “size” of the numerator and, or, the denominator, and seeing what happens to the “size” of the quotient.

such as quadratic functions, permits the user to compare expressions, tables and graphs. But Modellus also allows the user to create representations of phenomena where quadratic equations are used, such as launching a ball into the air. It also allows the user to explore and compare multiple contexts of the same phenomena, as can be seen in the example of Figure 3.37.

Multiple representations in Modellus are not limited to moving objects, graphs, equations, tables, etc. It is also possible to create visual interactive representations of mathematical relationships, such as the one shown on Figure 3.38.

From an educational point of view, Modellus incorporates both *expressive* and *exploratory* modes of learning activities (Bliss & Ogborn, 1989). In an expressive learning activity, students can build their own models and create ways of representing them. In an exploratory mode, students can use models and representations made by others, analysing how different things relate to one another.

Teachers and curriculum developers can take advantage of the educational design of Modellus, since the software can be used as an authoring language for creating visual representations—see, e.g., Lawrence & Whitehouse (2000; 2001) or Fiolhais et al.

(1996a) each with hundreds of examples of visual illustrations of physics concepts done with Modellus.

A teacher or curriculum developer can specify what is presented to the student in a certain learning situation. Since Modellus windows can be hidden and files can be protected with passwords, a Modellus example can show only what is appropriate for the student's knowledge level. For example, Figure 3.39 shows a file where high school physics students can "play" with satellites without any knowledge of the differential equations that were used to create the model (the *Model* window is hidden and the file is protected by a password). When a file is protected by a password, buttons on the *Animation* window are limited to those strictly necessary, such as measuring tools.

The pedagogical design of Modellus assumes that the computer is a cognitive tool, but not a replacement of the higher order human skills. Contrary to the design options of other educational systems, such as those usually adopted in intelligent tutoring systems, it is assumed that Modellus is a tool to "impart wheels to the mind" but the intelligence, emotion, culture, poetry, and art, reside in the user, not on the software. Modellus does not maintain any conversation with the user, or try to make any inference about the user's skills and purposes. It simply responds to user's actions. When using Modellus, it is assumed that conversations are activities made between students or between teachers and students, not between persons and software. Even if it was possible to create a system that can mantain a meaningful conversation, I agree with Schwartz (1995, p. 174) when he argues that "no student is ever going to believe that the computer that poses a question *cares* about his or her answer".

Learning with exploratory environments like Modellus can never take place spontaneously: *regulation* and *control* are fundamental. Written materials, with guided inquiry approaches, that students read, discuss with peers, confront conceptions and descriptions, and where students must also draw sketches and write (the process of writing is a «disaccelerator» of information, specially visual information, and can act as

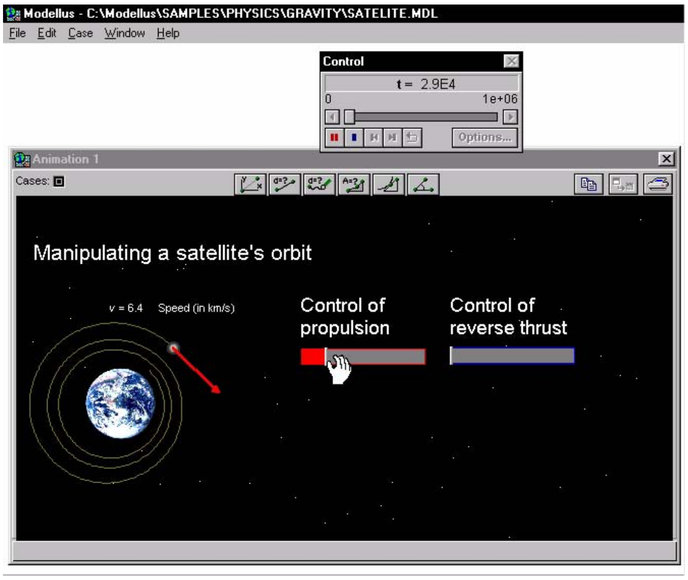


Figure 3.39 A Modellus file protected by a password. This example allow students to explore satellite motion without any knowledge of the differential equations used to create the model.

an accelerator of knowledge construction) are essential approaches when designing learning environments with Modellus.

Modellus can be used to help students understand scientific concepts in the sense given by Perkins (1993, p. 5), ie, “performance understanding”. A “performance perspective says that understanding a topic of study is a matter of being able to perform in a variety of thought-demanding ways with the topic, for instance to: explain, muster evidence, find examples, generalize, apply concepts, analogize, represent in a new way, and so on”. Understanding scientific concepts with Modellus must be *doing things* with them, not just repeating textbook explanations or writing equations out of any context. Performance-based understanding with Modellus is simoultaneously “hands-on” and “minds-on”: “hands-on” in the sense that students can use concrete objects, like images, to make meaning of concepts. And “minds-on” in the sense that this use involves abstractions ans its many representations. Interaction with Modellus is engagement-based interaction—students must create and manipulated abstract objetcs. And, as Laurillard, says, “engagement leads to reflection; reflection leads to understanding” (1995, p. 180).

3.5 A Model to Develop Exploratory Software for Learning Science (and Mathematics)

Exploratory computer tools are tools to overcome the boundary between lower and higher cognitive stages (Papert, 1980). This can be done because computers allow users to approach mathematical and physical *abstract* objects in a *concrete way.*

Early exploratory environments were based on computational metaphors, like programming. But programming, even in modern high level languages or in general mathematical packages, makes use of primitive statements (such as variable definitions, selections and repetitions) that are too “primitive” to allow meaningful exploration of most scientific ideas. With a programming language, it is possible to explore scientific ideas but the programming language itself behaves as a mediator that does not have the properties of the scientific ideas that we want to explore. For example, if we want to explore how the velocity of an object changes with time under certain circumstances, we must have direct access to a representation of velocity, such as a vector. With programming languages that can only be done with a reasonable programming effort, since programming languages are not domain specific and only have general primitives and procedures.

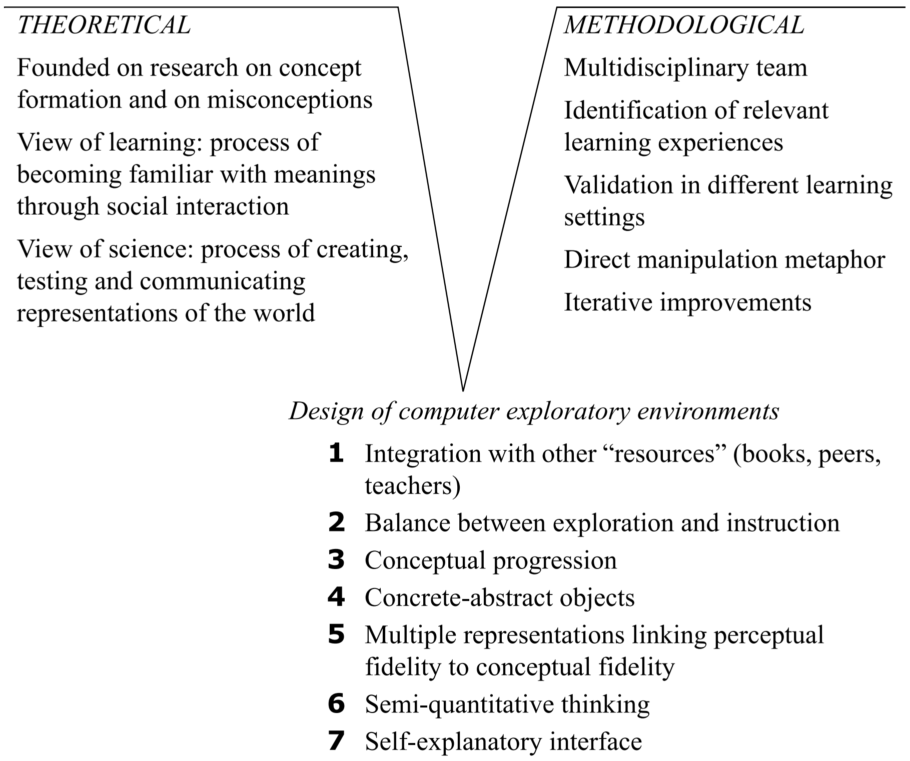
Exploratory environments, unlike programming languages, are *domain specific*. What a user can do with an exploratory environment depends on the domain. On the one hand, this gives very powerful primitive actions, such as showing a velocity vector just by clicking the mouse; on the other hand, however, it narrows the range of the capabilities of the software. But this is usually not a real problem because of the domain specificity of each exploratory environment.

Learning most of the scientific and mathematical ideas at secondary school can be seen as a process of *becoming familiar* with the ideas. A good understanding of an idea is most of the times a strong degree of familiarization with the idea. As Schank (1986, p. 5) pointed out, understanding "is not an all-or-none affair. People achieve degrees of understanding in different situations depending upon their level of familiarity with those situations." Familiarity with ideas is considered by many scientists, as mentioned in chapter 1, as the key process in understanding. Exploratory software must allow students to get a strong degree of *familiarization* with the basic ideas of the domain being explored. With exploratory software, students can see many situations, explore what happens in different conditions, discuss what happens if they change conditions, etc.; ie, they *can become more and more familiar* with the ideas, the consequences of the ideas and the representations of the world. When they become more familiar with new ideas and new representations, they can establish more meaningful relations with ideas they already have. Exploratory software can be a major way to foster familiarity with new ideas.

Figure 3.40 shows the structure of a model for designing exploratory software such as Modellus[1]. It has two lines of approach, one of which is *methodological* and the other *theoretical*.

Along the *theoretical* line, the model considers three issues:

1 The design of exploratory software should be based on research on concept



THEORETICAL

Founded on research on concept formation and on misconceptions

View of learning: process of becoming familiar with meanings through social interaction

View of science: process of creating, testing and communicating representations of the world

METHODOLOGICAL

Multidisciplinary team

Identification of relevant learning experiences

Validation in different learning settings

Direct manipulation metaphor

Iterative improvements

Design of computer exploratory environments

1. Integration with other "resources" (books, peers, teachers)
2. Balance between exploration and instruction
3. Conceptual progression
4. Concrete-abstract objects
5. Multiple representations linking perceptual fidelity to conceptual fidelity
6. Semi-quantitative thinking
7. Self-explanatory interface

Figure 3.40 A model to guide the design of computer learning exploratory environments for science (and mathematics).

---

[1] A previous version of this model has been published in 1993 (Teodoro, 1993a).

formation and on misconceptions in order to identify relevant learning experiences and sources of difficulties in concept formation. This assures a didactical basis for the software.

2. A specific view of learning is assumed. According to this view, "complete understanding, just like complete knowledge, is unlikely to be achieved (...) even in what seems the most elementary part of physical science" (Popper, 1977). *We understand when we are familiar with ideas and representations shared by members of a community.* Understanding scientific ideas is, then, a process of *enculturation*. This process is facilitated when learning occurs in the "zone of proximal development", as Vygotsky (1978, p. 86) defines it ("the distance between the actual developmental level as determined by independent problem solving and the level of potential development as determined through problem solving under adult guidance or in collaboration with more capable peers"). This assures a cognitive basis for the design of the learning experiences based on the software.
3. Science is assumed as a "process for producing knowledge [that] depends both on making careful observations of phenomena and on inventing theories for making sense out of those observations. (...) In science, the testing and improving and occasional discarding of theories, whether new or old, go on all the time." (American Association for the Advancement of Science, 1989, p. 9). In short, science is about *creating*, *testing* and *communicating representations* about the world, like theories and models, specially mathematical models. This assures that the learning experiences based on the software must reinforce the *production of knowledge from observations* and the *testing and improvement of models and theories* instead of the simple presentation of knowledge.

Along the *methodological* line, the model considers five issues.

1. The development of exploratory software is a team project, involving different specialists: at least, software designers, programmers, experienced teachers.
2. Exploratory software should be designed after the identification of the most relevant learning experiences in a certain domain. A relevant experience is related to the process of concept formation, either because it gives "anchors" to subsume concepts or because it shows a conflictual view with naïve thinking.
3. As learning takes place in many different settings, the software should be validated in the different settings where learning occurs: it cannot be designed only for classrooms. With the increasing diffusion of computers, at home, in resource centres, in libraries, etc., students can have experiences with exploratory software in many different places outside classrooms and outside teacher supervision.
4. Exploratory software should be based on graphical and direct manipulation interfaces, where the user controls his actions directly, not mediated by written commands.
5. Like all software, the design of exploratory environments is an iterative process with successive improvements based on user observation or usability studies and

feedback from students, teachers, and curriculum developpers.

As “output” of the model, seven relevant issues are considered:

1. Exploratory software by itself has very limited use. Exploratory software should be considered as a part of “learning packages” to foster “learning communities”. It is neither possible nor desirable to build exploratory software that is independent from other learning materials, such as written materials, in printed form or digital formats. Meaningful use of exploratory software involves the *exploration of what students already know*, not what they don’t know. Written materials, with a compelling design, are essential to present lines of argument, guide instruction, and guide personal and group work. Exploratory software should serve as a complement to books, allowing students to explore what they read and discuss, giving them the capabilities that no book has—this is the approach Advancing Physics (Ogborn & Whitehouse, 2000, 2001) adopted for the use of Modellus.
   As any other educational material, exploratory software is a resource for learners. Programs are like artist’s tools: tools can help artists but they don’t produce art. Only artists do. But exploratory software has a unique characteristic: when well designed, it can foster interactions between learners, in particular if students work in pairs or in small groups (Hoyles, Healy & Pozzi, 1994). Exploratory software can then help the formation of communities of learners that can explore, test and communicate ideas of science. Exploratory software must be “object and site for conversation” (Roth, 1996, 185).
2. Balancing exploratory learning and direct instruction is a fundamental issue in the design of “learning packages” and in the creation of good learning environments. Research shows that exploratory learning is difficult (a good synthesis of research can be found on de Jong 1998). Teachers should always bear in mind that *learners cannot explore what they don’t know already*! This statement can be seen as a corollary of Ausubel’s famous principle: “The most important single factor influencing leaming is what the leamer already knows. Ascertain this and teach him accordingly” (Ausubel, Novak & Hanesian, 1978, p. iv).
   The balance between exploratory learning and direct instruction must be managed by the curriculum developer and by the teacher. As all teachers know, novice learners tend to be distracted by surface features of specific representations. Exploratory software can increase distraction because surface features of a domain are usually more accessible.
3. Exploratory software must support conceptual progression, based on the nature of the subject and on didactical research. Users must be able to do “simple” explorations with the software or more and more complex ones, without major difficulties, as their knowledge increases. For example, a Modellus user can easily explore constant rate of change models with a simple linear function or explore the same model as an iterative process or even as a differential equation, as shown in Chapter 4.
4. The typical objects presented to the user in a computer exploratory environment for learning science are *concrete-abstract objects* (Hebenstreit, 1987). Users can

directly manipulate vectors, particles, forces, graphs, equations, geometrical figures, etc. Accordingly to Hebenstreit, *they are concrete only on the computer*. In reality, they *are abstract* since they are mental constructs, mental representations of certain properties of real or imagined objects. Exploratory environments must allow users to explore how these objects "work", making actions and the consequences of actions transparent for the user (Schwartz, 1997).

5. Multiple representations are one of the most important features of exploratory software. This feature gives students the possibility to interact with different coordinated representations of a phenomenon, such as equations, graphs, animations, videos, etc. Regardless of the importance of the use of multiple representations, they shall be used parsimoniously, since their abuse can easily lead to information overload in learners.
   Multiple representations are essential to help students progress from *perceptual fidelity* (what they observe) to *conceptual fidelity* (how science describes and represents the phenomenon). Studies of what distinguishes experts in a domain from novices (e.g., Chi, Feltovich, & Glaser, 1981) shows that experts have multiple ways of thinking about the domain, while novices have only one or a few ways. Experts switch from one representation to another and have the metaknowledge that allows them to know which representation to choose for which task and which representation to switch to while solving the task.
6. Another important issue about the design and use of exploratory software is the relation between *semi-quantitative* or *qualitative* representation and *quantitative* representation. It is known that experts tend to use more qualitative representations than novices (Chi, Feltovich, & Glaser, 1981). Exploratory software must allow students to focus reasoning on *qualitative* descriptions, not on algorithms and computations.
7. Finally, exploratory software must have a self-explanatory interface: the exploratory domain must be evident for the novice user and the software functionalitites must be easily accessible without formal training on the use of the software.

3.6 Coda

Design is a process of conceiving artefacts for human use. Software design is comparable to building design (Kapor, 1996). Both are team projects and in both the designer (software designer in one case and architect in the other) starts from a *concept* —or, as Liddle (1996) calls it, a *user's conceptual model*—, sometimes a *new concept*, that is based on what the user should do with the artefact and how he should interact with it. This process requires creativity, knowledge, and experience. *Knowledge* and *experience* not only about design techniques but, in the case of the design of exploratory software for learning science, fundamentally about how students learn and how computer tools can support learning. Creativity is an elusive concept but in science it necessarily involves recognition: "it is in the nature of science that the creative product will eventually be fully accepted and followed" (Elshout, 1998). It is too early to have any firm conclusion if Modellus is a creative product in this sense but I have good reasons to suppose that that is possible. Chapters 4, 5, and 6 show some of these reasons.

A personal conclusion after many years of software design is that "good design", whatever it means, is the result of many attempts, not a one shot process. Modellus, now used by thousands of teachers and students worldwide, was the product of many less successful products, much more specific and with many more limitations. But I felt that it was not possible to design it without this personal history of less successful trials.

Chapter 4 Modelling and Modellus in the Physics Curriculum: Some Examples and a Framework

> To create a world on the computer, and to watch it evolve, is a remarkable experience. It can teach one what it means to have a model of reality, which is to say what it is to think. It can show both how good and how bad such models can be. And by becoming a game played for its own sake it can be a beginning of purely theoretical thinking about forms. The microcomputer brings something of this within the reach of many pupils and teachers (Ogborn, 1994).

4.1 Introduction

In this chapter I exemplify how Modellus can help change the curriculum and I propose a framework for modelling, based on an analysis of the past and the present content of the physics curriculum.

Dissatisfaction with the physics curriculum has been a recurrent theme in the last decades. For example, Eric Rogers, in 1969, argued that “if we are far behind the times in our teaching, we shall be doing a damage to public understanding. So I make a strong plea that we try to accelerate” (Rogers, 1969, quoted in Jennison & Ogborn, 1994, p. 6). Rogers was referring to the gap between the topics that are taught in physics and the current knowledge, theories and world view. More recently, Shabajee & Postlethwaite (2000) argued that there is a “startling contrast” between the physics that is taught in schools “that deals prominently with batteries and bulbs, trolleys on ramps, and weightless strings and frictionless pulleys” and the twenty-century science.

It is not only the physics that is taught that is outdated. The way it is taught is also outdated. Computers are now probably the most ubiquitous tool in the making of physics (it is used for almost everything, from taking measurements, controlling experiments, and making models and simulations, to writing and communicating[1]) but it is very rare to find a physics curriculum where it has a similar place. I will try to show in the following sections how Modellus and modelling can contribute to change that. Some of the ideas presented have been recently put in practice in the context of the new Institute of Physics' Advancing Physics course (2000, 2001), where Modellus is considered a tool "integral to the course".

George Marx, a leading physics educator, asked some years ago "But how do we teach the art of model making instead of that of memorizing rules?" (Marx, 1994, p. 16). Probably we can teach this helping students to create their own models and using the models to test ideas and representations. The following section gives some examples.

4.2 How Can Modellus Help Change the Physics Curriculum? A Few Examples

4.2.1 Graphs as "story tellers"

Graphs and cartesian representation of data have been identified as a difficult topic for students, with many associated misconceptions (McDermott, Rosenquist & Zee, 1987). Typical misconceptions include the confusion between trajectory and shapes of position-time graphs. Difficulties with learning from graphs has also been identified in mathematics education research, with students and pre-service teachers (Ponte, 1984).

Cartesian graphs representing coordinates in two-dimensional space are easily confounded by learners with graphs of time dependent variables. Many educational strategies and tools have been devised to overcome these difficulties. A particular useful strategy, now used by many teachers, makes use of motion sensors and has been found very useful. E.g., Brassel (1987) studied the effect of a very brief treatment with this strategy with a kinematics teaching unit on high school physics students' ability to translate between a physical event and the graphic representation of it, and the effect of real-time graphing as opposed to delayed graphing of data. He concluded that a single class period was sufficient for high school physics students to improve their comprehension of distance and velocity graphs when compared with a paper-and-pencil control treatment, most of the improvement being attributable to the use of real-time graphing.

With Modellus, the student can make similar experiments, using the mouse as a motion sensor, exploring coordinates and graphs in real-time. The student starts with the definition of two variables to represent coordinates, such as x and y. Initial values are 0 pixels for each variable, but this can be changed in the Initial Conditions window. After creating an Animation window, he can then place a particle on the window with these coordinates, selecting the option "Trajectory". After starting an experiment, the mouse can be used to move the particle, seeing the trajectory as it moves (Figure 4.1).

---

[1] It is probably not just a coincidence that the World Wide Web was created in 1989 in the physics research community.

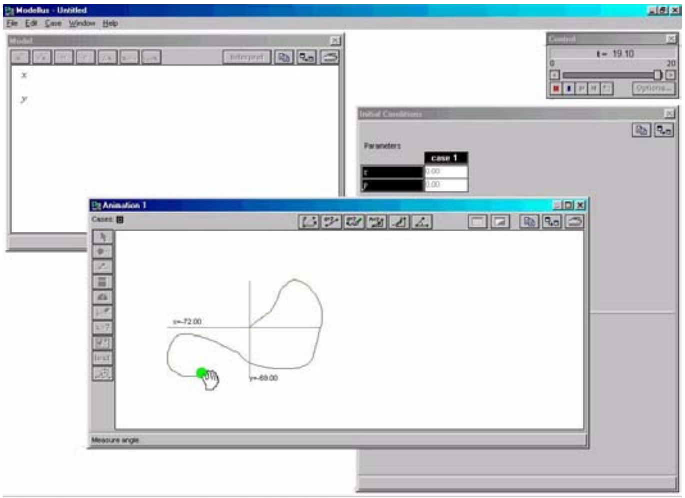


Figure 4.1 Coordinates and trajectory of an object moved with the mouse in a two-dimensional space.

The trajectory can also be seen simultaneously with a stroboscopic representation, if the student selects both “Trajectory” and “Tracking” when he place the particle on the Animation window. The stroboscopic representation is very useful to “feel” what it means to go “fast” or “slow” and how still images can represent motion.

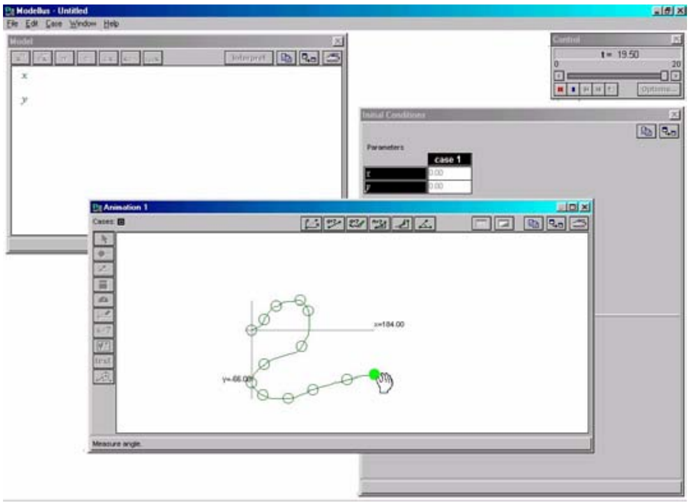


Figure 4.2 Trajectory and stroboscopic view of an object moved with the mouse.

The familiarity with these trajectory representations is a very important step before one can start making explorations with position-time graphs. Since there are two coordinates, it should be clear for students which coordinate are they seeing in a position-time graph. In particular, graphs of constant coordinates can be useful in order to become familiar which information is really in the graph. Figure 4.3 shows an example where the *y* coordinate has been fixed as 100 pixels and the *x* coordinate can move freely.

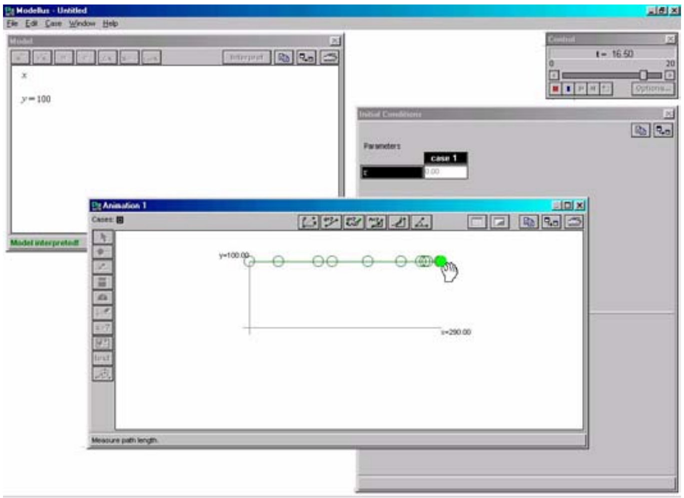


Figure 4.3 Moving the particle with the *y*-coordinate fixed.

To make a graph of the *x* coordinate as a function of time one just needs to click the graph button and select the appropriate variables. Moving the particle, the graph appears in real-time, similarly to the data logging and motion sensor systems (Figure 4.4).

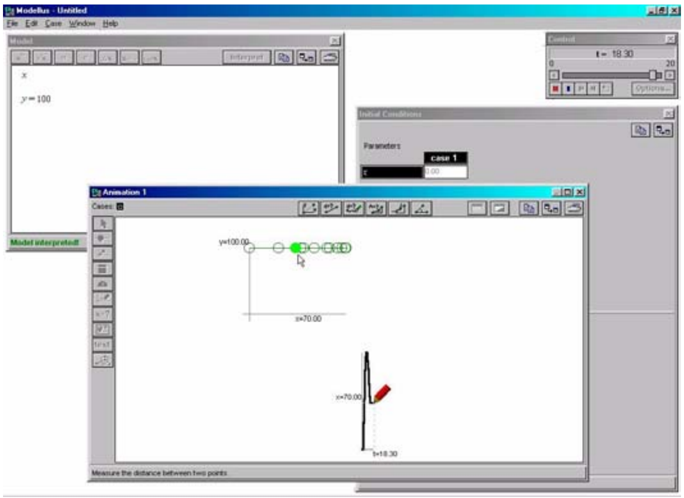


Figure 4.4 A real-time graph representing the *x*-coordinate of the moving particle. The graph scales are not appropriate and must be changed...

The graph scales are, by default, 1 pixel to 1 unit. In this case, they must be changed to have a better view of the relationship. This is done using the mouse (right button, the standard Windows button for “options”). Making the scale smaller, the graph enlarges, making the scale bigger, the graph shortens. Then, the time scale must be smaller (Figure 4.5). In some cases, the *x*-coordinate must also have a different scale, usually a bigger one.

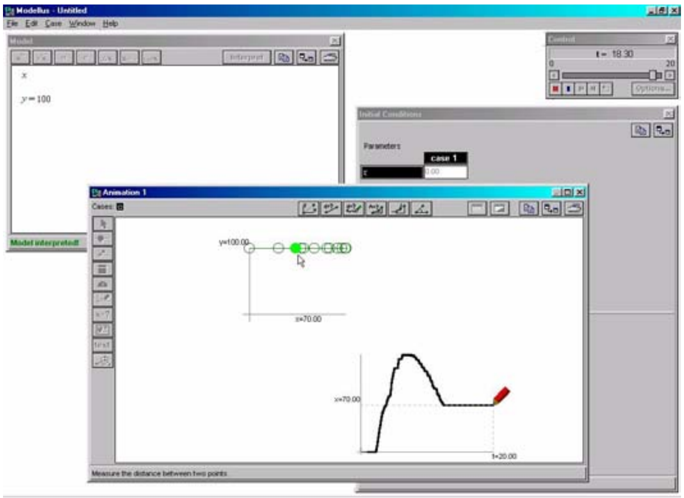


Figure 4.5 More appropriate scales to view the graph.

With such a model, the student has now a "mouse-based laboratory"[1] to experiment in real time with graphs of coordinates as a function of time. A student activity with this type of model can then evolve to make explorations of many different motions, representing one or two coordinates graphically (Figure 4.6).

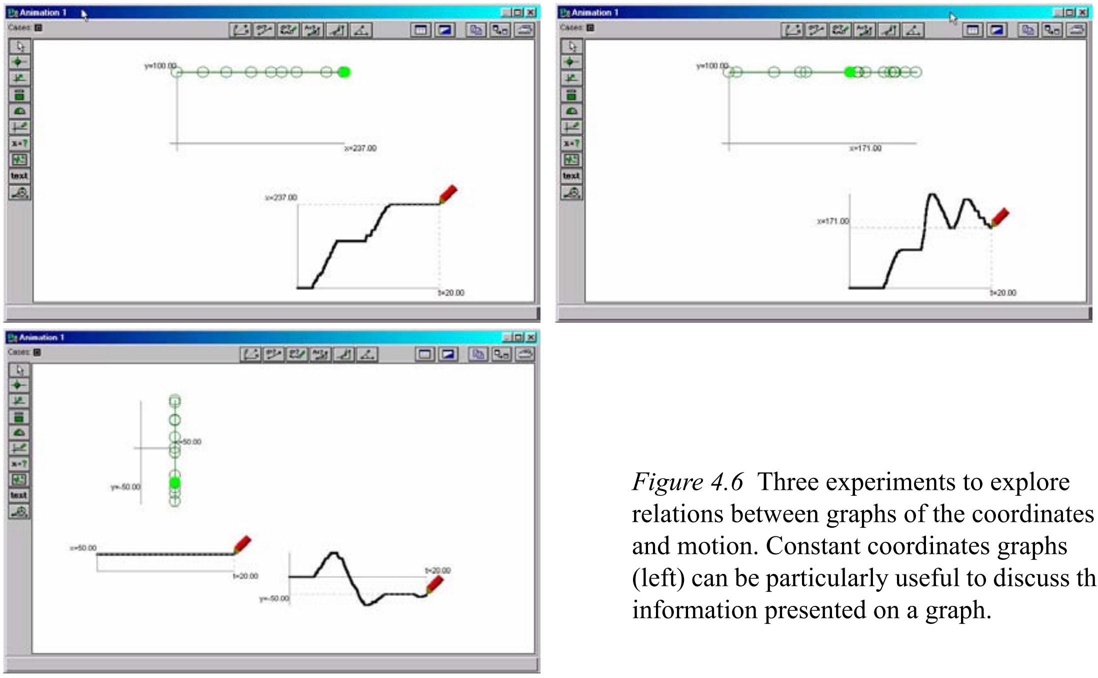


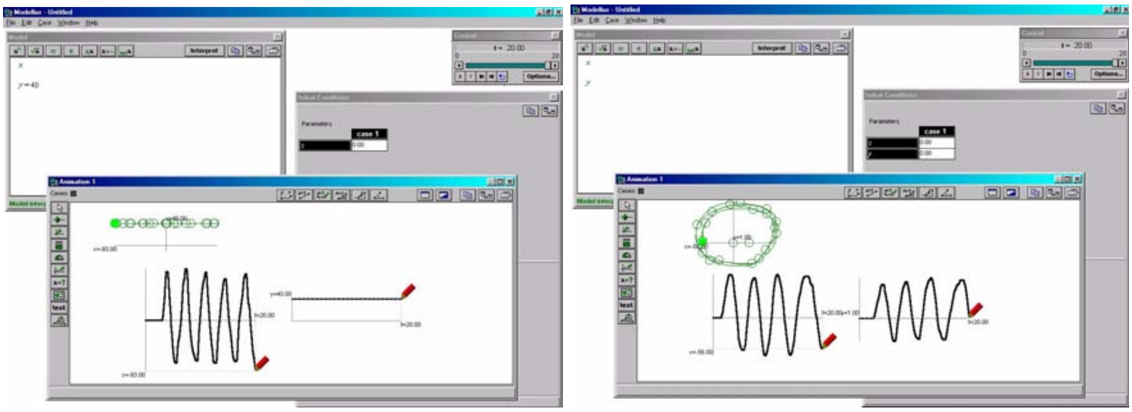
Figure 4.6 Three experiments to explore relations between graphs of the coordinates and motion. Constant coordinates graphs (left) can be particularly useful to discuss the information presented on a graph.

Similar experiments can be done to explore more complex movements, particularly periodic motion and circular motion (Figure 4.7). In this case, graphs have some

---

[1] The label "mouse-based laboratory" was suggested by Judah Schwartz (personal communication).

important features (e.g. amplitude, period, etc.) that can be introduced without the knowledge of more formal mathematics, such as trigonometric functions.

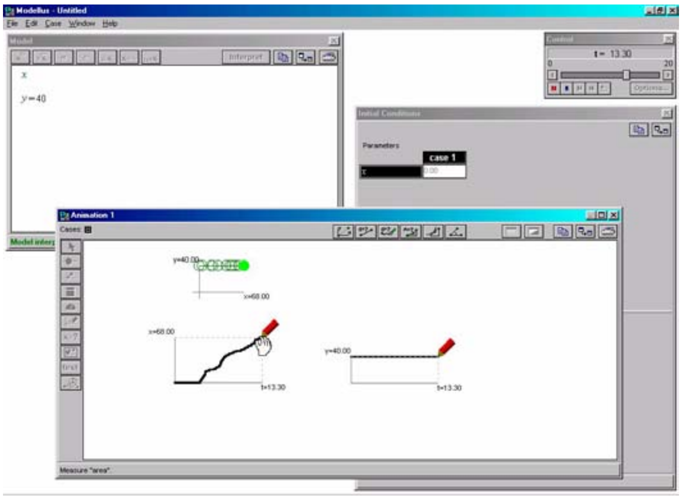


x

y=40

Figure 4.7 Experiments with “almost” simple harmonic motion and “almost” circular motion.

Graphical interpretation is an important skill in physics and in mathematics but is also important in our mediated world—it can even be considered a new kind of literacy (diSessa, 2000). Modellus can be an important didactic tool to improve this skill, allowing students to make simple experiments with graphs in real-time. It can even allow students, and teachers, to make “impossible” experiments, such as creating a motion from a graph. One just need to use the mouse to move the “pen” of the graph (Figure 4.8). These “thought experiments” are particular useful to play a sort of “multiple representation games”: students can go from the motion to the graph or from the graph to the motion.



x

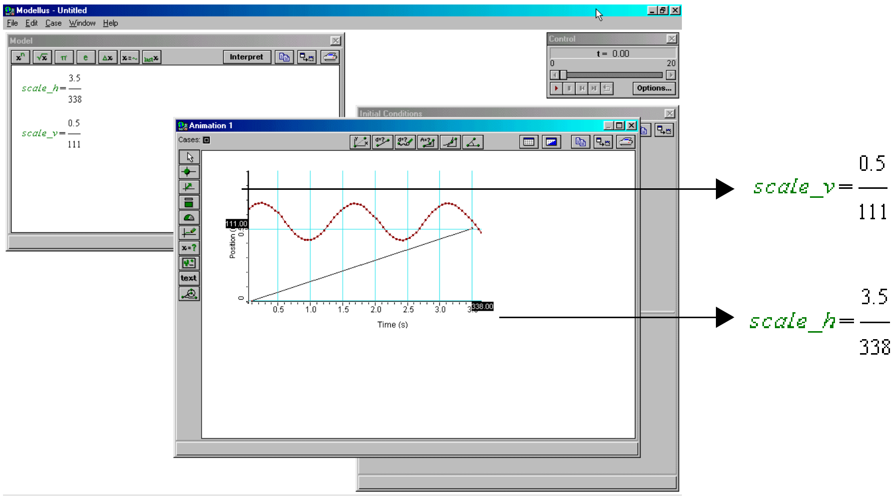
y=40

Figure 4.8 Obtaining a motion from a graph (notice the mouse over the pen of the *t*-*x* graph.)

4.2.2 Models from data: a different approach with Modellus

Constructing models from empirical data is a common activity in the physics curriculum. Modellus allows students to build models from graphs of experimental data, videos, and photos, with a different approach, where defining and using scales play an important role. Making the use of graph scales more explicit have been recommend by some researchers (e.g., Ponte, 1984; Friel, Curcio & Bright, 2001).

A typical activity of data analysis with Modellus can start with placing a graph (a *bmp* or *gif* image) on the background of the Animation window (the Modellus CD has a few hundreds graphs of experimental data obtained with data logging systems)[1]. Once the graph is placed, the first step is to find the appropriate factor scales, i.e., the factor that converts pixels into the graph units for each axis. This is done using the coordinates measuring tool—Figure 4.9. In this example, the horizontal factor scale is $\frac{3.5\text{ s}}{338\text{ pixels}}$. That is, one must multiply by this value any measurement in pixels on the horizontal scale. The vertical factor scale is $\frac{0.5\text{ m}}{111\text{ pixels}}$; multiplying by this value any measurement in pixels on the vertical scale the student gets the measurement in meters. These values can be used to define constants (in this case, *scale_v* and *scale_h*)—see Figure 4.9.



$scale\_h = \frac{3.5}{338}$

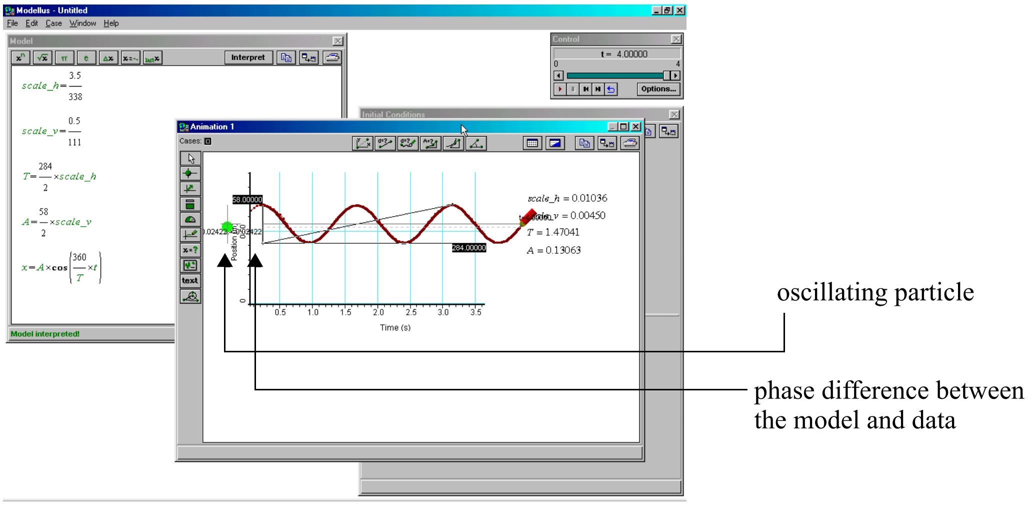
$scale\_v = \frac{0.5}{111}$

Figure 4.9 Finding the scales of the graph: in the horizontal axis, 3.5 s corresponds to 338 pixels; in the vertical axis, 0.5 m corresponds to 111 pixels. The factor scales are defined as constants in the Model Window.

Using again the measurement tools, and the factor scales, the student can then measure the appropriate parameters for the model—amplitude and period for the current example, an oscillation (Figure 4.10). With these values, he can finally define a function and place a graph of it over the experimental data, using the same scales.

---

[1] A detailed explanation of this activity is included on the Help file of Modellus (Workshop "Analyzing a Graph of Experimental Data and Creating an Animation").



$$scale\_h = \frac{3.5}{338}$$

$$scale\_v = \frac{0.5}{111}$$

$$T = \frac{284}{2} \times scale\_h$$

$$A = \frac{58}{2} \times scale\_v$$

$$x = A \times \cos\left(\frac{360}{T} \times t\right)$$

$scale\_h = 0.01036$

$scale\_v = 0.00450$

$T = 1.47041$

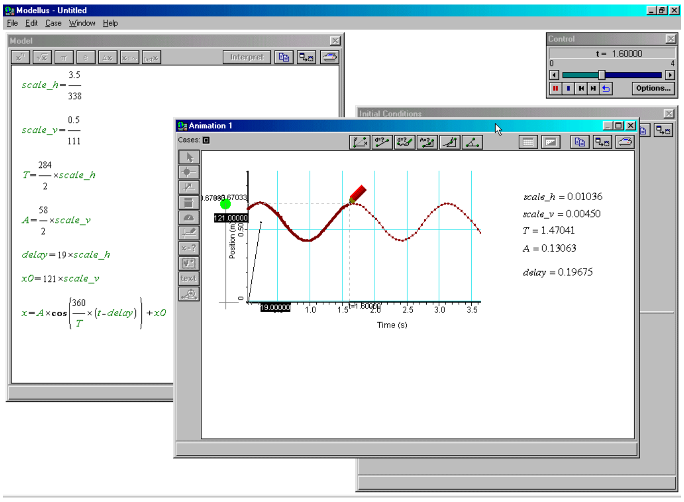
$A = 0.13063$

oscillating particle

phase difference between the model and data

Figure 4.10 Comparing the model and the data, and using the model to make an animation of the oscillator.

As can be seen on Figure 4.10, there is a phase difference between the model and the data. There is also the need to introduce a new parameter in order to make the equilibrium point higher than zero. The phase difference can be found measuring the initial value of the oscillator—and an inverse trigonometric function—or can more easily be found finding the delay in the initial value of the function—Figure 4.11.



$$scale\_h = \frac{3.5}{338}$$

$$scale\_v = \frac{0.5}{111}$$

$$T = \frac{284}{2} \times scale\_h$$

$$A = \frac{58}{2} \times scale\_v$$

$$delay = 19 \times scale\_h$$

$$x0 = 121 \times scale\_v$$

$$x = A \times \cos\left(\frac{360}{T} \times (t - delay)\right) + x0$$

$scale\_h = 0.01036$

$scale\_v = 0.00450$

$T = 1.47041$

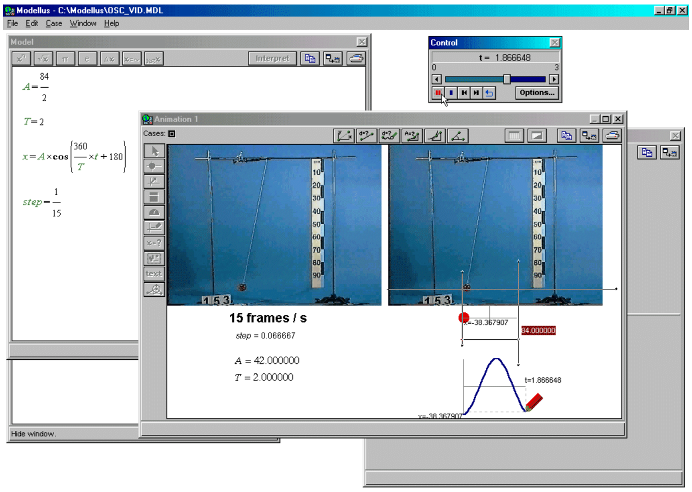
$A = 0.13063$

$delay = 0.19675$

Figure 4.11 A more complete model.

This example shows some important features of data analysis with Modellus when compared with other traditional tools, that use automatic data fitting with least-squares method. Using Modellus, students *need* to make measurements, find scales, compute parameters, etc. With automatic tools, they only *select* appropriate actions. Using Modellus, students can have a multiple step approach to the model, from less appropriate to more appropriate. And they can make an animation based on the model, analysing it in conjunction with the graph.

Models from data can also be done using a video, placed on the background of the Animation window[1]. In this case, it is necessary to find an appropriate time scale, usually finding the number of frames per second. An AVI video placed on the Animation is automatically synchronized with time. Defining a small step on the Control window and re-running the video step by step, the number of frames per second can easily be found. Once this parameter is found, the video can then be synchronized image by image and the relevant time parameters can then be easily measured. Figure 4.12 shows an oscillating pendulum, its model (the period is 2.0 s and the amplitude is 42 pixels), and an animation.



$$A=\frac{84}{2}$$

$$T=2$$

$$x=A\times\cos\left(\frac{360}{T}\times t+180\right)$$

$$step=\frac{1}{15}$$

t = 1.866648

15 frames / s

step = 0.066667

A = 42.000000

T = 2.000000

Figure 4.12 Using Modellus to make a model after taking measurements on a video.

In some cases, it can be useful to use Modellus in conjunction with a spreadsheet, such as Excel. In one of the Modellus manuals, there is presented a detailed explanation of how to make an integrated use of these tools analysing the motion of a Pasco fan cart that reduces its speed, stops, and then moves backwards. After synchronizing Modellus with the video, image by image, the user can take measurements of the position in each frame or instant (Figure 4.13), and then make a table in Excel.

---

[1] Modellus CD includes about 150 videos of experiments, thanks to Pasco Scientific and VideoPoint.

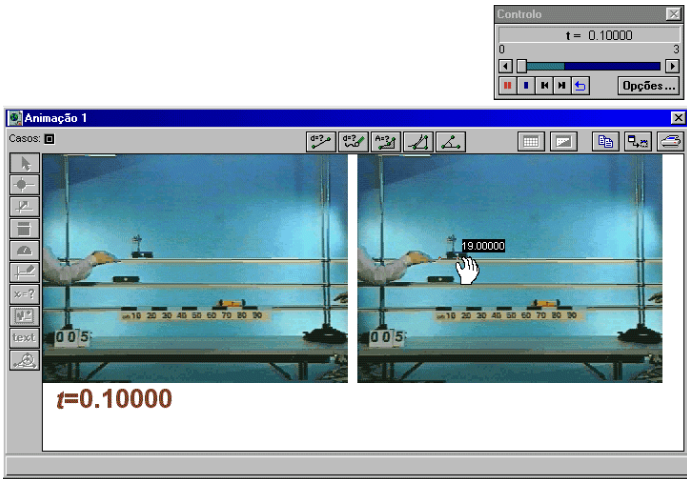
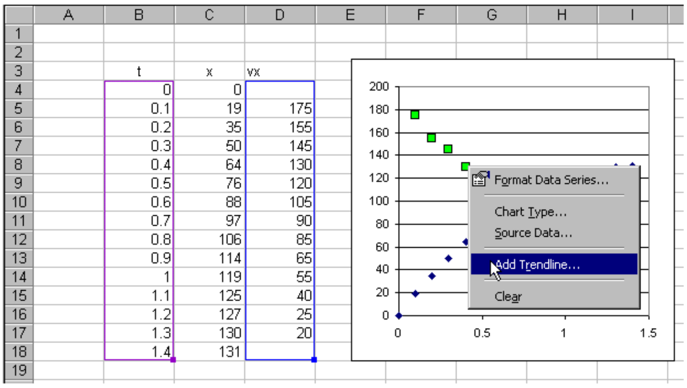


Figure 4.13 Making measurements of position and time frame by frame on a video.

From the table, the user computes change in position and makes graphs. Using the data fitting tools of Excel (Figure 4.14), he/she can find the parameters of the model.



	t	x	vx
4	0	0	
5	0.1	19	175
6	0.2	35	155
7	0.3	50	145
8	0.4	64	130
9	0.5	76	120
10	0.6	88	105
11	0.7	97	90
12	0.8	106	85
13	0.9	114	65
14	1	119	55
15	1.1	125	40
16	1.2	127	25
17	1.3	130	20
18	1.4	131	

Figure 4.14 Using the Excel data fitting tools to find the model of a data set.

Finally, using the parameters and the appropriate model, he/she can build an animation and analyse how good the model is. The animation can also include graphs, vectors, etc. (Figure 4.15).

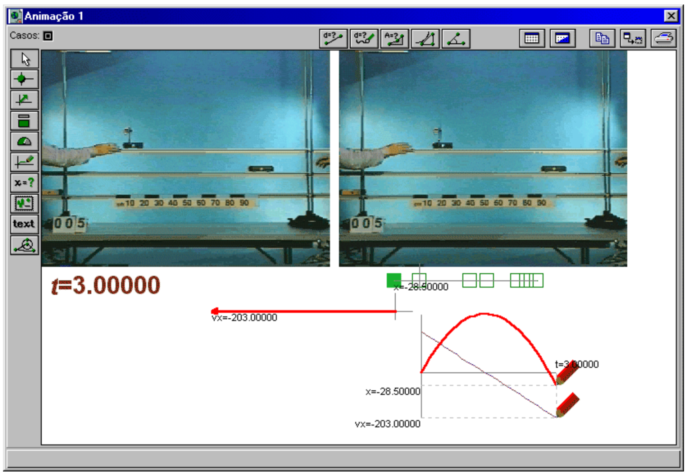
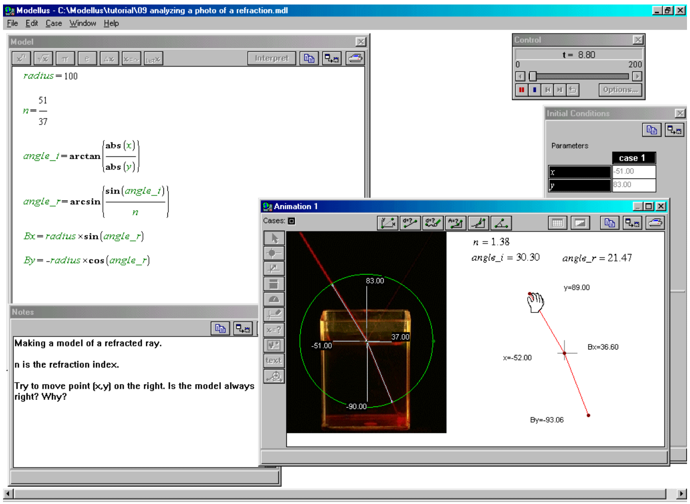


Figure 4.15 Comparing the model (position and velocity) and the video. The animation also includes vectors and graphs from the model.

This kind of analysis of experimental data can be done with specific software, such as VideoPoint (see chapter 2). But there are significant differences in both tools. In VideoPoint, most of the analysis is done using automatic procedures and the user cannot build any animation based on the model in order to analyse how well it fits data or to represent useful mathematical objects such as vectors.

Another way of using Modellus to take measurements and make models from images, used as experimental data, is shown on Figure 4.16. In this example, the measuring tools (and a circle, obtained with a graphical object) are used to find the refraction index. This index is then used to build a geometric model of a refracted ray; the incident ray can be manipulated to explore how the angle of refraction is related to the angle of incidence.



$radius = 100$

$n = \frac{51}{37}$

$angle\_i = \arctan\left(\frac{\mathrm{abs}(x)}{\mathrm{abs}(y)}\right)$

$angle\_r = \arcsin\left(\frac{\sin(angle\_i)}{n}\right)$

$Bx = radius \times \sin(angle\_r)$

$By = -radius \times \cos(angle\_r)$

	case 1
x	-51.00
y	83.00

$n = 1.38$

$angle\_i = 30.30$ $angle\_r = 21.47$

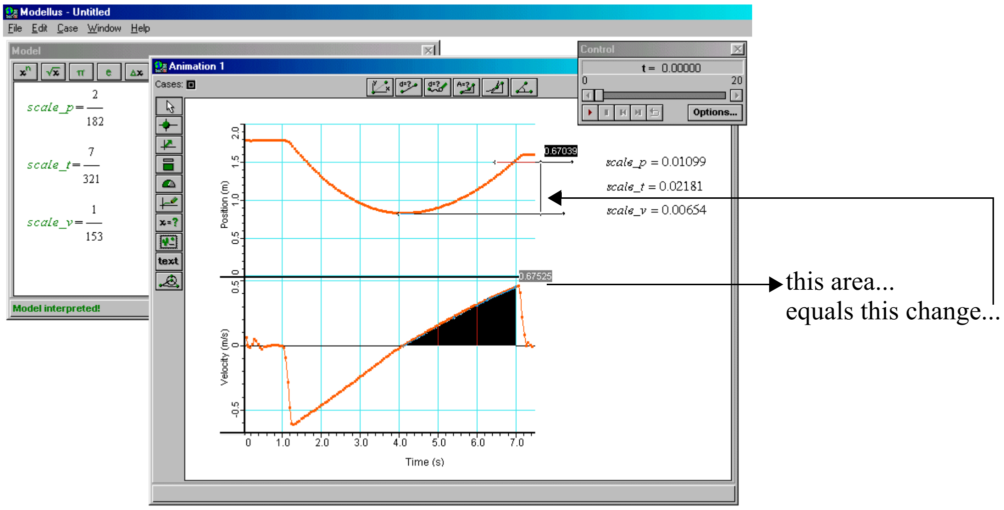
Making a model of a refracted ray.

n is the refraction index.

Try to move point (x,y) on the right. Is the model always right? Why?

Figure 4.16 Using a photo to measure the index of refraction and build a geometrical model of the incident and refracted ray.

There are more experimental contexts where measurement tools are particularly useful. An uncommon example is shown on Figure 4.17. In this case, the area measurement tool was used to find the area under a graph (an integral) and the distance tool checks how the area measures the change in position.



$scale\_p = \frac{2}{182}$

$scale\_t = \frac{7}{321}$

$scale\_v = \frac{1}{153}$

$scale\_p = 0.01099$

$scale\_t = 0.02181$

$scale\_v = 0.00654$

this area...
equals this change...

Figure 4.17 Measuring an integral on a velocity-time graph and comparing it with the change in the primitive function, a position-time graph.

These four examples of data analysis in Modellus show how it can be used as a tool for laboratory experiments. The main characteristics of this use are:

1. the *importance the user must give to factor scales*;
2. the *non-automatic way of finding parameters*;
3. and the *possibility to make animations from the models* to analyse how good are the models compared to the data.

4.2.3 Exploring phasors: using Modellus to "think with"

In a recent paper, Shabajee & Postlethwaite (2000) asked "What happened to modern physics?". Recognizing that twentieth century physics is almost absent in the physics curriculum, contrary to other sciences, like chemistry and biology, where recent topics have an important place, they argue that there is "an urgent need to include the concepts of 'twentieth century physics' within the curriculum". This is an important goal in the Advancing Physics course (Ogborn & Whitehouse, 2000, 2001) that uses Modellus as integral tool to the course. In this sub-section I will present some examples from the course that make use of phasors to explore a modern view of quantum mechanics. This view was popularized by Feynman in his famous short book *QED The strange theory of light and matter* (1985). It is a good example of how familiarization is a fundamental issue in "understanding" physics. Quantum mechanics is "impossible" to understand—one just gets familiar with it... Or, as Feynman wrote:

> What I am going to tell you about is what we teach our physics students in the third or fourth year of graduate school... It is my task to convince you not to turn away because you don't understand it. You see, my physics students don't understand it... That is because I don't understand it. Nobody does (p. 9).

The importance of *becoming familiar* is also stressed in the Advancing Physics course (2000):

> New ideas always seem peculiar at first, but you get used to them. That's how it is with quantum behaviour. Start young and you stop worrying sooner! (p. 158).

Units 6 (Wave Behaviour) and 7 (Quantum behaviour) have thirty seven Modellus files to illustrate concepts presented in the book. The initial models explore the idea of a "rotating arrow" and how it can be used to describe waves (Figure 4.18). More advanced models are used in activities where students can use phasors to interpret wave superposition and diffraction.

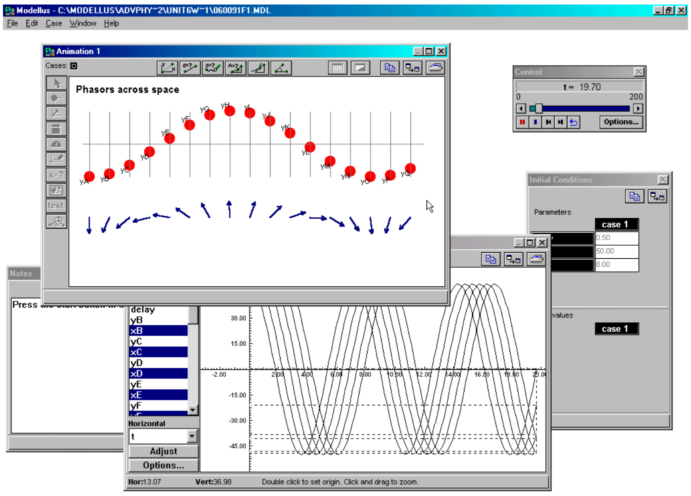


Figure 4.18 A model from the Advancing Physics course used to show students how phasors can describe wave propagation.

There are some very interesting models in Unit 7 (Quantum behaviour) that implements the Feynman's view of quantum mechanics applied to photons and electrons, light propagation, diffraction (Figure 4.19), reflection and refraction. In these models students can "see" how rotating "arrows" (in reality, complex numbers) are powerful tools to make predictions and explain phenomena. As with all visualisation tools, some students can take it too literally! Advice is given to students to be careful about "quantum objects" and its strange and weird behaviour. Modellus is used in these two units as a visualization tool that brings life to Feynman's ideas.

The editor of the CD of the course, in his answer to the Questionnaire about modelling and Modellus (chapter 5) wrote:

> Writing Modellus [models] is teaching me a lot about phasors that a degree in physics never did! Integrated in that you can see, by a number of visualisations, how the physics benefits from the concise expression and richness of the mathematics and how the mathematics is given real meaning by concrete interpretation. [Ian Lawrence]

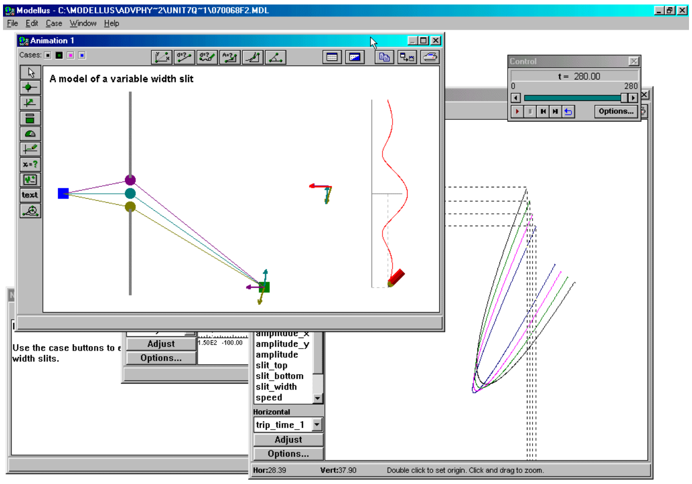


Figure 4.19 A model from the Advancing Physics course where students explore how phasors can interpret diffraction on a single slit.

Using "software to think with" is a powerful metaphor for the role of Modellus in physics teaching and learning. This is true for curriculum developers, teachers, *and* students. If one looks carefully at most of the models of these two units, they use basic mathematics (explicit common functions, such as linear functions; basic geometry, such as coordinates, distance between points in a plane, etc.; and elementary vector calculus, such as components, sum, etc.) that can be used by secondary school students. This means that they can also build this type of models—at least the less sophisticated ones. Certainly this is not possible for all Advancing Physics' Modellus examples (or in any other physics curriculum) but the tool is there for those who want to go deeper, getting more and more familiar with the strange properties of quantum objects. Students, as all of us, are initially "shocked" by these properties (like Planck, who was "shocked" by quantum theory), but when we become more familiar with them, when we put them to work, they seem as natural as those we are now familiar—even when we know that they were fought by many scientists when they were first proposed.

4.2.4 From simple multiplication to calculus and differential equations

Calculus has been a fundamental tool in physics since Newton and Leibniz. Accordingly to the reform movement in calculus—see 2.4.2—new calculus courses should (Gordon et al. 1994, p. 56):

1. cover fewer topics and give more emphasis on fundamental concepts;
2. place less emphasis on complex manipulative skills and emphasize modelling the real world;
3. promote experimentation and conjecturing;
4. teach students to think and reason mathematically, develop problem-solving skills;
5. make use of calculators and computers.

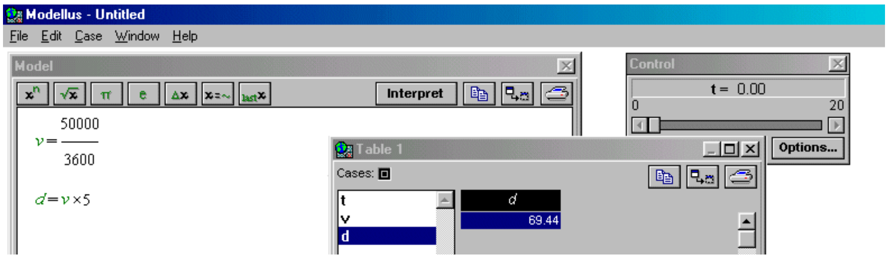
Modellus has the potentiality to be an important computer tool to introduce students to the *fundamental concepts* of calculus from a physics point of view (recent courses, such as Rex & Jackson, 2000, adopt this approach). It can be used to design learning sequences where concepts such as functions, derivatives, differential equations, iterative computations, etc., are approached in an integrated way, giving students the opportunity to explore different views of the same problem, placing "less emphasis on complex manipulative skills" and promoting "experimentation and conjecturing". In the Advancing Physics A2 course (Ogborn & Whitehouse, 2001), students are introduced to calculus concepts such as differential equations defined as "equations involving rates of change in quantities" and simple numerical methods using Modellus.

The following example illustrates how can Modellus be used to have an integrated view of some calculus concepts (it has been used in Modellus workshops given to students and teachers): a car is travelling in a straight line at a speed of 50 km/h. How much distance does it travel in five seconds?

This trivial problem can be solved easily with a direct computation after converting km/h in m/s ($\frac{50000\text{ m}}{3600\text{ s}} = 13.89\text{ m/s}$):

$$\begin{aligned} \text{distance} &= \text{speed} \times \text{time} \\ &= 13.89\text{ m/s} \times 5\text{ s} \\ &= 69.44\text{ m} \end{aligned}$$

Modellus can be used as a simple calculator to make this computation (Figure 4.20).



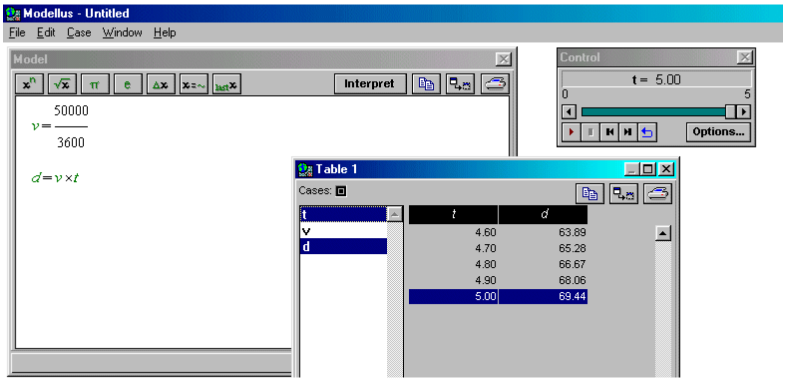
$v=\frac{50000}{3600}$

$d=v\times5$

d
69.44

Figure 4.20 Using Modellus to make a simple computation...

Another way to compute the distance is to define distance as a *linear function of time*, since it is assumed that speed is constant—Figure 4.21. After five units of time, *d* is 69.44 m, as expected.



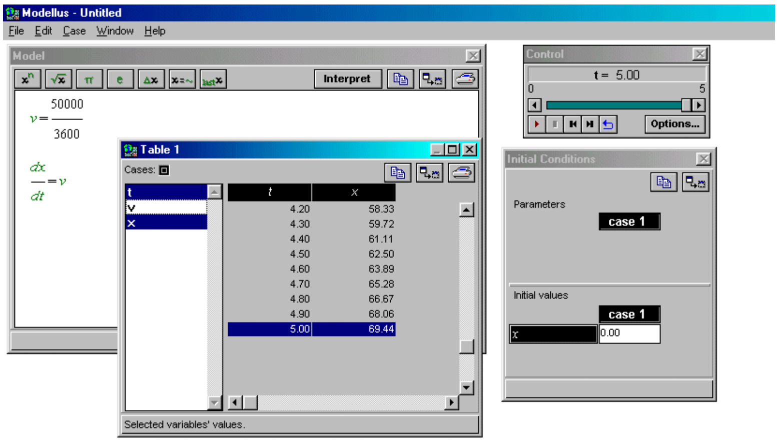
$$v=\frac{50000}{3600}$$

$$d=v\times t$$

t	d
4.60	63.89
4.70	65.28
4.80	66.67
4.90	68.06
5.00	69.44

Figure 4.21 Defining distance *d* as a function of time *t* and computing its value after five seconds.

A calculus approach to this computation can start from the idea that the rate of change of position is constant and equal to the speed. If *x* is the coordinate of the car, then we have “rate of change of *x* equal to *v*”. Figure 4.22 shows a Modellus model with a differential equation for this statement and the output after five units of time. This model can be used to explore why the user must give an “Initial” value for *x*, in the “Initial Conditions” window.



$$v=\frac{50000}{3600}$$

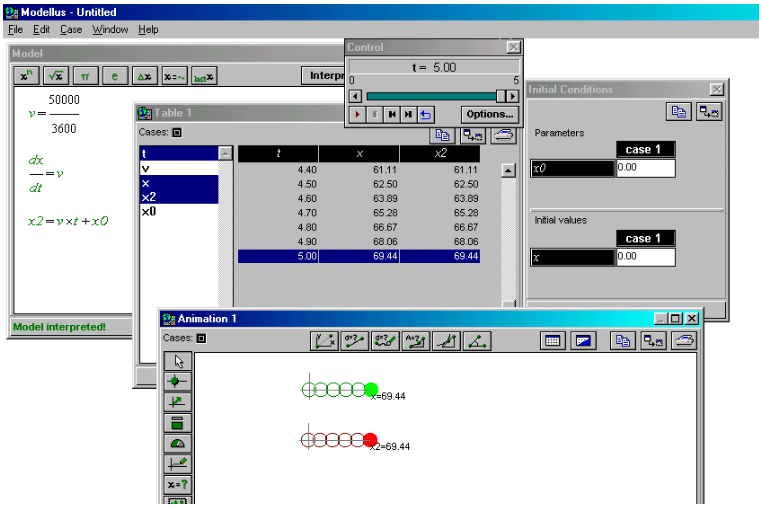
$$\frac{dx}{dt}=v$$

t	x
4.20	58.33
4.30	59.72
4.40	61.11
4.50	62.50
4.60	63.89
4.70	65.28
4.80	66.67
4.90	68.06
5.00	69.44

x 0.00

Figure 4.22 Using a differential equation to compute the position *x* after 5 seconds.

In any of these two dynamic models, one using a function and the other using a differential equation, the student can easily make an animation of the problem. The animation can be particularly useful to explore the meaning of the initial value for *x* and how to make the “function” model equivalent to the “differential equation” when the initial value for *x* is changed. It can even make the two models in the same file (using different labels for the variables)—Figure 4.23.



$$v = \frac{50000}{3600}$$

$$\frac{dx}{dt} = v$$

$$x2 = v \times t + x0$$

t = 5.00

Cases:

t	x	x2
4.40	61.11	61.11
4.50	62.50	62.50
4.60	63.89	63.89
4.70	65.28	65.28
4.80	66.67	66.67
4.90	68.06	68.06
5.00	69.44	69.44

Parameters

	case 1
x0	0.00

Initial values

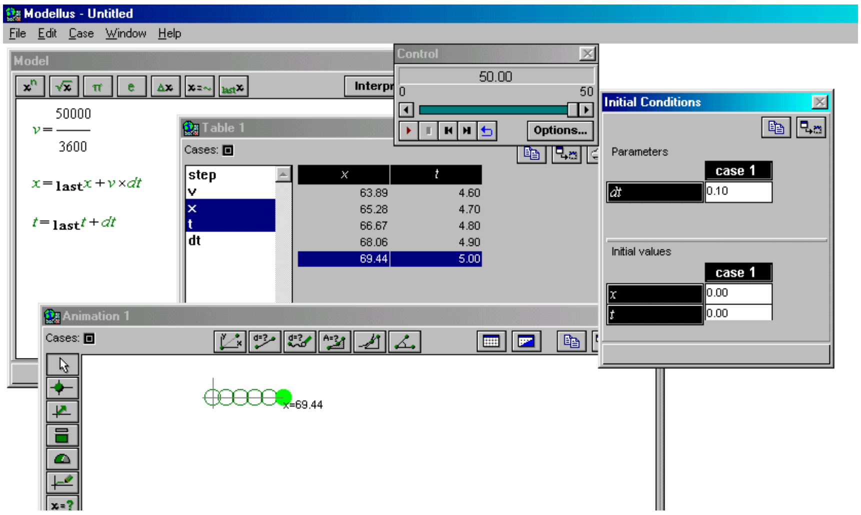
	case 1
x	0.00

Figure 4.23 Comparing a differential equation and a function...

It is possible to make yet another type of computation (an *iterative* computation) in Modellus to compute the distance. Considering *dt* as a small period of time, the “new *x* coordinate” after a “short” time *dt* is given by the “old *x*” plus the change in *x* during this short time *dt*:

$$\text{new } x = \text{old } x + \text{change}$$

Since the “change” in *x* during the short time *dt* is $v \times dt$ one can write an iterative Modellus model (Figure 4.24) where *x* iterates 50 times since *dt* is 0.1 and we need to compute the value of *x* after 5 seconds. Time *t* is also iterated in this model.



$$v = \frac{50000}{3600}$$

$$x = \text{last}\,x + v \times dt$$

$$t = \text{last}\,t + dt$$

50.00

Cases:

x	t
63.89	4.60
65.28	4.70
66.67	4.80
68.06	4.90
69.44	5.00

Parameters

	case 1
dt	0.10

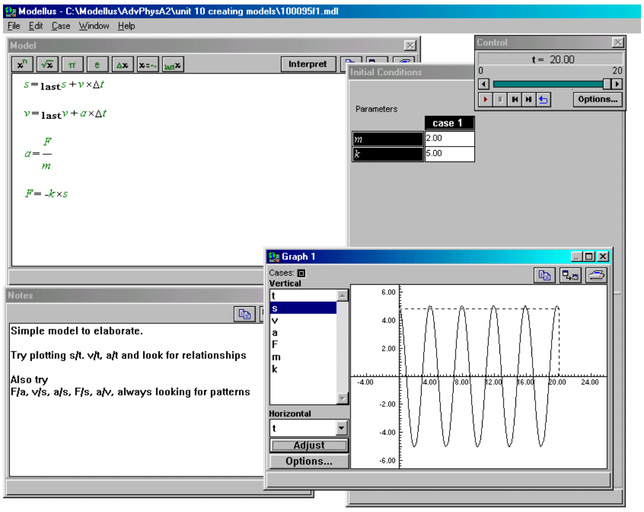
Initial values

	case 1
x	0.00
t	0.00

Figure 4.24 An iterative model to compute the distance travelled.

Iterative models like this one are extensively used in physics and applied mathematics in many contexts. Modellus makes the exploration of these iterative models an easy task to students and teachers—it is used in some units of the Advancing Physics *A2* course

(Figure 4.25). Iterative models are also used in classical books, such as *The Feynman Lectures on Physics*. With Modellus, to make a model of gravitation such as the one explained by Feynman in Chapter 9 of his *Lectures*, is a simple and straightforward task accessible to secondary school students—see, e.g., Fiolhais et al., 1996b, p. 149; or the models about geostationary satellites made by students during a "Ciência Viva" project at the IST, Lisbon, available on http://www.math.ist.utl.pt/cam/cviva1/ index.html (retrieved August 25, 2002) and reproduced in the Modellus web site.



$s = \text{last}\,s + v \times \Delta t$

$v = \text{last}\,v + a \times \Delta t$

$a = \dfrac{F}{m}$

$F = -k \times s$

Parameters

	case 1
m	2.00
k	5.00

Simple model to elaborate.

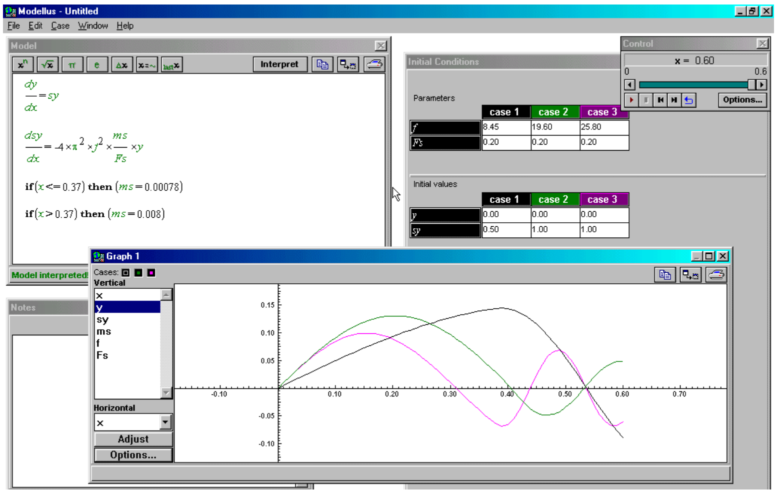
Try plotting s/t. v/t, a/t and look for relationships

Also try
F/a, v/s, a/s, F/s, a/v, always looking for patterns

Figure 4.25 An example of an iterative model used in *Advanced Physics A2* to explore oscillatory motion.

Many more examples could be given of how Modellus can also be used as a tool to explore models based on differential equations. A recent text on atomic physics written by Niedderer & Petri (2000) uses Modellus models to solve the Schrodinger equation. Accordingly to the authors, "The modelling of the Schrodinger equation can be substantially simplified and shortened" with Modellus. An example of a model from this text is given in Figure 4.26. The identification of the Schrodinger equation is straightforward if the user knows how to write second order differential equations in Modellus.

Having a tool to explore differential equations as "easy to use" as Modellus can also allow teachers and curriculum developers to introduce high school and undergraduate students to late twenty-century physics concepts, such as those related to chaotic systems and chaotic behaviour. This has also been done in the Advancing Physics A2 course for high school students and, e.g., by Veit & Mors (2000) to first year undergraduates—Figure 4.27.



$$\frac{dy}{dx} = sy$$

$$\frac{dsy}{dx} = -4 \times \pi^2 \times f^2 \times \frac{ms}{Fs} \times y$$

if$(x <= 0.37)$ **then** $(ms = 0.00078)$

if$(x > 0.37)$ **then** $(ms = 0.008)$

x = 0.60

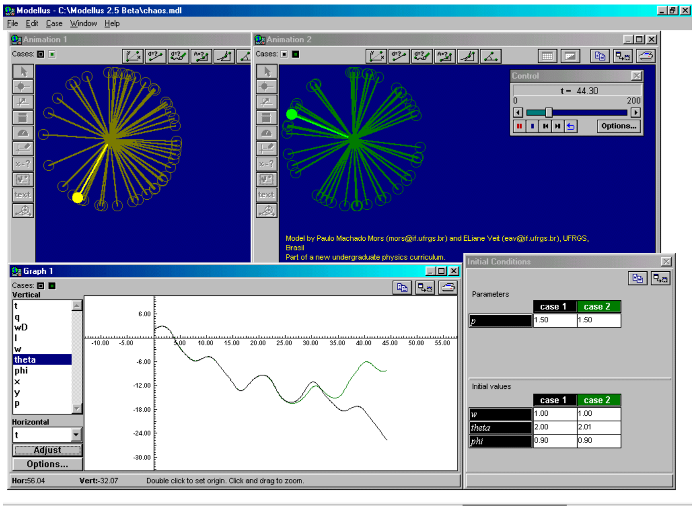
Parameters

	case 1	case 2	case 3
f	8.45	19.60	25.80
Fs	0.20	0.20	0.20

Initial values

	case 1	case 2	case 3
y	0.00	0.00	0.00
sy	0.50	1.00	1.00

Figure 4.26 Solving the Schrodinger equation with Modellus (Niedderer & Petri, 2000).



Model by Paulo Machado Mors (mors@if.ufrgs.br) and ELiane Veit (eav@if.ufrgs.br), UFRGS, Brasil
Part of a new undergraduate physics curriculum.

Hor:56.04 **Vert:**-32.07 Double click to set origin. Click and drag to zoom.

Parameters

	case 1	case 2
p	1.50	1.50

Initial values

	case 1	case 2
w	1.00	1.00
theta	2.00	2.01
phi	0.90	0.90

Figure 4.27 An example of the chaotic behaviour of a pendulum used in a first year undergraduate course (From Veit & Mors, 2000).

4.3 Modelling in the Physics Curriculum: A Framework

4.3.1 The physics curriculum: old and new challenges

In the second half of the nineteen century, scientific disciplines challenged the dominance of the classical-literary curriculum at the secondary and college level. They were recognised as curriculum subjects only at the end of the century and “after fierce controversy” over its status and credits as subjects (Jenkins, 1991). The main reasons for the introduction of science were the industrialization of societies and the development of science as a professional activity. Courses tended to stress practical use of scientific knowledge, particularly in secondary schools, rather than any systematic study of science. Early methods of teaching were based on lecture and on the textbook; practical work and laboratory as an integral part of teaching were only introduced in the early twenty century (Jenkins, 1991).

The emergence of physics as an academic and school subject took more than a century (Barnett, 2000). And, accordingly to Jenkins (1996), physics as a school subject has been "constructed largely for educational purposes":

> Its emergence from fields like electricity and magnetism, heat, light and sound, mechanics and properties of matter does, of course, owe something to developments in fundamental understanding of the natural world but it was a long time before school physics ceased to be conceptualised, institutionalised, taught and assessed in terms of these contributing fields of understanding (p. 5).

The school physics curriculum suffered significant changes during the 1960's and 1970's when teams of scientists and teachers joined efforts to develop new texts and new approaches, inspired in most cases by the work of cognitive psychologists such as Bruner, Gagné and Piaget. Bruner wrote in his famous book *The process of education* (1960) about the importance of the structure of the subject as the main factor in curriculum design. The emphasis of the new courses, such as the PSSC (Physical Science Study Committee, 1960) or the Nuffield (Nuffield Foundation, 1967), were "on pure science, not on applications of science, and on important scientific processes and conceptual schemes" (Lunetta, 1991). Mathematics was an important aspect in the courses. In the early 1980s, Nuffield Advanced Physics, a revision of the two-year course for the final years of secondary school in the UK, introduced for the first time numerical methods (difference equations and iteration) and software as an integral part of the course (Ogborn, 1984).

Project Physics, initiated in the United States in 1964 under the direction of experienced physics educators, philosophers and historians of science, also had international impact (e.g., it is available in Portuguese). The course "intended to increase the appeal of physics to a broader range of high school students by emphasizing the humanistic roots and consequences of physics" integrating "history, culture, technology, and people in the development of physical ideas" (Lunetta, 1991). Physics was seen as an human intellectual endeavour. Mathematical skills were reduced in order to broaden the appeal to students with less academic abilities.

Forty years after the beginning of the curriculum development wave, physics educators are still facing the same problems *and* new challenges due to the developments of physics and technology and to a changing society's view on education (Lijnse, 1998). How can the curriculum balance "the structure of science" with an informed view of the use of technology and its impact on society? How can the curriculum answer the needs of the more academic talented students with the learning difficulties of the majority? Is it possible to teach current physics ideas and how they were developed or are they simply too difficult to be accessible to students? Can the curriculum balance acquaintance with many ideas with a deeper understanding of a few? What is the role of practical, laboratory work, and computers in the curriculum? Is it possible to balance autonomy and meaningful learning with examinations? These are only a few of the many challenges faced not only by physics educators and curriculum developers but also by physics education researchers.

A new wave of renovation in the physics curriculum started in the 1990s inspired by the Standards movement (mainly in the US) and by professional societies, such as the Institute of Physics (UK) and the American Association for the Advancement of Science. New curricula, like Active Physics (Eisenkraft, 1998) in the US or the Advancing Physics in the UK, present physics as a much more practical subject, linked to everyday phenomena, technology, and engineering, promoting student investigations outside traditional physics topics (e.g., image analysis, digital measurement, relativity), but including the essential of the classic physics curricula, such as motion, with new perspectives based on using computers to develop and test models.

These new curricula try to broaden the appeal of physics, not only for those interested in it, but mainly for students that will pursue careers in other sciences and in engineering, or even in other subjects outside science and technology. Mathematics is seen as "fundamental to the pleasure and power that physics has to offer" (Ogborn, 1999) but without an excessive formalism. Algebraic expressions, equations, and graphs are approached as a way of representing ideas symbolically. Students with insufficient mathematical backgrounds have special support. Graphing calculators and computer tools (spreadsheets and other modelling tools) help students work with complex mathematical ideas, promoting visualization and direct manipulation of abstract ideas.

Laboratory work is *not exclusively inquiry/discovery* oriented. It is more concerned with demonstrating concepts and making ideas "come real" (Ogborn, 1999). Stories and "scenarios" are frequently used to introduce physics ideas and views of the world. Project and group work are also used frequently, promoting collaborative and constructivist approaches. Writing, discussing, explaining, and supporting points of view, are common tasks in learning activities. Sources of information are not restricted to the classroom book (smaller than traditional course books). Students need to use multiple sources of information, including digital documents locally or from the Internet.

4.3.2 Experimentation and meaningful learning

It is unquestionable that experimentation and direct observation is inherent to any physics curriculum. One can question its role but not its place in the curriculum. Experimentation can be done by students or by teachers in more open or in more closed formats depending on goals and contexts. When done by students, usually in laboratories, it is usually considered as "practical work". Accordingly to Rogers & Wild (1996):

> Practical work is essential not just for learning material content, but for pupils to make their own personal contact with scientific work, with its delight and sorrows. They need to meet their own difficulties like any professional scientist and enjoy their own successes, so that the relation of scientific knowledge to experiment is something they understand (p. 130).

But learning from practical work and observation is more complex that it usually seems. Driver (1983) argues that "the slogan 'I do and I understand' is commonly used in support of practical work in science teaching" but in many cases it is more appropriate to speak of "I do and I am even more confused" (p. 9). Meaningful learning from observation is not an easy task. Almost all observations require previous conceptual knowledge to make sense of them. This conceptual knowledge is critical for the understanding of the purposes of a practical task, to the conduct of the task, and to the analysis and interpretation of data. Learners cannot discover "the theoretical models and scientific conventions (...). They need to be presented. Guidance is needed to help children assimilate their practical experiences into what is possibly a new way of thinking about them" (Driver, 1983, p. 9).

Driver's strong statement about *guidance* in practical work (and in science teaching in general) can be considered as a reaction against the role attributed to exploration, direct observation, and experimentation promoted by the *inquiry curricula* developed in the 1960s. Inquiry-oriented curricula engage students in the investigative nature of science. It can be more or less guided, but in any case it involves activity and skills in *search* for knowledge. But Driver (and many others authors) argue that knowledge must be *presented.* The role of practical work is not to give students the opportunity to *discover* science. It is the opportunity to extend the knowledge of phenomena, to exemplify principles and ideas, to gain experience in design, planning, executing, recording, interpreting, discussing, and arguing (Driver, 1993; Millar, Le Maréchal & Tiberghien, 1998).

The inquiry curricula, associated with discovery learning, were inspired by Bruner's ideas, notably the *The act of discovery* paper (1961), where he argues that discovery learning is superior to learning based on expository modes. Bruner identifies benefits in four areas: (1) information is deeply processed and better rooted in memory, being available more easily for practical and problem solving situations; (2) intrinsic motivation; (3) discovery as a very useful art, that needs to be practised; (4) cognitive structures more suited to the learner's own interests, that can be retrieved more easily.

A very different position has been assumed by other cognitive theorists such as e.g. Ausubel (1963, 2000). Ausubel's most common critique of discovery learning is that although it can be effective in certain specific situations, for the most part it is cumbersome and overly time-consuming. Discovery learning also demands from the teacher a greater contextualization in order to have a better chance of retention than rote memorization of a procedure. Accordingly to Ausubel (2000), most meaningful learning is associated with reception learning, not with discovery learning:

> "the acquisition of subject-matter knowledge in any culture is primarily a manifestation of reception learning (...) The principal content of what is to be learned is typically presented to the learner in more or less final form through expository teaching (...) The learner is simply required to comprehend the material and to incorporate it into his cognitive structure so that it is available for either reproduction, related learning, or problem solving at some future date (p. 6).

In opposition to Bruner, Ausubel argues that "discovery learning can also be, and in most classrooms typically is, rote in nature because it does not conform to the conditions of meaningful learning" (p. 5). Ausubel also shows that *meaningful reception learning is an active process*, not a passive one, and requires cognitive analysis in order to define "which aspects of existing cognitive structure are most relevant to the new potentially meaningful material", a "reconciliation with existing ideas in cognitive structure" and a "reformulation of the learning material in terms of the idiosyncratic intellectual background and vocabulary of the particular learner" (Ausubel, p. 5).

A now widely accepted principle is that learning goes beyond memorization. Meaningful learning requires learners to actively construct meaning for themselves. But this active process is not merely a function of learner activity, particularly in experimental settings. It is a complex process involving the learner, the teaching materials and resources, the community of learning, the teachers, and the family. Creating *personal* knowledge is, then, a *social* process, a *situated* social process (Brown, Collins, & Duguid, 1989). *Personal* in the sense that it is assumed by each learner. *Social* in the sense that learning depends on constant interactions with others, peers and non-peers. *Situated* in the sense that learning cannot be separated from how it is learned and used:

> The activity in which knowledge is developed and deployed (...) is not separable from or ancillary to cognition. Nor is it neutral. Rather, it is an integral part of what is learned. Situations might be said to co-produce knowledge through activity (Brown, Collins, & Duguid, 1989, p. 32).

This approach is well exemplified in the narratives presented by Roth (1995) about how students construct knowledge in physics classrooms, moving "between various mental representations", including "symbolic mathematical, descriptive, experimental, phenomenal and conceptual forms", through a "complex web of interactions" where they develop ideas through conversations in order to clarify concepts and make connections, assisted by a teacher that assume a role of "graduate student advisor" to support "cognitive apprenticeship"—a term coined by Brown, Collins, & Duguid (1989) to describe cooperative and social learning environments such as these.

Experimentation is part of the true essence of learning physics. But it cannot be confused with discovery learning. As Ogborn (1999) wrote, it should be seen as an opportunity to make ideas "come real", in situated meaningful contexts.

4.3.3 Experimenting with computers

The last quarter of the XX century saw the generalization of the use of computers in almost all human activities. In science, computers are now probably the most ubiquitous tool. But the impact of using computers in science and technology goes beyond its role as a tool. Accordingly to the National Research Council (1989, p. 36), "scientific computation has become so much a part of everyday experience of scientific and engineering practice that it can be considered a third fundamental methodology of science—parallel to the more established paradigms of experimental and theoretical science."

Computers helped to create or develop many new fields of science and technology in the last decades. Complex measurements, such as those made in particle accelerators; the storing and processing of very large quantities of data, such as data from space gathered with telescopes; large scale weather simulations, etc., cannot be envisaged without computers. A somewhat similar impact is under way in school laboratories. New laboratory learning computer-based tools are now available. Complex mechanical devices for laboratory teaching and learning[1] or special purpose teaching equipment (e.g., the famous Atwood machine) are being substituted by sensors, software and computers—real time data from many types of experiments suddenly become accessible to school students.

The impact of scientific computation goes beyond the traditional concept of experimentation, linked to control, action, or measurement on tangible objects. Computers can extend the nature of the concept of experiment to non-tangible objects, not only in research (Galison, 1997) but also in teaching. In research, sophisticated computer software able to make numerical and, or, symbolic computations are now part of the everyday work of many physicists. The software is used to test ideas, make predictions, compute physical properties, analyse images. Mathematical models are the non-tangible objects used as fundamental objects to make new types of experiments on the computer. A similar process can happen in physics education: students can use simple or relatively complex mathematical models to make experiments with them, fundamentally to test ideas and make predictions, such as described in section 4.2.

Making *computer experiments* in education is not a completely new idea. Dorn (1977), considers computer modelling as a type of computational experiment that can motivate and develop the student's intuition, and indeed encourage the student to make "educated guesses" or conjectures about phenomena. What is new is that *new genres of software*, such as modelling software and spreadsheets, make computer experiments available to all students, not only to those who know how to use programming languages. These new genres of software make computational experiments easy enough to be done as part of the physics curriculum, as exemplified in Advancing Physics (Ogborn & Whitehouse, 2000, 2001).

4.3.4 Which computer tools for the physics curriculum?

This thesis is about Modellus and modelling in the curriculum. But it should be clear that I consider Modellus just as one of a (small) set of computer tools that must be used as integral part of a modern physics curriculum. What type of computer tools must, then, be used?

First of all, *data logging software*, used in conjunction with data logging hardware to make measurements, plot graphs in real time, fit models to data, etc. A good example of such tool is *Pasco Science Workshop* or its successor *Data Studio*. This type of tool is used mainly in laboratory activities but can also be used on its own, without measurement hardware, to analyse data.

Second, a *spreadsheet*, such as Excel, is an appropriate and powerful tool to manipulate tables of data. Spreadsheets are particular useful to make data transformations, iterations, plot graphs from tables of data, and also to fit models to data.

---

[1] I remember a popular 8 mm film-loop of late 1970s showing a small car moving linked to a *complex electromechanical device* that could draw real time graphs of position, speed and acceleration...

Third, a *computer algebra system*, such as *Mathcad*, is particularly useful to solve equations, numerically and symbolically, plot graphs from explicit functions, make symbolic computations (such as simplifying expressions, computing derivatives or primitives), and make general computations using variables or just numerical values.

Fourth, *a modelling system*, such as Modellus. Its main purpose is to "make experiments with mathematical objects", namely functions, iterations, differential equations, vectors, etc. A modelling system must allow not only the construction of graphs and tables, as most modelling systems do, but also the construction of animations of mathematical relationships, as Modellus does.

Finally, in certain circumstances, other specific types of software can be useful: specific simulations that illustrate complex behaviour (e.g., wave properties); image analysis tools that allow the student to manipulate computer images and extract information from them; other modelling tools, such as those used to explore cellular automata models.

At first sight it seems to be impossible for a student or a teacher to cope with all these types of software. This was certainly true before the standardization of the software interface in late 1990s. The now common conventions used in almost all types of software allow a quick learning of most of it, in particular if its *concept* is clear for the user, and *relevant examples* are given to introduce it. This means that computer tools can be introduced and used in the curriculum in the context of the problems they are required for and not necessarily after specific training on its interface and use. The introduction and use of computer tools in the curriculum is, then, mainly a process of explaining the *concept* of the software (see section 3.2 for the meaning of *software concept*) and the provision of one or more specific examples that illustrate the concept (Roth, 1995).

This set of computer tools that include measurement, mathematical and modelling tools are part of a "physics student toolkit", a toolkit that has some similarities with the "physicist toolkit" described by Krieger (1987). But this toolkit is not only of invaluable use in physics: it can also be used in other physical and biological sciences (particularly the data logging software) and in mathematics (especially the computer algebra system and the modelling tool).

4.3.5 Physics curriculum and modelling (with some notes on the mathematics curriculum)

There is probably no general accepted definition of curriculum. A useful working definition is the Schwab's (1983) definition, which stresses the importance of "serious reflection and communal decision" in the development of curriculum:

> Curriculum is what is successfully conveyed to differing degrees to different students, by committed teachers using appropriate materials and actions, of legitimated bodies of knowledge, skill, taste, and propensity to act and react, which are chosen for instruction after serious reflection and communal decision by representatives of those involved in the teaching of a specified group of students who are known to the decision makers (p. 240).

The curriculum considers questions related to knowledge, learning, students and motivation. Should knowledge focus on content or process? Should learning activities be structured for the individual students or for groups? Should motivation be essentially intrinsic or essentially extrinsic? The way the teacher handles the essentials of these dilemmas determines the nature of the curriculum (Bransford, Brown & Cocking, 2000).

Modelling in the physics curriculum has the following fundamental assumptions:

1. Observation is increasingly *technology-mediated observation* (Mosterin, 1998). Direct observation through the senses has become an exception, rather than the rule, in science (and also in our daily life, thanks to the media industry). Hacking (1983) pointed out that even though an experimental apparatus is laden with the theory behind the apparatus, observations remain robust despite changes in the theory of the apparatus or in the theory of the phenomenon. Physics learners must be familiar with technology-mediated observation—e.g., with electronic sensors for direct measurement of physical quantities or computer tools to make measurements on images—from the very beginning, even when they are not familiar with the theory subsumed by the apparatus.
2. Knowledge construction in physics depends on the tools learners use. *Acting and thinking with tools* are seen as fundamental to "(...) build an increasingly rich implicit understanding of the world in which they use the tools and of the tools themselves. The understanding, both of the world and of the tool, continually changes as a result of their interaction. Learning and acting are interestingly indistinct, learning being a continuous, life-long process resulting from acting in situations" (Brown, Collins, & Duguid, 1989, p. 34).
3. Physics is about *making models* of the physical world, not about how things are. As Feynman said on his Nobel lecture (1965), "The only true physical description is that describing the experimental meaning of the quantities in the equation—or better, the way the equations are to be used in describing experimental observations."
4. Quantitative thinking, as an essential part of physics, should be taught gradually and explicitly. *Mathematics* must be "used in all aspects of scientific inquiry"—it is "essential to ask and answer questions about the natural world": "observations and investigations should become increasingly quantitative, incorporating the use of computers and conceptual and mathematical models" (National Research Council, 1996).
5. Understanding physics is for the most part a process of *familiarization* with the language, concepts and ideas of physics. Familiarization has been mentioned by famous physicists as an important process in the construction of personal knowledge. Planck's autobiographic (1950) assertion that new revolutionary ideas are accepted because their opponents die and a "new generation grows up familiar with it" or Feynman statement that his graduate "physics students don't understand it" (QED theory—quantum electrodynamics) and that he himself didn't understand it— "I don't understand it. Nobody does"—shows us how difficult it is to make sense of the physical view of the world—a completely different view from common sense. As the Advancing Physics curriculum points out, students must start young in order to *get used to new ideas*—and they will stop worrying sooner...

> Familiarization is, then, an important issue in learning complex ideas of science. But it is more than just *getting used to new ideas*. It should be a process of being familiar with the ideas and *its limitations and contexts*—new ideas can not be taken too literally.

Physics teaching is a challenge to transform a "functional *science sauvage*" (Holton, 1994) into a scientific world view. In a functional *science sauvage* view,

> (...) the facts of nature form a seemingly infinite, atomistic, unconnected set; material bodies come to a stop unless they continue to be propelled; electricity flows through wires as water does through pipes, only much faster; space is a big container in which matter appeared at the beginning of time; time is everywhere the same and marches on inexorably on its own; notions of probability and scaling are minimal; the pattern of cause and effect works most of the time, but unfathomable and magical things do occasionally intervene; science provides truths, but now and then everything previously known turns out to have been entirely wrong, and a revolution is needed to establish the real truth. And so forth (Holton, 1994, pp. 158-159).

On the other hand, a *scientific world view* recognizes science as a process *and* a body of knowledge, a process that creates knowledge, from observation and from thinking, where statements are subject to careful scrutiny by their proponents and by others, and, indirectly, by Nature. It recognizes that science creates and tests explanations about phenomena using observations, experiments, and theoretical and mathematical models, and that scientific ideas are always subject to criticism and change, and can, in principle, be improved (National Research Council, 1996).

Most scientific curricula have not given enough attention to "the nature of mathematical inquiry as a modelling process" (American Association for the Advancement of Science, 1993, p. 335). Even the word *modelling* is relatively new in the physics curriculum. This is probably due to the fact that only in recent decades was there an increase in the degree of abstraction of science (Holton, 1996)—more abstraction means more modelling interpretations and less concrete and tangible interpretations. Holton warns us of the dangers of the lag between the increase of abstraction in science and common practices in education:

> (…) while some time lag between new discoveries and their wider dissemination has always existed, the increase in degree of abstraction, and *in tempo*, of present-day science, coming precisely at a time of inadequate educational effort, has begun to change a lag into a discontinuity (Holton, 1996, p. 56).

Ogborn (1999) presents the "obvious" view of how the curriculum is organized in order to support student progression in theoretical and abstract thinking in science:

> (...) progression of students in theoretical thinking in science has to go through the following foot-dragging and painful sequence of stages:
>
> 1. learn some arithmetic (in elementary school)
> 2. learn, or fail to learn, some algebra (in high school)
> 3. learn some calculus (maybe only at college)
> 4. learn (or not) about finite-difference approximations to calculus
> 5. use computational models of processes (maybe at graduate school)
>
> A good proportion of the whole population drops out at the first stage, and only the very few

> who study mathematics, science or engineering at university get to stage 3 and beyond. Mathematics is generally held to be too difficult for most, and mathematics with calculus and computers is often saved up for university and graduate school, where most of us never arrive. As a result, the population at large has no idea how computers guide space-craft or predict the course of the economy. Worse, the limitations of computational models are hidden from public view; certainly people cannot distinguish those successes and failures of models which depend on the smartness of people and those which depend on the 'smartness' of computers. Mathematics which most of us can't do is needed before one can make models with that mathematics. It all seems so blindingly obvious. *Only experts need apply* (p. 5).

According to Ogborn, this "natural progression" is far from obvious. He argues that when making models, students begin to think mathematically and that students should use "computers to make models and, through that process, learn mathematics, rather than having to learn mathematics in order to get started " (p. 5). The "natural progression" should then be:

> | | |
> |---|---|
> | 1. Make some computer models | learn some arithmetic |
> | 2. Make some more computer models | learn some algebra |
> | 3. Analyze some computer models | learn some calculus |
>
> (...)
>
> The idea is that making models can help make mathematics. Mathematics is *not* needed to make models. But making models *is* learning, in one special way, to 'think mathematics'—and in a way which many people *can* do. *Not only experts need apply* (p. 6).

This progression does not make clear some aspects of the type of mathematical objects used in each step. Some clarifications seem to be needed. So, a possible progression can be:

1	Make some computer models, based on direct computations	learn some arithmetic
2	Make some more computer models, based on functions	learn some algebra
3	Make some more computer models, based on differential equations and iterations	learn some calculus and numerical methods

This proposed shift from a traditional "natural progression" to a new computer-based "natural progression" is far from being consensual, in the present time, not only because of the unavailability of computers for *all* students but also because it is not clear what are the effects of an intensive computer use when learning basic arithmetics and algebra. But some authors, such as Ralston (1999), pointed out that traditional paper and pencil arithmetic (PPA)

> is doomed to relative failure in a world where arithmetic is almost universally done using calculators and where even the dimmest child will see that attaining skill in PPA has almost no value in non-academic pursuits (p. 174).

Ralston argues that PPA must be substituted by machine computation, since PPA is not a useful professional and life skill. He also argues that the important basic skills in computation should be *mental arithmetic, estimation and problem solving*, not the mechanical use of algorithms, something that has never been very successful—Ralston quotes studies that shows that PPA has never been successful in the past, before the advent of calculators and computers. And, more important,

> there is no research evidence—quite the contrary— that calculator use impedes children's understanding of arithmetic or acquisition of later mathematics (Ralston, 1999, p. 178).

The progression described above assumes that *number sense* and *symbol sense* together with *skill in problem solving in meaningful contexts* are the most important learning goals in the school curriculum where computers and modelling are ubiquous. This will take time, due to insufficient hardware availability, inadequate teacher education, school organizational problems, etc., but it can be clearly envisioned as a goal for the future. The need for adequate computer tools—such as those described in subsection 4.3.4 is then evident.

But these goals and the progression described above cannot be envisioned only as structuring the physics curriculum. Other science subjects are also affected. And the mathematics curriculum is *profoundly* affected. Curriculum research, curriculum design, and curriculum implementation in physics cannot be independent of curriculum research, design and implementation in mathematics. They are completely interdependent and a coherent approach is needed, not only in secondary school physics but in college physics. This is now being done in some of the innovative curriculum developed in late 1990s in the UK and in the US, as mentioned above, but also in other countries, such as Portugal, at least in the new mathematics curriculum (the Portuguese mathematics curriculum for secondary schools describes Modellus as "an unavoidable tool in the mathematics curriculum"). This curriculum considers experimental work and mathematical modelling as essential for meaningful learning in mathematics:

> "meaningful learning in mathematics cannot exclude typical characteristics of experimental work, and skills developed through mathematics must give a contribute to lay the foundations of knowledge and ways of thinking about experimental science" (Departamento de Ensino Secundário, 2001, p. 5).

The curriculum also describes the importance of school mathematics laboratories and the use of sensors and data logging software. For an inattentive reader, it may even seem, in some pages, a document describing a physics curriculum. Technology use (calculators, computers, data logging) is mentioned many times as essential curriculum tools. The Portuguese mathematics curriculum reflects a trend of the new educational mathematical thinking: the NCTM Standards for School Mathematics (NCTM, 2000) describes technology and computer tools as an essential component of the mathematics learning environment and considers physical experiments and model building as common mathematical learning activities.

4.4 Coda

This chapter presented some examples of how Modellus can be used to design new physics curricula, in secondary schools but also in undergraduate studies. A fundamental characteristic of such new curricula is the use of computer tools, for different purposes, but with a common vision that can be synthesized as:

1. Mathematical modelling of physical phenomena is an essential part of physics learning and should be taught explicitly, in physics and mathematics, in a coherent and coordinated way;
2. Semi-quantitative thinking, estimation, number sense, and symbol sense are fundamental approaches to make meaningful use of computer tools;
3. Visualization and concrete manipulation of abstract concepts can be enhanced by computer tools, supporting student reasoning and exploration, empowering their knowledge construction, in collaborative guided environments.

It is not the ambition of this thesis to have a complete definition of what such new curricula should be. There is much work to do in the next decades, on theoretical thinking about the nature of the curriculum and learning with computer tools, and on the practical aspects of curriculum design and implementation. But the message seems clear: computer tools and technology are fundamental to support concrete thinking and "concrete thinking is not inferior to, but a valuable and often indispensable alternative to hypothetic-deductive/abstract reasoning" (Roth, 1995).

Chapter 5 Teachers, Modelling and Modellus

> One last word and I am done. I have said nothing about computers, which seems strange in this day and age. I really have nothing to say about them, aside from the fact that I love them and my life would be much more tedious without them. They can be a boon to scientific consciousness and, besides, they have reintroduced the servant in an era when the sages all said we would forevermore he servantless. Best of all, we can construct programs that can “simulate” what we might with great cost and effort do in our heads or on paper, and, in so doing, making us aware of what it is that we must still do ourselves in our own heads (Bruner, 1992, p. 12).

5.1 Introduction

This chapter reports opinions about modelling and Modellus from secondary and university teachers, belonging to different scientific fields and countries. The data were collected through an email questionnaire, answered by 75 subjects, a subset of the 117 that acknowledged a message sent to 239 subjects that contacted me by email, at least once either in 1997 or 1998, asking questions about Modellus. Table 5.1 shows the number of messages sent, mail errors, messages acknowledged and filled questionnaires received by country.

The questionnaire was written in Portuguese for Portuguese speaking subjects, in Spanish for Spanish speaking subjects, and in English for all other subjects.

Table 5.2 summarises the field of work and teaching level of the 75 subjects who answered the questionnaire.

The respondents are, for sure, not representative of the teacher population. They are a subset of teachers and researchers interested in using and creating new tools for teaching and learning. By no means, does the data gathered reflect what teachers, as a whole, think of Modellus. That is why there is no statistical treatment of the answers. For each

Table 5.1 Number of messages sent, mail errors, messages acknowledge and filled questionnaires received

	A	B	C	D	
	Messages sent	Mail errors	Messages acknowledge and questionnaires sent (A)	Questionnaires received (B)	D/C (%)
Portugal	123	11	50	30	60.0
Brazil	48	4	22	13	59.1
UK	11		11	9	81.8
USA	8		7	5	71.4
Colombia	10		5	4	80.0
Spain	5		4	4	100.0
Netherlands	6	2	4	4	100.0
Chile	8		4	2	50.0
Germany	2		2	2	100.0
Canada	1		1	1	100.0
Ecuador	2		1	1	100.0
Mexico	2		0		
Greece	6		3		0.0
Australia	1		1		0.0
Guatemala	1	1	0		
Mozambique	1	1	0		
Italy	1		0		
Paraguay	1		1		0.0
Argentina	2		1		0.0
	239	19	117	75	59.1

question, I looked for answers that could help understand how different teachers and researchers see the assumptions behind Modellus and how they evaluate it according to those assumptions.

5.2 Questionnaire about Modellus and Modelling

The email questionnaire has 11 questions:

- the first four questions ask about the current work of the respondent and his/her familiarity with Modellus;
- questions 5 and 6 asks about more integration between Physics and Mathematics;

Table 5.2 Teaching level and area of the questionnaire respondents

	N	%
High School, Physics/Chemistry	19	25.3
High School, Mathematics	12	16.0
High School, Mathematics/Physics/Chemistry	1	1.3
High School, Technical disciplines	4	5.3
High School, Informatics	2	2.7
University, Physics	13	17.3
University, Mathematics	2	2.7
University, Physics/Mathematics	1	1.3
University, Engineering/Pharmacy	3	4.0
University, Physics Education/Science Education	6	8.0
University, Mathematics Education	8	10.7
University, Educational Research	4	5.3
	75	100.0

- question 7 asks if Modellus can help the reification of abstract objects;
- question 8 asks about the quality of Modellus user interface, in particular for non-computer experts;
- question 9 asks if the respondents have used Modellus with students and how they evaluate that use;
- question 10 asks respondents to evaluate the claim that Modellus can help students make experiments with abstract objects;
- finally, question 11 is an open question where respondents can express opinions about any other aspect, namely potentialities and threats of using Modellus.

These questions were chosen to reflect the most important issues raised by the use of Modellus, in particular:

- the integration between Physics and Mathematics, the reification of abstract objects and the experimentation with this type of objects;
- the readiness of use of Modellus by non-computer experts.

More questions could be asked but it was considered important to keep the time of answer to a minimum, to assure a bigger response rate and to follow the informal rules of email (short messages are considered by most users as part of the Internet etiquette).

The following is a literal transcription of the questionnaire:

`Dear Colleague,`

`Following my previous message, this is the short questionnaire about Modellus I asked you to answer. The estimated time to`

answer is about 20 to 30 minutes. Thanks again for your cooperation.
Please insert your answers after the questions.

1. How do you describe your current work (schoolteacher, university teacher, curriculum developer, educational researcher, student, etc.; use one or more job descriptions, if necessary)?
----

2. If you are involved in teaching physical or mathematical sciences, what subject(s) do you teach currently (e.g. physics, mathematics, chemistry, numerical methods, computing, dynamical systems)?
----

3. If you are involved in teaching, at what level do you teach (e.g., secondary school, college/university)?
----

4. How do you evaluate your familiarity with Modellus (e.g., «I've made my own models»; «I've run models created by others but my knowledge about Modellus is not enough to create or change them»; «I have only looked at it very superficially»; etc.)
----

5. Modellus was designed to help teachers of physical sciences and mathematics to work with their students with a more integrated view of these subjects. Do you think this is a reasonable goal of the teaching of physical sciences and mathematics? If necessary, say why or why not.
----

6. Do you think Modellus can help attain that goal? Why?
----

7. A common problem in some teaching environments is the absence of reification or concretization of abstract objects, such as an oscillator or a parabolic trajectory. One of the main goals of Modellus is to give students and teachers a tool to work with «formal objects» in a concrete way. Do you think this goal can be attained with Modellus?
----

8. Can you evaluate Modellus user interface? Is it friendly enough for students and teachers with a non-expert knowledge to use computers and software?
----

9. Have you ever used Modellus with your students? If your answer is «yes», can you please specify what they have done

```
with it and the way you assess that?
----------------------------------

10. From a traditional point of view, an experiment involves
concrete devices such as pendulums, clocks, sensors, etc. From
a less traditional viewpoint, an experiment is something you
can also do with abstract objects such as particles, vectors,
functions, differential equations, etc., using a computer, for
example. Modellus was created to allow students and teachers
(particularly at secondary school and undergraduate levels) to
make this kind of experiment, without the complexities of
programming languages or of powerful computer systems such as
Mathematica, Mathcad or Matlab. How do you evaluate this claim
about Modellus?
----------------------------------

11. If you have any other comments about Modellus and its
potentialities and threats, please use the space below.
----------------------------------

Please email me this file as soon as possible
(vdt@mail.fct.unl.pt). Thanks again for your co-operation.
Vitor Duarte Teodoro
```

This questionnaire was validated by a panel of three researchers, working in the field of computers in education. They analysed the questions and the goals of the questionnaire; their comments were used to make minor corrections in the text.

The full text of the answers can be read on the Modellus web page (http://phoenix.sce.fct.unl.pt/modellus, following the link “Reviews”). Some of the quoted answers in the following section, originally in Portuguese and Spanish, were translated into English.

5.3 A Reading of the Answers to the Questionnaire

5.3.1 Modellus as a tool for learning

Modellus is the outcome of an original design. But, as all new tools, it has predecessors, like *Dynamic Modelling System* (Ogborn, 1984), and predecessors are clearly recognised in some of the questionnaires’ answers:

> I have much more familiarity with the original Dynamic Modelling System and I see Modellus as an updated version of that [Hugh Wylam, Secondary School, Physics, UK].

A good computer tool must have

> all the essential features and nothing dispensable [Eliane A. Veit, University, Physics, Brazil]

allowing constructivist approaches both to teaching and learning, helping change teaching:

> Modellus is an exemplary constructivist tool. Besides facilitating learning, it can help change teaching [Eliane A. Veit, University, Physics, Brazil].

Modellus is recognised as a tool for learning, both for teachers and students alike, as noted by several respondents:

> (…) writing models with Modellus is teaching me a lot about phasors that a degree in physics never did! [Ian Lawrence, Secondary School, Physics/Curriculum Developer, UK].
>
> I've just finished a course with Modellus for mathematics teachers where most of the examples were from physics. During the course, teachers created models and (…) this helped reveal teachers' difficulties (…) [Verónica Gitirana Gomes Ferreira, University, Mathematics Education, Brazil].

5.3.2 Integrating physics and mathematics

As noted several times in the preceding chapters, one of the main visions behind Modellus is a much closer integration between the teaching of Mathematics and Physics. Questions 5 and 6 asked explicitly about how important is this issue and how Modellus could help in attaining the goal of integrating Science and Mathematics:

```
Question 5:
Modellus was designed to help teachers of physical sciences
and mathematics to work with their students with a more
integrated view of these subjects. Do you think this is a
reasonable goal of the teaching of physical sciences and
mathematics? If necessary, say why or why not.
```

```
Question 6:
Do you think Modellus can help attain that goal? Why?
```

How do questionnaire respondents see this vision and the relevance of Modellus to fulfil it?

All respondents, either with a Physics/Chemistry or with a Mathematics background, agreed with the importance of teaching and learning these subjects in a more integrated way. Several reasons are pointed out, such as a fundamental one—that mathematical reasoning is an essential feature of physics in many contexts:

> Much of science, especially physics, is at the same moment mathematical. In doing theoretical physics, or in analysing an experiment, there simply is no real distinction. Physicists do not so much 'use' mathematics, as 'do mathematical style physics' at these times [Jon Ogborn, University, Science Education].
>
> After all, the sciences are all interconnected, therefore it does not make sense to teach them as they were tight compartments [Branca Silveira, Secondary School, Mathematics, Portugal].
>
> Mathematics is an essential framework for all science teaching and learning [Robert Lewis, University, Education, UK].
>
> (…) The proper role of mathematical modelling is an important part of physics teaching [Philip Britton, Secondary School, Physics, UK].

> (…) One cannot have basic understanding of physical phenomena without the help of mathematics, and, on the other side, physics can give excellent contexts to facilitate the comprehension of many important mathematical concepts [Claudio Pérez Matzen, University, Physics Education, Chile].

Besides this fundamental reason, some answers mention that an integrated approach makes Mathematics and Physics more interesting, motivating and meaningful for students:

> I think that should be a main goal of math and science teaching. The integration of these subjects makes them more meaningful and more interesting for students. Unfortunately this process of teaching these subjects with a more integrated view started only a few years ago in the Netherlands [Jan Rasenberg, University, Physics/Chemistry Education, Netherlands].

Working closely with *formulae* and *physical representations*, an “invented reality”, is considered a unique experience for learners:

> Modellus is one of the few tools that show the real integration of mathematics and physics. The close and dynamic coupling of working with formulae and physical systems gives a unique experience to learners [Ton de Jong, University, Education, Netherlands].
>
> It shows relationship between mathematics and ‘reality’; is a very quick and performant tool for creating animations, especially in kinematics and dynamics [Jean-Pierre Rolland, Secondary School, Physics/Chemistry, Canada].
>
> (…) You construct the formulas and see the effects in a ‘realistic’ simulation. In a practical experiment, or in most other simulation software (i.e., *Interactive Physics*) there is no direct link to the theories and formulas [Elwin Savelsbergh, Secondary School, Physics, Germany].

Using Mathematics to analyse physical phenomena or solve physics problems helps students distinguish between “solutions” and “specific solutions”:

> In many calculus-based courses students are often not able to distinguish between the really important ‘equations’ and the huge number of specific solutions for specific phenomena. The limited number of essential laws, which is necessary to work out solutions, determines the structure of a domain. Familiarity with them helps to make crosslinks between domains. Also, System Dynamics methods can be applied in a variety of domains outside physics [Horst P. Schecker, University, Physics Education, Germany].

The learning of some important concepts in Mathematics, such as the derivative, can benefit from this integration, since they usually use Physics as starting points:

> There are mathematical concepts, such as the derivative, that usually appeal to physics (…). So, this integration can only facilitate the learning of these two subjects [Francisco Timóteo, Secondary School, Physics/Chemistry, Portugal].

The integration between Physics and Mathematics is also considered useful for the development of *conceptual understanding* in Mathematics, as mentioned by several respondents:

> Interdisciplinarity between Physics and Mathematics helps the student wake up to more qualitative ideas, to conceptual understanding, of mathematical objects and its properties. Understanding, for example, the sine function (its amplitude, period and frequency) turns out to be much easier when working with physical phenomena that can be modelled by this type of function [Verónica Gitirana Gomes Ferreira, University, Mathematics Education, Brazil].

Using Modellus one can model many complex phenomena with numerical solutions, based more on physical reasoning than on analytical solutions of differential equations, which need advanced Calculus knowledge to be found. This important feature of Modellus is clearly identified by some respondents:

> A problem that physics teachers often found is that students don't know enough mathematics to solve many physics problems; this is essentially due to the lack of knowledge of analytical and numerical methods to solve differential equations. This compels the teacher to present only elementary models, or to introduce drastic approximations to allow students to be able to solve them. (…) In this sense, Modellus gives the solution of physics models without the knowledge of advanced methods [Miguel Angel González Rebollo, University, Physics, Spain].

5.3.3 Working with "formal objects" in a concrete way

As mentioned in the preceding chapters, creating and manipulating concrete and directly manipulable representations of "formal objects", such as functions, vectors, graphs, differential equations, etc., is perhaps the most important goal of Modellus. The questionnaire asked explicitly about how successful Modellus was about this issue:

```
Question 7:
A common problem in some teaching environments is the absence
of reification or concretization of abstract objects, such as
an oscillator or a parabolic trajectory. One of the main goals
of Modellus is to give students and teachers a tool to work
with 'formal objects' in a concrete way. Do you think this
goal can be attained with Modellus?
```

Most users agreed, more or less explicitly that Modellus was an important contribution to help students reify abstract objects:

> YES! (…) So far, as I can see this now, Modellus is the best tool available (especially for secondary education) to attain that goal [Jan Rasenberg, University, Physics/Chemistry Education, Netherlands].

> Absolutely [Ton de Jong, University, Education, Netherlands].

> It helps develop reasoning and abstract skills, and to create an observational attitude and analysis skills before experimentation [Elisa Maria Vaz Gomes Figueira, Secondary School, Mathematics, Portugal].

One answer noted how Modellus symbolic objects can be seen as "formal reasoning" objects:

> I think this is exactly correct. For me, 'formal reasoning' (Piaget sense) is precisely concrete reasoning (Piaget sense) with symbolic objects (Modellus sense) [Jon Ogborn, University, Science Education].

Some respondents recognised reification as a particular difficult problem, impossible to be easily solved and needing "many different tools" to overcome:

> Not by itself. It takes many different tools to attack this particularly difficult problem [Robert Beichner, University, Physics Education, US].

Other respondents noted that it is not necessary to use models if one can use “the real thing”. But even they agree that some objects can only be shown with computer models:

> I find it not necessary to use models if you can show things in real. But indeed there are subjects that you can only show on the blackboard or in a computer model. In that case it is really good to have that kind of models [Raoul Majewski, Secondary School, Physics, Netherlands].

There are also some respondents that point out that it is important to concretise formal objects, but this cannot be done at the expense of destroying students’ abstract skills or the relevance of ‘formal objects’ in the development of Science:

> Completely, since one do not destroy students’ abstract skills, or the value of ‘formal objects’ in Science [Bernardo Brotas Carvalho, University, Physics, Portugal].

One must also be aware not to overestimate the importance of graphical instruments (such as Modellus animations), since it is more important to concentrate on the model itself:

> Working with the graphical visualisation facilities of Modellus complements the display of results in graphs and tables. However, I do not overestimate the value of the graphical instruments. In most cases I concentrate on the model itself and the output in a graph [Horst P. Schecker, University, Physics Education, Germany].

Reification using Modellus objects is also considered relevant for creating “mental images” that can help conceptual understanding of mathematical models, since students can use Modellus’ *metaphorical objects* to do qualitative explorations and study cause-effect relationships:

> The ‘objects’ given by the software allow the simulation of physical phenomena, surely given “concretization” to problems. The possibility of manipulating or animating the ‘metaphorical objects’, allow qualitative explorations (…). The possibility of manipulating the conditions of the model and the observation of cause-effect relationships, make propitious the construction of mental images that give more meaning, even comprehension, to the mathematical model [NN, University, Mathematics Education, Brazil].

“Learning visually”, as opposite to learning from “equations” was also mentioned by other respondent:

> The potentialities of Modellus in my field are particularly interesting since I can help students visualise the evolution of chemical concentrations in time, allowing better learning when compared with the rude and less motivating use of differential equations [Amilcar Falcão, University, Biomathematics, Portugal].

An interesting point raised by one respondent is “persuasion”: Modellus concretization of ‘formal objects’ can help *persuade* students about the validity of these objects to represent natural phenomena:

> I think that with Modellus one can deal with physical phenomena close to reality, and this helps ‘persuade’ individuals that they are building their knowledge [José Gabriel Evangelista Silva, Secondary School, Mathematics, Brazil].

5.3.4 How good is the user interface of Modellus?

Creating new software in now much more easily done than it used to be a decade ago. Graphical user interfaces are now common and most users know the conventions used (pull-down menus, buttons, dialogue boxes, direct manipulation, etc.). Modellus conforms to most of these conventions but since it is unique in some aspects it could be confusing for some users. Question 8 asked explicitly for an evaluation of Modellus's user interface, in particular for those users who are not computer experts:

```
Question 8:
Can you evaluate Modellus user interface? Is it friendly
enough for students and teachers with a non-expert knowledge
to use computers and software?
```

Most respondents praised the simplicity and usability of the user interface:

> A very good interface. It is user-friendly [Patrick T. Tam, University, Physics, US].
>
> It is easy to use, also for people who have never worked with it [Raoul Majewski, Secondary School, Physics, Netherlands].
>
> Very easy to use. I tested it (…) with an 'A' level student who had never seen Modellus before. Within 15 minutes, he had created a simple model of the motion of an electron in a cross E field [Bob Cooke, University, Physics/Mathematics, UK].
>
> The interface is quite user-friendly. It would be practically impossible to do it more simple, without losing important resources [João Goedert, University, Physics, Brazil].
>
> I note that non-computer literate students became efficient enough in less then 1 hour [Jean-Pierre Rolland, Secondary School, Physics/Chemistry, Canada].
>
> My pupils have taken about two one hour lessons to feel confident in altering and proposing models [Philip Britton, Secondary School, Physics, UK].
>
> Surely the interface is much more user-friendly than the ones from other software like Matlab; Modellus, on the contrary, inspires the user to use his intuition [Thiago Roland Belard Jr., University, Physics, Brazil].
>
> This should not pose serious problems; learners are increasingly becoming used to a variety of interfaces; this one is simpler and more intuitive than many [Robert Lewis, University, Education, UK].

Not all respondents said that users agreed with the simplicity of the interface:

> Some teachers with little computer knowledge managed to build examples with the program, but others considered it very complex [Izôlda Maia, Secondary School, Informatics Education, Brazil].

Some presented criticisms, such as not enough integration with MS Windows and a certain difficulty to attain enough level of sophistication:

> It could be more user-friendly, perhaps with a much more closed integration with MS Windows [Nuno André Catarino, University student, Mechanical Engineering, Portugal].
>
> The interface is user-friendly and allows progressive learning of the different capabilities of the program, but it is not easy to attain a certain level of model sophistication… Time and work are necessary.

> For those who are not computer experts, especially in this kind of software (I have teacher training workshops with Modellus as reference) it is not easy to start using the program [António Bernardes, Secondary School, Mathematics, Portugal].

One respondent noted that it is important to educate students to be always aware of what they are doing, preventing them to get lost on the screen:

> You've to get used to all the numbers and things you can change on the interface. Take care students don't get lost in all the things there are on the screen [C. Vreman, University, Educational Technology, Netherlands].

Another respondent pointed out that non-computer experts need to be convinced of the benefits of Modellus before getting involved:

> Not sure. I think you probably need to be convinced of the benefits before getting involved if you are not a computer-phile [Hugh Wylam, Secondary School, Physics, UK].

An interesting point raised by some respondents is that, at the beginning, Modellus can be seen as difficult and strange. But, with some practice, it is software that *becomes* user-friendly:

> I think that at a first contact, for those who do not have lots of practice with computers, the diversity of windows can seen awkward. After this first impression, and with a little more practice, the interface looks user-friendly and flexible [Adelina Precatado, Secondary School, Mathematics, Portugal].

> I can tell my own experience. When I had my first contact with the software, I was familiar with other programs for teaching mathematics (Logo, Cabri, Sketchpad, Graphmatica, Imagiciel). I must confess that, in the beginning, I saw the screen too much 'polluted' with windows: model window, graph window, animation window, initial conditions window… I had the manual with me, and after doing the 'guided visit', I felt quite at ease with the program (…).
>
> After being able to command the software, the impression of 'pollution' become abundance of possibilities (…). I can say that it is a software that becomes user-friendly as soon as one is more and more familiar with it [NN, University, Mathematics Education, Brazil].

> At first, it seems to cause some adverse reactions, with judgements that assume difficulty of use. Nevertheless, after exploring a few examples, students manage to overcome these difficulties [António Domingos, University, Mathematics Education, Portugal].

One of the most important points taken into consideration in the design of Modellus was that expressions should be written on the Model Window as they would be written on paper, avoiding complex syntax rules typical of programming environments. This point is clearly identified by several respondents as one of the most successful aspect of Modellus:

> Modellus user interface is quite user-friendly, but the most important aspect I see is that expressions are written as on paper or on the blackboard. The fact that the student doesn't need to know how to program is important—this is a necessary condition for a program like Modellus but not sufficient. Other programs also show this characteristic: the great difference to the credit of Modellus is that it can be used to help the student understand the language of mathematics, in the same way as the teacher or the scientists usually use it [Eliane A. Veit, University, Physics, Brazil].

> Modellus allows the use of mathematics and objects to make interactive graphical representations with a minimum of effort, since it is not necessary to know any programming language to be able to do that [Jose M. Zamarro, University, Physics, Spain].

Finally, one respondent raised the issue of "generation". Teachers tend to have more difficulties with novelties, like computers, while students see computers not as novelties but just as normal tools of their own generation:

> I think it is more user-friendly to students than teachers, but this is a problem of generation! [Cremilde Ribeiro, Secondary School, Physics/Chemistry, Portugal].

5.3.5 Students using Modellus

Modellus has been used by some respondents with students, from high school to college and student-teachers. Question nine asked what kind of activities respondents have done with students and how they have assessed that:

```
Question 9
Have you ever used Modellus with your students? If your answer
is «yes», can you please specify what they have done with it
and the way you assess that?
```

One of the respondents who regularly uses Modellus with students of different courses wrote:

> I have used Modellus in three types of courses:
>
> - Secondary teacher training courses. Practically, teachers only used ready-made models. Although they were delighted with the capabilities of Modellus, they didn't have the initiative to create models, even after being encouraged to do so.
> - Initial teacher training courses. Students were very receptive. They have done different examples of motion (…).
> - Engineering courses. We are creating a tutorial course for General Physics (Mechanics). The text goes with modellus files of different types (illustration of concepts, simulations, conceptual experiments, modelling). During the first application of this course, receptiveness was very high and, no doubt, Modellus facilitated learning. Student evaluation of Modellus was very positive. Nevertheless, students created very few models. Perhaps the major reason why this happened was that the examination didn't used the computer, since it was a classical Halliday and Resnick problems, similar to those classes that didn't use Modellus. Another limitative factor was that the course included many topics and it was not possible to spend much time on each topic [Eliane A. Veit, University, Physics, Brazil].

The difficulty of making students and teachers create their own models, instead of just running ready-made models, is also mentioned by other respondents:

> The biggest difficulty faced by the teacher is to make the student create with the program, as opposed to the passive attitude of using what is ready. As a matter of fact, this also happens with many of our colleagues [Paulo Machado Mors, University, Physics, Brazil].

> Since I've been working in teacher training, many times I get worried since teachers, notwithstanding being delighted to build models, they do not allow their students to use Modellus to explore physical and mathematical concepts. Indeed, they use models as ready-made simulations where the student only sees the model as part of a teacher presentation, freeing students from the opportunity to investigate concepts and create their

> own models [Verónica Gitirana Gomes Ferreira, University, Mathematics Education, Brazil].

And, when teachers create models, these models can be too much simplistic, at first:

> I asked teachers to evaluate the software and to build a model. They considered the software to be a good one, understanding reasonably well what its potential in the teaching of mathematics was. But the models they built were too simple [Edla Maria Ramos, University, Mathematics Education, Brazil].

Other respondents also noted that teachers tend to use only ready-made models, planning to change in the direction of students building their own models in the future:

> Several models have worked very well as directed simulation assignments. Students are able to navigate very easily through the multiple representations I create to have them investigate, for example, ranges of collisions, damped oscillation, and driven oscillator. I have not yet assigned students the task of creating models, but intend to [Jay P. Kopp, University, Physics, US].

This is perhaps a common trend. At first, students use prepared models, analysing and making changes on them, and after becoming familiar with the software, they can also start creating models:

> First I let students study some physics concepts with prepared models. Second, I let them change/adapt some models. Third, I asked them to develop some (simple) models by themselves. Some students are using modellus now for more advanced research and make more complex models [Jan Rasenberg, University, Physics/Chemistry Education, Netherlands].

Other respondent pointed out that Modellus use is compatible with graphical calculators, which are now required by many courses:

> I used it in due course, studying quadratic functions in the 10th grade and solving problems involving rational functions, in the 11th grade, with the same students. I evaluate it very positively (…). The fact that the use of Modellus can be articulated with the use of graphical calculators is an advantage. When solving problems, in which the students find the model, they can continue to think about it and explore new aspects, either at home or in following lesson, without computers, with the help of the graphical calculator [Adelina Precatado, Secondary School, Mathematics, Portugal].

Indeed, for Modellus to be successfully used the implementation of significant changes in a course is necessary, as one of the respondents pointed out, after using Modellus for a year:

> After one year of experimentation with some students, I completely redesigned a course of kinematics that fully integrate Modellus. In this course, Modellus is used for: presenting concepts, experimenting on concepts, testing hypothesis, complementing real labs, substituting real labs, creating models (by students). (…) This new course was in force (but in 'beta-testing') since December 1998… and the student's response looks good (wait & see)! [Jean-Pierre Rolland, Secondary School, Physics/Chemistry, Canada].

A typical problem with using Modellus with students is the unavailability of the necessary organisational conditions, such as laboratories with enough computers:

> I've used Modellus with students, and I haven't used it as many times as I would like, because my school does not have the necessary conditions [António Gomes, Secondary School, Mathematics, Portugal].

One respondent noted how useful Modellus can be to help foster advanced aspects of Mathematics, both for teachers and students:

> I've directed the activities to focus on mathematical applications, and not simply of mathematics by its own sake, since this seems to be the most important aspect of the use of the program. This has been very interesting because it show student, future teachers, that although they will not use differential equations in high school, this knowledge can help them explore real phenomena qualitatively, through simulations, differently from those 'fabricated' mathematical applications without any reality (of the type intellectual skill described by quadratic functions, as I have seen in schoolbooks!). In this sense, the software brings new possibilities to the mathematical content in school, topics that would be impossible to study without it [NN, University, Mathematics Education, Brazil].

And can any specific point of didactic interactions be identified? One respondent pointed how a similar tool fostered "better discussion":

> I have used it [Modellus] with colleagues, and I have used similar modelling tools with students. I use them mainly to build iterative models. In a Ph.D. research we had evidence that using a similar tool to Modellus (though not as good), led students to better discussions and made them select much more often a question (not mentioning models) about the value of differential equations, in a national exam where they had free choice [Jon Ogborn, University, Science Education].

Some respondents also acknowledge how students were pleased and enthusiastic when using the program:

> Students reacted very well, they interacted with the program and their peers, either members of the same group or with other members of the class, they learned and, in general, they considered the work as very positive and enjoyable. Some expressed this as 'This way we learn!', 'We should use the computer in physics lessons more often' [Margarida Afonso, Secondary School, Physics/Chemistry, Portugal].

For certain types of students, particularly college students studying science and engineering courses, Modellus was clearly conceived as a transition tool to professional tools:

> The big question is whether students will learn any skills from using it which are useful to them as in learning Matlab and the like. Our plan is to have our courses above the introductory level introduce Matlab. A question will be whether familiarity with the user-friendly Modellus eases the transition into a more powerful solver. I don't really know how to evaluate this, but would be very interested in any evidence or studies elsewhere [Jay P. Kopp, University, Physics, US].

As noted by another respondent, a useful aspect of using Modellus is that it can help teach more in less time, using less class time:

> In general, I think Modellus is a very wise program to teach physics, being compatible with current trends (…) (less class time and more personal time at home and in libraries). Teachers need to explain *more in less time*, and so the need to use supporting tools becomes indispensable [Jerónimo Vida Manzano, University, Physics, Spain].

5.3.6 Making conceptual experiments

As the concept of "conceptual experiment" is fundamental to understand how students and teachers can use Modellus, I was particularly interested in looking at what respondents thought about it. Question ten, after a short introduction, without mentioning the word “conceptual” *explicitly*, asked what respondents thought about making experiments with “abstract objects”:

```
Question 10:
From a traditional point of view, an experiment involves
concrete devices such as pendulums, clocks, sensors, etc. From
a less traditional viewpoint, an experiment is something you
can also do with abstract objects such as particles, vectors,
functions, differential equations, etc., using a computer, for
example. Modellus was created to allow students and teachers
(particularly at secondary school and undergraduate levels) to
make this kind of experiment, without the complexities of
programming languages or of powerful computer systems such as
Mathematica, Mathcad or Matlab. How do you evaluate this claim
about Modellus?
```

There was not a clear agreement about using the word “experiment” in this context, in spite of the fact that more than half of the answers clearly supported the idea that creating models with Modellus can be, in certain circumstances, considered as experiments:

> Effectively, I (really the students!) use intensively Modellus for experimenting with VECTORS and FUNCTIONS, and because of this, Modellus seems a unique tool! [Jean-Pierre Rolland, Secondary School, Physics/Chemistry, Canada]

> Yes, it is a very good ‘experimental interface’ in this sense. The pupils really do say ‘I wonder what would happen if...?’ [Philip Britton, Secondary School, Physics, UK].

> Modellus is intuitive to use and once the students have been persuaded that ‘an experiment’ need not involve, necessarily, physical equipment it is an ideal tool. One can vary parameters that cannot be altered ‘in real life’ e.g., what’s the effect of changing ‘g’ on the trajectory of a particle? This can expand the way in which a student views the problem in a way not possible in a traditional lab [Bob Cooke, University, Physics/Mathematics, UK].

> Modellus is by far and away the best program that I have seen for attempting this important objective [Hugh Wylam, Secondary School, Physics, UK].

On the contrary, one respondent was very clear, expressing the idea that ‘computer experiments’ are not true experiments:

> They can make some ‘experiments’. The results can help their visualisation and intuition. But, computer ‘experiments’ are not experiments [Patrick T. Tam, University, Physics, US].

The word “experiment” is considered a doubtful word for the idea expressed in the question. One respondent, with a considerable background of research in the field, prefers to say that Modellus “experiments” are just idealisations:

> I am not convinced that the term ‘experiment’ is a good one. But there certainly are ‘theoretical experiments’ where one does not know beforehand what the result will be—even in research. I think that the Modellus ‘experiments’ are idealisations, and that it is important to know what an ideal case will do or not do [Jon Ogborn, University, Science Education].

But another respondent, who also has a considerable background of research in the field, admits the validity of the word “experiment”, specifying that Modellus experiments are “experiments with ideas”:

> The claim can be fulfilled much easily with Modellus than with Matlab or other computer algebra tools. The 'overhead' of such systems is too big. I would clearly prefer Modellus as a tool 'to experiment with ideas', as I call it [Horst P. Schecker, University, Physics Education, Germany].

"Concrete" versus "thought experiments" are both considered as vital and indispensable, being Modellus particularly useful for "thought experiments":

> Notwithstanding my idea that concrete experiments (…) are vital, I also think that 'thought experiments' are indispensable. In this regard, Modellus can play an important role and it has undeniable superiority (…) at least for secondary education [Adelina Precatado, Secondary School, Mathematics, Portugal].

5.3.7 New features suggested by users

Question eleven was an open question where respondents could write about any other issue or specifically about potentialities and threats of the use of Modellus:

```
Question 11:
If you have any other comments about Modellus and its
potentialities and threats, please use the space below.
```

Some respondent pointed out the importance of comparing real data with models, what can easily be done with Modellus, not only with graphs but also with many types of images from experiments:

> The best use that I can foresee is combining real world data collection with modeling. Is it possible to input real data (from a sonic ranger, for example) into Modellus for comparison? [Robert Beichner, University, Physics Education, US].

> Might useful understanding be obtained from getting a data table to drive an animation—then you could use the animating power to compare the behaviour of the mathematics and the (data logged) data? [Ian Lawrence, Secondary School, Physics/Curriculum Developer, UK].

This issue of comparing real data with models was also raised by another respondent, considering that a statistical module/data fitting tool could be very useful:

> I regret that Modellus doesn't have a statistic module, to analyse experimental data and to make curve fitting (regression); this would allow students to link and relate those "traditional" experiments and "thought" experiments much more closely as well as discuss a more complete view of mathematical modelling with students [Adelina Precatado, Secondary School, Mathematics, Portugal].

Introducing new features, such as the possibility of having more multimedia features and the capability of being integrated with an Internet browser, were also raised:

> If possible, I think it would be excellent to add multimedia features, but only if these additions do not complicate the interface and the use of the program [Claudio Pérez Matzen, University, Physics Education, Chile].

> (…) What about integrate Modellus with an Internet browser? This (…) would extend a lot Modellus potentialities, since distance learning tends to assume a decisive role [João Goedert, University, Physics, Brazil].

But “keeping it simple” is mentioned by a few respondents; it is important to add new features, but the simplicity of use must be a priority, specially when compared with professional tools:

> (…) But keep it simple to use... don’t make a ‘heavy tool’ of it... for more advanced research people can use other tools (…) [Jan Rasenberg, University, Physics/Chemistry Education, Netherlands].

One “heavy user” made a detailed list of the new features he would like to see in the software:

> (...) When we have the Moon, we want the Sun! It's always possible to add something, but please, keep in mind the SIMPLICITY (and effectiveness) of the actual version (1.0). Here is my grocery's list:
>
> — in programming language, if... then... ELSE; for... next; an INTEGER function; enhance the ARCTAN function so it returns a value in a 0 - 360 or 0 – 2 pi format;
>
> — in an Animation Window: the possibility to manage objects in layers; possibility to control decimal places individually, for each Digital Meter or new Plotter; possibility to adjust co-ordinates of a point of a Geometric Object, simply by dragging it with the mouse;
>
> — some new features, like: a simple Drawing Tool, for drawing lines, boxes, ellipses; a Check Box; a combo Check Boxes;
>
> — in a Graph Window: possibility to control for each axs; decimal places; exponential threshold; possibility to hide variables; possibility to write a Title;
>
> — in a Table window: possibility to control decimal places for each variable; possibility to hide variables;
>
> — have little bugs repaired: the Grid Spacing is not kept when saving; the possibility to work with DATA, and not only with functions;
>
> — the possibility to have a New Text Window; the possibility to name a window (Plot, Animation or Graph); the possibility to manage the windows in layers... and a Millennium Gift: a French Interface!
> [Jean-Pierre Rolland, Secondary School, Physics/Chemistry, Canada]

5.4 Coda

The previous section presented some relevant comments about Modellus and its assumptions, made by teachers (secondary and university) and educational researchers, from different scientific areas and countries.

As one respondent pointed out, the first fundamental idea about Modellus is that it cannot be considered as an alternative to laboratory experiments:

> (…) Software like Modellus cannot be seen as an alternative to laboratory experiments, but as a complementary tool for teachers and students analyse mathematical models of physical phenomena [António Luís Ferreira, University, Physics, Portugal].

The second fundamental idea about Modellus is that there are many ways of integrating it into the curriculum. From “heavy use”, implying a *complete restructuring* of the course, to casual use only to illustrate one or more specific points. To use Modellus in more than casual contexts, it is necessary to have at least three conditions:

1. Organisational conditions like computer laboratories;
2. A good command of the software;
3. Written materials to support student work in groups or individually.

Is all this effort worthwhile? As I mentioned in chapter 1, many innovations seem to be worthwhile at a certain time, but years later they are completely abandoned. However, since it is now clear that computer tools are indispensable tools in knowledge production, in almost all fields, we can be confident that computers will tend to be more and more used in schools.

The respondents cannot be considered as representative of all teachers and researchers. They are innovators, familiar with computers in science and mathematics teaching, interested in analysing and creating new ways of teaching as well as new tools. In general, their answers clearly support the ideas behind Modellus. But, as one respondent pointed out, Modellus is "potentially extremely useful"—and I emphasise "potentially":

> I think Modellus is potentially extremely useful in science teaching at school and university level. [Jon Ogborn, University, Science Education]

Like all tools, there are many ways of using Modellus. The use of the tool can easily be disseminated but the assumptions behind it are much more difficult to disseminate, as the teachers and researchers who have been involved with learning with computers know, at least since Papert and Logo. One respondent explicitly pointed out how important teacher training is:

> I think Modellus cannot be satisfactorily explored without persistence in teacher training (…) starting in college [Luís Jorge Morais, Secondary School, Mathematics, Portugal].

Chapter 6 Students Using Modellus

> The student should run the computer, not the other way round (Redish & Wilson, 1993).

6.1 Introduction

This chapter describes two interventions using Modellus, one with 11th grade secondary school students and the other with college students from the Faculty of Sciences and Technology of Nova University, Lisbon. In the first study, twelve secondary school students, participating in a summer course, used Modellus with an exploratory-guided approach, making models from real data or from pure theoretical thinking. In the second study, ten students in their second year of a Bachelor of Science course (future teachers of Biology and Geology) used Modellus also with an exploratory-guided approach for three days.

As mentioned in previous chapters, most teachers tend to use Modellus with models made by them—students only run the models and analyse and discuss what they see on the screen. This was not the approach adopted in both interventions, since I was interested in getting data about how difficult it was for students to create and explore their own models. The data obtained clearly supports the assumption that students, even with relatively low knowledge of computer software, can easily start using Modellus, creating their own models and animations, with minor difficulties. Their main difficulties seem to be related to their knowledge of physics and mathematics, not to Modellus itself.

6.2 Some Methodological Reflections

Choosing a research methodology is not an easy task. The “two classical paradigms” (Husén, 1998) come from different traditions and serve different purposes. According to

Husén, the choice of paradigm is "determined by what kind of knowledge one is searching for". What kind of knowledge am I searching for with this thesis?

First of all, I'm interested in *creating a new tool* for learning, a tool with a sound basis both on *educational research* and on *computer science*, particularly computer software interface design. The arguments for this goal were presented in the chapters 2, 3 and 4.

Secondly, I'm interested in having reliable information *if students can use the tool*, as creators, not just as simply loading and running examples made by teachers or curriculum developers.

Thirdly, I'm also interested in getting information about *differences students identify* when they learn physics with and without the tool.

Finally, I'm searching for justified *opinions from experienced teachers* and educational researchers—those with "connoisseurship" (Eisner, 1994) of computer tools—about how the tool fits its purposes. This was done in Chapter 5, on a basis of a written questionnaire answered by 75 teachers and researchers from 19 countries.

This chapter refers to goals two and three. As discussed in sub-section 1.4.3, I was not interested in showing that using Modellus to teach physics is "better" than using "traditional instructional methods", whatever "better" and "traditional instructional methods" means. Thus, there was no reason to make experimental or quasi-experimental studies comparing students using Modellus to others not using it in order to measure any superiority of a "new teaching method". As argued by Solomon (1994), using a computer tool is not a *teaching method.* The same tool can support multiple teaching methods, ranging from expository methods to guided discovery and pure discovery.

The studies described in this chapter can be considered in the framework of the new methods of cognitive psychology (Gardner, 1987). These methods are framed on the qualitative research tradition (Gall, Borg & Gall, 1996) and their goal is to investigate the *processes* and the *mental structures used by individuals in different situations*. Accordingly to Olson & Torrance (1996), this framework has more recently an increased concern with what educators can "understand" and "share" with learners:

> Even the more objective "cognitive processes" have yielded pride of place to a concern with children's beliefs, goals, plans, and values and the ways in which educators can understand, address, and ultimately, share them (p. 5).

These studies are an effort to understand *if* and *how* students can use Modellus. This understanding is fundamental to the purpose of any knowledge arrived at in educational research: "to provide a basis for action, be it policy or methods of teaching in the classroom" (Husén, 1998). However, it should be clear that these studies are only a very small fraction of what can be done in order to understand how students (and teachers) can use Modellus. To attain that in a satisfactory way it is necessary a larger research effort, with much more students from different backgrounds and settings and in more real contexts of learning.

6.3 Pilot Work

The two interventions related in this chapter had been preceded by many observations of students using exploratory software, developed in late 80s and early 90s, both in classrooms and individually. Some of these observations have been reported briefly elsewhere (Teodoro, 1992), but most have never been reported.

For example, during the academic year of 1990/91, I observed a class of 10th grade students using *Newton* (Teodoro, 1993) for about ten weeks. *Newton* is an exploratory environment for learning dynamics and kinematics (described in chapter 3). The activities were careful designed to guide the students. Class records and tests showed that students had difficulties making sense of what they saw on the screen, in particular stroboscopic representations of motion and the links (or absence of links) between graphs and trajectories.

In 1991/92, I taught for two months two classes of 11th graders. Students used *Newton* and *Dinamix*, a MS-DOS modelling system developed at FCTUNL and published by the Portuguese Ministry of Education (Lobo et al., 1993). Students were able to use both programs, in inquiry contexts about motion, working in small groups. Seven of the eight groups were skilled enough to use the software to predict the range of a projectile, previous to an experimental activity, were they used metallic tracks and small spheres to check their own predictions.

At the time these observations were made most students were unfamiliar with personal computers. For most it was their first experience with PCs. It was difficult for them to manage multiple windows, understand conventional actions with buttons (such as multiple click or click and drag on the Macintosh, in the case of the students who used *Newton*), etc. This background knowledge about computers is not comparable with the background knowledge current students have since in the late 90s the graphical windows interface become almost common knowledge for school and college students, and for most professionals.

Pilot work was particularly useful for testing the ability of students to combine the use of written materials with exploratory learning on the computer. The typical learning environment involved the introduction to a specific topic, conceptually and experimentally, *without the use of computers*. After this introduction, students used the software, following structured worksheets, with an exploratory approach. This means that they were typically asked to:

1. discuss actions before using the software;
2. predict observations;
3. compare predictions and observations;
4. draw sketches of way they observe on the screen;
5. make “what-if” investigations about parameters and models;
6. write conclusions.

It was not an easy task for students to combine the use of the software and write and discuss what they were doing. Typically, they tended to use the software in a low reflexive approach. This was taken in special consideration in the following studies, where the discussion of the activities took the most important fraction of the time used for each topic.

6.4 First Study: Modellus Used by Secondary School Students

6.4.1 Subjects, research questions and description of the intervention

This study was done with twelve secondary school students (3 girls and 9 boys), who had finished 11th grade, gathered together in the College of Sciences and Technology for a week (five half-days), working in pairs in a computer lab. These students answered by email a call for participation in a Summer Course on *Modelling with Computers*, published in a well-known weekly newspaper[1]. Most students had good or reasonable knowledge about how to use a computer, but only about half of them considered their knowledge of physics and mathematics good or reasonable (detailed data are synthesized on the Appendix).

The research questions for this study were:

- — Can students create their own models and animations?
- — What advantages and disadvantages do students identify when using Modellus to learn simple mathematical models of motion?

The course programme included:

1. An introduction to data logging, using motion sensors.
2. An introduction to Modellus (writing functions for the position coordinates of a particle; domain and range of a function; making graphs and tables; animating the motion of a particle; use of “if... then” clauses for piecewise-defined functions; data-analysis with the graphical analysis tools of Modellus). This introduction was spread over the first two days.
3. Modelling constant linear motion with linear functions.
4. Modelling accelerated linear motion with quadratic functions.
5. Modelling oscillatory motion with trigonometric functions.

After the introduction to data logging and real time graphs, students experimented different types of motion and observed the corresponding graphs with position-time and of velocity-time, obtained from their own motion in front of the sensor or from the

---

[1] The course was possible thanks to “Ciência Viva”, a project from the Ministry of Science and Technology to promote science and technology among young students.

motion of the Pasco fan cart[1]. The fan cart was particularly useful for obtaining constant acceleration motion.

The students worked from simple to more complex models, such as linear functions for representing constant velocity motion and quadratic and trigonometric functions to model constant acceleration and oscillatory motions. Since some students were unfamiliar with trigonometric functions, they were given a short introduction to the rotation of a radius, the graphical representation of the coordinates of its extreme point and the definition of the sine and co-sine functions.

The typical approach followed for each topic included:

1. a short introduction for the entire group using a screen projector;
2. a discussion of what the students were asked to do;
3. work in pairs, taking notes when necessary and getting specific help if needed;
4. final discussion, with each group presenting and confronting results and conclusions.

In addition to the exploration of abstract mathematical models, in order to become familiar with the parameters of the linear, quadratic and trigonometric functions, students used real data obtained with the motion sensor and stroboscopic images (Figure 6.1). They were asked to take measurements, using Modellus measuring tools, find parameters, define functions, and compare the models with the data superimposing the model graph or the motion on the image.

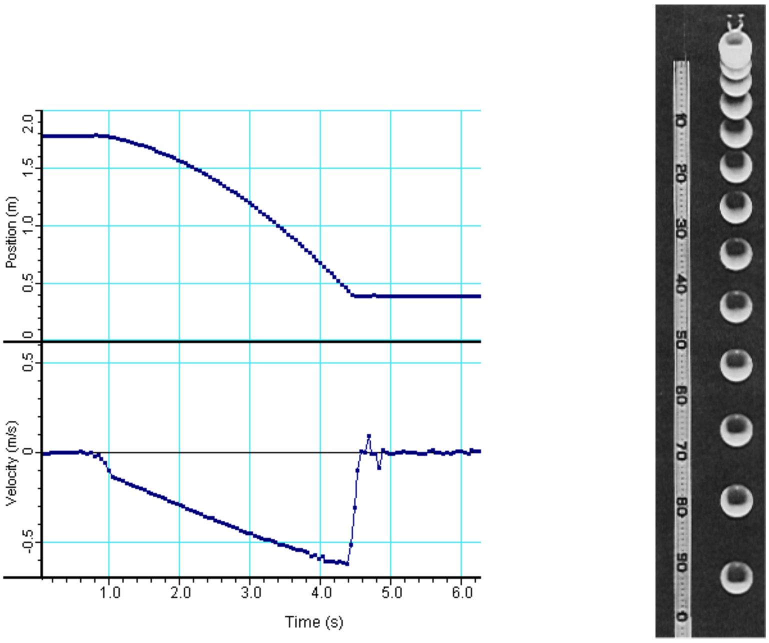


Figure 6.1 A typical graph (obtained with the Pasco motion sensor and the fan cart) and a typical stroboscopic photo (free fall of a small ball) analysed by students.

---

[1] Pasco fan cart is a low-friction cart with a wooden fan blade and an electric motor that can apply an almost constant force.

All students succeeded in doing all the proposed activities, receiving specific support when necessary. Most of this support was related to physical and mathematical aspects of the problems they were faced with, not with problems related with the use of the software.

6.4.2 Questionnaire and results

After the five half days of work in the computer lab, the students answered a short written questionnaire about the work they have done (Table 6.1).

Table 6.1 Items of the Questionnaire about the work done with Modellus (secondary school students)

1	How do you describe your knowledge and skills about the use of computers?
2	How do you evaluate your knowledge about the topics of physics and mathematics approached in these activities?
3	Was it difficult to use Modellus? Why?
4	What advantages or disadvantages do you think the use of Modellus have in this kind of activities?
5	The software helped you understand better the way physics describe motion? Or, on the contrary, it was of any help? Why?

The Appendix synthesizes the answers to this questionnaire. Eleven of the twelve students answered that Modellus was "easy to use". One mentioned that it was easy to use but it "can be complex if the user doesn't know the theory and the formulae". Guidance, companionship and specific help were mentioned by half of the students as essential to the ease of use. As mentioned on the previous sub-section, most students reported they had good or reasonable knowledge about how to use a computer, but only about half of them considered their knowledge of physics and mathematics good or reasonable.

All students identified one or more advantages in using Modellus. The most frequent advantages reported the help it can provide to make computations and visualizations. Three students reported that it could help understanding phenomena and two that it helped them to think. One student gave a detailed reason specifying the importance of multiple representations:

> It helps solving problems that were difficult to solve using traditional approaches, thanks to the use of animations, that can include vectors, graphs, displacements, among other objects, helping understanding the problems since one can see simultaneously distinct phenomena, motion, change, etc. It is possible to represent the same problem in different ways [NC].

Eight of the twelve students mentioned disadvantages. Two of them allude to the negative influence the use of Modellus can have in the training of computations and the other six gave other reasons, ranging from an alert to the danger of using it "for hours and hours" to the use of an English interface.

The last item of the questionnaire asked students to evaluate how useful Modellus was to help them understand the way physics describes motion. All students wrote that Modellus was useful for them to attain that goal. One student is very specific about how Modellus was an "intellectual mirror" for him:

> This program gave me a new vision about motion that until now I only had reached thanks to mental projection. Intrinsically, it is a mirror that reflects our reasoning and makes possible to see it over and over again [NC].

Other students reported how important was for them to explore multiple representations in order to make physics more concrete:

> (...) It shows how the real motion and the formulae are connected, using graphs and animations, making physics less abstract [CD].

> It helped since we could observe the experiments, and sees the results in multiple ways, tables, graphs, etc. [PC].

6.4.3 Discussion

The type of activities proposed to students was relatively demanding. They were asked to use models in real experimental contexts, something they were unused to. All the students used real-time graphs, data logging, and sensors for the first time. It was also the first time that they took real measurements to make models of motion. All students were able, with support but working on their own, in pairs, to make the models they were asked to do, computing factor scales, finding parameters, creating functions (including piecewise-defined functions) and comparing the models with real data. The students considered Modellus easy to use, thanks to its simple interface and to the learning environment where they could easily get support and discuss difficulties. It is also clear for students the importance of previous knowledge when using exploratory tools like Modellus. They were conscious that one could only make meaningful use of it if they were skilled enough in the meaning of the parameters and the specific use of the different mathematical models. As one student said, "if you don't know the content, it is difficult to write models" [AB]. Modellus was recognized, at least by some of the students, as a "cognitive artefact" that can reflect what they think, providing ways to create and explore multiple representations.

This study shows that high school students can start creating models with Modellus after a brief introduction to its features, if they have enough knowledge of the physics and mathematics needed to create them. It also shows that students recognize that Modellus can be an important tool for them to think about how physics describes motion using mathematical models. They do not clearly identify any significant disadvantage—the disadvantages they mentioned can easily be overcome with good learning environments, where meaningful learning is promoted.

6.5 Second Study: Modellus Used by Undergraduate University Students

6.5.1 Subjects, research questions, description of the intervention, and questionnaire

This second study was done with ten[1] undergraduate university students (7 girls and 3 boys), who had just finished the 3rd semester of a BSc in Teaching of Natural Sciences at the College of Sciences and Technology. These students where volunteers for a short course on pre-calculus and basic kinematics that took place in the break between two semesters. The course was offered by the Science Education department to help students overcome difficulties in basic concepts needed to succeed in college courses of physics and mathematics. Six students reported they had basic or reasonable knowledge about how to use a computer and four weak or bad knowledge. Most of them (seven) reported they had reasonable knowledge of secondary school mathematics, but only one reported reasonable knowledge about secondary school physics. None of the students succeeded in their first physics course at college (detailed data are synthesized on Appendix)—this was a criterion for being admitted to the course.

The specific research questions for this study were:

- Can students create their own models and animations?
- Do students agree that Modellus can promote a more integrated approach to physics and mathematics?
- Do students agree that Modellus can help them work more concretely with formal objects?
- What differences do students identify when solving problems without and with Modellus?

The students worked for three days, individually, in a computer lab, following the activities proposed in a 77 page book (in Portuguese) available on the Modellus web page called “Functions and description of motion in space: a brief introduction with Modellus”. There are nine activities included in this book:

1. A ball that moves... (Introduction to Modellus controls and windows and to linear functions.)
2. What is the story of the motion of *Santa Claus*? (Semi-quantitative introduction to graphs and position in a linear axis.)
3. Throwing a ball into the air, with a parabola... (Parabolas and vertical motion with constant acceleration.)
4. Experiments with vectors. (Introduction to vectors, components and magnitude.)
5. Parametric equations of motion. (Two-dimensional motion and graphs.)

---

[1] Two more students started the course but since they didn’t participate in all the activities they were not considered as participants.

6. Parametric equations of a projectile. (Projectile motion, including the equation of the trajectory.)
7. Radians, degrees and rotation. (Using radians and degrees to measure the rotation of a radius vector.)
8. Rotation, sine and co-sine of an angle. (Relation between the rotation of a radius vector and the sine and co-sine of an angle.)
9. Circular motion. (Introduction to the parametric equations of the circular motion, with and without constant speed.)

These activities gradually introduce the most common features (except differential equations and iterations) of Modellus using an exploratory-guided approach. They also include questions for students to solve. At the end of each activity, the students were gathered together to discuss it and the solutions to the questions. The students didn't have time to do all of the activities. Activity 6 was the last one completed by all; some started activities 7 and 8.

On the beginning of the second day (after activity number 3), they solved a simple physics' "braking car problem" *without the use of the computer* (Table 6.2).

Table 6.2 "Braking car problem"

A 1000 kg car took 10 s to brake in a straight line when its speed was 80 km/h. Find the distance travelled by the car until it stopped.
Explain and schematize your reasoning, even if you are not sure if it is correct.
Formulae you may need to solve the problem:

$$x = x_0 + v_{0x}t + \frac{1}{2}a_x t^2 \; ; \; v_x = v_{0x} + a_x t \; ; \; v_x = \lim_{\Delta t \to 0} \frac{\Delta x}{\Delta t} \; ; \; a_x = \lim_{\Delta t \to 0} \frac{\Delta v_x}{\Delta t}$$

On the end of the third day, they solved the same problem again, now using Modellus, answered a nine question written questionnaire (Table 6.3), and made an open comment about the differences they found when solving the problem without and with Modellus. The worksheet given to students to write the solution of the problem included a few problem solving hints, such as: (1) make a diagram of the motion; (2) make a semi-quantitative analysis of the problem; (3) do not mix trajectory with graph; (4) review the solution in order to check if it makes sense.

Table 6.3 Items of the Questionnaire about the work done with Modellus (College students)

1	How do you describe your knowledge and skills about the use of computers?
2	How do you evaluate your knowledge about the topics of physics and mathematics approached in these activities, *when you finished secondary school?*
3	*Now*, how do you evaluate your knowledge about the topics of physics and mathematics approached in these activities?

Table 6.3 Items of the Questionnaire about the work done with Modellus (College students) (Continued)

4	Modellus has been designed to help teachers of physical sciences and mathematics work with their students with a more integrated vision of these two scientific areas. Do you think this a suitable goal for teaching physics and mathematics? If necessary, given your reasons.
5	Do you think Modellus can help reach that goal? Why?
6	A common problem with some teaching and learning environments is the absence of reification of abstract objects, such as, e.g., an oscillator or a parabolic trajectory. One of the main goals of Modellus is to give, both to students and teachers, a tool to work concretely with "formal objects". Do you think this goal can be attained with Modellus?
7	What is your evaluation of the Modellus' interface? Is it enough user-friendly for students and teachers not particular knowledgeable about computers and software?
8	From a traditional viewpoint, an experiment implies the use of concrete objects such as pendulums, stopwatch, sensors, etc. From a less common viewpoint, an experiment is something that one can also do with abstract objects such as particles, vectors, functions, differential equations, etc., using a computer. Modellus has been designed to allow students and teachers (namely at secondary schools and undergraduate studies) do this last type of experiments, without the difficulties of using more complex tools, such as programming languages. How do you evaluate this claim about Modellus?
9	If you want to make any other comment about Modellus and its potentialities and risks, use the following space. You can also write comments about any other aspect of the work done in the last three days.

6.5.2 Results

The Appendix gives a complete description of the answers of the Questionnaire and to the "braking car problem" without and with Modellus.

It was not an easy task for students to carry out all the proposed activities. Most of them reported they had enough knowledge on how to use a computer; but all of them reported they were under-prepared in physics, with only two mentioning the same in mathematics (Table A.10). None of the students had physics in the twelve grade and the physics semester they had in College seemed to have almost no influence on their background knowledge necessary to do the activities.

All students reported some improvement after the intervention (Table A.11), but not enough for them to be confident on the subject. One student reported that "the activities helped him start interpreting problems from a graphical and spatial perspective" [CAL] and another mentioned that "Modellus helped him to make the mental projection of an experiment" [TJS].

Nine of the ten students considered that physics and mathematics could be taught in a more integrated way than it is taught in schools and colleges (Table A.12). All of them recognize that Modellus can help reach this goal (Table A.13)—the "improvement of abstract skills" and the "visualizations" being the main reasons for that help. One student pointed out that "using Modellus one cannot differentiate when it is maths or physics" [RB]. All students agree that Modellus can help them work more concretely with formal objects (Table A.14)—one reported that this is "probably the more satisfactory goal attained with Modellus" [AMP].

All students considered that Modellus is user-friendly enough for students and teachers not particularly knowledgeable about computers and software (Table A.15), but three reported that “only with support”. The same general agreement was reported on the question about the claim that Modellus can help users make experiments with abstract objects, such as particles, vectors, functions, differential equations (Table A.16), without the complexities of more generic tools, such as programming languages.

As described on the previous subsection, the students where asked to solve the “Braking car problem” (Table 6.2) on the beginning of the second day without a computer and at the end of the intervention with Modellus. Four of the ten students were not able to solve the problem without the computer; with Modellus, and after two more days of work with the guided activities, all solved the problem (Table A.6).

Table A.7 summarizes students’ comments about the differences on solving the problem without and with Modellus, as seen by students. Nine of the ten students report “better visualization” as an important difference, the “control of results” being the second most reported difference. Visualization is considered an essential aspect of the process of understanding for several reasons, as exemplified by the following student’s comment:

> “With the help of Modellus we have a better visualization of the problem, it allows us to reduce the degree of abstraction and have a better notion of the expected result. Just with paper and pencil, the abstraction effort must be bigger and we must appeal to the calculator.” [APL].

One student [CMC] considered that “it is more complicated to solve on the computer” because

> “using paper and pencil, we can reason and manage variables in order to have a faster understanding of the problem, but on the computer everything must be more schematized and exact”.

However, this student recognizes that “nevertheless, solving with the computer allows us to visualise graphs and diagrams better than with paper and pencil.” The difficulty this student identify is also mentioned by another one [TJS], who talks about the importance of being familiar with the computational rules of Modellus to take advantage of it:

> “Initially it seems more complicated to solve the problem with Modellus since I wasn’t familiar with its ‘computational rules’. Later it becomes more easy, even dispensing the use of a calculator. Essentially, one needs to get familiar with such rules and its reason to exist. After this is done, it is simple.”

6.5.3 Discussion

These results show that undergraduate students under-prepared in physics can easily use Modellus to create their own models with linear, quadratic, and parametric functions. Students agreed that Modellus support a more integrated approach to physics and mathematics than they had in school.

Students recognized the importance of previous knowledge to take advantage of Modellus. The advantages are related to the visualization capabilities that can help improve abstract skills and reasoning. Students agreed that Modellus helped them work more concretely with formal objects, reducing the abstractness of mathematical models.

As reported by students, the work they have done with Modellus was insufficient to overcome the under-preparation they had in secondary school physics, even though all were able to solve the "braking problem" at the end of the intervention while at the beginning of the second day only six were able to do it. Students agree that more work is necessary to overcome the under-preparation they brought from secondary school.

6.6 Comments on the two Studies

The studies reported in the previous sections were done in a context out of normal courses and with voluntary students. By no means can these students be considered as representative of secondary school or undergraduate students. It is clear that the conclusions of the studies must be taken with caution but I am inclined to expect that if the studies were done in normal classroom contexts the conclusions would be similar. Most students were not particularly skilled in physics and mathematics, and they were not computer enthusiasts. It is not easy to make these type of studies in normal classroom contexts, for reasons such as: (1) the curriculum is usually too big and restrictive to allow the researcher to have the opportunity to make a significant change; (2) computer labs, are scarce in schools and in colleges; (3) students are pressed by examinations and usually think that topics and work not directly assessed in the examinations are not useful; etc.

The main research question of the two studies was "can students create models and animations with Modellus?" Obviously, this question assumes that students should be able to use Modellus without a lot of specific instruction and training given by a teacher, but only with support from written materials and, or, short classroom introductions where physics, modelling and the specific functionalities of the software are approached in an integrated way. The data gathered in the two studies support the view that students find Modellus easy to use and that they can use it to create, explore models, and solve problems. This is an important conclusion: Modellus can be integrated in any physics curriculum without a significant learning workload for students.

The other research question refers to the claim that Modellus can be a "cognitive artefact" for students. As described in Chapters 1 and 3, it was designed to support student reasoning, allowing them to work concretely and visually with abstract mathematical objects, provide ways of making multiple representations, confront data and models, and face thinking and implications of what they are thinking. The data gathered in the studies support these claims: most students report that Modellus helped them reach all or some of these goals. Some recognize that they used it as an "intellectual mirror" or as tool to help make "the mental projection of an experiment". Most reported how useful it was as a tool to visualize physical ideas and models, reducing the gap between abstract models and concrete phenomena.

Will all students that use Modellus use it as those that participated in the two studies? Clearly they will not. In chapter 1, I quoted Pryluck's distinction between "inherent" and "imposed" characteristics of a medium (Pryluck, 1968). The imposed characteristics are situations of exposure to the medium, such as the learning environment created and directed by the teacher. These imposed characteristics are essential to define and characterize the use of exploratory tools. Within a different learning environment, for example, a non-exploratory environment, students will probably see Modellus with other characteristics.

Chapter 7 From Theory to Practice: Computers, Modelling and Modellus in Education

Information is not knowledge, knowledge is not wisdom, wisdom is not insight.
Clarke, A. C. (1993).

7.1 Introduction

This final chapter is both a synthesis of the work and ideas presented in the previous chapters and a personal reflection on the way ahead, in order to integrate computer tools like Modellus in the physics curriculum.

More than thirty years ago, Oettinger (chairman of the Harvard University Program on Information Resources Policy), writing about the mythology of innovation, concluded that the user, more than the tool, is the key factor in innovation:

> In short, computers are capable of profoundly affecting science by stretching human reason and intuition, much as telescopes or microscopes extend human vision. I suspect that the ultimate effects of this stretching will be as far-reaching as the effects of the invention of writing. Whether the product is truth or nonsense, however, will depend more on the user than on the tool (Oettinger, 1969, p. 36).

The "user versus the tool" is a recurring theme in the use of computers in education, and other areas, with most of, if not all, analysts of innovation. And all seem to agree that tools can empower users, but that *is not* an inevitable consequence of the use of tools.

Educational innovation is a difficult enterprise. Constraints for change come from many sources. Typical classrooms are still very similar to classrooms at the beginning of the massification of schools, in the late 1900s. Leon Lederman, a Nobel physics laureate, is probably right when says that "classrooms of today all too often appear and function as they did 100 years ago" (Lederman, 1998). Central and local governments, as well as supranational authorities like the European Union authorities, spent millions of euros in promoting the use of new technologies in schools but the most common classroom practices are still based on lecturing, questioning and recitation, both in science and in other subjects. For example, a recent study on Portuguese schools concluded that the "most common practice in classrooms deals with solving exercises, expositive transmission of knowledge, demonstrations accompanied by questions and correcting tests and homework" (Martins et al., 2002, p. 184).

The "complexity of the real world (…) is extremely difficult" (Edelson, 1998). It is not an easy task to overcome the complexities and difficulties schools and teachers have to change their practices. It will take *time*, *purpose*, *effort*, and *guidance*. But I'm sure it can be done.

7.2 Modelling in Physics Education: the Way Ahead

7.2.1 The place of models in science and mathematics education

Before Galileo and Newton, most of the ideas in physical sciences were expressed only in words (or through geometrical representations), even when authors used proportional reasoning, but with ill-defined concepts and variables, like Aristotelian concepts of force, speed, energy, etc. Mathematics as the language of nature is usually credited to Galileo. For Galileo, and Newton, nature was inherently mathematical, and mathematics was the key to the understanding of reality behind the appearance of natural phenomena.

For a common dictionary (e.g., the *Oxford Concise Dictionary*, Oxford University Press, 1996), the word *model* means "a simplified (often mathematical) description of a system etc., to assist calculations and predictions." Models are created to describe natural systems, capturing only the essence of major objects and processes. A model is only a simplified representation of a system and does not pretend to represent all the features of the system. Many models in the physical sciences are dynamic systems models, i.e., models that establish some sort of mathematical relation between physical quantities and time, considered as an independent variable.

Accordingly to Webb & Hassell (1988), there are five families of models:

1 dynamic systems models;

2 spatial distribution models;

3 qualitative models of logical reasoning;

4 probabilistic event models;

5 data analysis models.

All of these types of models are important in physics, but dynamic systems, i.e., the type of models that can be built with Modellus, are of particular relevance. A simple example of a dynamic system model is the model of an object, considered as a particle (an object with mass but no dimensions), moving with constant velocity (since velocity is a vector quantity, both the direction and the magnitude are constants). The distance travelled by the particle, *s*, can be expressed as a linear function of time, *t*,

$$s = vt$$

Another way of expressing this model is to represent the *x*-coordinate in a certain reference frame, where the direction of *Ox* is coincident with the trajectory of the particle, as a linear function of time,

$$x = v_x t + x_0$$

In this equation, v_x represents the scalar component of the velocity in the *Ox* direction and x_0 represents the value of *x* when time is zero.

We can also represent this model using the concept of instantaneous rate of change. Since the instantaneous rate of change of the *x*-coordinate is constant, we can write that

$$\frac{dx}{dt} = v_x \, .$$

These three models are equivalent, in certain conditions, and they are used in different stages of learning. Typically, the first is used with junior high school students, the second with senior high school students and the third with college students. Both the first and the second model are functions and the third one is a first-order differential equation.

None of these models are "explanations" of the motion of the object. Explaining a motion is describing the forces that are responsible for the motion. This mean that explaining a motion with constant velocity is just saying that the sum of all forces acting on the particle is zero. But this *explanation* is really a *mathematical description* of the interaction between the system (the particle) and its environment.

We can consider yet another equivalent model for the motion of a particle with constant velocity. Using the equal sign as a means to represent that the left side of the equation is substituted by the right side, we can write that

$$x = x + v_x dt$$

where *dt* is a "small" time interval. This model, a difference equation, is of particular interest in certain conditions, especially when we are interested in discussing how computers can use rates of change to make simple or complex computations. Some authors (e.g., Ogborn, 1984) and more recently curricula such as Advancing Physics (Ogborn & Whitehouse, 2000, 2001) give special attention to difference equations as a simple way of introducing students to calculus concepts, such as rate of change, differential equations, integration, etc.

Models are mental constructions distilled from theory and data about the physical world. For example, we can use data (time and position) of a motion with constant velocity to adjust a function such as $x = v_x t + x_0$. From a few assumptions and definitions, Newton was able to use the then new precisely defined concept of force (the instantaneous rate of change of the momentum) to explain uniform rectilinear motion. It should be noticed that, as Bertrand Russell pointed out (1948), scientists use mathematical models to describe/explain phenomena not because they know the physical world very well but because they know very little: they can only discover mathematical properties of physical entities.

The argument of this thesis is that powerful computer modelling tools, i.e., computer software that allow the user to create and explore computer-based models without writing a program in a computer programming language, can be crucial tools both for learning by reception (students can explore models built by the teacher or another person) and for learning by doing (students can build and explore their own models). Computer modelling software have the characteristics of *powerful learning environments*, as defined by De Corte (1989), and may lead to better understanding of scientific ideas and processes. For example, modelling tools can (Webb & Hassell, 1988):

1. help raise the level of cognitive process, by encouraging students to think at a higher level, generalising concepts and relationships;
2. encourage pupils to define their ideas more precisely;
3. provide pupils with opportunities to test their own cognitive models and detect and correct inconsistencies.

7.2.2 Modellus: making "conceptual experiments" with a computer

Modellus is a computer tool that allows students and teachers to perform conceptual experiments using mathematical models expressed as functions, derivatives, rates of change, differential equations and difference equations. From a computational point of view, Modellus can be regarded as a computer microworld for both students and teachers alike, based on a non-programming metaphor: in the "Model window" the user can write mathematical models, almost always in the same way as he would write on paper. There is no new language to be learned, just a few syntax rules about how to write explicit functions, differential equations and iterations.

Two essential features of Modellus are *multiple representations* and *direct manipulation*. Multiple representations mean that the user is able to create, see, and explore different representations of the same model. Direct manipulation means that the user can interact directly with representations, using the mouse and a common graphical interface, without the mediation of written language.

Modellus incorporates both *expressive* and *exploratory* modes of learning activities (Bliss & Ogborn, 1989). In an expressive learning activity, students can build their own models and create ways of representing them. In an exploratory mode, students can use models and representations made by others, analysing how different things relate to one another. Teachers, as well as, curriculum designers and developers, can use it as an authoring language for creating visual representations. This has been done extensively, for example, by the Advancing Physics curriculum in the UK.

Modellus design, discussed on Chapter 3, assumes that the computer is a *cognitive tool*, a tool to support student reasoning, not to show him or her how to reason. Research has given evidence that students and adults have persistent misconceptions or alternative conceptions about scientific matters, even in the common basic ideas (Pfundt & Duit, 1991). Most of these misconceptions are surely associated with the way science and mathematics is taught in schools: teaching tends to overemphasise verbal learning. Objects, particularly mathematical objects, are taught as abstract entities that students cannot "experiment". Computer tools like Modellus can help change the emphasis in a direction where mathematical objects are *objects-to-think-with* (Papert, 1980), objects to make experiments, and objects to study interactively. Learning can then become more *concrete*—mathematical objects behave like "real" objects—and simultaneously they maintain their abstractness: they represent the essential features of phenomena, not the whole phenomena.

Computer-based modelling tools have some evident relevant limitations. For example, teachers must be aware that *learning* with these tools *does not take place spontaneously*, just by exploring the software: *regulation* and *control* are fundamental (de Jong & Joolingen, 1998). In the studies described on Chapter 6, regulation and control of learning was done by teacher intervention and by the use of written materials, with guided inquiry approaches, where students read, discuss with peers, confront conceptions and descriptions, and write (the process of writing was used as a "deccelerator" of information, specially visual information, and an accelerator of knowledge construction). Another limitation is that for certain students and teachers, computer and software user interface can be an obstacle. But this limitation is becoming less important, since computer software have now common standards, familiar to most users.

7.2.3 Modelling in the curriculum: a horizontal and a vertical perspective

In Chapter 4, section 4.3, I proposed a framework for modelling in the physics curriculum. According to this framework, computer modelling tools have an important role on all levels of the school and college curricula. The proposed "natural progression" for modelling, adapted from Ogborn (1999), can be expressed as:

1	Make some computer models, based on direct computations	learn some arithmetic
2	Make some more computer models, based on functions	learn some algebra
3	Make some more computer models, based on differential equations and iterations	learn some calculus and numerical methods

This progression relies on an intensive use of computers to make models using direct computations, functions and differential equations and iterations, as exemplified with Modellus on sub-section 4.2.4. As I've argued in Chapters 2 and 4, this progression implies a deep coordination of the mathematics and the science curriculum (particularly physics) at all levels of schooling. This large scale coordination effort will probably be made on the next few decades, as computers become a ubiquitous tool in learning.

Semi-quantitative thinking, estimation, number sense, symbol sense, visualization and concrete manipulation of abstract concepts will more likely substitute routine and algorithmic approaches, both in physics and mathematics learning, as claimed in Chapter 4. Curriculum developers now face the challenge of embedding computer tools and these new approaches in the curriculum.

7.2.4 Modellus: a preview of the next version

The design and implementation of Modellus version 3 is being done using Java technologies. This new version will follow new standards (*Java Look and Feel Design Guidelines*, 2nd Ed., 2001) and will present a different interface, more tools and web integration, but with the same *software concept* (see section 3.4). Figure 7.1 shows a preview of the new version.

The new version will be able to:

1 Run as a stand-alone application.

2 Run under a browser, from a server.

3 Create files that can be placed and manipulated in a web page.

4 Use URL links.

The new interface is based on multiple windows that can collapse as “sticky notes” in order to facilitate screen management and problem solving planning. Models can be represented as *algebraic, iterative and differential equations* or as *diagrams* (in the Model window), similar to those used in visual modelling tools like *Stella* or *Powersim*. Besides “accumulators”, “rates of change”, and parameters, it will be possible to use icons to explicitly represent functions and all types of variables and variables dependencies.

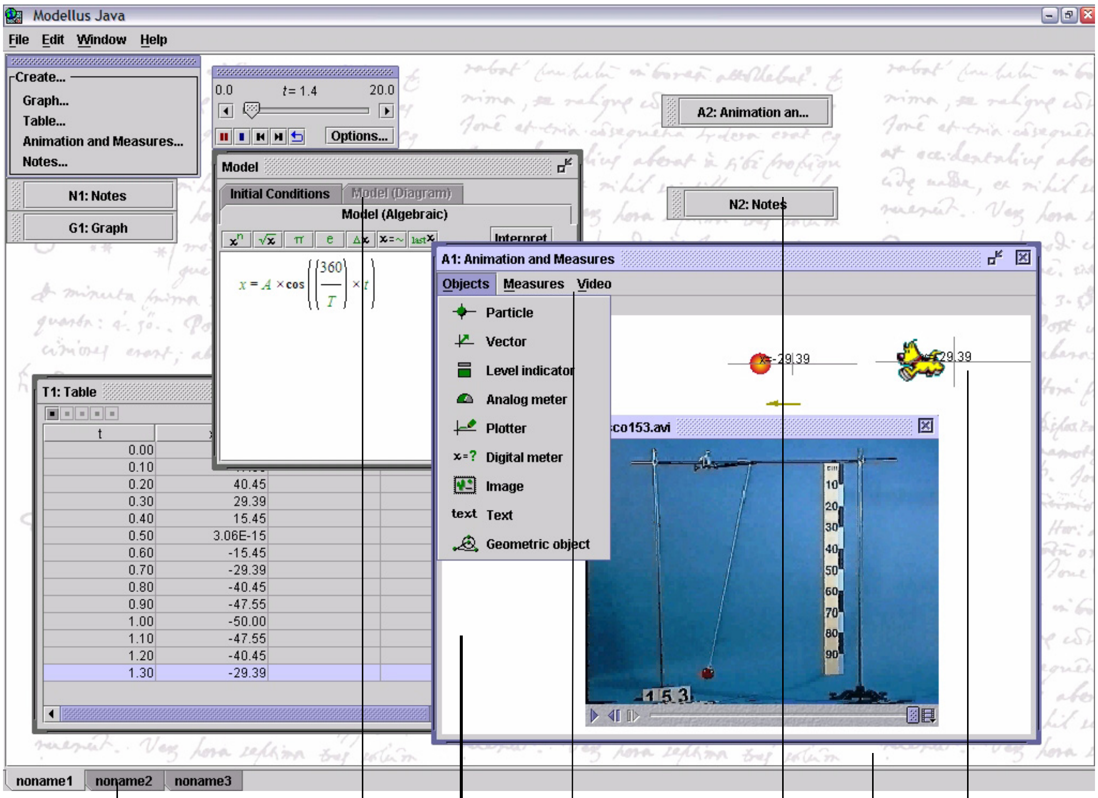
The Model Window will have an improved syntax. For example, it will allow “if... then... else...” relations, vectors, and more functions. Graphs and tables will be available to define parameters’ values in different ranges. Drag and drop from the Model Window, as well as from other windows, will be used to create animations, graphs, tables, etc. The Animation Window will have more objects, including fields, 3D trajectories, simple electronic and mechanical devices, waves, particle animations, conics and other geometrical shapes, etc.

Image and video analysis functionalities will be available on specific windows that can transfer measurements automatically to Data Sets Windows. Data Sets can then be analysed in Data Analysis Windows, in order to find “best-fit” models and graphs. Best-fit models will be expressed as functions or as differential equations. Data Sets Windows will also accept data from other applications, like data logging software and spreadsheets. All windows will easily export data and images to other applications, in standard formats.

The use of more direct manipulation features will reduce the number of buttons and dialog boxes. For example, graphs will be manipulated directly, to change variables, scales and limits, as well as some visual characteristics like colour and type.

The parser used in the Model Window will be completely redesigned in order to give immediate feedback and contextual help when a new line is introduced, avoiding the need for the “Interpret” button.

All local actions, in each window or object, will be controlled locally. The application menu will be reduced to a minimum: file reading, saving and protecting, undo, cut, copy, paste, etc., and selecting between multiple opened files. Password protection of certain windows will also be implemented.



$$x = A \times \cos\left(\left(\frac{360}{T}\right) \times t\right)$$

t	x
0.00	
0.10	
0.20	40.45
0.30	29.39
0.40	15.45
0.50	3.06E-15
0.60	-15.45
0.70	-29.39
0.80	-40.45
0.90	-47.55
1.00	-50.00
1.10	-47.55
1.20	-40.45
1.30	-29.39

Multiple files accessible through tabs.

Models can be created as algebraic or as diagrams.

Windows can be collapsed as sticky notes.

New features on the Animation Window: drag and drop from other windows to use defined variables as object properties; new objects: fields, 3D trajectories; physics components such as electronic, thermal or mechanical devices; etc.

Background can have images.

Particles can be different types of animated objects.

New data, image and video analysis tools available on the Animation Window to allow experimental data analysis transferred from data acquisition software.

Figure 7.1 Modellus version 3: a preview.

A common observation made by some Modellus users, specially beginning users, is that Modellus has “too many windows”. The preview shown in Figure 7.1 also has many windows but it must be noted that a typical file will not have all those windows open: it will, probably, have only three or four windows, or “sticky notes” that correspond to collapsed windows.

7.3 Students and Teachers using Modellus

7.3.1 What teachers say… and do

In Chapter 5, I reported opinions of teachers from schools and colleges. The data were collected through an email questionnaire with ten short answer items, answered by 75 subjects, a subset of the 117 that acknowledged a message sent to 239 subjects that contacted me by email, at least once either in 1997 or 1998, asking questions about Modellus. The respondents were from 12 different countries. These respondents are innovators, familiar with computers in science and mathematics teaching, interested in analysing and creating new ways of teaching— and some have been also involved in the development of new tools.

Most answers clearly support the ideas behind Modellus, namely the importance of:

— concrete reasoning with formal objects;

— reification of mathematical ideas;

— experimentation with abstract objects;

— integration between physics and mathematics;

— visualisation in learning physics and mathematics.

Respondents also agreed about the readiness of Modellus to be used by non-computer experts, either teachers or students. Some used it with students and reported no major difficulties.

Using Modellus to make “conceptual experiments” is criticized by some respondents: they argue that computer experiments are not real experiments, since they are “idealisations”. But other respondents consider this claim as a reasonable claim, since one can “experiment with ideas”.

As one respondent pointed out, Modellus can be “potentially extremely useful” in the physics curriculum. E.g., it can, according to some respondents:

— “help develop reasoning and abstract skills, and to create an observational attitude and analysis skills before experimentation”;

— “help [students] explore real phenomena qualitatively, through simulations, differently from those ‘fabricated’ mathematical applications without any reality”.

As it happens with other tools, there is not a unique way of using Modellus. On the one hand, it can be used just to *show* animations to students that illustrate physical or mathematical ideas, such as the examples that are part of the teachers' guides of *Física 9* and *Física 10* (Fiolhais et al., 1997a and 1997b) and of *Química 10* (Santos & Teodoro, 1997), 9th and 10th grade textbooks published in Portugal. Similar use is made in most chapters of *Advancing Physics* (Ogborn & Whitehouse, 2000; Lawrence & Whitehouse, 2000), in the UK. On the other hand, it can be fully integrated in the curriculum and students must use it to follow what is presented in the textbook. This is partially done in some chapters of *Advancing Physics*, particularly chapters 10 ("Creating Models") and 11 ("Out into Space") and chapter 1 ("Motion of a Particle") of *Fisica 12* (Fiolhais et al., 1996).

Modelling is now recognized by curriculum developers as a "powerful tool for learning abstract concepts, because it facilitates concrete thinking with imagined objects" since by studying how models behave "students can develop familiarity with physical laws and their mathematical expression", helping them to think and to learn how to think better and appreciate both the power and limitations of models (Lawrence & Whitehouse, 2001).

7.3.2 What students say… and can do

Chapter 6 describes two interventions using Modellus, one with 11th grade secondary school students and the other with college students from the Faculty of Sciences and Technology of Nova University, Lisbon. In the first study, twelve secondary school students, participating in a summer course, used Modellus with an exploratory-guided approach, making models from real data or from pure theoretical thinking. In the second study, ten students attending the second year of a Bachelor of Science course (future teachers of Biology and Geology) used Modellus also with an exploratory-guided approach for three days.

In both interventions I was interested in getting data about how difficult it was for students to create and explore their own models. The data obtained support the assumption that students, even with relatively low knowledge of computer software, can easily start using Modellus, creating their own models and animations, with minor difficulties. Their main difficulties seem to be related to their knowledge of physics and mathematics, not to Modellus itself.

In both studies students worked from simple to more complex models, such as linear functions for representing constant velocity motion and quadratic and trigonometric functions to model constant acceleration and oscillatory motions. The teaching approach followed for each topic included:

1. a short introduction for the entire group using a screen projector;
2. a discussion of what the students were asked to do;
3. work in pairs, taking notes when necessary and getting specific help if needed;
4. final discussion, with each group presenting and confronting results and conclusions.

Students reported that Modellus helped them to make computations and visualisations. Some were explicit about how useful it was to help them “represent the same problem in different ways” and as “a mirror that reflects (...) reasoning”.

The students considered Modellus easy to use and were aware of the importance of previous knowledge when using it: “if you don’t know the content, it is difficult to write models”, as one student wrote.

The main research question of the two studies—“can students create models and animations with Modellus?”—had a positive answer: the data gathered in the two studies show that students find Modellus easy to use and that they can use it to create, explore models, and solve problems, without a significant learning workload about the specific functionalities of the software.

The other research question—“can Modellus be a used as a ‘cognitive artefact’?”—i.e., a tool to support reasoning, work concretely and visually with abstract mathematical objects, make multiple representations, confront data and models, and face thinking and implications of thinking, also got a positive answer. Most students reported that Modellus helped them to reach all or some of these goals. Some of them recognize that they used it as an “intellectual mirror” or as tool to help them to make “the mental projection of an experiment”, reducing the gap between abstract models and concrete phenomena.

Both studies were done in a context out of normal courses and with voluntary students. Participating students cannot be considered as representative of secondary school or undergraduate students, but their comments and performances are useful to understand the potentialities and limitations of Modellus. Certainly, students who use Modellus can use it in many different ways and in different learning environments. Students can even use it to practice advanced problem solving, as in the case of the secondary school Portuguese students who participated in a “Ciência Viva” project at the “Instituto Superior Técnico”, in 1998, using Modellus to “put satellites in orbit”, with differential equations (http://www.math.ist.utl.pt/cam/cviva1/index.html [retrieved August 25, 2002]) or to explore advanced topics, such as general relativity and the precession of planets, as was the case of the Spanish college student who wrote a web essay on this topic (http://www.geocities.com/relatividad_2000 [retrieved August 25, 2002]).

7.4 Curriculum Change and Computers

7.4.1 Forty years of curriculum change

In the 60s, science education reformers in the USA and other countries concentrated their efforts on curriculum development. PSSC Physics, BSSC Biology, CBA Chemistry and many other projects required the work of hundreds of people, from schools and universities. In Portugal, in the late 1960s and early 1970s, there was a similar effort, supported by OECD, which produced new curricula for Mathematics, Biology and Geology. Thirty years later, in the 90’s, educators and scientists, pushed by politicians, developed standards about what students should know and be able to do, and what and how they must learn (NCTM 1989, 2000; American Association for the Advancement of Science, 1993; National Research Council, 1996).

Many remarks have been made about the new curricula and standards. One of the most common deals with the fact that teachers are “intimidated by the time, content and preparation demands of hands-on learning” (Tressel, 1994) proposed by the reformers. As a consequence, only more motivated teachers, with better resources, are able to implement and maintain interest in the new curricula. Tressel is also very critical about computers in education: “after 30 years, the promise of computer education remains just that” and one can’t “find a single valid evaluation of the efficacy of computer aided instruction” with the possible exception of “an occasional research project such as the *Geometric Supposer*”. This severe criticism about computers in education is clearly dated, at least in science and mathematics education, where most if not all curriculum guidelines now stress the importance of computers (and calculators) as tools for learning, recognizing that spreadsheets, data logging systems, modelling, etc., should be used across the curriculum.

It is useful to note that, according to Tinker (1996), innovation in science and mathematics teaching didn’t happen within schools, schools of education or in institutions devoted to curriculum development, since it required “the involvement of scientists and educators working together. (...) It may be that all educational innovations that lead to fundamental change require a mix of talent not found within the educational establishment”. A similar view is expressed by Gago (1990).

The renovation of the physics curriculum started in the late 1990s, inspired by the Standards movement and by professional societies, such as the Institute of Physics (UK) and the American Association of Physics Teachers. In these more recent curricula (see Chapter 4), physics is presented as a practical subject, connected to everyday phenomena and technology, incorporates new topics (e.g., image analysis, digital measurement, relativity), and promotes the use of computers to make measurements, graphs and develop and test models. Mathematics has a new role in the physics curriculum, with less formalism. Laboratory work is more concerned with demonstrating concepts and illustrating ideas, but student project work is commonly recommended to do simple research and development projects.

7.4.2 Curriculum development and support

Portugal doesn’t have a tradition of large scale research and development effort in curriculum development. Currently, the educational authorities publish curriculum guidelines, which are developed by groups of teachers from schools and universities. These guidelines are then used by textbook authors to create schoolbooks, adopted by local schools. Schoolbooks, and examinations, particularly the 12th grade examination, determine what is taught in schools.

Curriculum development requires much larger multidisciplinary teams and a larger time span. Specialists from different fields and settings are needed to collaborate in the development and to validate choices. Curriculum materials and activities need to be tested extensively, for accuracy and viability, by teachers and subject specialists. *Advancing Physics* and *Salters Horners Advanced Physics* are two examples of curriculum development projects in the UK that can inspire a model for curriculum development, not only for schools but also for colleges.

Organization and funding is a relevant issue for curriculum development projects. But if educational authorities have funds to buy computers and equipment for school laboratories, supporting curriculum development is at least as important. Creating new ways of teaching, particularly with computers, can't be done by curriculum guidelines or by textbook authors. Curriculum materials, grounded on research and extensively validated, are as essential as hardware and software to promote the curriculum integration of computers.

Even "simple" innovations can take a long time to be widely adopted. For example, it took hundreds of years to use books as personal objects in schools. Something similar may be true for using computers in education, particularly in physics education: it seems clear now that computers have an undeniably ubiquitous role. But as a "complex" innovation, which requires knowledge and skills, as well as policy and organizational measures, it will probably take one or more generations until they are as common as schoolbooks are now. And, as schoolbooks did, they will transform education. But this transformation will probably be slower and longer that we can now imagine. The transformation is more related to access to information and to ways of producing and communicating knowledge and less related to any major change in the process of teaching, where personal interactions are a key factor, even with adults, as Brown and Duguid have shown (2000).

7.4.3 Some policy and organizational considerations

Policy documents and implementation projects, from governments and supranational authorities like the European Union, stress the importance of the use of computers in the curriculum. The E-Europe initiative, according to the *E-Learning Action Plan*, target a ratio of 15 pupils per on-line computer for educational purposes in EU schools by the end of 2003 and supports the evolution of school curricula with the aim of integrating new learning methods based on information and communication technologies by the end of 2002. The recognition that computer skills are a key component of literacy is wide-spread (e.g., the new curriculum guidelines for 10th grade physics in Portugal have about one hundred http addresses to be used by students and teachers!).

Some policy analysts, such as Ernst (1997), consider that "computer use offers an example of technology (largely because of cost and limited useful life) not yet 'ripe' for mass use in early education". But, as Ernst also recognizes, this situation is "bound to change" in the next years. However, improving education can never be strictly related with disseminating computers in schools. Some visionaries, like the well known Bill Gates, in his book *The road ahead* (1995) tried to show that "technology will be pivotal in the future role of teachers". But they also modestly recognize that technology "won't replace or devalue any of the human educational talent needed for the changes ahead:

committed teachers, creative administrators, involved parents, and, of course, diligent students".

For decades "reformers" of different origins have argued that technology of one sort or another is about to revolutionize teaching and learning (Cuban, 2001). However, as Cuban argues, the fact is that teaching and learning are intensely personal activities, and at best technology helps to *facilitate* the interaction between teacher and learner. On the other hand, it's important to say that the most important problems of education (engaged learners, teachers and communities; adequate funding; professional development of teachers and other staff; accountability; resource management; parental and societal involvement in school communities; democratic participation; etc.) have little to do with more technology in schools, as Cuban also points out. Naïve conceptions of educational change assume that educational change depends on the change of independent variables, such as teaching methods, teaching tools and materials, school environments, etc. But, according to Salomon (1992), decades of educational thinking and practice show that there are no independent variables. All variables are mutually dependent when we think of learning environments.

Organisational aspects and resource management in schools play a determining role on how and how often computers are used in schools (Becker, 2000; Cuban, 1989, 2001). For Tinker (1996), school organizational inertia is the most important factor that explains the limited implementation of computers in school science. Usually, governments tend to make investments in equipment but investments in resource management and maintenance are not considered. A typical school can now have 30 or more computers, in laboratories, in resource centres, in libraries, etc. Schools need professional staff to manage and maintain equipments and support its educational use. As recently as 1999, the Portuguese government published a law (DL 515/99) that regulates the careers of technical personnel in schools, such as the careers of laboratory support personnel. But there is no mention of computer support personnel. It seems that legislators aren't aware yet of the importance of computers in schools. If we want schools of the XXI century to be different from XIX century schools, technical personnel for resources management must be considered as being essential as teachers are now.

Embedding computers in physics (and mathematics) education, in high school and college, must be an explicit goal of educational policy. The use of computers in science and technology, as well as in most human activities, are now so widespread that it is difficult, if not impossible, to argue that computers should not be used in science and mathematics education. Technologies, old and new, helped to change the way people see the world, communicate, learn, and build identities. With the current stage in technology dissemination, education cannot discuss whether to "use or not to use" computers. The discussion can only be about "best practices" and "bad practices", about "empowering users" or "deskilling users".

As with all educational innovations, it is not possible or desirable to define a universal "algorithm" that can *guarantee* a successful process to embed computers in education. But research in policy and in education can suggest some useful elements to define and implement such a process. For example, the classic RD&D model (Research, Develop & Disseminate) based on techno-rationality, has serious limitations and is insensitive to school and teachers cultures, as Lieberman (1998) concludes, in the introduction to the 1998 *International Handbook of Educational Change*. And, as Fullan (1998) wrote, "change can't be managed".

In the first wave of curriculum development, forty years ago, "educating the teachers" to use the curricula had reduced importance, particularly in the USA. Developers aimed to make "teacher proof curriculum materials" but, more recently, teachers are recognized as playing a relevant role both in development and implementation and there is plenty of evidence that teachers change the ideas when they teach them (Korthagen, 2001). The sense of ownership is probably the most crucial question in curriculum innovation:

> One of the strongest conclusions to come out of decades of studies of the success and failure of a wide variety of curriculum innovations is that innovations succeed when teachers feel a sense of ownership of the innovation: that it belongs to them and is not simply imposed on them (Ogborn, 2002, p. 143).

According to Korthagen (2001, p. 3), "the problem of educational change, and particularly of teacher education, is first of all a problem of dealing with the natural emotional reactions of human beings to the threat of losing certainty, predictability or stability. This affective dimension is too much neglected in the technical-rational approach." He proposes the following basic tenets of a *realistic* approach:

1. The starting points are concrete practical problems and the concerns experienced by teachers in real contexts.
2. It promotes the systematic reflection of teachers on their own and their students' wanting, feeling, thinking and acting, on the role of context, and on the relationships between those aspects.
3. It builds on the personal interaction between the teacher educator and the teacher and on the interaction amongst the teachers.
4. It takes gestalts of the teacher as the starting point for professional learning, using theory not as a reduction or simplification of formal academic knowledge, but as perceptual knowledge, personally relevant and closely linked to concrete contexts.
5. It has a strongly integrated character (integration of theory and practice and integration of several disciplines.)

A policy for changing physics education must be grounded in a new vision of "ownership" of innovation, a vision that recognizes teachers, particularly experienced teachers, as essential to create, define and assess the goodness of innovative ideas and approaches. This doesn't mean that scientists, educational researchers and curriculum developers have any relevant role. As established institutions, schools have a great inertial organization and must interact with other systems that can help change. But outside institutions don't change schools: change is an *internal* commitment, not an external intention.

This policy has a clear concept: schools are learning communities in which teachers pursue clear, shared purposes for student learning, engage in collaborative activities to achieve their purposes, and take personal and collective responsibility for student learning. Students work, discuss, listen, present their views and ideas, and support other students, particularly those who are having difficulties. Teachers are frequent learners and they are able to learn from their students, from other people in the school, and from sources outside the school. Teachers are particularly interested in monitoring and support students with learning problems. In physics education, instruction and hands-on activities are balanced. Scientific inquiry (getting evidence, testing ideas and models, supporting conclusions, etc.) is normal practice. And tools, particular computer tools, are used to support inquiry, to search and get information, and to communicate and share knowledge.[1]

Such a policy should:

1. Be part of a global policy to improve the quality of education, valuing democratic and participatory principles and recognizing change as an internal process, focusing on a sense of ownership of innovation by those who are in the front line of the educational enterprise.
2. Promote curriculum and professional development as participatory activities, guided by practical problems but grounded on relevant educational research and theory.
3. Promote schools as supportive and rich environments, where committed teachers can teach and manage learning environments with support from other professional staff.

7.5 Coda: old and new processes, old and new technologies

Common educational practices, like lecturing, homework, reading, writing, and hands-on activities had hundreds of years of development and refinement. These practices have been used by generations of teachers and students. And they will certainly be used by generations to come. Teaching and learning has always been an intense personal activity and it will probably always be (Cuban, 2001). Computers will help change education, as they have changed most human activities, but this change will be more concerned with the way learners can access, create, and share knowledge.

In a technology rich environment, some old processes still maintain their power, even when the new rich environments seem to supersede the old processes. For example, in a technology environment, long division algorithms are completely useless. The same is true for solving most equations and many other routine computational processes, as I have argued in Chapter 1. But estimating solutions (which implies the mastery of arithmetic tables...) and verifying results, are absolutely fundamental knowledge skills

---

[1] This vision is almost the opposite of naïve visions presented by some early visionaries of the use of technology that, based on a techno-rationality view, saw the school of the future as places where the “goals of education are fully realized through computer technology. (...). At the heart of this school is the computer system, offering individualized education to each student. The student displays are large three-dimensional color displays in the basic learning dome (Bork, 1977, p. 22, referring to George Leonard's book, *Education and Ecstasy*, 1968).

needed to make sense of the use of the technology. *Making sense* of computations, of data, of symbols, of models, is as old as scientific thinking. It will be always a powerful process in technology rich environments. These new environments can supersede certain old processes, like sophisticated numerical or symbolical calculations, but cannot replace others, like estimation and verification.

Modellus (available in Portuguese, Brazilian Portuguese, English, Spanish, Greek, Polish, and Slovak) is now a tool used by thousands of students and teachers in schools and colleges in many places in the world, from Brazil to the UK, from China to Colombia, from Poland to Portugal. Frequently, I receive email messages like the one I received from Professor Luo Xingkai in June 2002:

> From: "LUO Xingkai" <xkluo@mailbox.gxnu.edu.cn>
> To: <modellus@mail.fct.unl.pt>
> Subject: Greeting and appreciation from Guilin, China
> Date: Fri, 14 Jun 2002 08:55:45 +0800
> Dear Colleagues,
>
> I often visit your wonderful website for Modellus. I do appreciate it very much for your significant contribution to the science education and frequently introduce your work to our Chinese colleagues.
> In order to learn your work more conveniently I am herewith writing to you for requesting a CD-Rom version of your program and the related materials.
> Our institute is one of the leading working group in the field of science education in Mainland China. For more detail please visit our website at: http://www.ipe.gxnu.edu.cn or http://www.risechina.org
>
> With best regards.
>
> --
> Prof. Dr. LUO Xingkai
> Director, Research Institute of Science Education
> Guangxi Normal University
> Guilin, 541004, P. R. China
> Tel: + + 86-773-5833180 Fax: + + 86-773-5838963
> E-mail: xkluo@mailbox.gxnu.edu.cn URL: http://www.ipe.gxnu.edu.cn

This message, only possible due to the extraordinary Internet revolution that took place in the last two decades, initially from scientific and educational institutions and later spreading to all other institutions, shows clearly how a *new* technology helped change an *old* process (communication between colleagues)—Professor Xingkai received an answer the same day he sent his message, with the new Modellus version 2.5 setup file attached (the CD and the printed manual will take at least a week to arrive in China). It also shows how *old* technologies (printed documents) live together with *new* technologies (the printed manual and many other documents are included on the CD and can be downloaded from the Internet).

Is Modellus a “significant contribution to science education” as Xingkai wrote? It has been around for seven years (the first time it was published was in June 1996: more than two hundred teachers of physics and mathematics joined at the Campus of Caparica for the first public presentation). It has received prizes in Portugal and in the US, including a prize from the journal *Computers in Physics*, from the American Institute of Physics. It has been reviewed in many different countries and published in seven languages. An Internet search using Google (http://www.google.com) locates thousands of pages relating to Modellus. It has more than 700 registered users, from 36 different countries. A significant number of mathematics teachers in Portugal use it with some regularity and there are courses and workshops promoted by the Portuguese Association of Mathematics Teachers. The Portuguese Ministry of Education supported its publication and dissemination. There is a senior high school curriculum in the UK that uses it extensively as an integrated tool. *Physics Education*, a leading journal in the field, has recently published papers where authors make use of Modellus. A Brazilian university publisher will publish a college course based on Modellus examples and activities. A German physics education department made available on the web a small book about quantum mechanics that has many examples made in Modellus. And so on.

It's too early to know how significant Modellus is for improve science education. This thesis presents theoretical arguments, based on how science is done and learnt, to support the claim that modelling (and Modellus) can improve physics teaching and learning, making it more concrete but still maintaining its abstractness, supporting cognition as a distributed process, shared both with other individuals and with tools and artefacts, situated in a particular context of intentions, social partners, and tools, extending human intelligence, and enabling students and teachers to perceive and think in ways they could not manage without tools. It has also given evidence that experienced teachers and researchers recognize that modelling is a powerful approach to learn physics and that students can use Modellus without a significant learning workload about the software and can make use of it as a "tool-to-think-with".

The purpose of any knowledge arrived at in educational research is, as Husén (1998) wrote,

> "to provide a basis for action, be it policy or methods of teaching in the classroom".

I have done my best to give a contribution to that end.

Appendix Results from Students Questionnaires

A.1 Results from the First Study

Table A.1 Item 1: Synthesis of students’ answers

How do you describe your knowledge and skills about the use of computers?

	RC	JB	NC	ML	CD	PFC	PC	VH	CB	AB	PP	FL
“Reasonable”		√	√	√		√			√		√	
“Good”	√							√				√
“Basic”					√		√					
“Weak”										√		

Table A.2 Item 2: Synthesis of students’ answers

How do you evaluate your knowledge about the topics of physics and mathematics approached in these activities?

	RC	JB	NC	ML	CD	PFC	PC	VH	CB	AB	PP	FL
“Good”	√		√	√		√			√			
“Insufficient”		√					√				√	√
“Next to nothing”					√					√		
“Reasonable”								√				

Table A.3 Item 3: Synthesis of students’ answers

Was it difficult to use Modellus? Why?

	RC	JB	NC	ML	CD	PFC	PC	VH	CB	AB	PP	FL
“No, it is easy to use”	√	√	√	√	√	√	√	√	√		√	√
“No, good guidance”	√						√				√	
“No, I was helped when it was necessary”	√	√										
“No, thanks to a good companionship”		√					√					
“No, except some equations”		√										
“No, but it can be complex if the user doesn’t know the theory and the formulae”										√		

Table A.4 Item 4: Synthesis of students' answers

What advantages or disadvantages do you think the use of Modellus have in this kind of activities?

	RC	JB	NC	ML	CD	PFC	PC	VH	CB	AB	PP	FL
advantages												
"excellent help on computations"	√						√	√	√			√
"help on visualization of the phenomena"			√	√		√				√		
"helps the understading of the phenomena"	√		√		√							
"help on thinking"		√				√						
"help on multiple representations of the phenomena"			√									
"helps problem solving"											√	
disadvantages												
"can promote the absence of the training of computations"			√	√								
"English interface"		√										
"the user can work with it for hours and hours"						√						
"demand certain knowledge that some people don't have"										√		
"it can be difficult for those who don't understand about computers"							√					
"it can avoid students to understand the problems"												√
"None"	√											

Table A.5 Item 5: Synthesis of students' answers

The software helped you understand better the way physics describe motion? Or, on the contrary, it was of any help? Why?

	RC	JB	NC	ML	CD	PFC	PC	VH	CB	AB	PP	FL
"It helped a lot"	√	√									√	√

Table A.5 Item 5: Synthesis of students' answers (Continued)

The software helped you understand better the way physics describe motion? Or, on the contrary, it was of any help? Why?

	RC	JB	NC	ML	CD	PFC	PC	VH	CB	AB	PP	FL
"It helped to visualise and to concretize our knowledge"				√	√	√						
"It helped to understand better how physics describe motion"									√	√		
"It gave me a new vision of motion"			√									
"It reflects our reasoning"			√									
"It helped to see the experiences from multiple perspectives"							√					
"It clarified my mind"								√				

A.2 Results from the Second Study

A.2.1 The "Braking car problem"

Problem: "A 1000 kg car took 10 s to brake in a straight line when its speed was 80 km/h. Find the distance travelled by the car until it stopped."

Table A.6 Student's answers to the "Braking problem"

	Without computer		With computer	
ACF	Correct	Drawing: yes Graph: yes	Correct	Drawing: yes Graph: yes
			Student comment: "With Modellus and the computer it is possible to get a bigger degree of abstraction, supported by graphical demonstrations and instant animations, besides quick and easy computations that give a set of conditions that optimize problem solving".	
AMP	No. Error in the computation of the initial speed. Acceleration not computed.	Drawing: incomplete Graph: incomplete	Correct	Drawing: incomplete Graph: no
			Student comment: "The difference between using paper and using Modellus is that with Modellus one can visualize the motion of the car, the acceleration and the velocity vector. That is, it becomes easy to understand the movement."	
APL	Correct	Drawing: incomplete Graph: no	Correct	Drawing: no Graph: no
			Student comment: "With the help of Modellus we have a better visualization of the problem, it allows us to reduce the degree of abstraction and have a better notion of the expected result. Just with paper and pencil, the abstraction effort must be bigger and we must appeal to the calculator."	
AM	Correct solution but incorrect analysis of the solution. Wrote: "the result is physically absurd, probably due to a wrong solution of the problem"	Drawing: incomplete Graph: no	Correct	Drawing: no Graph: no
			Student comment: "In my opinion, the differences between the two methods are fundamentally due to the issue of 'imagination', that is, with the possibility that we have that, solving the problem with Modellus, be in a closer contact with the circumstances of the problem, particularity that doesn't occur when we solve the problem just using 'paper and pencil'. On the other way, the required knowledge of Modellus allow faster problem solving processes, an important advantage, if we take into account that most problems take a long and stressful time to solve only with 'paper and pencil'. In short, one can conclude that using Modellus, and knowing well how it works, problem solving (at least the one we solved) become much simpler than using paper and pencil."	

Table A.6 Student's answers to the "Braking problem" (Continued)

	Without computer		With computer	
CAL	Correct	Drawing: yes Graph: no	Correct	Drawing: no Graph: no
			Student comment: "From the moment one can control Modellus potentialities it is simpler to solve a problem of this type, since one can dispose all the necessary information, whatever the instant in time, beyond allowing to corroborate through graphs and tables values computed with 'paper and pencil' ".	
CMC	Incorrect	Drawing: no Graph: no	Correct	Drawing: incomplete Graph: no
			Student comment: "In my opinion, it is more complicated to solve on the computer. Nevertheless, solving with the computer allow us to visualise graphs and diagrams better that with paper and pencil. Using the traditional method, using paper and pencil, we can reason and manage variables in order to have a faster understanding of the problem, but on the computer everything must be more schematized and exact."	
NLP	Incorrect	Drawing: yes Graph: yes	Correct	Drawing: yes Graph: yes
			Student comment: "Solving the problem with Modellus, it becomes more easy to visualize what we are solving; just with paper and pencil this is more difficult. Knowing the topic, it is possible to solve the problem checking the result with the help of Modellus."	
RB	Incorrect	Drawing: yes Graph: yes	Correct	Drawing: yes Graph: yes
			Student comment: "Although I think that even with the help of Modellus I got it wrong, at least it helped me visualized better what I wanted, helping to connect the reasoning, accordingly with what I was asked for. Regarding Physics I, it is a pity we can't even use the calculator. The computer can help a lot if the 'mechanics' of the problem is internalised. If we don't have the knowledge of how to solve the problem, the computer is useless."	
SM	Correct	Drawing: yes Graph: no	Correct	Drawing: no Graph: no
			Student comment: "With the help of Modellus, it is possible to see the car breaking, is useless to imagine. With the pencil, we have to imagine, potentially making errors in vectors, constants, that didn't happen with the help of Modellus."	

Table A.6 **Student's answers to the "Braking problem" (Continued)**

	Without computer		With computer	
TJS	Correct	Drawing: no Graph: yes	Correct	Drawing: no Graph: no
			Student comment: "Initially it seems more complicated to solve the problem with Modellus since I wasn't familiar with its 'computational rules'. Later it becomes more easy, even dispensing the use of a calculator. Essentially, one needs to get familiar with such rules and its reason to exist. After this is done, it is simple."	

Table A.7 **Summary of student comments about differences on solving the "Braking problem" without and with Modellus**

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
"Better visualization"	√	√	√	√		√	√	√	√	
"Control of results"			√		√	√	√	√		
"Graph visualization"	√				√	√				
"Quick/easy computations"	√			√						√
"Less abstraction"	√		√							
"Animation features"	√	√								

A.2.2 Questionnaire about the work done with Modellus

Table A.8 Items of the Questionnaire about the work done with Modellus

1	How do you describe your knowledge and skills about the use of computers?
2	How do you evaluate your knowledge about the topics of physics and mathematics approached in these activities, *when you finished secondary school?*
3	*Now*, how do you evaluate your knowledge about the topics of physics and mathematics approached in these activities?
4	Modellus has been designed to help teachers of physical sciences and mathematics work with their students with a more integrated vision of these two scientific areas. Do you think this a suitable goal for teaching physics and mathematics? If necessary, given your reasons.
5	Do you think Modellus can help reach that goal? Why?
6	A common problem with some teaching and learning environments is the absence of reification of abstract objects, such as, e. g., an oscillator or a parabolic trajectory. One of the main goals of Modellus is to give, both to students and teachers, a tool to work concretely with “formal objects”. Do you think this goal can be attained with Modellus?
7	What is your evaluation of the Modellus’ interface? Is it enough user-friendly for students and teachers not particular knowledgeable about computers and software?
8	From a traditional viewpoint, an experiment imply the use of concrete objects such as pendulums, stop-watch, sensors, etc. From a less common viewpoint, an experiment is something that one can also do with abstract objects such as particles, vectors, functions, differential equations, etc., using a computer. Modellus has been designed to allow students and teachers (namely at secondary schools and undergraduate studies) do this last type of experiments, without the difficulties of using more complex tools, such as programming languages. How do you evaluate this claim about Modellus?
9	If you want to make any other comment about Modellus and its potentialities and risks, use the following space. You can also write comments about any other aspect of the work done in the last three days.

Table A.9 Item 1: Synthesis of students’ answers

How do you describe your knowledge and skills about the use of computers?

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
“Basic knowledge”	√		√			√				√
“Weak”		√		√			√			
“Reasonable”					√			√		
“Bad”									√	
“Dislike computers”		√								

Table A.10 Item 2: Synthesis of students' answers

How do you evaluate your knowledge about the topics of physics and mathematics approached in these activities, *when you finished secondary school?*

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
"Reasonable knowledge in mathematics"		√	√	√		√	√	√	√	
"Under prepared in mathematics"	√				√					
"Under prepared in physics"	√	√			√	√	√			
"Almost any knowledge in physics"		√	√	√				√		
"Physics only until 9th grade"	√	√								
"Insufficient knowledge in physics in spite of acceptable classifications"									√	
"Reasonable knowledge but difficulties to connect ideas"										√

Table A.11 Item 3: Synthesis of students' answers

Now, how do you evaluate your knowledge about the topics of physics and mathematics approached in these activities?

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
"Better but still with insufficient knowledge"				√		√			√	
"A new perspective but still with insufficient knowledge"	√							√		
"A little better"		√								
"Better knowledge in physics and better problem solving skills"					√					
"Better knowledge in mathematics"					√					
"No problems with the math, but physics is still a problem"							√			
"More able to connect ideas"										√

Table A.12 Item 4: Synthesis of students’ answers

Modellus has been designed to help teachers of physical sciences and mathematics work with their students with a more integrated vision of these two scientific areas. Do you think this a suitable goal for teaching physics and mathematics? If necessary, given your reasons.

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
“Yes, these areas demand abstract skills that can be improved with the computer”	√				√		√		√	√
“Yes, since it allows a more easy transmission of knowledge”				√	√			√		
“Not a appropriate goal”		√								
“Yes”			√							
“Modellus is nice and useful but one still needs to know the basic knowledge necessary to take advantage of it”						√				

Table A.13 Item 5: Synthesis of students’ answers (Questionnaire about the work done with Modellus)

Do you think Modellus can help reach that goal? Why?

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
“Yes, can improve the development of abstract skills”		√			√		√			√
“Yes, since it is very important see what we are learning, not just listening”							√		√	
“Yes, the user can explore formulas, graphs, tables, etc.”	√									
“Yes, the user can progress individually, but with support”			√							
“Yes, since it allows a more easy transmission of knowledge”				√						
“Yes, but it cannot be considered as something that solves all difficulties students have”						√				
“Yes, using Modellus one cannot differentiate when it is maths or physics”								√		

Table A.14 Item 6: Synthesis of students’ answers (Questionnaire about the work done with Modellus)

A common problem with some teaching and learning environments is the absence of reification of abstract objects, such as, e. g., an oscillator or a parabolic trajectory. One of the main goals of Modellus is to give, both to students and teachers, a tool to work concretely with “formal objects”. Do you think this goal can be attained with Modellus?

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
“Yes”	√						√			√
“Yes, entirely”			√	√						
“Yes, probably is the more satisfactory goal attained with Modellus”		√								
“Yes, the student can make experiments and changes, promoting faster and better learning”					√					
“Yes, but only if the students have some knowledge to understand what they do”						√				
“Yes, it helps visualization”								√		
“Yes, it helps concretization”									√	

Table A.15 Item 7: Synthesis of students’ answers (Questionnaire about the work done with Modellus)

What is your evaluation of the Modellus’ interface? Is it enough user-friendly for students and teachers not particular knowledgeable about computers and software?

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
“Yes”			√		√	√		√		
“Yes, with support”		√					√		√	
“Yes, it is user-friendly but it should be able to communicate more knowledge to the user”	√									
“Yes, fairly”				√						
“Yes, but it needs some time”										√

Table A.16 Item 8: Synthesis of students' answers (Questionnaire about the work done with Modellus)

From a traditional viewpoint, an experiment imply the use of concrete objects such as pendulums, stop-watch, sensors, etc. From a less common viewpoint, an experiment is something that one can also do with abstract objects such as particles, vectors, functions, differential equations, etc., using a computer. Modellus has been designed to allow students and teachers (namely at secondary schools and undergraduate studies) do this last type of experiments, without the difficulties of using more complex tools, such as programming languages. How do you evaluate this claim about Modellus?

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
"Attained, the user can transform abstract experiments in visual and concrete experiments"			√				√			√
"Attained"						√		√	√	
"Attained, the user can visualize problems minimizing abstraction"	√									
"Attained, particularly for experiments with objects in motion"		√								
"Attained, it allows more complete interpretation of basic situations"				√						
"Attained, you don't need to know how to program to make experiments with the computer"					√					

Table A.17 Item 9: Synthesis of students' answers (Questionnaire about the work done with Modellus)

If you want to make any other comment about Modellus and its potentialities and risks, use the following space. You can also write comments about any other aspect of the work done in the last three days.

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
"The excessive use is a risk since we cannot use it on exams"					√					
"The activities we have done were too much intensive"						√				
"Classes like these are more motivating, if we know what we are doing"							√			

Table A.17 Item 9: Synthesis of students' answers (Questionnaire about the work done with Modellus) (Continued)

If you want to make any other comment about Modellus and its potentialities and risks, use the following space. You can also write comments about any other aspect of the work done in the last three days.

	ACF	AMP	APL	AM	CAL	CMC	NLP	RB	SM	TJS
"The systematic use have the risk of making students able to think only if they have a computer"							√			
"This software, and teachers able to use it, should be in schools when I was a student"								√		

Author Index

Index

References

Advancing Physics Project. (1998). Briefings for working groups. London: Institute of Physics.

American Association for the Advancement of Science. (1989). *Science for all americans (Project 2061)*. Washington: American Association for the Advancement of Science.

American Association for the Advancement of Science. (1993). *Benchmarks for science literacy (Project 2061)*. N.Y.: Oxford University Press.

American Association for the Advancement of Science. (1998). *Blueprints for reform, science, mathematics, and technology education (Project 2061)*. NY: Oxford University Press.

Apple, M. W. (1991). The new technology: is it part of the solution or part of the problem in education? *Computers in the Schools, 8*(1/2/3), 59-81.

Atkin, J. K. (1975). *The digital computer in physics education: a description of achievements and trends in modern teaching.* Unpublished Master, University of Sheffield.

Ausubel, D. P. (1963). *The psychology of meaningful verbal learning*. New York: Grune & Stratton.

Ausubel, D. P., Novak, J. D., & Hanesian, H. (1978). *Educational psychology: a cognitive view.* N. Y.: Holt, Rinehart and Winston.

Ausubel, D. P. (2000). *The acquisition and retention of knowledge: A cognitive view.* Dordrecht: Kluwer Academic Publishers.

Bak, P. (1996). *How Nature works*. NY: Springer Verlag.

Barnett, M. (2000). The emergence of physics as an academic subject. In A. Morris (Ed.), *Shaping the future, Revitalising physics education*. Bristol and Philadelphia: Institute of Physics Publishing.

Brasell, H. (1987). The effect of real-time laboratory graphing on learning graphic representations of distance and velocity. *Journal of Research in Science Teaching, 24*(4), 385-395.

Baugsto, B., Byrknes, A., Krakenes, A., & Myrtveit, M. (1993). Powersim, the complete software tool for dynamic simulation (Version 1.1) [Windows]. Manger, Norway: ModellData.

Becker, H. J. (2000). *Findings from the teaching, learning, and computing curvey: is Larry Cuban right?* (Internal Report): University of California at Irvine.

Beichner, R. J., DeMarco, M. J., Ettestad, D. J., & Gleason, E. (1990). VideoGraph: a new way to study kinematics. In E. F. Redish & J. S. Risley (Eds.), *Computers in physics instruction* (pp. 244-245). Redwood, CA: Addison-Wesley.

Berger, C. F., Lu, C. R., Belzer, S. J., & Voss, B. E. (1994). Research on the uses of technology in science education. In D. L. Gabel (Ed.), *Handbook of research on science teaching and learning* . NY: MacMillan.

Bliss, J., & Ogborn, J. (1989). Tools for exploratory learning. *Journal of Computer Assisted Learning, 5*(1), 37-50.

Boohan, R. (1994). Interpreting the wold with numbers: An introduction to quantitative modelling. In H. Mellar & J. Bliss & J. Ogborn & C. Tompsett (Eds.), *Learning with artificial worlds: Computer based modelling in the curriculum*. London: Falmer Press.

Bork, A. (1975). Effective computer use in physics education. *American Journal of Physics, 43*(1).

Bransford, J., Brown, A., & Cocking, R. (2000). *How people learn: Brain, mind, experience and school: Expanded Edition*. Washington: National Academy Press.

Brna, P. (1987). *Confronting science misconceptions with the help of a computer.* Unpublished Ph.D. Thesis, University of Edinburgh.

Brooker, F. E. (1949). Communication in the modern world. In N. B. Henry (Ed.), *The forty-eighth yearbook of the National Society for the Study of Education: Part I, Audio-visual materials of instruction* . Chicago: The University of Chicago Press.

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher, 18*(1), 32-42.

Brown, J. S., & Duguid, P. (2000). *The social life of information*. Boston, Mass.: Harvard Business School Press.

Brungardt, J. B., & Zollman, D. (1995). Influence of intervideodisc instruction using simultaneous-time analysis on kinematics graphing skills of high school physics students. *Journal of Research in Science Teaching, 32*, 855-869.

Bruner, J. S. (1961). The act of discovery. *Harvard Educational Review, 31*(1), 21-32.

Bruner, J. (1992). Science Education and Teachers: a Karplus Lectures. *Journal of Science Education and Technology, 1*(1), 5-12.

Bruner, J. (1996). *The culture of education*. Cambridge, MA: Harvard University Press.

Buchmann, M. (1992). Dilemmas and virtues in research communication. *Curriculum Inquiry, 22*(3), 313-329.

Calverley, G., Fincham, D., & Bacon, D. (1998). Modernisation of a traditional physics course. *Computers & Education, 31*(2), 151-169.

Caramazza, A., McClossky, M. A., & Green, B. (1980). Curvilinear motion in the absence of external forces: Naive beliefs about the motion of objects. *Science, 210*, 1139-1141.

Carlsen, D. D., & Andre, T. (1992). Use of a microcomputer simulation and conceptual change text to overcome student preconceptions about electric circuits. *Journal of Computer-Based Instruction, 19*(4), 105-109.

Carvalho, M. (1994). *Um estudo da utilização do computador pelos professores de física.* Unpublished Master Thesis, Universidade do Minho, Braga.

Carvalho, R. (1947). Sôbre a correlação entre a matemática e a física no ensino liceal. *Gazeta de Matemática, VIII*(31), 11-13.

Carvalho, R. (1985). *História do ensino em Portugal*. Lisboa: Gulbenkian.

Chi, M. T., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science, 5*, 121-152.

Clark, R. C. (1990). Instructional media and technology research. *International Journal of Educational Research, 14*(6), 487-580.

Clark, R. E. (1992). Facilitating domain-general problem solving: computers, cognitive processes and instruction. In E. De Corte, M. C. Linn, H. Mandl, & L. Verschaffel (Eds.), *Computer-based learning environments and problem solving* . Berlin: Springer-Verlag.

Clark, R. E., & Salomon, G. (1986). Media in teaching. In M. C. Wittrock (Ed.), *Handbook of research on teaching* (3rd ed.,). N. Y.: MacMillan.

Clérigo, F., & Teodoro, V. D. (1994). *Geometria Descritiva [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Cockburn, A., & Greenberg, S. (1998). The design and evolution of TurboTurtle, a collaborative microworld for exploring Newtonian physics. *International Journal of Human Computer Studies, 48*(6), 777-801.

Collins, A. (1991). The role of computer technology in restructuring schools. *Phi Delta Kappan, 73*, 28-36.

Cooper, A. (1995). *About face, the essentials of user interface design*. Foster City, CA: Programmers Press.

Cuban, L. (1986). *Teachers and machines, the classroom use of technology since 1920*. N. Y.: Teachers College Press.

Cuban, L. (1989). Neoprogressive visions and organizational realities. *Harvard Educational Review, 59*(2), 217-222.

Cuban, L. (1993). Computers meet classroom: classroom wins. *Teachers College Record, 95*(2), 185-210.

Cuban, L. (2001). *Oversold and underused: Computers in the classroom*. Cambridge, Mass.: Harvard University Press.

Davies, B. (1997). Physics like you've never had before. *Physics Education, 32*(6), 418-421.

Davis, B., Porta, H., & Uhl, J. (1994). *Calculus&Mathematica®*. Reading, Mass.: Addison-Wesley.

De Corte, E. (1989). Learning with new information technologies in schools: perspectives from the psychology of learning and instruction. *Journal of Computer Assisted Learning, 6*(2), 69-87.

De Corte, E. (1994). Technology-supported learning in a stage of transition: The case of mathematics. In R. Lewis & P. Mendelsohn (Eds.), *Lessons from learning* (pp. 113-123). Amsterdam: North-Holland.

De Corte, E., Verschaffel, L., & Lowyck, J. (1998). Computers and learning. In T. Husén & T. N. Postlethwaite & B. R. Clark & G. R. Neave (Eds.), *Education: The complete encyclopedia (CD-ROM)*. Oxford: Pergamon Press.

De Jong, M. L. (1999). *Mathematica for calculus-based physics*. Menlo Park, CA: Addison-Wesley.

de Jong, T. (1992). Learning and instruction with computer simulations: learning processes involved. In E. d. Corte & M. C. Linn & H. Mandl & L. Verschaffel (Eds.), *Computer-Based Learning Environments and Problem Solving*. Berlin: Springer-Verlag.

de Jong, T., & Joolingen, W. R. v. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of Educational Research, 68*(2), 179-201.

Departamento de Ensino Secundário. (2001). *Programa de matemática*. Lisboa: Departamento de Ensino Secundário: Ministério da Educação.

Deus, J. D., Pimenta, M., Noronha, A., Peña, T., & Brogueira, P. *(2000). Introdução à física* (2 ed.). Lisboa: McGraw-Hill. ed.). Lisboa: McGraw-Hill.

Dickey, E. (1998). (A review of) Modellus: Interactive Modeling with Mathematics. *The Mathematics Teacher, 91*(6), 528-530.

diSessa, A. A. (1982). Unlearning aristotelian physics: A study of knowledge-based learning. *Cognitive Science, 6*(1), 37-75.

diSessa, A. A. (1997). *Twenty reasons why your should use Boxer (instead of Logo).* Paper presented at the Sixth European Logo Conference: Learning & Exploring with Logo, Budapest.

diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy.* Cambridge: MIT Press.

diSessa, A. A., Hoyles, C., & Noss, R. (Eds.). (1995). *Computers and Exploratory Learning*. Berlin: Springer.

Doerr, H. M. (1996). Stella ten years later: a review of the literature. *International Journal of Computers for Mathematical Learning, 1*, 201-224.

Domingos, A. (1994). *A aprendizagem de funções num ambiente computacional com recurso a diferentes representações.* Unpublished Master Thesis, Universidade Nova de Lisboa, Monte de Caparica.

Dorn, W. S. (1977). Modeling and computing in the social-behavioral sciences. In A. Jones & H. Weinstock (Eds.), *Computer-based science instruction* (Vol. 24, pp. 141-183). Leyden: Noordhoff.

Douglas, R. G. (Ed.). (1986). *Toward a lean and lively calculus*. Washington, DC: The Mathematical Association of America.

Driver, R. (1983). *The pupil as scientist?* Milton Keynes: Open University Press.

Ebison, M. G. (1993). Newtonian in Mind but Aristotelian at Heart. *Science & Education, 2*(4), 345-362.

Edelson, D. C. (1998). Realising authentic science learning through the adaptation of scientific practice. In B. J. Fraser & K. G. Tobin (Eds.), *International handbook of science education*. Dordrecht: Kluwer Academic Publishing.

Eisenberg, J. (1995). The limits of educational research: Why most research and grant plans in education are futile and wasteful. *Curriculum Inquiry, 25*(4), 367-380.

Eisenberg, M. (1995). Creating software applications for children: some thoughls about design. In A. A. diSessa, C. Hoyles, & R. Noss (Eds.), *Computers and exploratory learning*. Berlin: Springer.

Eisenkraft, A. (Ed.). (1998). *Active physics (several volumes)*. Armonk, NY: It's About Time, Inc.

Eisner, E. W. (1994). *The educational imagination: On the design and evaluation of school programs* (3rd ed.). New York: Macmillan.

Elshout, J. (1998). Creativity. In T. Husén & T. N. Postlethwaite & B. R. Clark & G. R. Neave (Eds.), *Education: The complete encyclopedia (CD-ROM)*. Oxford: Pergamon Press.

Ernst, M. L. (1997). *At the heart of evolving literacy: A framework for action* (Report). Cambridge: Harvard University Program on Information Resources Policy.

Eylon, B.-S. (1998). Physical sciences: secondary school programs. In T. Husén & T. N. Postlethwaite & B. R. Clark & G. R. Neave (Eds.), *Education: The complete encyclopedia (CD-ROM)*. Oxford: Pergamon Press.

European Commission. (2001). *The e-learning action plan*. Brussels: European Commission.

Feiman-Nemser, S., & Floden, R. E. (1986). The cultures of teaching. In M. C. Wittrock (Ed.), *Handbook of research on teaching* . NY: Macmillan.

Féteira, E. (1996). *"Ratos" em movimento: um estudo para computador.* Unpublished MSc Thesis, Universidade Aberta, Lisboa.

Feurzeig, W. (1993). Visualization in educational computer modelling. In T. d. Jong & H. Spada & D. Tawne (Eds.), *The use of computer models for explication, analysis and experiential learning*. Berlin: Springer-Verlag.

Feynman, R., Leighton, R. L., & Sands, M. L. (1963). *The Feynman lectures on physics* (Vol. I). Addison-Wesley: Reading, Mass.

Feynman, R. P. (1982). Simulating physics with computers. *International Journal of Theoretical Physics, 21*, 467-488.

Feynman, R. P. (1985). *QED - The strange theory of light and matter.* Princeton: Princeton University Press.

Finegold, M., & Gorsky, P. (1992). Using computer simulations to restructure students' conceptions of force. *Journal of Computers in Mathematics and Science Teaching, 11*, 163-178.

Fiolhais, C., Silva, L., Valadares, J., & Teodoro, V. D. (1996a). *Física 12.º ano*. Lisboa: Didáctica Editora.

Fiolhais, C., Silva, L., Valadares, J., & Teodoro, V. D. (1996b). *Física 12.º ano, Manual de actividades*. Lisboa: Didáctica Editora.

Fiolhais, C., Silva, L., Valadares, J., & Teodoro, V. D. (1997). *Física 9.º ano, Guia do Professor*. Lisboa: Didáctica Editora.

Forman, G. (1988). Making intuitive knowledge explicit through future technology. In G. Forman & P. B. Pufal (Eds.), *Constructivism in computer age*. Hillsdale, N. J.: Lawrence Erlbaum.

Friel, S. N., Curcio, F. R., & Bright, G. W. (2001). Making sense of graphs: Critical factors influencing comprehension and instructional implications. *Journal for Research in Mathematics Education, 32*(2), 124–158.

Fullan, M. (1998). The meaning of educational change: A quarter of a century of learning. In A. Hargreaves, A. Lieberman, M. Fullan, & D. Hopkins (Eds.), *International handbook of educational change* (pp. 242-260). Dordrecht/Boston/London: Kluwer

Gago, M. (1990). *Manifesto para a ciência em Portugal*. Lisboa: Gradiva.

Galison, P. (1997). *Image & logic, a material culture of microphysics*. Chicago: The University of Chicago Press.

Galitz, W. O. (1997). *The essential guide to user interface design*. NY: John Wiley & Sons, Inc.

Gall, M. D., Borg, W. R., & Gall, J. P. (1996). *Educational research, an introduction*. White Plains, N.Y.: Longman.

Gardner, H. (1987). *The mind's new science*. N. Y.: Basic Books.

Garrison, J. (1997). An alternative to Van Glasersfeld's subjectivism in science education: Deweyan social constructivism. *Science & Education, 6*(6), 543-554.

Gastineau, J. E., Gjertsen, M. H., Risley, J. S., Engelhardt, P. V., Schultz, S. F., Braught, G., Laws, P. W., & Cooney, P. J. (1998). Excel spreadsheet tutorial with Workshop Physics tools. Raleigh, NC: Physics Academic Software, American Institute of Physics.

Gates, W. H. (1995). *The road ahead*. NY: Penguim.

Giddens, A. (1991). *Modernity and self-identity.* Cambridge: Polity.

Giere, R. N. (1989). *Explaining science: a cognitive approach (Science and its conceptual foundations)*. Minneapolis: University of Minnesota Press.

Goodstein, D. (1992). The Science Literacy Gap: A Karplus Lecture. *Journal of Science Education and Technology, 1*(3), 149-155.

Goodstein, D., & Olenick, R. (1988). The making of the mechanical universe. *American Journal of Physics, 56*, 779.

Gordon, S. P., Hughes-Hallett, D., Ostebee, A., & Usiskin, Z. (1994). Report of the content workshops. In A. E.Solow (Ed.), *Preparing for a new calculus: conference proceedings* (pp. 55-59). Washington, D.C.: Mathematical Association of America.

Grant, C. R., & Antueno, E. D. (1997). *Graphic Logo* (Version 4.4). Buenos Aires: Fundaustral. Retrieved August 25, 2002, from the World Wide Web: http://www.nalejandria.com/fundaustral/ingles/graplogo.htm.

Gregory, B. (1988). *Inventing reality: physics as language*. NY: Wiley.

Grouws, D. A. (Ed.). (1992). *Handbook of research on mathematics teaching and learning*. NY: Macmillan.

Hacking, I. (1983). *Representing and interventing: Introductory topics in the philosophy of natural science*. Cambridge: Cambridge University Press.

Hargreaves, A., Lieberman, A., Fullan, M., & Hopkins, D. (1998). *International handbook of educational change*. Boston, Mass.: Kluwer Academic Publishers.

Harvard Educational Technology Center. (1988). *Making sense of the future*. Cambridge, Mass.: Harvard Graduate School of Education.

Hebenstreit, J. (1987). *Simulation et pédagogie: Une rencontre du troisième type*. Gif Sur Yvette: École Superieure d'Electricité.

Hennessy, S., Twigger, D., Driver, R., O'Shea, T., O'Malley, C., Byard, M., Draper, S., Hartley, R., Mohammed, R., & Scanlon, E. (1995). A classroom intervention using a computer-augmented curriculum for mechanics. *lnternational Journal of Science Education, 17*(2), 189-206.

High Performance Systems. (1988). *Stella*. Hannover, NH: High Performance Systems.

High Performance Systems. (1997). *Stella* (Version 5). Hannover, NH: High Performance Systems.

Hiller, J. R., Johnston, I. D., & Styer, D. F. (1995). *Quantum mechanics simulations, The Consortium for Upper-Level Physics Software*. NY: John Wiley & Sons.

Holton, G. (1994). *Science and anti-science*. Cambridge, Mass.: Harvard University Press.

Holton, G. (1996). *Einstein, history, and other passions: The rebellion against science at the end of the twentieth century.* NY: Addison-Wesley.

Hooper, R. (1990). Computers and sacred cows. *Journal of Computer Assisted Learning, 6*(1), 2-13.

Horwitz, P., & Barowy, B. (1994). Designing and using open-ended software to promote conceptual change. *Journal of Science Education and Technology, 3*(3), 161-185.

Hoyles, C. (1995). Exploratory software, exploratory cultures? In A. A. diSessa, C. Hoyles, & R. Noss (Eds.), *Computers and exploratory learning* . Berlin: Springer.

Hoyles, C., Healy, L., & Pozzi, S. (1994). Groupwork with computers: An overview of findings. *Journal of Computer Assisted Learning, 10*(4), 202-215.

Huggins, E. R. (2000). *Physics 2000*. Dartmouth College. Retrieved August 25, 2002, from the World Wide Web: http://www.physics2000.org.

Hurley, J. (1985). *Logo physics*. N. Y.: CBS.

Husén, T. (1988). Research paradigms in education. In J. P. Keeves (Ed.), *Educational research, methodology, and measurement: An international handbook* . Oxford: Pergamon Press.

Hutchinson, J. R., & Huberman, M. (1994). Knowledge dissemination and use in science and mathematics education: a literature review. *Journal of Science Education and Technology, 3*(1), 27-47.

Jenkins, E. W. (1991). History of science education. In A. Lewy (Ed.), *The international encyclopedia of curriculum* (pp. 903-906). Oxford: Pergamon.

Jenkins, E. W. (1996). *Historical and policy-related research and the practice and rationale of school science education.* Paper presented at the British Educational Research Association Annual Conference, Lancaster.

Jennison, B., & Ogborn, J. (Eds.). (1994). *Wonder and delight, Essays in science education in honour of the life and work of Eric Rogers 1902-1990*. Bristol: Institute of Physics Publishing.

Johnson, D. C., Cox, M. J., & Watson, D. M. (1994). Evaluating the impact of IT on pupil's achievement's. *Journal of Computer Assisted Learning, 10*(2), 138-156.

Johnson-Laird, P. N. (1983). *Mental models*. Cambridge, MA: Harvard University Press.

Jones, A., & Weinstock, H. (Eds.). (1977). *Computer-based science instruction* (Vol. 24). Leyden: Noordhoff.

Joyce, B. R. (1974). The madic lantern: metaphor for humanism. In D. R. Olson (Ed.), *Media and symbols: the forms of expression, communication, and education* . Chicago: The University of Chicago Press.

Junqueira, M., Valente, S., Teodoro, V. D., & Seabra, P. (1993). *Thales [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Kahn, B. (1985). *Computers in Science*. Cambridge: Cambridge University Press.

Kapor, M. (1996). A software design manifesto. In T. Winograd (Ed.), *Bringing design to software*. Reading, Mass.: Addison-Wesley.

Key Curriculum Press. (1995). *The geometer's sketchpad Version 3 [Software for Windows]*. Berkeley: Key Curriculum Press.

Key Curriculum Press. (2000). *Fathom Version 1.1 [Software for Windows]*. Berkeley: Key Curriculum Press.

Knowledge Revolution. (1998). *Interactive physics* (Version 5). San Mateo, CA: Knowledge Revolution.

Knowledge Revolution. (1997). *Modellus, interactive modeling with mathematics*. San Mateo, CA: Knowledge Revolution.

Koop, J. (1991). *Spreadsheet physics study units for university physics*. N. Y.: John Wiley.

Korthagen, F. A. J. (2001). *Linking practice and theory: The pedagogy of realistic teacher education.* Paper presented at the Annual Meeting of the American Educational Research Association, Seattle.

Kozma, R. B. (1991). Learning with media. *Review of Educational Research, 61*, 179-211.

Krieger, M. H. (1987). The physicist's toolkit. *American Journal of Physics, 55*(11), 1033-1038.

Landsheere, G. (1988). History of educational research. In J. P. Keeves (Ed.), *Educational research, methodology, and measurement: an international handbook* . Oxford: Pergamon Press.

Laurillard, D. (1995). Multimedia and the changing experience of the learner. *British Journal of Educational Technology, 26*(3), 179-189.

Lawrence, I., & Whitehouse, M. (Eds.). (2000). *Advancing physics AS CD-ROM.* Bristol: Institute of Physics.

Lawrence, I., & Whitehouse, M. (Eds.). (2001). *Advancing physics A2 CD-ROM.* Bristol: Institute of Physics.

Laws, P. W. (1997). *Workshop physics*. NY: Wiley.

Lax, P. D. (1984). In praise of calculus. *The College Mathematics Journal, 15*(5), 378-380.

Lederman, L. M. (1998). *ARISE: American Renaissance in Science Education* (FERMILAB-TM-2051). Batavia, Illinois: Fermi National Accelerator Laboratory.

Levin, J. A., & Waugh, M. (1988). Educational simulations, tools, games, and microworlds: Computer-based environments for learning. *International Journal of Educational Research, 12*, 71-79.

Lévy, P. (1994). *As tecnologias da inteligência: O futuro do pensamento na era da informática*. Lisboa: Instituto Piaget.

Liddle, D. (1996). Design of the conceptual model. In T. Winograd (Ed.), *Bringing design to software*. Reading, Mass.: Addison-Wesley.

Lieberman, A. (1998). Introduction, *International handbook of educational change*. Boston, Mass.: Kluwer Academic Publishers.

Lijnse, P. L. (1998). Curriculum development in physics education. In Andrée Tiberghien & E. L. Jossem & J. Barojas (Eds.), *Connecting research in physics education with teacher education*. The International Commission on Physics Education.

Lobo, M. S., Clérigo, F., Martins, F., Lobo, L. S., & Teodoro, V. D. (1993). *Dinamix, Modelação de sistemas dinâmicos em computador [Software for MS-DOS]*. Lisboa: GEP, Ministério da Educação.

Luetzelschwab, M., Laws, P., Gile, M., & Cooney, P. (1997). VideoPoint (Version 2.1). Roseville, Ca: Lenox Softworks / Pasco Scientific.

Lunetta, V. N. (1991). Physics programs. In A. Lewy (Ed.), *The international encyclopedia of curriculum* (pp. 930-935). Oxford: Pergamon.

Mandelbrot, B. B. (1992). Fractals and the rebirth of experimental mathematics. In H. Peitgen & H. Jurgens & D. Saupe (Eds.), *Fractals for the classroom: part one, introduction to fractals and chaos* (pp. 1-16). New York: Springer-Verlag.

Mandinach, E. B. (1988). *Self-regulated learning substudy: Systems Thinking and Curriculum Innovation (STACI) Project* (Technical Report TR88-25). Cambridge: Educational Technology Center (Harvard Graduate School of Education).

Mandinach, E. B. (1989). Model-building and the use of computer simulation of dynamic systems. *Journal of Educational Computing Research, 5*(2), 221-243.

Mandinach, E. B., & Cline, H. F. (1994). *Classroom dynamics: implementing a technology-based learning environment*. Hillsdale, NJ: Erlbaum.

Martins, A. (Coord.). (2002). *O livro branco da Física e da Química*. Lisboa: Sociedade Portuguesa de Física e Sociedade Portuguesa de Química.

Marx, G. (1994). Shortcut to the future. In B. Jennison & J. Ogborn (Eds.), *Wonder and delight, Essays in science education in honour of the life and work of Eric Rogers 1902-1990*. Bristol: Institute of Physics Publishing.

McDermott, L. C. (1993). Guest comment: how we teach and how students learn — A mismatch? *American Journal of Physics, 61*(4), 295-298.

McDermott, L. C., Rosenquist, M. L., & Zee, E. H. v. (1987). Student difficulties in connectiong graphs and physics: examples from kinematics. *American Journal of Physics, 55*, 503-513.

Mellar, H., Bliss, J., Boohan, R., Ogborn, J., & Tompsett, C. (Eds.). (1994). *Learning with artificial worlds: computer based modelling in the curriculum*. London: Falmer Press.

Microsoft. (1995). *The Windows interface guidelines for software design*. Redmond: Microsoft Press.

Millar, R., Le Maréchal, J.-F., & Tiberghien, A. (1998). *A map of the variety of labwork* (Working Paper 1). York, UK: European Commission, Targeted Socio-Economic Research Programme, Project PL 95-2005 (Labwork in science education).

Mosterín, J. (1998). Technology-mediated observation. *Techné: Journal of the Society for Philosophy and Technology (http://scholar.lib.vt.edu/ejournals/SPT), 4*(2), 68-80.

National Council of Teachers of Mathematics. (1989). *Curriculum and evaluation standards for school mathematics*. Reston,VA: NCTM.

National Council of Teachers of Mathematics. (1998). *Principles and standards for school mathematics: discussion draft*. Reston, VA: National Council of Teachers of Mathematics.

National Council of Teachers of Mathematics. (2000). *Principle and Standards for School Mathematics*. Reston,VA: NCTM.

National Research Council. (1989). *Everybody counts*. Washington: National Academy of Sciences.

National Research Council. (1996). *National Science Education Standards*. Washington: National Academy Press.

National Science Teachers Association. (1996). *Scope, sequence and coordination of secondary school science: A framework for school science*. Washington, DC: NSTA.

Nickerson, R. S. (1995). Can technology help teach for understanding? In D. N. Perkins, J. L. Schwartz, M. M. West, & M. S. Wiske (Eds.), *Software goes to school* . NY: Oxford University Press.

Niedderer, H., & Petri, J. (2000). *Mit der Schrödinger-Gleichung vom H-Atom zum Festkörper*. Bremen: Universität Bremen Institut für Didaktik der Physik.

Norman, D. A. (1991). Cognitive artefacts. In J. M. Carroll (Ed.), *Designing interaction: Psychology at the Human-Computer Interface* . Cambridge: Cambridge University Press.

Noss, R. (1995). Computers as commodities. In A. A. diSessa, C. Hoyles, & R. Noss (Eds.), *Computers and exploratory learning* . Berlin: Springer.

Nuffield Foundation. (1967). *Physics*. London: Longman, Harlow and Penguin.

Oettinger, A. G. (1969). *Run, computer, run: The mythology of educational innovation*. Cambridge, MA: Harvard University Press.

Ogborn, J. (1984). A microcomputer modelling system and the teaching of problem structure, *Recherche en Didactique de la Physique* . Paris: Éditions du CNRS.

Ogborn, J. (1994). Modelling clay for computers. In B. Jennison & J. Ogborn (Eds.), *Wonder and delight, Essays in science education in honour of the life and work of Eric Rogers 1902-1990*. Bristol: Institute of Physics Publishing.

Ogborn, J. (1997). The Institute of Physics 16-19 Physics Initiative - just beginning... *Physics Education, 32*(6), 422-424.

Ogborn, J. (1999). Modeling clay for thinking and learning. In W. Feurzeig & N. Roberts (Eds.), *Modeling and simulation in science and mathematics education* (pp. 5-37). New York: Springer.

Ogborn, J. (2002). Ownership and transformation: teachers using curriculum innovation. *Physics Education, 37*(2), 142-146.

Ogborn, J., & Holland, D. (1989). *Cellular modelling system*. London: University of London Institute of Education.

Ogborn, J., & Whitehouse, M. (Eds.). (2000). *Advancing physics AS*. Bristol: Institute of Physics.

Ogborn, J., & Whitehouse, M. (Eds.). (2001). *Advancing physics A2*. Bristol: Institute of Physics.

Olson, D. R. (1986). Intelligence and literacy: the relationships between intelligence and the technologies of representation and communication. In R. J. Sternberg & R. K. Wagner (Eds.), *Practical intelligence: Nature and origins of competence in the everyday world* (pp. 338-360). NY: Cambridge University Press.

Olson, D. R., & Torrance, N. (Eds.). (1996). *The handbook of education and human development*. Oxford: Blackwell.

Orhum, E. (1995). Design of computer-based cognitive tools. In A. A. diSessa, C. Hoyles, & R. Noss (Eds.), *Computers and exploratory learning* . Berlin: Springer.

Osborne, J. (1990). Sacred cows in physics—towards a redefinition of physics education. *Physics Education, 25*(4), 189-196.

Oxford University Press. (1996). *The Concise Oxford Dictionary.* (Ninth ed.). Oxford: Oxford University Press.

Papert, S. (1980). *Mindstorms: children, computers and powerful ideas*. N. Y.: Basic Books.

Papert, S. (1994). *The children's machine: rethinking school in the age of the computer.* Harvester: Wheasheaf.

Pea, R., & Cuban, L. (1998). *The pros and cons of technology in the classroom*. Retrieved August 25, 2002, from the World Wide Web: http://www.tappedin.org/info/teachers/debate.html.

Perkins, D. N., Schwartz, J. L., West, M. M., & Wiske, M. S. (Eds.). (1995). *Software goes to school, teaching for understanding with new technologies*. NY: Oxford University Press.

Peters, P. C. (1982). Even honours students have conceptual difficultites with physics. *American Journal of Physics, 50*(6), 501-508.

Pfundt, H., & Duit, R. (1991). *Bibliography: Students' alternative frameworks and science education*. Kiel: Institute for Science Education.

Physical Science Study Committee. (1960). *Physics*. Boston: Heath & Co.

Planck, M. (1950). *Scientific autobiography and other papers*. London: Williams & Norgate.

Ponte, J. P. (1984). *Functional reasoning and the interpretation of cartesian graphs.* Unpublished Doctor of Education Thesis, University of Georgia, Athens.

Ponte, J. P. (1994). *MINERVA Project, introducing NTI in education*. Lisboa: DEPGEF.

Popper, K. (1977). *The self and its brain*. London: Routledge & Kegan Paul.

Popper, K. (1989). *Conjectures and refutations*. London: Routledge.

Pryluck, C. (1968). Structural analysis of pictures as a symbol system. *AV Communication Review, 16*, 372-402.

Raizen, S. A. (1991). The reform of science education in the U.S.A. Déjà vu or de novo? *Studies in Science Education, 19*(1-41).

Ralston, A. (1999). Let's abolish pencil-and-paper arithmetic. *Journal of Computers in Mathematics and Science Teaching, 18*(2), 173-194.

Redish, E. F. (1994). The Implications of Cognitive Studies for Teaching Physics. *American Journal of Physics, 62*(6), 796-803.

Redish, E. F., & Wilson, J. M. (1993). Student programming in the introductory physics course: M.U.P.P.E.T. *American Journal of Physics, 61*, 222-232.

Repenning, A. (1993). Agentsheets: A tool for building domain-oriented dynamic visual environments. Unpublished PhD thesis, University of Colorado.

Resnick, M. (1994). *Turtles, termites, and traffic jams*. Cambridge, MA: MIT Press.

Resnick, L. B., & Collins, A. (1998). Cognition and learning. In T. Husén & T. N. Postlethwaite & B. R. Clark & G. R. Neave (Eds.), *Education: The complete encyclopedia (CD-ROM)*. Oxford: Pergamon Press.

Rex, A., & Jackson, M. (2000). *Integrated physics and calculus* (2 vols.). Reading, MA: Addison-Wesley Longman.

Ribeiro, C., & Rocha, M. d. L. (1993). *Novas perspectivas no ensino da física e da química*. Lisboa: GEP: Ministério da Educação.

Robertson, J. (1998). Paradise lost: children, multimedia and the myth of interactivity. *Journal of Computer Assisted Learning, 14*(1), 31-39.

Robinson, N. (1992). Evaluating simulations and games: an economist's view. *Simulations/Games for Learning and Instruction, 22*, 308-325.

Rogers, L., & Wild, P. (1996). Data-logging: effects on practical science. *Journal of Computer Assisted Learning, 12*(3), 130-145.

Roitman, J. (1998). A Mathematician Looks at National Standards. *Teachers College Record, 1*(Fall1), 22-44.

Ronen, M., Langley, D., & Ganiel, U. (1992). Integrating computer simulations into high school physics teaching. *Journal of Computers in Mathematics and Science Teaching, 11*, 319-329.

Ronen, M. (1995). A 3-D motion tracing system in High School Physics teaching. *Journal of Computer Assisted Learning, 11*(3), 141-156.

Ross, S. (1993). Computers and physics teaching. *Physics Education, 28*(2), 69.

Roth, W.-M. (1995). *Authentic school science: knowing and learning in open-inquiry science laboratories*. Dordrecht ; Boston: Kluwer Academic Publishers.

Roth, W.-M. (1996). The co-evolution of situated language and physics knowing. *Journal of Science Education and Technology, 5*(3), 171-191.

Russell, B. (1948). *Our knowledge of the external world*. London: Allen & Unwin.

Salomon, G. (1974). What is learned and how it is taught: the interaction between media, message, task, and learner. In D. R. Olson (Ed.), *Media and symbols: the forms of expression, communication, and education* . Chicago: The University of Chicago Press.

Salomon, G. (1990). Studying the flute and the orchestra: controlled vs. classroom research on computers. *International Journal of Educational Research, 14*, 521-532.

Salomon, G. (1992). Effects *with* and *of* computers and the study of computer-based learning environments. In E. De Corte, M. C. Linn, H. Mandl, & L. Verschaffel (Eds.), *Computer-based learning environments and problem solving* . Berlin: Springer-Verlag.

Salomon, G., Perkins, D. N., & Globerson, T. (1991). Partners in cognition: extending human intelligence with intelligent technologies. *Educational Researcher, 20*(3), 2-9.

Santos, A. M. N. d., & Teodoro, V. D. (1997). *Química 10.º ano, Guia do Professor.* Lisboa: Didáctica Editora.

Schank, R. C. (1986). *Explanation patterns, understanding mechanically and creatively.* Hillsdale, N. J.: Lawrence Erlbaum.

Schecker, H. (1993). Learning physics by making models. *Physics Education, 28*(2), 102-106.

Schwab, J. J. (1983). The pratical 4: something for curriculum professors to do. *Curriculum Inquiry, 13*(3), 239-265.

Schwartz, J. L. (1989). Intellectual mirrors: A step in the direction of making schools knowledge-making places. *Harvard Educational Review, 59*(1), 217-222.

Schwartz, J. L. (1993a). Introduction: The many faces of educational innovation. In J. L. Schwartz, M. Yerushalmy, & B. Wilson (Eds.), *The geometric supposer: What is the case of?* . Hillsdale, NJ: Erlbaum.

Schwartz, J. L. (1993b). A Personal View of the Supposer: Reflections on Particularities and Generalities in EducationaI Reform. In J. L. Schwartz, M. Yerushalmy, & B. Wilson (Eds.), *The geometric supposer: what is a case of?* . Hillsdale, NJ: Erlbaum.

Schwartz, J. L. (1997). What happened to the voice of the author? *Journal of Science Education and Technology, 6*(2), 83-90.

Schwartz, J. L., Yerushalmy, M., & Wilson, B. (Eds.). (1993). *The geometric supposer: what is a case of?* Hillsdale, NJ: Erlbaum.

Schwarz, C., & Beichner, R. (1997). *Interactive journey through physics*. Upper Slader River, NJ: Prentice Hall.

Shabajee, P., & Postlethwaite, K. (2000). What happened to modern physics? *School Science Review, 81*(297), 55-55.

Shamos, M. (1995). *The myth of scientific literacy.* New Brunswick, NJ: Rutgers University Press.

Shneiderman, B. (1982). The future of interactive systems and emergence of direct manipulation. *Behaviour and Information Technology, I*, 237-256.

Shneiderman, B. (1983). Direct manipulation: a step beyond programming languages. *IEEE Computer, 16*(8), 57-69.

Shneiderman, B. (1992). *Designing the user interface*. Reading, Mass.: Addison-Wesley.

Shuchat, A., & Shultz, F. (1994). *The joy of Mathematica®*. Reading, Mass.: Addison-Wesley.

Shulman, L. S. (1986). Paradigms and research programs in the study of teaching: A contemporary perspective. In M. C. Wittrock (Ed.), *Handbook of Research on Teaching* . NY: MacMillan.

Silva, J. S. (1975). *Guia para a utilização do compêndio de matemática* (Vol. 2-3). Lisboa: Gabinete de Estudos e Planeamento do Ministério da Educação.

Sparkes, R. A. (1984). *The ZX Spectrum in science teaching*. London: Hutchison & Co.

Squires, D., & McDougoll, A. (1996). Software evaluation: a situated approach. *Journal of Computer Assisted Learning, 12*(3), 146-161.

Stoll, C. (1995). *Sillicon snake oil: second thoughts on the information highway.* NY: Doubleday.

Tam, P. T. (1996). *A physicist's guide to Mathematica*. NY: Academic Press.

Tao, P.-K., & Gunstone, R. F. (1999). Conceptual change in science through collaborative learning at the computer. *International Journal of Science Education, 21*(1), 39-57.

Taylor, R. P. (Ed.). (1980). *The computer in the school: tutor, tool, tutee*. N. Y.: Teachers' College Press.

Taylor, R. P. (2000, 4-6 December). *A digital world in the school access, collaborate, communicate, and experience: dominant types of use since 1980, an update to the computer in the school: tutor, tool, tutee.* Paper presented at the RIBIE 2000, Viña del Mar, Chile.

Tempel, M. (1998). The state of the turtle. *Logo Update, 6*.

Teodoro, V. D. (1990). Computadores, aprendizagem e ensino da física: O computador como ‘laboratório conceptual’. *Análise Psicológica, VIII*(1), 75-82.

Teodoro, V. D. (1992). Direct manipulation of physical concepts in a computerized exploratory laboratory. In E. d. Corte, M. C. Linn, H. Mandl, & L. Verschaffel (Eds.), *Computer-Based Learning Environments and Problem Solving.* Berlin: Springer-Verlag.

Teodoro, V. D. (1993a). A model to design exploratory software for science and mathematics. In T. d. Jong & H. Spada & D. Tawne (Eds.), *The use of computer models for explication, analysis and experiential learning*. Berlin: Springer-Verlag.

Teodoro, V. D. (1993b). The computer as a conceptual lab: learning dynamics with an exploratory environment. In D. L. Ferguson (Ed.), *Advanced technologies in the teaching of mathematics and science* . Berlin: Springer-Verlag.

Teodoro, V. D., & Batalha, J. C. (1990). *Funções [Software for MS-DOS]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Batalha, J. C. (1993). *Mendeleieff [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Batalha, J. C. (1993). *Variações [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Clérigo, F. (1993). *Galileo [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Clérigo, F. (1994). *Funções e derivadas [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Clérigo, F. (1994). *Rato em movimento [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Clérigo, F. (1994). *Vrum... vrum... jogos de movimento [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Santos, V. d. (1990). *Espaço [Software for MS-DOS]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Seabra, P. (1990). *Cinemática [Software for MS-DOS]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Seabra, P. (1993). *Jogos de funções [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Seabra, P. (1994). *Estatística: Explorar conceitos básicos [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Seabra, P. (1994). *Jogos de funções II [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Silva, J. C. (1990). *Galileo [Software for Macintosh]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Vieira, J. P. (1994). *Jogos de campos de força [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., & Vieira, J. P. (1994). *Outro mundo [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., Gonçalves, P., & Melo, A. (1990). *Projécteis [Software for MS-DOS]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., Ribeiro, C., Rocha, L., & Oliveira, E. (1993). *Faraday [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., Romano, F., & Cardoso, T. (1994). *Sinais e osciloscópio [Software for Windows]*. Lisboa: GEP, Ministério da Educação.

Teodoro, V. D., Vieira, J. P., & Clérigo, F. C. (1997). *Modellus, interactive modelling with mathematics*. San Mateo, CA: Knowledge Revolution.

Teodoro, V. D., Vieira, J. P., & Clérigo, F. C. (2000). *Modellus, interactive modelling with mathematics (Version 2.01)*. Monte de Caparica: Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa.

Tinker, R. F. (1996). Introduction, *Microcomputer-based labs: educational research and standards*. NY: Springer-Verlag.

Tonks, D., & Armitage, S. (1997). A framework for understanding learning from management simulations. *Journal of Computer Assisted Learning, 13*(1), 48-58.

Travers, M. D. (1996). *Programming with Agents: New metaphors for thinking about computation.* Unpublished PhD thesis, Massachusetts Institute of Technology, Cambridge.

Tressel, G. W. (1994). Thirty years of "Improvement" in precollege math and science education. *Journal of Science Education and Technology, 3*(2), 77-88.

Veit, E., & Mors, P. (2000). *Física, um curso introdutório*. Porto Alegre: Instituto de Física, UFRGS.

Viennot, L. (1979). *Raisonnement spontanné en dynamique elémentaire*. Paris: Hermann.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes*. Cambridge: Harvard University Press.

Wandersee, J. H., Mintzes, J. J., & Novak, J. D. (1994). Research on alternative conceptions in science. In D. L. Gabel (Ed.), *Handbook of research on science teaching and learning*. NY: MacMillan.

Watkins, J., Augousti, A., & Calverley, G. (1997). Evaluation of a physics multimedia resource. *Computers & Education, 28*(3), 165-171.

Webb, M., & Hassell, D. (1988). Opportunities for computer based modelling and simulation in secondary education. In F. Lovis & E. D. Tagg (Eds.), *Computers in Education*. Amsterdam: North-Holland.

White, B. Y. (1983). Sources of Difficulty in Understanding Newtonian Dynamics. *Cognitive Science, 7*(1), 41-66.

White, B. Y. (1993). ThinkerTools: causal models, conceptual change, and science education. *Cognition and Instruction, 10*, 1-100.

White, B. Y. (1984). Designing computer games to help physics students understand Newton´s laws of motion. *Cognition and Instruction, 1*(1), 69-108.

Wilson, J. M. (1997). Some possible futures for the introductory physics course. In J. Wilson (Ed.), *Conference on the introductory physics course, on the occasion of the retirement of Robert Resnick* (pp. 21-31). NY: Wiley.

Winograd, T. (1996). *Bringing design to software*. Reading, Mass.: Addison Wesley.

Wright, J. C., & Wright, C. S. (1998). A commentary on the profound changes envisioned by the National Science Standards. *Teachers College Record, 100*(1), 122-143.

Wullenweber, M. (1996). *Albert*. Berlin: Spinger.

Zimmerman, R. L., & Olness, F. I. (1995). *Mathematica® for physics*. Reading, Mass.: Addison-Wesley.